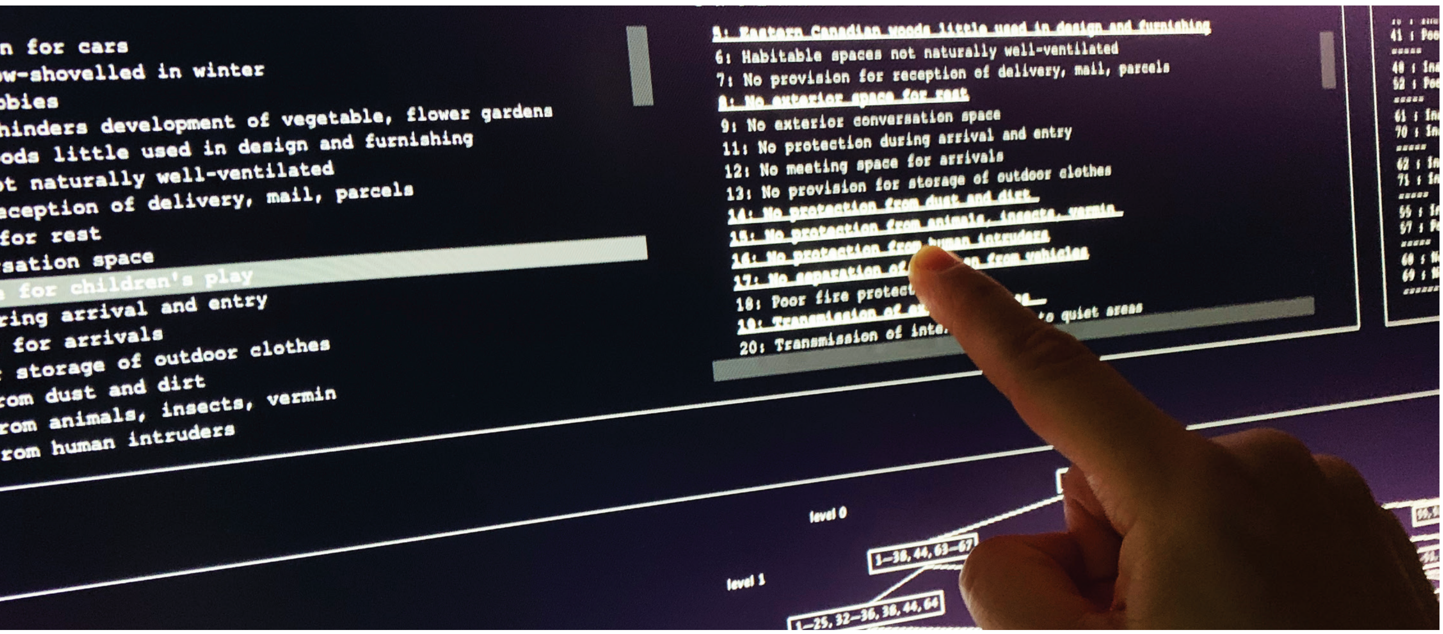
Setting Historic Computer Systems in Motion

Theodora Vardouli
McGill University

Mobilities and Reenactments of the Hierarchical Decomposition System 2 (HIDECS 2, 1962)



1

1 Interactive reconstruction of the historic HIDECS program using a list of 72 “misfits” (situations to avoid) in designing a Canadian house, listed in a 1966 paper by Allen Bernholtz and Edward Bierstone

ABSTRACT

This paper engages digital humanities methods, namely data visualization and software reconstruction, to shed new light on a landmark computer system for design decision-making presented in 1962 by Christopher Alexander and Marvin Manheim. The paper puts the historic computer system in motion in two interconnected ways. First, it moves the system outside the context of its initial development and tracks its trajectories and adaptations within a larger network of researchers in North American postwar research institutions. It does this through a newly built database on activity in “rational design methods” from 1966–1971, which the paper also details. Second, it reenacts a hybrid of the system’s multiple versions unearthed within the Design Methods Network in a contemporary technical context and makes it available for manipulation by contemporary audiences. The paper begins with a brief overview of the HIDECS 2 system and its significance. Then, it discusses the development of the Design Methods Network database and its use to identify implementations, applications, and versions of the HIDECS 2 system developed outside the context of its origin. Finally, the paper presents the interactive reconstruction of a hybrid version of HIDECS that synthesizes features discovered through querying the database, and its presentation in a public exhibition on histories and contemporary practices of computer-aided design. Ultimately, the paper contributes new insights on the history of this impactful computer system and offers productive historiographic methods for the study of other computer systems and programs from the early years of design and architectural computing.

INTRODUCTION

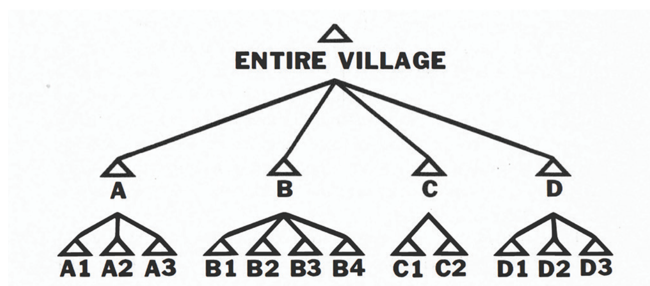
This paper focuses on a historical computer system for design decision-making developed by architect and mathematician Christopher Alexander and civil engineer Marvin Manheim in 1962 and consecutively reworked by various researchers in North American research institutions operating under the broad umbrella of “rational design methods” (Moore 1966). The development and uses of the first version of the system, known as Hierarchical Decomposition System 2 or HIDECS 2, have been well documented (Upitis 2013; Steenson 2017; Vardouli 2020; Cristobal Olave 2021). The current paper adds new insights to the study of this important artifact in the history of architectural computing by deploying methods from the interdisciplinary field of digital humanities, namely data visualization and software reconstruction (Cardoso Llach and Donaldson 2019). These insights include a renewed understanding of the program as a mobile and malleable technical artifact whose trajectory extended beyond its initial setting of conception and development. New insights also emerge from the reconstruction of a hybrid form of the program—one that crossbreeds different contexts of implementation and application as well as distinct technical conditions. The reconstruction takes the form of an interactive artifact available for manipulation by contemporary audiences.

Using data visualization, the paper situates HIDECS 2 within an intellectual ecology consisting of multiple researchers and institutions who adopted and adapted it. An interactive software reconstruction of HIDECS 2 using the programming language Java reenacts a hybrid version of the system that synthesizes versions and adaptations as it moved across different institutions and research settings, while also reinterpreting it in a new programming language and computer hardware. Presented in the public exhibition *Vers un imaginaire numérique* (Centre de Design de l’UQAM, Montreal, 2021) and displayed together with historical documents around the program’s inception and development, the interactive reconstruction of HIDECS 2 activates critical debates around software as a product of specific historical settings. On one level, the interactive nature of the reconstruction serves a instructive function, allowing audiences to acquire an intuitive understanding of its algorithmic logic and the design process it modeled. The reconstruction also operates on a critical register: it renders visible tensions and collusions between messy, at times even arbitrary, data and the ostensibly objective algorithms that structured them for design and dramatizes the notion of “failure” as a constitutive category of the program’s computational logic. That is in juxtaposition with the positive categories of “goals” and “optima” that drive much of contemporary data-driven design.

