Massachusetts Institute of Technology

THRESHOLDS REVISITED

Author(s): DANIEL CARDOSO LLACH and FARZIN LOTFI-JAM

Source: *Thresholds*, 2022, No. 50, Before After (2022), pp. 289-300

Published by: Massachusetts Institute of Technology

Stable URL: https://www.jstor.org/stable/10.2307/48821184

JSTOR is a not-for-profit service that helps scholars, researchers, and students discover, use, and build upon a wide range of content in a trusted digital archive. We use information technology and tools to increase productivity and facilitate new forms of scholarship. For more information about JSTOR, please contact support@jstor.org.

Your use of the JSTOR archive indicates your acceptance of the Terms & Conditions of Use, available at https://about.jstor.org/terms



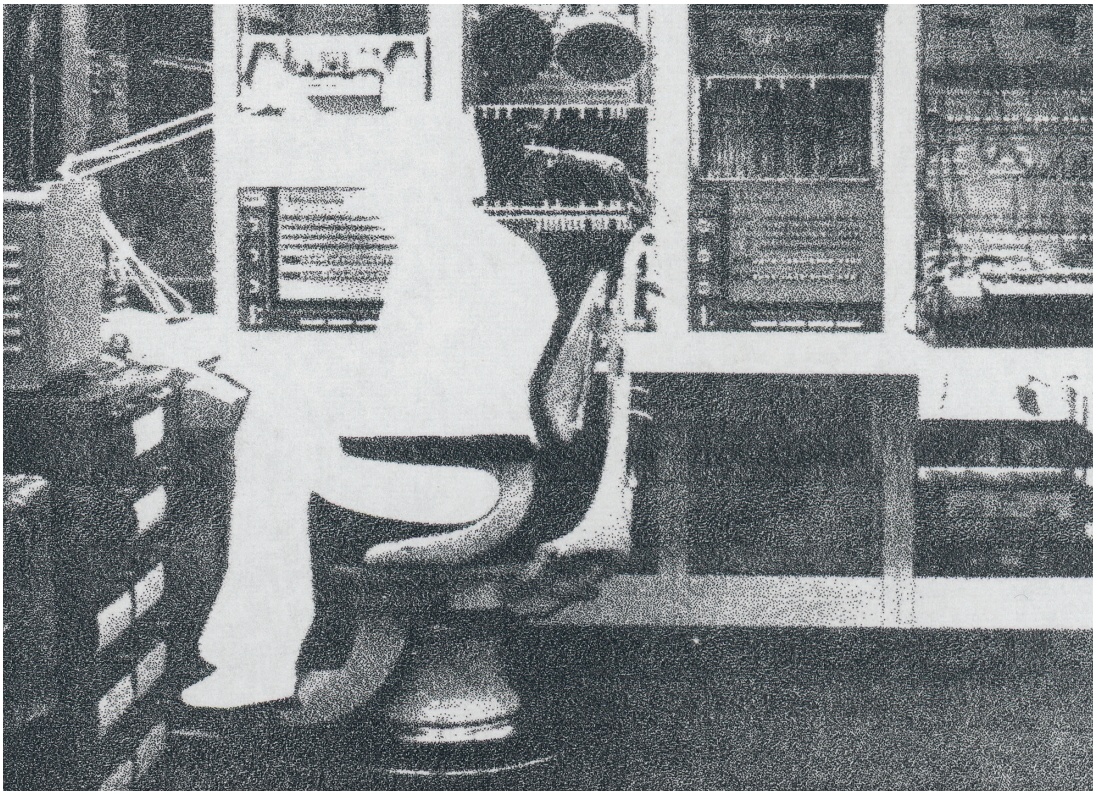
Massachusetts Institute of Technology is collaborating with JSTOR to digitize, preserve and extend access to *Thresholds*

This content downloaded from
71.60.57.98 on Mon, 20 Jul 2026 18:25:57 UTC
All use subject to https://about.jstor.org/terms

THRESHOLDS REVISITED
INERTIA OF AN AUTOMATED UTOPIA: DESIGN COMMODITIES AND AUTHORIAL AGENCY 40 YEARS AFTER "THE ARCHITECTURE MACHINE"

DANIEL CARDOSO LLACH
IN CONVERSATION WITH
FARZIN LOTFI-JAM

It was over a decade ago that Daniel Cardoso Llach first engaged with the subject of techno-utopian architects and their use of software as tools of creation in his 2011 essay, "Inertia of an Automated Utopia." He revisits the essay here to relate and reflect upon technological imaginaries of design both as historically constituted situations and as ongoing discursive and political projects. Farzin Lotfi-Jam works through the same lens to reanimate the piece through new commentary that critically observes the ways in which computation has been made to penetrate even the most intimate aspects of daily life.



doi.org/10.1162/thld_a_00762 © 2022 Daniel Cardoso Llach and Farzin Lofti-Jam

This content downloaded from
71.60.57.98 on Mon, 20 Jul 2026 18:25:57 UTC
All use subject to https://about.jstor.org/terms

01 FLJ: I admire this article as it has a continued relevance today and provides a critical overview of an early application of computer technologies within architecture. It analyzes the origins of enduring technological narratives that position the computer as the utopian solution to complex social problems. As the article shows, rather than providing such promised fixes, techno-utopian imaginaries have instead paved the way for the penetration of computation into all spheres of life, a transformation whose effects continue to demand critical scrutiny.