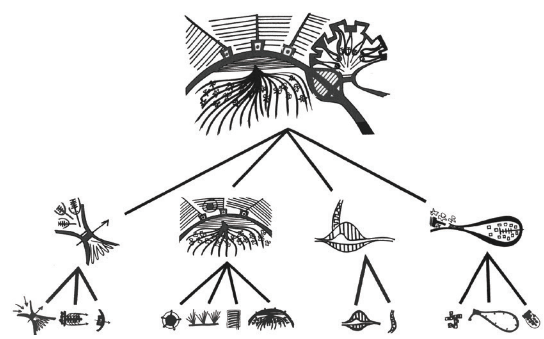
BACKGROUND: A BRIEF HISTORY OF HIDECS 2

Although not developed specifically for architectural or urban design, HIDECS 2 was one of the first computer systems applied to architectural decision-making. The system became known to architectural audiences through Christopher Alexander’s influential book *Notes on the Synthesis of Form*, published in 1964 and based on Alexander’s doctoral dissertation at Harvard University. In the book Alexander famously presented an implementation of the computer system for the determination of so-called “components” of an agricultural village of six hundred people in Bavra, India. Components were groups of design considerations (or “requirements”) that the designer addressed through simple schematic drawings (or “diagrams”). The computer system was developed to identify these components and indicate the structure by which the “diagrams” should be combined to produce a complete design that responded to the requirements.

HIDECS 2 achieved this by analyzing relationships between design requirements in terms of the data they shared and the ways they influenced each other. This analysis allowed grouping together requirements that were strongly connected and separating ones that were relatively independent to each other. Following this process, HIDECS 2 broke down (“decomposed”) an unstructured set of requirements into



2



3

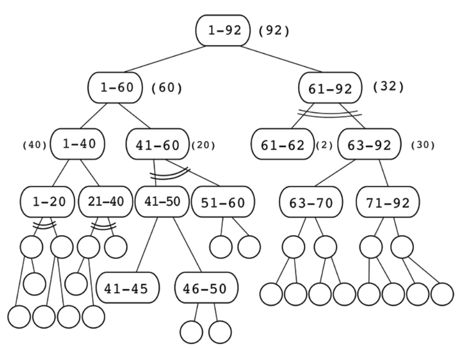
2 Tree decomposition of 141 design requirements for an agricultural village in India into 12 subgroups using HIDECS 2

3 Composition of diagrams, defined by the designer, addressing the subgroups defined by HIDECS 2 to produce the design of the village

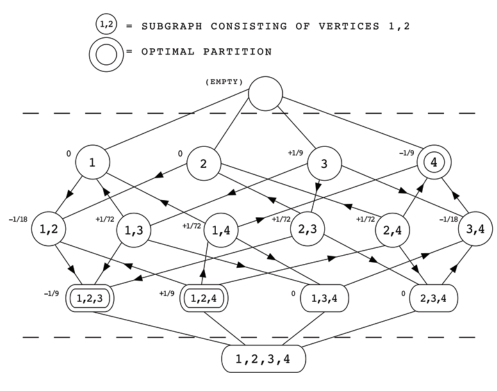
a hierarchical tree: a graph in which every two vertices are connected by exactly one path. The tree indicated the order by which a designer ought to combine the component diagrams they developed for each grouping of requirements.

For instance, in the "Worked Example" of the village in India presented in *Notes*, HIDECS 2 broke down 141 requirements including wildly diverse statements such as "67. Drinking water to be good, sweet" and "117. Spread of information about birth control, disease, etc." into 12 independent components (Alexander 1964, 140-41) that Alexander recomposed to produce a design for the village.

Although in *Notes* HIDECS 2 was presented as the implementation of a theory of design presented in the book, the computer system preceded and, as scholars of Alexander have convincingly argued, influenced the theory (Upitis 2013). HIDECS 2 was developed during a consultancy at the Civil Engineering Systems Laboratory from 1960–1962 in collaboration with civil engineer Marvin L. Manheim. It was implemented in the IBM 709 of the MIT Computation Center, under the control of the Fortran Monitor System in use at the Center during the second half of 1961. Since the beginning of his doctoral studies, Alexander had been concerned with the organization of empirical data related to design and how this organization could indicate a well-ordered sequence of decision-making steps for the designer (Vardouli 2020). Alexander called this decision-making sequence a design "program" (1964, 69). HIDECS 2 computed the design "program" through first, making trial cuts of an initial unordered graph (the set of requirements) into subgraphs, then calculating an "INFO" parameter based on the number of links that the partition cut and the number of vertices at each side of the partition, and finally performing a heuristic optimization method to minimize the INFO parameter. The smaller INFO, the more independent the groups partitioned, achieving the principle of breaking down the design problem in independent components.



4



5

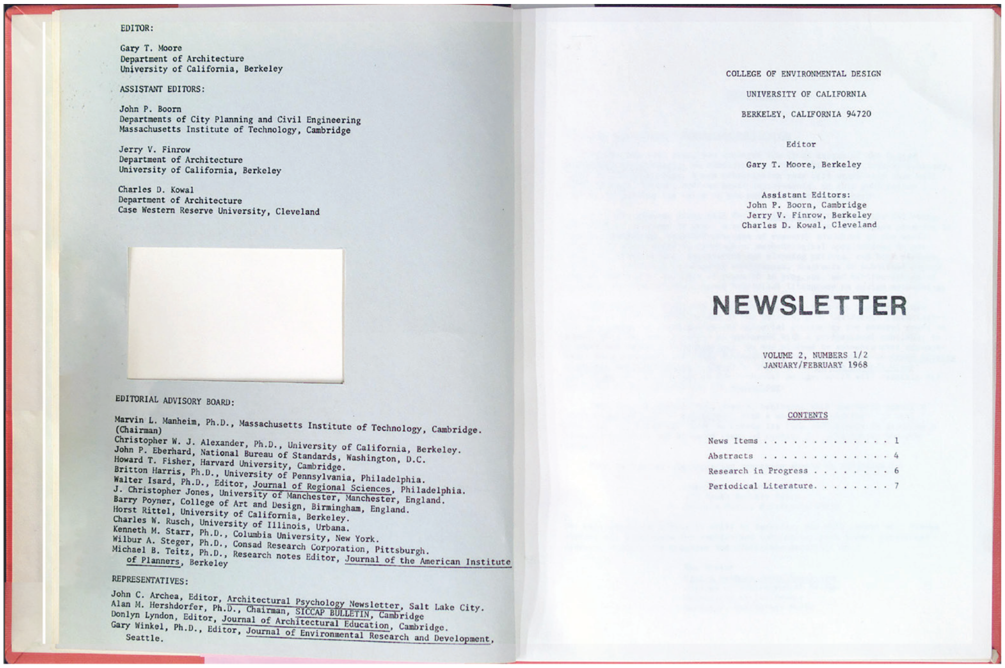
4 Tree drawn to represent the output of HIDECS 2 (printed as verbal statements) after analyzing relationships between 92 requirements

5 HIDECS 3 replaced the hierarchical tree with a semi-lattice

Aside from the village in India, HIDECS 2 was also used to locate a section of the I-91 Interstate Highway System in Western Massachusetts (Alexander and Manheim 1962b). This project brought forward tensions arising between the analysis to identify design subproblems for which to develop diagrams and the synthesis of the diagrams, which Alexander and Manheim envisioned as transcending the mere combination of discrete entities (Alexander and Manheim 1962b, 91–92). The history of HIDECS 2 continues with the development of a new version, HIDECS 3, by Alexander in June 1963. HIDECS 3 accounted for anomalies stemming from hierarchical decomposition and instead approached decomposition by searching for maximal simplices. Simplices were shape-like topological constructs like triangles and tetrahedra defined by a set of requirements (represented as vertices of a graph). HIDECS 3 searched for triangles or tetrahedra whose number of vertices was not smaller than other triangles or tetrahedra in the graph (Alexander 1963; Vardouli 2017). The crucial attribute of this method was that a vertex (representing a design requirement) could be part of multiple subsystems. This method produced as its output not a tree, but instead a "lattice." Alexander discussed the theoretical implication of this move from trees to lattices in the awarded 1965 article "A City is Not a Tree," which ushered a new image of the city as a complex system not as a single hierarchy but as multiple overlapping ones.

METHODS: DIGITAL HISTORY TACTICS FOR THE STUDY OF COMPUTER-AIDED DESIGN PROGRAMS

Historians and technology studies scholars have long urged to move beyond the "inventor" narratives and to study technological artifacts, including computer programs, in their multiple and diverse contexts of use (for example, Pinch and Bijker 1980; Oudshoorn and Pinch 2003). This becomes challenging with influential artifacts in the history of design computing such as HIDECS 2, which resulted in publications and demonstration


EDITOR:

Gary T. Moore
Department of Architecture
University of California, Berkeley

ASSISTANT EDITORS:

John P. Boorn
Departments of City Planning and Civil Engineering
Massachusetts Institute of Technology, Cambridge

Jerry V. Finrow
Department of Architecture
University of California, Berkeley

Charles D. Kowal
Department of Architecture
Case Western Reserve University, Cleveland

EDITORIAL ADVISORY BOARD:

Marvin L. Manheim, Ph.D., Massachusetts Institute of Technology, Cambridge. (Chairman)
Christopher W. J. Alexander, Ph.D., University of California, Berkeley.
John P. Eberhard, National Bureau of Standards, Washington, D.C.
Howard T. Fisher, Harvard University, Cambridge.
Britton Harris, Ph.D., University of Pennsylvania, Philadelphia.
Walter Isard, Ph.D., Editor, Journal of Regional Sciences, Philadelphia.
J. Christopher Jones, University of Manchester, Manchester, England.
Barry Poyner, College of Art and Design, Birmingham, England.
Horst Rittel, University of California, Berkeley.
Charles W. Rusch, University of Illinois, Urbana.
Kenneth M. Starr, Ph.D., Columbia University, New York.
Wilbur A. Steger, Ph.D., Consad Research Corporation, Pittsburgh.
Michael B. Teitz, Ph.D., Research notes Editor, Journal of the American Institute of Planners, Berkeley

REPRESENTATIVES:

John C. Archea, Editor, Architectural Psychology Newsletter, Salt Lake City.
Alan M. Hershdorfer, Ph.D., Chairman, SICCAP BULLETIN, Cambridge
Donlyn Lyndon, Editor, Journal of Architectural Education, Cambridge.
Gary Winkel, Ph.D., Editor, Journal of Environmental Research and Development, Seattle.

COLLEGE OF ENVIRONMENTAL DESIGN
UNIVERSITY OF CALIFORNIA
BERKELEY, CALIFORNIA 94720

Editor
Gary T. Moore, Berkeley

Assistant Editors:
John P. Boorn, Cambridge
Jerry V. Finrow, Berkeley
Charles D. Kowal, Cleveland

NEWSLETTER

VOLUME 2, NUMBERS 1/2
JANUARY/FEBRUARY 1968

CONTENTS

News Items	1
Abstracts	4
Research in Progress	6
Periodical Literature	7

6

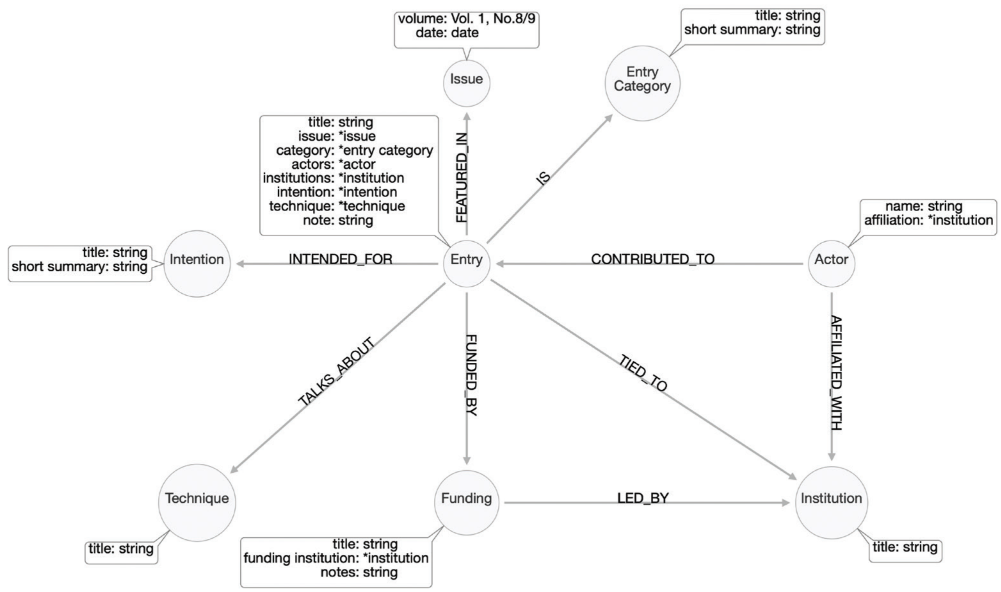
6 Front matter of the *Design Methods Group Newsletter*, Volume 2, Number 1-2, 1968

projects but were not taken up by professional architects and designers, thus mostly remaining academic experiments. How to mobilize such historic computer programs beyond the confines of their invention? This paper presents two interconnected methods for historically mobilizing HIDECS 2 drawing from the interdisciplinary field of digital humanities, specifically digital history and software reconstructions. First, it presents a method for tracking mobilities of HIDECS 2 within a cross-institutional and cross-disciplinary network of North American academics who developed little known subroutines and implementations of the computer system. Then, it discusses an interactive software reconstruction developed to reenact HIDECS 2 in an alternate technical and cultural setting opening it up for new uses and critical readings. "Mobilizing" here refers both to a historical revealing of HIDECS 2 as mobile but also to rendering of the system capable for action in the present through a reenactment of a hybrid version that recalls its multiple versions and implementations.