Inertia of an Automated Utopia[01]

Design Commodities and Authorial Agency 40 Years after "The Architecture Machine"

Daniel Cardoso Llach[02]

In a concise history of architects' embrace of computer-aided design (CAD) Robert Bruegmann notes that in the early days of CAD two camps – Nicholas Negroponte and Yona Friedman in one, and SOM in another – exemplified diverging conceptions of the role of computers in architecture.[1] Whereas SOM pioneered the development of software tools to automate aspects of the design of high-rise buildings, such as cost/area calculation for maximized revenue, Negroponte and Friedman developed socio-technical utopias where computers become partners of people in the design of a "more humane" environment. They envisioned computers as "liberators" of design expertise, allowing people to bypass traditional architects and planners – professionals they saw as elitist middlemen – and the generalizing assumptions about dwellers that, in their view, drove the contemporary housing market. Nicholas Negroponte's influential book *The Architecture Machine*, published in 1970 by the MIT Press, synthesized this provocative view of CAD in a collection of speculative scenarios and technological artifacts that projected design as an amiable conversation between humans and computers. While an array of fields celebrates *The Architecture Machine*'s pioneering enunciation of key computational paradigms, including gestural and windows-based interfaces, in this article I focus on aspects of *The Architecture Machine*'s cultural and institutional context to unfold its dimension as a social – not to mention architectural – critique. I show that by swapping the social roles of architects and dwellers through intelligent

Figure 1. Photograph of an early CAD system altered by the author. Original from Nicholas Negroponte, *The Architecture Machine* (Cambridge: MIT Press, 1971), 55. Courtesy of MIT Press, Cambridge, Mass.

Daniel Cardoso Llach, "Inertia of an Automated Utopia: Design Commodities and Authorial Agency 40 Years after 'The Architecture Machine.'" *Thresholds*, no. 39 (2011): 39.

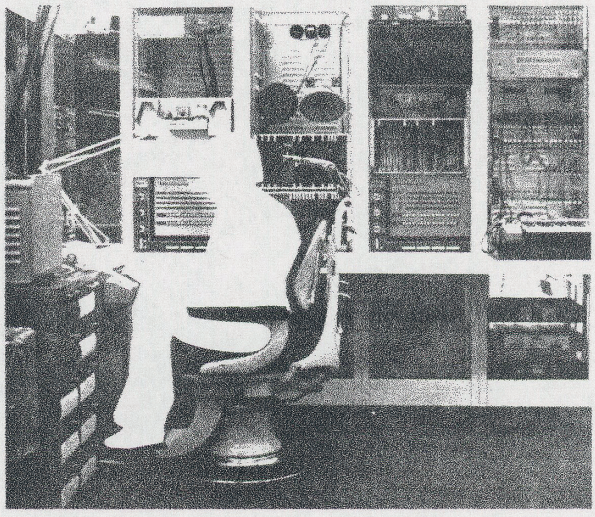
This content downloaded from
71.60.57.98 on Mon, 20 Jul 2026 18:25:57 UTC
All use subject to https://about.jstor.org/terms

machines, Negroponte sought to de-stabilize traditional conceptions of architectural authorship and that, by construing computers as social subjects, he aimed at redefining a contemporary debate about human-machine interaction. The terms of this redefinition continue to influence and limit our expectations about design, architecture, and technology.

From "perfect slaves" to "mechanical partners"

> Let us build machines that can learn, can grope, and can fumble, machines that will be architectural partners, architecture machines.[2]

When *The Architecture Machine* was published, computers were unique, expensive devices that only an academic and government elite had access to. However, the notion of widespread personal computing was already part of the American popular imagination, fueled in part by the awe-inspiring wartime advances in electronics and by the contemporary promotion of technology and automation as patriotic endeavors.[3] The MIT School of Architecture, where Negroponte's Architecture Machine Group was housed, was literally (and physically) immersed in an intense climate of technoscientific advancement. MIT – the recipient of the largest sums of federal research funds during the Cold War – was the epicenter of research on mechanical sensors and actuators, cathode-ray tube monitors and computer graphics.[4] The work of Steven Coons, one of



02 DCL: It is a rare opportunity to be able to reflect on a piece of writing a decade or more after its publication. At the time I was a PhD candidate who had just embarked upon a year-long period of fieldwork. I aspired to contrast the images of design advanced by techno-utopian visionaries such as Steven A. Coons and Nicholas Negroponte with the embodied realities of contemporary designers and builders using software to support architectural work. In retrospect, I think that writing it at that precise moment was key for me to begin to define a place of enunciation—a position from which to describe, narrate, and reflect upon—technological imaginaries of design both as historically constituted and as ongoing discursive and political projects.

The article informed one of the chapters in my doctoral dissertation and re-appeared a few years later in an expanded form in my book *Builders of the Vision*. However, instead of expanding the essay here with the new materials that I added over time, I think it is more in the spirit of this issue to make only a few minor editorial changes to the original text, and to instead add reflections, commentaries, and updated references as footnotes. One reason for this decision is that I think that despite its limitations—and an ever-shifting technological landscape—the essay's critical observations and analyses still hold relevance today.