Mobilities Within the Design Methods Network

Alexander's work was received with avid interest from a short-lived but influential movement that proliferated across sites of architectural education and research in the second half of the 1960s under the umbrella of "rational design methods" (Montgomery 1970). Design methods filtered interwar mandates of "rational architecture"—buildings designed under the tenets of material economy or functional purpose—through the lens of a goal-oriented, rule-based rationality characteristic of postwar intellectual life (Erickson et al. 2015; Vardouli 2020). Design methods research took place mainly in the setting of British and North American universities. Working with mathematicians and engineers, architects developed methods that viewed architectural design as a stepwise process, amenable to mathematical analysis, and directed toward articulable goals. This produced a new disciplinary focus for architecture, from the physical form of the final artifact (object, building, city) to the steps and decisions leading to it, ultimately paving the path for algorithmic and computational approaches to design. Ardent engagement with mathematical analysis and rigorous theory produced not only new styles of studying and talking about design and architecture, but also a proliferation of societies and associations during the 1960s, founded with the mission to advance communication among design methods activity performed in disparate settings.

Acknowledging design methods as an intellectual ecology whose study requires attention to mobilities and exchanges across institutions and geographies as opposed to their study in isolation, the CoDEx research team at McGill University led by the author, produced an interactive digital database mapping the design methods network in North America. We built the database using the entries of a monthly periodical headquartered at the University of California Berkeley and founded in 1966 with the aim to establish a network of researchers working on design methods in North America and around the world. The *Design Methods Group Newsletter*, as the periodical was called, is a rich resource for identifying cross-disciplinary and cross-institutional transactions, tentative computer experiments, technical languages, controversies, and trends, giving a lively view of research-in-the-making throughout the six published volumes of the *Newsletter* from 1966-1971.



7

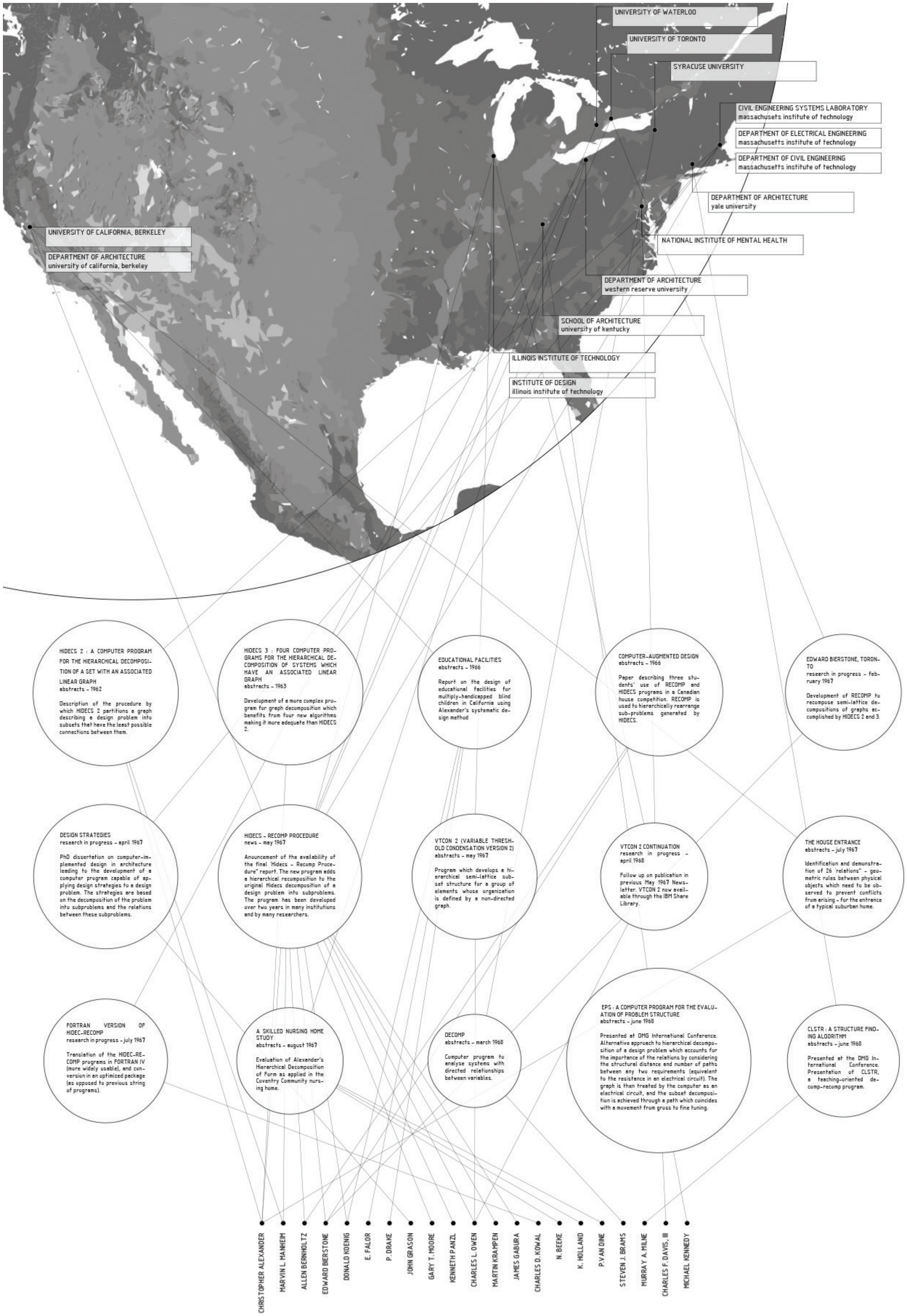
7 Database architecture of the Design Methods Network in Neo4j