One theme the article develops is that technological visions of design operate at different registers, including but extending far beyond arguments about their function. They "carry a blueprint" not merely for how artifacts such as buildings ought to be produced, but also for how work should be organized—including who has a say in its organization—and what values are deemed important. This makes the technological experimentalism of ArcMac, and design technologies more generally, interesting from both an architectural and a science and technology studies perspective—a disciplinary juncture that shaped (and continues to shape) my work. I wanted to, on the one hand, call attention to the fact that ArcMac's experimentalism was, at least on the surface, a "critical" project, and on the other hand, offer tools to put the effectiveness, and perhaps the sincerity, of that project into question.

This content downloaded from
71.60.57.98 on Mon, 20 Jul 2026 18:25:57 UTC
All use subject to https://about.jstor.org/terms

Negroponte's academic advisors, had a strong influence on the ideas of *The Architecture Machine*. As a professor of mechanical engineering, Coons led important developments in computer graphics and computer-aided manufacturing and was an enthusiastic advocate for the use of computers in art, architecture and design.[5] Coons conceived design as an iterative process of representation and analysis where a "creative stage," in which a design representation is created, was followed by a "mechanical stage," in which the representation is tested against different metrics – such as mechanical stress and overall structural performance. Coons believed that the "creative stage" was better suited for human designers while the "mechanical stage" could be successfully automated.[6] In *The Architecture Machine*, Negroponte would adopt the technologies that Coons helped develop as the constitutive elements of his vision, but not without rejecting the dualism between creativity and analysis that rendered computers as the "perfect slaves" of a design process.[7] Negroponte instead used a language that endowed computers with human-like attributes such as curiosity and judgment, and construed human-machine interaction as the "cohabitation of two intelligent species,"[8] as a symbiotic relationship that gave origin to an "extended designer," and as an "ecology of mutual design complementation, augmentations, and substitution."[9] Despite this key rhetorical difference, Coons' themes are noticeable throughout *The Architecture Machine*.[03] Among these the notion of adaptability deserves special analysis. Coons distinguished between two different CAD philosophies: one that relied on a large set of specialized procedures and another that relied on a single one that *adapted* to its context.[10] In *The Architecture Machine*, Negroponte echoed the contrast between these two philosophies by opposing the notion of *flexibility* – the capacity of a system to be extended within a fixed set of constraints – to the notion of *adaptability* – the ability of a single computational procedure to find solutions to different problems. In Coons' and Negroponte's view, user interfaces and programming languages could provide flexibility, but not "true" adaptability:

> We have a condition where each designer is creating his own library of services out of the problem-oriented [programming] language. Once created, note that these operations are no less rigid than the predefined package of design commodities.[11]

40

Daniel Cardoso Llach, "Inertia of an Automated Utopia: Design Commodities and Authorial Agency 40 Years after 'The Architecture Machine.'" *Thresholds*, no. 39 (2011): 40.

03 DCL: I would write this differently today. Like most people, the historical actors portrayed here were not as consistent in their positions as my framing might suggest. For example, both Coons and Negroponte (along with many of their contemporaries) used metaphors of human-machine collaboration or subservience when the context suited. They deployed these metaphors tactically depending on the audience they sought to persuade. Their theoretical stances were as pliable and adaptive as the computer-aided design systems that they imagined.

This content downloaded from
71.60.57.98 on Mon, 20 Jul 2026 18:25:57 UTC
All use subject to https://about.jstor.org/terms

However, the means and methods to achieve adaptability were never discussed, and thus the concept remains as a suggestive device, a black box enclosing a computer's potential to revert the perverse effects of industrial standardization and of the notion of *l'homme type* on architectural and urban spaces.[12]

Clashing with the World or, "Ted, many conflicts are occurring"

Instead of presenting a single artifact, as its title suggests, *The Architecture Machine* developed a collection of scenarios and themes, the majority of which fell into one of two categories. In the first, computers were regarded as artificial experts that guide the dweller through design choices until a unique and optimal residential solution is reached—a sort of architectural "expert system."[13] In the second, a computer network was offered as a communication platform between a community and a human designer. The network's central processor would interpret the community's preferences into clear spatial guidelines to be followed by a human architect, thus penetrating what Negroponte considered "the designer-dweller dissonance that exists in today's housing problem."[14] As a proof of concept for the first, "expert system" category, Negroponte presented URBAN5, a computer-aided design system comprising a cathode-ray tube monitor, a keyboard, an optic pen, and a small console with several controls—including a notorious "PANIC" button. Described as an experiment to test the desirability of natural language as a communication medium between humans and machines, URBAN5 received text commands from the user and interpreted them as geometric and spatial constraints [Figure 3]. The program would then calculate a response based on user inputs, a sort of architectural version of the early language-processing program Eliza.[15] URBAN5's responses were displayed on the cathode-ray tube monitor as compositions of three-

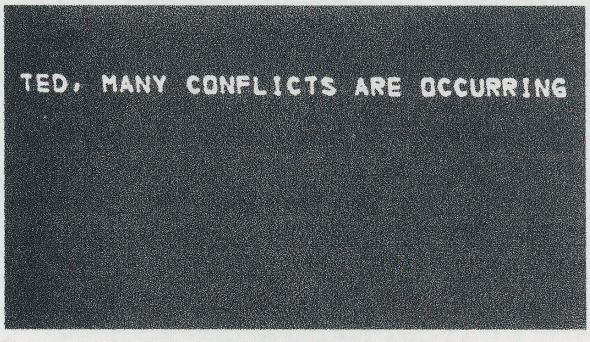


Figure 2. Early CAD system URBAN5 alerts the user "Ted" of design conflicts. From Negroponte, *The Architecture Machine*, 85. Courtesy of MIT Press, Cambridge, Mass.