8 Mapping mobilities of HIDECS 2 and its successors within the Design Methods Network

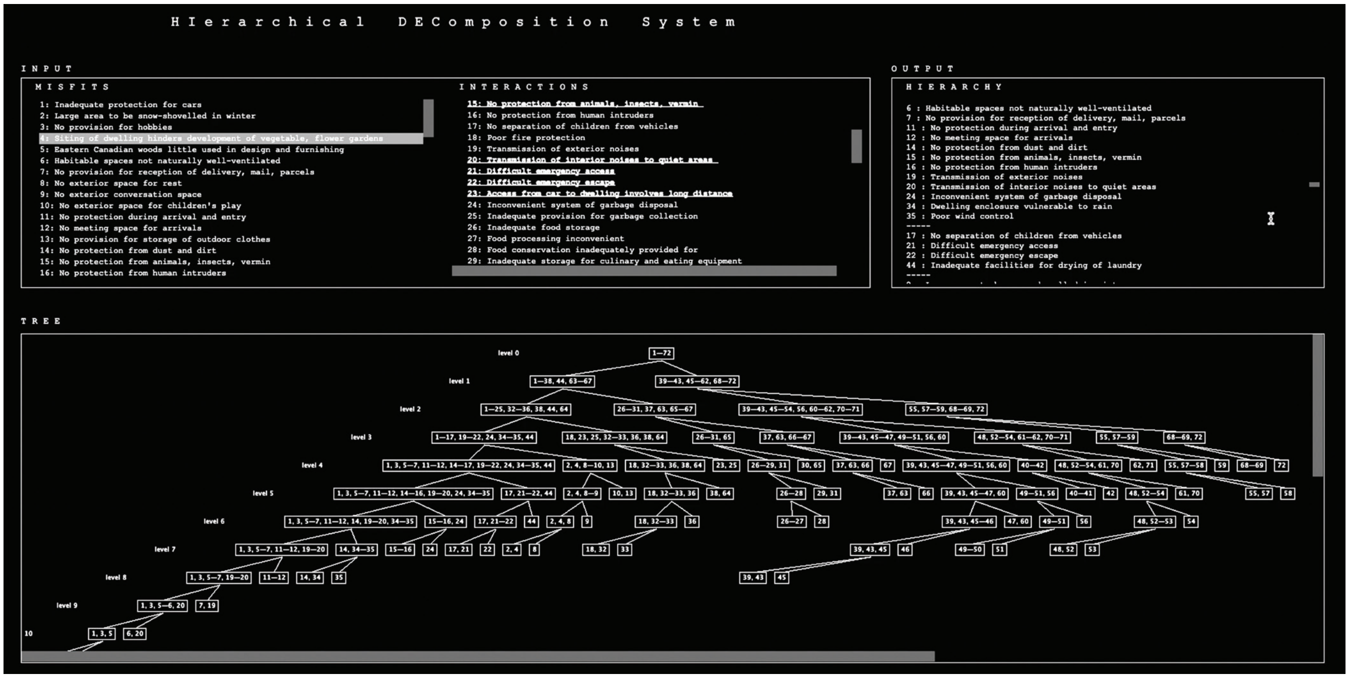
We used the *DMG Newsletter* as the foundation of the Design Methods Network database, which we began building using the HEURIST data management system (https://heurist-network.org). From each *Newsletter* entry we extracted the following data: 1. "title" (entry title), 2. "entry category" (the newsletter accepted news, abstracts, work in progress outlines, reviews and criticisms, bibliographies of design methods, and computer program abstracts), 3. "actor" (entry author), 4. "institution" (location that each entry refers to), 5. "funding" (funding organization associated to the entry). We also added interpretative information such as "technique" and "intention," following the study of each entry's content. We use "intention" to account for the conceptual drivers of each entry and statements for what the author was trying to accomplish. "Technique," in turn, reflects the algorithmic, mathematical, and calculative tools enlisted to achieve these goals and how concepts were operationalized. The assignments of "intention" and "technique" were performed based on the researcher's interpretation of each entry's content and awareness of the recurrence of certain keywords and themes. We are currently exploring ways to combine the *close reading* of the entries that we performed with *distant reading* (Moretti 2013) techniques that would allow us to trace the prominence of certain themes or keywords in a more systematic way. To improve usability and functionality of the database, we transferred the entries to the graph database system Neo4j (https://neo4j.com). The use of directed graphs allowed us to also characterize relationships and dependencies between the different data types and led us to refine the database architecture as indicated in Figure 7.

The motivation behind building the database was to generate visualizations that revealed connections between actors, sites, or techniques. We have explored both static images made by hand that condense findings from the database within graphic conventions of maps, timelines, and diagrams of relationships, as well as dynamic visualizations such as digital stories using the markdown-based app Obsidian (https://obsidian.md). The Design Methods Network database can be mined for historical projects concerned with mobilities of technical practices within academic research networks. In the case of HIDECS 2, the database revealed 15 related entries by 21 authors who engaged the development, implementation, or application of the system in 14 distinct institutions.

Figure 8 shows a static visualization, made by hand and for printing, of manifestations of the HIDECS 2 system and its successors within the Design Methods Network. The figure focuses on one specific technique ("hierarchical decomposition") and visualizes its associated entries, actors, institutions, and funding organizations. As elaborated on in the "Results and Discussion" section of the paper, the visualization establishes HIDECS as a mobile entity that can be followed across distinct geographic, institutional, and material contexts. A closer look at the associated data shows a multiple and diverse engagement with the system: some projects entailed the development of specific subroutines for decomposition and recomposition such as DECOMP, RECOMP, and VTCON 2 as well as adjusting these routines to languages and computer systems other than FORTRAN and the IBM 709. Other entries focused on the practical advantages and limitations of the program in specific application contexts such as the Coventry Community Nursing home. Finally, other entries discussed versions of the system, such as CLSTR, adapted for use in educational settings. Contexts identified in this study, such as the University of Toronto in which a version of HIDECS 3