04 FLJ: I want to find out more about computer expert systems and the emergence of the architect as a techno-social expert at MIT. Not to fall into an MIT recursive reference loop, but it might be a good time to revisit *A Second Modernism.** The publication offers an account of architecture's postwar ambition, which the text sums up as a preference for technical expertise over artistic intuition. This impetus is used by architecture practices to transform architecture itself into a research-oriented and technologically complex discipline of design expertise.

* Arindam Dutta, ed. *A Second Modernism: MIT, Architecture, and the "Techno-Social" Moment.* (Cambridge, MA: SA+P Press, Department of Architecture, Massachusetts Institute of Technology, 2013).

This content downloaded from
71.60.57.98 on Mon, 20 Jul 2026 18:25:57 UTC
All use subject to https://about.jstor.org/terms

05 DCL: A great deal of brilliant scholarship on the subject of ArcMac has been produced since "Inertia" was published—including an in-depth historical chapter in Molly Wright Steenson's book *Architectural Intelligence*, and Morgan Ames' recent *The Charisma Machine*, as well as chapters by Felicity Scott and Mary Louise Lobsinger in *A Second Modernism*. Steenson's book and Scott's chapter in particular offer invaluable context about the institutional context of the group, including details about its funding.

06 FLJ: In a more recent revisiting, Orit Halpern places this experiment within a larger context of urban and racial inequalities: "Situated within a moment of extreme urban crisis and violence, the Architecture Machine Group attempted to turn the external traumas of American racism and economic crisis into an interactive simulation and to advance computing as the solution to these structural problems."*

* Matteo Pasquinelli and Orit Halpern, "The Trauma Machine: Demos, Immersive Technologies and the Politics of Simulation," *Alleys of Your Mind: Augmented Intelligence and Its Traumas* (Lüneburg: Meson Press, Hybrid Publishing Lab, Centre for Digital Cultures, Leuphana University of Lüneburg, 2015).

07 FLJ: I am reminded here of Wendy Hui Kyong Chun's critique of transparency and computation. Chun highlights the difficulty and complexity in defining software through a discussion of the history of programming, and later, through a discussion on the development of structured programming. By minimalizing contact with the actual machine, structured programming resulted in the eradication of "distinction between programmers and users" and in the "gradual demotion of programmers." Furthermore, discussing the aspect of software, Chun advocates "making the invisible visible" while suggesting that software is "about translating between computer-readable code and human-readable language."*

* See: Wendy Hui Kyong Chun, "On Software, or the Persistence of Visual Knowledge," *Grey Room* 18 (January 2005): 26–51.

dimensional cubes.[16] In case of ambiguity in the inputs, URBAN5 would request further specification from the user; in case of conflicting constraints, URBAN5 would present gentle alerts on the screen [Figure 2]. However, URBAN5 did not fulfill Negroponte's expectations because it was unable to enact what were thought to be the attributes of a truly adaptable architecture machine.[05]

> Playing is learning, but URBAN5 has not been sufficiently sophisticated actually to frolic; instead it has inexhaustibly printed garbage.... URBAN5 suggests true dialogue, suggests an evolutionary system, suggests an intelligent system – but, in itself, is none of these.[17]

An example of the second, "participatory" category is the "Hessdorfer Experiment," in which three African-American men from "Boston's ghetto area" were asked to interact with a machine – comprising a keyboard and a printer head remotely operated by one of the researchers – that interrogated them about their "urban needs" [Figure 4]. Negroponte celebrated that during the experiment the subjects (who thought they were communicating with an autonomous machine) "did not type uncalled-for remarks" and argued that the subjects engaged the machine in a way that would not have taken place if the interlocutor had been a "white planner or politician."[18]

The experiment is presented in the book[06] as a sign of the computer's potential to foster a more transparent conversation with members of a community. The fact that the book is outdated from a racial, nomenclatural point of view is self-evident, but the kind of neutrality Negroponte was willing to attribute to the machine was worthy of notice. In the Hessdorfer Experiment, the machine is assumed as a neutral device that helped the researchers conceal their personas and, quite specifically, their whiteness.[07]

David Noble, writing on technology discourse in the post-war US, informs us that "technology rarely fulfills the fantasies of their creators. As people are fallible, so too are their machines, however perfect, complete, and automatic the designs."[19] Yet, even failed technologies carry a blueprint of their maker's time, desires, and questions. URBAN5 and the machine in the Hessdorfer Experiment present design as a collaborative, quantifiable and evolving dialogue with always-adapting machines. These two technological experiments also reveal a desire for design to become a more rational and social endeavor devoid of the imposing

42

Daniel Cardoso Llach, "Inertia of an Automated Utopia: Design Commodities and Authorial Agency 40 Years after 'The Architecture Machine.'" *Thresholds*, no. 39 (2011): 42.

This content downloaded from
71.60.57.98 on Mon, 20 Jul 2026 18:25:57 UTC
All use subject to https://about.jstor.org/terms



Figure 4. One of the subjects in the "Hessdorfer Experiment" interacts with the system. From Negroponte, *The Architecture Machine*, 56. Courtesy of MIT Press, Cambridge, Mass.

subjectivity of the human architect. Moreover, it expresses an anxiety about the social role played by the disciplines of architecture and planning, conveyed through a populist desire to give people, through technology, the right to design their own environments.[20]

Rephrasing Authorship Through Social Machines

Aside from a post-structural fantasy of "killing the author," the symbolic demotion, or reconfiguration, of the architect advanced in *The Architecture Machine* should be understood as a techno-rationalist's reaction to the individualistic notion of the architect as a sole creator, inherited by architecture from *L'Ecole de Beaux-Arts*.[20] The notion of a machine that learns by itself was instrumental in affording Negroponte the space to ignore questions about the machine's nature as a designed artifact, and to inspire visions of machine agency in computer-aided design:

> When a designer supplies a machine with step-by-step instructions for solving a specific problem, the resulting solution is unquestionably attributed to the designer's ingenuity and labors.... Whenever a mechanism is equipped with a processor capable "of finding a method of solution," the authorship of the answer probably belongs to the machine.[22]

08 FLJ: Good point.

09 DCL: This historical explanation sounds somewhat reductive and opportunistic.

FLJ: How does this idea around neutral machines as de-authored creators map onto contemporary ways algorithms are framed?

This content downloaded from
71.60.57.98 on Mon, 20 Jul 2026 18:25:57 UTC
All use subject to https://about.jstor.org/terms

And yet, *The Architecture Machine* sought a larger goal beyond disciplinary critique. At the most basic level, it sought to challenge the terms of the contemporary debate about the relationship between humans and machines. In contrast with cultural critics concerned with the de-humanizing aspects of a technological society and with the rise of an unquestionable scientific ideology, *The Architecture Machine* portrayed computers – and CAD – as vehicles for social and environmental activism. Key to the advancement of this vision was a rhetoric that phrased machines as social entities mirroring the attributes of humans while avoiding inconvenient questions about the programmers' culpability as designers. By re-positioning machines as humans, Negroponte aimed at overcoming – rhetorically, at least – the mechanical-organicist polarity that concerned those who saw technology and people in opposing camps.[23] But his move to equate humans and machines reproduced the polarity: using a symmetric language that obscured the machine's complex ties to its makers, Negroponte broke the bridge between the two.[24] Considering, with ~~Louis~~ [10] Mumford, technologies as embodiments of social relations, and *The Architecture Machine's* abundant academic progeny, we can think of Negroponte's socio-technical utopia as inaugurating its own sort of technocratic idealism.[11]

Final Notes: Some Futures Never Get Old

Negroponte's cultural and moral entrepreneurship carried a critique of the "gentleman architect," an idea predicated on the notions of houses and urban spaces as personal commodities and on the inherent value of uniqueness. By proposing to automate the production of uniqueness – via machines that amortize the time and labor involved – Negroponte gave the notion of adaptability between user and context its contemporary wrapping, and anticipated a desire (and a market) for digitally mass-customized goods. In short, he turned the *act* of design into a valuable commodity.

By provocatively construing the computer as a social actor capable of de-stabilizing the relationships between buildings, dwellers and architects, Negroponte aimed at shifting the balance in architecture's political economy, giving it a techno-rationalist – and populist – turn. Interestingly, some aspects of this shift continue to underpin the sensibility and language of today's design utopians, who craft their visions in terms only superficially different from those advanced in *The Architecture Machine*. In the new

Daniel Cardoso Llach, "Inertia of an Automated Utopia: Design Commodities and Authorial Agency 40 Years after 'The Architecture Machine.'" *Thresholds*, no. 39 (2011): 43.

10 DCL: Replace "Louis" with "Lewis."

11 DCL: The argument here is simply that technopopulism is, as far as these experiments are concerned, technocracy in disguise. The notion that technology can replace politics is the ideological core of these experiments, and a theme which can be seen to structure Negroponte's oeuvre since the late 1960s until the more recent One Laptop Per Child project. The common thread is the idea that technologies can bypass conventional structures for education, deliberation, and design (in the post Epstein context we should also add "oversight"), enabling a more direct "democracy" and "participation." What is missing is a critical acknowledgment that any technology able to support meaningful participation must first be extended and infrastructural—and thus the result of a quintessentially "top-down" operation.