UNIVERSITY OF WATERLOO
UNIVERSITY OF TORONTO
SYRACUSE UNIVERSITY
CIVIL ENGINEERING SYSTEMS LABORATORY
massachusets institute of technology
DEPARTMENT OF ELECTRICAL ENGINEERING
massachusetts institute of technology
DEPARTMENT OF CIVIL ENGINEERING
massachusetts institute of technology
DEPARTMENT OF ARCHITECTURE
yale university
UNIVERSITY OF CALIFORNIA, BERKELEY
DEPARTMENT OF ARCHITECTURE
university of california, berkeley
NATIONAL INSTITUTE OF MENTAL HEALTH
DEPARTMENT OF ARCHITECTURE
western reserve university
SCHOOL OF ARCHITECTURE
university of kentucky
ILLINOIS INSTITUTE OF TECHNOLOGY
INSTITUTE OF DESIGN
illinois institute of technology
HIDECS 2 : A COMPUTER PROGRAM FOR THE HIERARCHICAL DECOMPOSITION OF A SET WITH AN ASSOCIATED LINEAR GRAPH
abstracts - 1962
Description of the procedure by which HIDECS 2 partitions a graph describing a design problem into subsets that have the least possible connections between them.
HIDECS 3 : FOUR COMPUTER PROGRAMS FOR THE HIERARCHICAL DECOMPOSITION OF SYSTEMS WHICH HAVE AN ASSOCIATED LINEAR GRAPH
abstracts - 1963
Development of a more complex program for graph decomposition which benefits from four new algorithms making it more adequate than HIDECS 2.
EDUCATIONAL FACILITIES
abstracts - 1966
Report on the design of educational facilities for multiply-handicapped blind children in California using Alexander's systematic design method
COMPUTER-AUGMENTED DESIGN
abstracts - 1966
Paper describing three students' use of RECOMP and HIDECS programs in a Canadian house competition. RECOMP is used to hierarchically rearrange sub-problems generated by HIDECS.
EDWARD BIERSTONE, TORONTO
research in progress - february 1967
Development of RECOMP to recompose semi-lattice decompositions of graphs accomplished by HIDECS 2 and 3.
DESIGN STRATEGIES
research in progress - april 1967
PhD dissertation on computer-implemented design in architecture leading to the development of a computer program capable of applying design strategies to a design problem. The strategies are based on the decomposition of the problem into subproblems and the relations between these subproblems.
HIDECS - RECOMP PROCEDURE
news - may 1967
Anouncement of the availability of the final "Hidecs - Recomp Procedure" report. The new program adds a hierarchical recomposition to the original Hidecs decomposition of a design problem into subproblems. The program has been developed over two years in many institutions and by many researchers.
VTCON 2 (VARIABLE THRESHOLD CONDENSATION VERSION 2)
abstracts - may 1967
Program which develops a hierarchical semi-lattice subset structure for a group of elements whose organization is defined by a non-directed graph.
VTCON 2 CONTINUATION
research in progress - april 1968
Follow up on publication in previous May 1967 Newsletter. VTCON 2 now available through the IBM Share Library.
THE HOUSE ENTRANCE
abstracts - july 1967
Identification and demonstration of 26 "relations" - geometric rules between physical objects which need to be observed to prevent conflicts from arising - for the entrance of a typical suburban home.
FORTRAN VERSION OF HIDEC-RECOMP
research in progress - july 1967
Translation of the HIDEC-RECOMP programs in FORTRAN IV (more widely usable), and conversion in an optimized package (as opposed to previous string of programs).
A SKILLED NURSING HOME STUDY
abstracts - august 1967
Evaluation of Alexander's Hierarchical Decomposition of Form as applied in the Coventry Community nursing home.
DECOMP
abstracts - march 1968
Computer program to analyse systems with directed relationships between variables.
EPS : A COMPUTER PROGRAM FOR THE EVALUATION OF PROBLEM STRUCTURE
abstracts - june 1968
Presented at DMG International Conference. Alternative approach to hierarchical decomposition of a design problem which accounts for the importance of the relations by considering the structural distance and number of paths between any two requirements (equivalent to the resistance in an electrical circuit). The graph is then treated by the computer as an electrical circuit, and the subset decomposition is achieved through a path which coincides with a movement from gross to fine tuning.
CLSTR : A STRUCTURE FINDING ALGORITHM
abstracts - june 1968
Presented at the DMG International Conference. Presentation of CLSTR, a teaching-oriented decomp-recomp program.
CHRISTOPHER ALEXANDER
MARVIN L. MANHEIM
ALLEN BERNHOLTZ
EDWARD BIERSTONE
DONALD KOENIG
E. FALOR
P. DRAKE
JOHN GRASON
GARY T. MOORE
KENNETH PANZL
CHARLES L. OWEN
MARTIN KRAMPEN
JAMES GABURA
CHARLES D. KOWAL
N. BEEKE
K. HOLLAND
P. VAN DINE
STEVEN J. BRAMS
MURRAY A. MILNE
CHARLES F. DAVIS, III
MICHAEL KENNEDY

8



9 A screenshot from the HIDECS interactive reconstruction

was adapted for the development of Canadian housing under a National Research Council Grant, became the launching pad for thinking about the interactive reconstruction of the computer system.

Reconstructing the HIDECS Hybrid

Design and computing scholar Daniel Cardoso Llach has put forward software reconstruction as a key tactic in a media archaeological approach to computer-aided design. He and Scott Donaldson have framed software reconstruction "as a method to shed new light into the material, gestural, and sensual dimensions of computer-aided design technologies" (Cardoso Llach and Donaldson 2019). Some projects developed under the related project "Experimental Archaeology of CAD" have paid attention to the hardware of early computer aided design systems to highlight the embodied, as opposed to just cognitive, interaction between users and the systems, as well as to their spatial, aesthetic, material and design sensibilities. Other projects, such as the Coons Patch reconstruction, have focused on making visible and palpable abstract mathematical concepts (Cardoso Llach and Donaldson 2019). Our reconstructive work of HIDECS 2 aligns more with the latter approach, which we characterize as a software *reenactment* to convey liberties we take in materializing an abstract algorithm into an interactive artifact. HIDECS 2 was not an interactive system: as Upitis (2013) has detailed, the system's input was a deck of punch cards and its output was a sheet of printed verbal statements. Users of the system then had to translate these statements manually into the iconic tree or semi-lattice. Although these material conditions were key for the development and function of the system, we decided to temporarily set them aside and focus on HIDECS 2 as the computer implementation of a theory and method for designing.

A similar approach was taken by Pablo Miranda Carranza, who in 2020 published a Python implementation of HIDECS 2. His intention was, as he wrote, to "deemphasise the material and technical conditions behind the code and foreground the abstractions and concepts implemented" (Miranda Carranza 2020). Exploring different algorithms for finding a minimum cut in a graph, as HIDECS 2 was programmed to do, Miranda Carranza concluded that Alexander and Manheim did not look for an efficient solution to the graph partition problem but sought to demonstrate the intractability of a design problem without the heuristic algorithms that they put forward. Our reconstruction of HIDECS carves a middle ground between seeing the system solely through the algorithmic abstractions it performed or exclusively through the material settings and conditions in which it was embedded. We aimed to produce a reconstruction that was, on the one hand, *instructive*, communicating the fundamental tenets of HIDECS as design methodological and theoretical proposition, and on the other hand, *critical*, rendering visible biases and assumptions embedded into the system.

To make the reconstruction, we used the 1962 report of HIDECS 2 by Alexander and Manheim in conjunction with a 1966 paper by Allen Bernholtz and Edward Bierstone from the University of Toronto that applied a version of HIDECS in problems of Canadian housing. The paper was included in the Proceedings of the 1966 Design and Planning Conference at the University of Waterloo, which was the founding event of the Design Methods Group. As such, our reconstruction of the HIDECS computer system holds together multiple sites and events of historical significance. The blending of these disparate but related contexts renders the reenactment a *hybrid*. This is both in the more direct sense of cross-polinating two distinct things for the production of a third, but also in a broader and perhaps theoretically more evocative sense of creating an assemblage that is capable of holding together, but also producing, difference in its unpredictable uses by audiences today.