This content downloaded from
71.60.57.98 on Mon, 20 Jul 2026 18:25:57 UTC
All use subject to https://about.jstor.org/terms

design utopias the emphasis on the idea of adaptability is no longer deployed through the fantasy of an artificial mind, but through biologically-inspired, computationally obtained variation. The attacks on a designer's subjectivity do not come from human-like machines, such as those advanced in Negroponte's vision, but from the unquestionable, faceless wisdom of an algorithmic conception of the natural. The electronic sketchpads of cathode-ray tube monitors are now replaced by elaborate associative software platforms, plasma screens and layer deposition 3-D printing technologies. Negroponte's mechanistic idioms have given way to a blend of biological and computational metaphors that reflect a new dominant technoscientific paradigm, yet reproduce the rationalist drive (now called "performance-based") that *The Architecture Machine* synthesized.[12]

The imaginaries set in motion by *The Architecture Machine* – its visions of a technologically enhanced, authorless, participatory planning and design – relate closely to the cultural enterprise of construing technology as a democratizing force, and as a means for uplifting supposedly under-privileged, "third-world" communities (the contemporary equivalents to the carefully chosen subjects in the Hessdorfer Experiment).[13] Yet, the desire for a better environment through an expanded, participatory, and technoscientific practice of design remains unsatisfied today, just as when *The Architecture Machine* was published. As SOM had predicted, the CAD industry evolved in an entirely different direction, more in tune with the realities of the market than with the automated utopias of ethical robot designers.[14] Given the distance between its ambitions of social change and the irony of its deployment into popular culture, quantifying the success of *The Architecture Machine*'s technological utopia in terms of social "impact" would be unproductive. But it is worth noting that inverting the equation from the gentleman architect into the gentleman dweller does not fundamentally change the nature of the transaction.

12 DCL: Today, the phrase "data-driven" has largely taken the place of "performance-based." They each invoke a different set of aspirations, but the implication in both cases is that design is a measurable resultant of tractable, scientific methods and operations. The shift is due in part to the increased dominance of machine learning methods over other forms of computing over the last decade.

13 FLJ: Powerful point.

14 DCL: Fueled by machine learning methods, a resurgence of conversational AI research in design promises to rehabilitate some of these aspirations and engender a new batch of architecture machines founded upon the assumption of human-machine symmetry and thus in the concealment of the agencies involved in their production.

This content downloaded from
71.60.57.98 on Mon, 20 Jul 2026 18:25:57 UTC
All use subject to https://about.jstor.org/terms

1 Skidmore, Owings and Merrill (SOM) is an architectural firm founded in Chicago in 1936. The firm has built a vast number of skyscrapers throughout the world, and its name carries—in the architectural and engineering worlds—an instant association with "glass-box" corporate architecture. SOM's software was called "BOP" (Building Optimization Program). See Robert Bruegmann, "The Pencil and the Electronic Sketchpad," in *Architecture and Its Image*, ed. Eve Blau and Ned Kaufman (Cambridge: MIT Press, 1989).

2 Nicholas Negroponte, *The Architecture Machine* (Cambridge: MIT Press, 1970), 121.

3 For an expanded discussion about the politics of technological discourse in post-war US, see Paul Edwards, *The Closed World: Computers and the Politics of Discourse in Cold War America* (Cambridge: MIT Press, 1996) and David Noble, *Forces of Production* (New York: Oxford University Press, 1984). For an early mention to the "personal computer" see "Pocket Computer May Replace Shopping List", *The New York Times*, November 3, 1962.

4 Noble, *Forces of Production*, 11 & 56.

5 Coons oversaw the first interactive computer graphics display, the *Sketchpad*, developed by his student Ivan Sutherland. Ivan Sutherland, "Sketchpad: A man-machine graphical communication system" (Ph.D. Diss., Massachusetts Institute of Technology, 1963).

6 About the testing stage Coons wrote: "These are all essentially mechanical operations, however, and it is quite clear that at least in principle, the computer can be made to deal with all of them." Steven A. Coons, "An Outline of the Requirements for a Computer-Aided Design System," in *Spring Joint Computer Conference, Detroit, Michigan* (New York: ACM, 1963), 301.

7 In a 1966 lecture Coons described computers and numerically controlled manufacturing technology as "perfect slaves" who would take over the most laborious and repetitive aspects of artistic and engineering production—a statement that notoriously resembled Alberti's ideas about the distinction between the architect (who draws) and the "skilled craftsman" (who makes). Daniel Cardoso Llach, "What matters: Certain assumptions in digital design culture," (paper presented in the Harvard University First International Conference on Critical Digital, Cambridge, Massachusetts, April 18-19, 2008).

8 See Negroponte, *The Architecture Machine*, 7 & 17.

9 MIT's technological impetus inspired in Negroponte visions of anthropomorphic machines with human-like organs that "receive direct sensory information from the real world" and that "must see, hear and read, and must take walks in the garden." Negroponte, *The Architecture Machine*, 27.

10 Coons was wary about the risk of narrowing the space of design alternatives in a computer-aided design system by overloading the software—and the user—with a large set of very specialized procedures. See Coons, "An Outline of the Requirements for a Computer-Aided Design System," 301.

11 Negroponte, *The Architecture Machine*, 27.

12 The notion of *l'homme type* (French for "man-type", or "typical man") was a collection of anthropometric assumptions about architecture's "end-user" advanced by advocates of modern architecture, especially in the first half of the twentieth century, to facilitate (and legitimize) standardized residential housing design. Abiding by these assumptions could only yield, in Negroponte's view, impersonal and banal architecture.

13 Expert systems are a type of computer program that performs a human expert's task. Such systems are founded on the notion of "knowledge capture": the idea that a "knowledge engineer" can study and encode in a computer program the decision-making process and expertise of a human subject. For a sharp perspective on the culture of "knowledge engineering" and artificial intelligence see Diana Forsythe and David J. Hess, *Studying Those Who Study Us: An Anthropologist in the World of Artificial Intelligence* (Stanford: Stanford University Press, 2001).

14 Negroponte, *The Architecture Machine*, 55.

15 Eliza is a computer program that uses a simple "pattern-matching" algorithm to assign pre-canned phrases as responses to sentences entered as text by the user. The interaction with the program gave the user the impression—generally for a brief time—of having a conversation with a human (the program simulates being a Rogerian psychotherapist). Eliza was written in 1966 by Joseph Weizenbaum at MIT, and is considered an early example of automated language processing. There are many available online implementations of the program. One can be found at "Eliza Test", accessed January 10, 2011, http://chayden.net/eliza/Eliza.html.

16 Today we would call these cubes "voxels," a word that wasn't available at the time *The Architecture Machine* was written.

17 Negroponte, *The Architecture Machine*, 93.

18 Negroponte, *The Architecture Machine*, 50.

19 Noble, *Forces of Production*, 325.

20 Cf. Nicholas Negroponte, *Reflections on Computer Aids to Design and Architecture* (New York: Petrocelli/Charter, 1975). While a few years later, in a less enthusiastic tone, Negroponte would admit the naïveté of these experiments, the notion of empowerment through technology remained a constant of his work.

21 In fact, Bruegmann places Negroponte in a tradition of rationalism that goes back to J.N.L. Durand. Bruegmann, "The Pencil and the Electronic Sketchpad," 140.

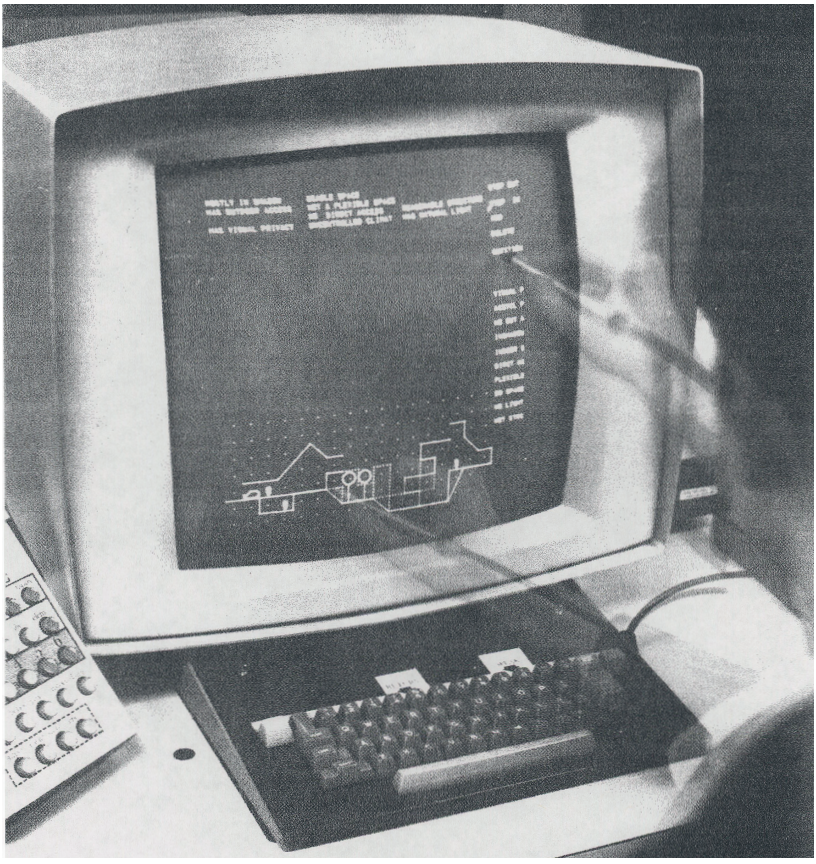
22 Negroponte, *The Architecture Machine*, 11.

23 "What needs to be articulated, regardless of the format of the man-machine relationship, is the goal of humanism through machines.... The concern is to avoid dehumanizing a process whose aim is definitely humanization." See Negroponte, *The Architecture Machine*, 7. Matthew Wisniosky's history of engineers' temporary reconciliation with social theory during the fifties and sixties captures and elaborates on the contrast between these two postures about technology and society by distinguishing between two philosophies, a *theory of technological change*, which attributes agency to technology as a semi-autonomous force, and a *theory of technological politics*, which considers technology as an "embodiment of social relations." Matthew Wisniosky "Engineers and the Intellectual Crisis of Technology" (Ph.D. diss., Princeton University, 2005), 6.

24 Lucy Suchman coined the term "dissymmetry" to overcome idioms of absolute symmetry or absolute asymmetry between humans and machines. See Lucy Suchman, *Human-Machine Reconfigurations* (Cambridge: Cambridge University Press, 2007).

Daniel Cardoso Llach, "Inertia of an Automated Utopia: Design Commodities and Authorial Agency 40 Years after 'The Architecture Machine.'" *Thresholds*, no. 39 (2011): 44.