We coded the graph partition algorithm in Java using hill climbing and implemented interactive features using touch events. The "user" of our reconstruction is presented with a touch screen organized in an INPUT and OUTPUT frame, which include a "Misfits" and "Interactions" frame and a "Hierarchy" and "Tree" frame respectively. As I discuss in the following section, the notion of "misfits" was central in Alexander's theory and in the logic of HIDECS 2. Unlike traditional architectural briefs that typically list desired situations, goals, or qualities to achieve, Alexander proposed, and Bernholtz and Bierstone took on, the idea that the design brief ought to list all situations that an architect ought to *avoid*.

The "Misfits" frame is prepopulated with 72 requirements for Canadian housing listed in the 1966 paper by Bernholtz and Bierstone, such as "16. No protection from human intruders," "22. Difficult emergency escape," and "54. Poor love-making facilities for parents." The "Interactions" frame default state includes interactions between these requirements as they were identified in the same paper. Based on these default conditions, the reconstruction shows the calculated "Hierarchy" decomposing these requirements into subgroups and automatically draws the corresponding "Tree." Users can use the interactive touch screen to change the links between requirements by clicking on one of the 72 requirements in the "Misfits" window and then clicking on requirements in "Interactions" window to select or deselect them. Every touch event triggers the program to compute a new hierarchy and a new tree. Users can also type in new requirements using a keyboard, which are automatically added in the "Misfits" and in the "Interactions" frames. The interactive reconstruction (missing the keyboard implementation) was shown in the 4,000 sq ft exhibition *Vers un imaginaire numérique* in Montreal, alongside historical material showing the origins and trajectories of HIDECS 2 and its subsequent versions.

10 The hybrid HIDECS interactive reconstruction and accompanying historical material in the gallery space

RESULTS AND DISCUSSION: HISTORIOGRAPHIC AND THEORETICAL IMPLICATIONS

Considering HIDECS 2 within the Design Methods Network destabilizes it as a fixed or singular system and reveals it to be a mobile and malleable entity that can be followed across multiple contexts. This is in keeping with approaches in adjacent fields in the humanities that foreground specific artifacts as objects to be followed as they move between diverse settings. Anthropologist Arjun Appadurai, for instance, has compellingly proposed that focusing on things and their trajectories rather than on social actors and settings can shed light on the human operations that enliven them (1986). In the history of science, Lorraine Daston has spoken of "biographies of scientific objects" as shedding light on the formation of knowledge through attentiveness to the fluctuating cultural, material, and theoretical meanings of objects of scientific inquiry (2000). Hans-Jörg Rheinberger (1997), in turn, proposed tracking "things embodying concepts"—what he termed "epistemic things"—as a way to navigate sites of knowledge production.

In a similar spirit, mobilities and agencies of techniques and technological artifacts within the Design Methods Network enable threading together disparate contexts while preserving their social, intellectual, and institutional specificities. In other words, the mobilities of HIDECS, its versions, implementations, failures and applications, do not only shed new lights on the computer system itself and provide new sites and cases for historical scrutiny, but they also weave together disparate and heterogeneous settings as parts of the same story—a story about tensions between human reason and algorithmic rationality (Vardouli 2017) and negotiations between empirical data collection and the algorithmic orders that were argued to undergird that data.

The interactive reconstruction of HIDECS hybridizes key contexts of its development and adaptation, while also highlighting some of its key theoretical categories. One of these categories, which puts HIDECS in stark contrast with contemporary data-driven design projects that are predicated on goals and optima, is the notion of “failure.” Through its list of 72 “misfits” reconstruction activates questions about what it means to consider design through a list of situations to avoid and to use algorithmic rationality not to maximize efficiency or productivity of various kinds but instead to avoid a failed architecture. The historical documents accompanying the reconstruction, such as for example a sketch a so-called “failure card” from 1961 also position failure as a foundational technical category pertaining to the organization of building-related information (Alexander 1960a; 1960b). “Failure” denoted a kind of physical event that prevented a need from being satisfied (for example, sleep prevented by bioclimatic discomfort). Sometimes failures shared data, other times they were corrected by the same operations, and other times the correction of one failure aggravated the other. Similar relations of overlap, reinforcement, or conflict were then established among the failures’ corresponding requirement. Depending on the data they shared, failures could be established as “interacting” in certain ways. Grouping together failures that interacted more strongly and setting apart those that were independent from each other, would reveal what Alexander would refer to as the non-arbitrary “structure” of the design problem, that HIDECS 2 was designed to compute (Vardouli 2017).

The interactive reconstruction dramatizes the contingent relationship between data and their organization. The list of 72 misfits loaded in the reconstruction reads like a cornucopia of aphorisms, intriguing in its unapologetic arbitrariness—why 72 misfits, and why these 72? It soon becomes evident that the semantic content of these requirements matters little for the algorithm and that the key consideration are their stated relationships. One could type in non-design related, even nonsensical, statements and the program would unproblematically incorporate them in its tree calculations. At the same time, the reconstruction renders visible the multiple layers of bias and human intervention in an algorithmic process: humans collect the data, describe them as “misfits”, and establish their relationships.

CONCLUSION

In this paper, I have presented two interconnected methods that complement text-based scholarship on historic computer systems with digital humanities approaches such as database visualization and with software reconstructions. These methods have mobilized the historic computer system of HIDECS 2 in two cross-fertilizing ways: first, by tracking its trajectories and adaptations outside its initial contexts of inception and second, by reconstructing a hybrid of the system’s multiple versions and making it available for manipulation and critical contemplation.

Responding to the call behind the *Hybrids & Haecceities* conference theme to dissolve binary conditions and immanent hierarchies and to juxtapose specificities to totalizing abstractions, the paper casts the historic computer program as a *hybrid* absorbing but also revealing the specificities of the heterogeneous sites it traversed. Setting HIDECS 2 *in motion* brings forward critical arguments about cultural meanings and social exchange of computer systems and algorithmic techniques, entanglements of institutional and intellectual formations, and stories about theoretical commitments cloaked under algorithmic objectivity and software pragmatism.

Moving forward, we are planning further work on the software reconstruction to implement the recomposition process: the use of the tree to combine user-defined schematic diagrams. We anticipate this to be challenging and exciting in terms of the interface design, and to present opportunities for further hybridizing the system with the contemporary digital landscape (for instance, one possibility is using web databases for identifying diagrams instead of the users developing them). We are also eager to delve into some of the contexts we have identified by mapping the HIDECS trajectory within the Design Methods Network and to consider digital storytelling as a means for narrating the history of this landmark computer program in its many reconfigurations in the second half of the 1960s.