This content downloaded from
71.60.57.98 on Mon, 20 Jul 2026 18:25:57 UTC
All use subject to https://about.jstor.org/terms



Daniel Cardoso Llach, "Inertia of an Automated Utopia: Design Commodities and Authorial Agency 40 Years after 'The Architecture Machine.'" *Thresholds*, no. 39 (2011): 41.

15 FLJ: This Edwards publication is the go-to source for a critical analysis of the origin story of the electronic computer and its relationship to Cold War doctrines of command and control.

This often leads to a contemporary discussion that frames computational and algorithmic surveillance and security techniques within a Cold War military logic. Most recently, I've found the work of Andrea Miller exceptionally productive in expanding the story of computational control to include a longer historical arc of settler colonial forms of domination. In an online publication, she writes: "Thinking with a genealogy of the algorithm as protocol ... I want to offer a reflection on what it might mean to dwell in the gap between digital and non-digital modes of colonialist computation, sorting, and classification."*

* Andrea Miller, "Protocological Violence and the Colonial Database," *Antipode* (May 2017): 1.

16 DCL: Replace with Steven Anson Coons, "Computer, Art & Architecture" *Art Education* 19, no. 5 (May 1, 1966): 9–11, https://doi.org/10.2307/3190804.

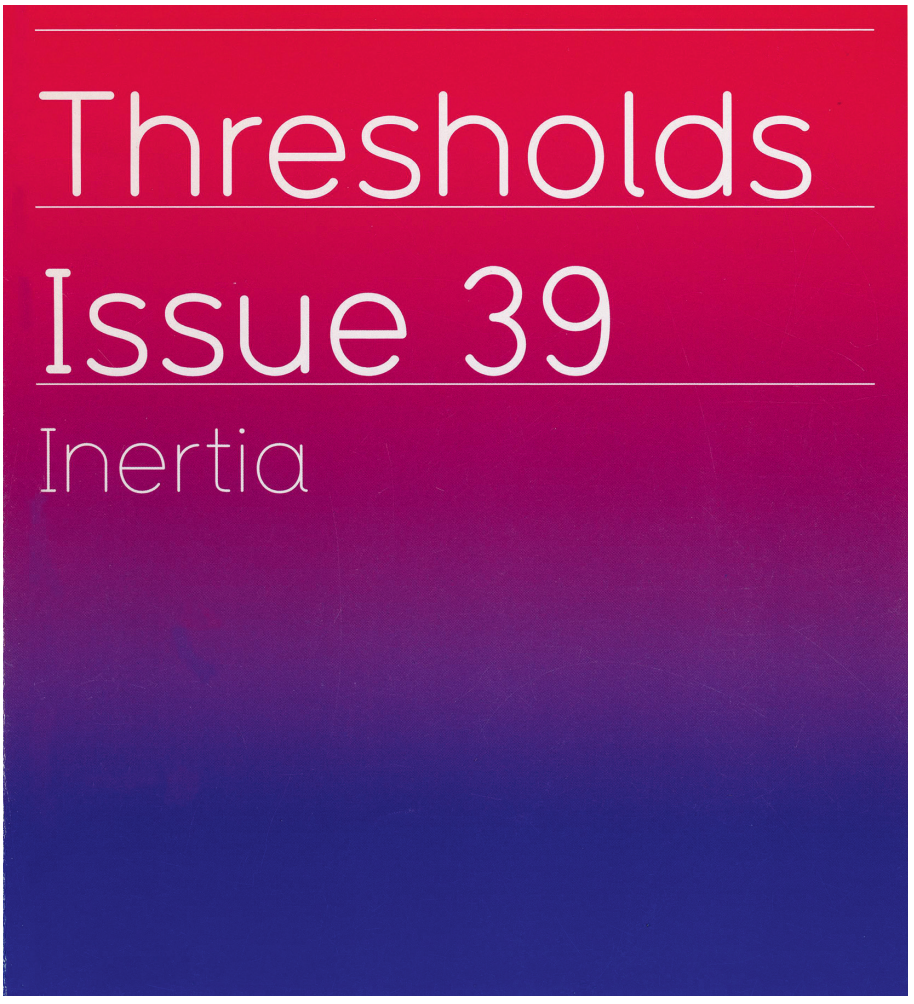
FLJ: Expanded discussion of this article's arguments.

17 FLJ: I'm thinking about Timothy Mitchell's work here in the *Rule of Experts* where Mitchell examines technology, governance, and violence in modern Egypt.*

* Timothy Mitchell, *Rule of Experts: Egypt, Techno-Politics, Modernity* (Berkeley, CA: University of California Press, 2012.)

This content downloaded from
71.60.57.98 on Mon, 20 Jul 2026 18:25:57 UTC
All use subject to https://about.jstor.org/terms

Cover design by Ian Dapot. Featured quote by Adam Fulton Johnson, "Introduction," *Thresholds*, no. 39 (2011): 5.



"We talk about inertia as if we understand it—but what are we observing when we see inertia? Unhindered motion? Freedom? Absolute stability? A fundamental 'desire' in an object?" —Adam Fulton Johnson, 2011, *Thresholds* 39: Inertia

When does friction productively disrupt momentum? Write your thoughts down here, pass your journal around, and contemplate with your friends.

This content downloaded from
71.60.57.98 on Mon, 20 Jul 2026 18:25:57 UTC
All use subject to https://about.jstor.org/terms