ACKNOWLEDGMENTS

I would like to gratefully acknowledge Jiaqi (Arlene) Chen for her work on the HIDECS reconstruction, Max Leblanc and George-Étienne Adam for their work on the Design Methods Network database and visualizations, and Eliza Pertigkiozoglou for her feedback on the database structure and querying and for reconstructing drawings from the HIDECS 2 and 3 reports. I would also like to warmly thank the anonymous reviewers for their detailed and constructive comments. This work has been made possible by a grant by the Social Sciences and Humanities Research Council of Canada.

REFERENCES

Alexander, Christopher. 1960a. “Letter to Chermayeff Re: Failure Cards.” Box 4, Folder “Alexander, Christopher, 1958-1966.” Serge Ivan Chermayeff Architectural Records and Papers, 1909-1980. Department of Drawings & Archives, Avery Architectural and Fine Arts Library, Columbia University.

Alexander, Christopher. 1960b. “Letter to Chermayeff Re: Failures Interlock, IBM Group Extraction.” Box 4, Folder “Alexander, Christopher, 1958-1966.” Serge Ivan Chermayeff Architectural Records and Papers, 1909-1980. Department of Drawings & Archives, Avery Architectural and Fine Arts Library, Columbia University.

Alexander, Christopher, and Marvin L. Manheim. 1962a. *HIDECS 2: A Computer Program for the Hierarchical Decomposition of a Set Which Has an Associated Linear Graph*. Cambridge, Mass.: Massachusetts

Institute of Technology, Civil Engineering Systems Laboratory Publication 160.

Alexander, Christopher, and Marvin L. Manheim. 1962b. *The Use of Diagrams in Highway Route Location: An Experiment*. Cambridge, Mass.: Massachusetts Institute of Technology, Civil Engineering Systems Laboratory Publication 161.

Alexander, Christopher. 1963. *HIDECS 3: Four Computer Programs for the Hierarchical Decomposition of Systems Which Have an Associated Linear Graph*. Cambridge, Mass.: Massachusetts Institute of Technology, Civil Engineering Systems Laboratory Publication Report R63–27.

Alexander, Christopher. 1964. *Notes on the Synthesis of Form*. Cambridge, Mass.: Harvard University Press.

Alexander, Christopher. 1965. "A City Is Not a Tree." *Architectural Forum* 122 (1): 58–62 (Part I) and 122 (2): 58–62 (Part II).

Appadurai, Arjun, ed. 1986. *The Social Life of Things: Commodities in Cultural Perspective*. Cambridge, U.K.: Cambridge University Press.

Bernholtz, Allen, and Edward Bierstone. 1966. "Computer-Augmented Design." *Design Quarterly*, Special Issue: Design and the Computer 66/67: 40–51.

Cardoso Llach, Daniel, and Scott Donaldson. 2019. "An Experimental Archaeology of CAD: Using Software Reconstruction to Explore the Past and Future of Computer-Aided Design." In *Computer-Aided Architectural Design*, edited by J.-H. Lee. Singapore: Springer. 105–119.

Cristobal Olave, Diana. 2020. "The Computer-Aided Rough Patterns of Christopher Alexander." *Nexus Network Journal* 23: 113–128.

Daston, Lorraine. 2000. *Biographies of Scientific Objects*. Chicago: University of Chicago Press.

Erickson, Paul, Judy L. Klein, Lorraine Daston, Rebecca Lemov, Thomas Sturm, and Michael D. Gordin. 2015. *How Reason Almost Lost Its Mind: The Strange Career of Cold War Rationality*. Chicago: University of Chicago Press.

Miranda Carranza, Pablo. 2020. *HIDECS2 Python.* GitLab. Accessed May 28, 2022. http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1565218&dswid=-7330.

Moretti, Franco. 2013. *Distant Reading*. London: Verso.

Montgomery, Roger. 1970. "Pattern Language - The Contribution of Christopher Alexander's Centre for Environmental Structure to the Science of Design." *Architectural Forum* 132 (1): 52–59.

Moore, Gary, ed. 1966. "What Is the Design Methods Group?" *DMG Newsletter* 1 (1): 1.

Oudshoorn, Nelly, and Trevor Pinch, eds. 2003. *How Users Matter: The Co-construction of Users and Technologies*. Cambridge, Mass.: The MIT Press.

Pinch, Trevor J., and Wiebe E. Bijker. 1984. "The Social Construction of Facts and Artefacts: Or How the Sociology of Science and the Sociology of Technology Might Benefit Each Other." Social Studies of Science 14 (3): 399–441.

Steenson, Molly Wright. 2017. Architectural Intelligence: How Architects and Designers Created the Digital Landscape. Cambridge, Mass.: The MIT Press.

Rheinberger Hans-Jörg. 1997. *Toward a History of Epistemic Things: Synthesizing Proteins in the Test Tube.* Writing Science Series. Stanford, Calif.: Stanford University Press.

Vardouli, Theodora. 2017. Graphing Theory: New Mathematics, Design, and the Participatory Turn. Doctoral Dissertation, Massachusetts Institute of Technology.

Vardouli, Theodora. 2020. 'Bewildered the Form-Maker Stands Alone': Computer Architectures and the Quest for Design Rationality. In Vardouli T and Olga Touloumi (eds.) Computer Architectures: Constructing the Common Ground. Routledge Research in Design, Technology and Society Series.

Upitis, Alise. 2013. Alexander's choice: How Architecture Avoided Computer-aided Design c. 1962. In Dutta, A. (ed.) A Second Modernism: MIT, Architecture, and the "Techno-Social" Moment. Cambridge, Mass: The MIT Press.

IMAGE CREDITS

Figures 2, 3: *Notes on the Systhesis of Form* by Christopher Alexander, Cambridge, Mass.: Harvard University Press,

Figures 4, 5: Reconstructed drawings by Eliza Pertigkiozoglou from Alexander and Manheim (1962a).

Figure 6: *Design Methods Group Newsletter*, Volume 2, Numbers 1-2, January-February 1968.

All other drawings and images by the author and the CoDEx research group, McGill University.

Theodora Vardouli is an Assistant Professor at the Peter Guo-hua Fu School of Architecture, McGill University. She is co-editor of *Computer Architectures: Constructing the Common Ground* (Routledge 2020) and *Designing the Computational Image, Imagining Computational Design* (forthcoming AR+D Publishing), and co-curator of the exhibition "Vers un imaginaire numérique" (Centre de Design de l'UQAM, Montreal 2021). Her in-progress book (under contract with the MIT Press) examines postwar architecture's relationship with mathematics with focus on the physical and symbolic prevalence of graphs. Vardouli is an External Examiner at the Bartlett (UCL, UK) and an Editorial Board member of the journal *Technology, Architecture and Design* (TAD).