Thinking Inversely in Engineering Design: Towards an Operational Definition of Generative Design Thinking

John Z. Clay[1], Xingang Li[1], H. Onan Demirel[2], Molly H. Goldstein[3], Rundong Jiang[4], Charles Xie[4], Darya L. Zabelina[5], Zhenghui Sha*[1]

[1]*Walker Department of Mechanical Engineering, University of Texas at Austin*
[2] *School of Mechanical, Industrial, and Manufacturing Engineering, Oregon State University*
[3]*Grainger College of Engineering, University of Illinois at Urbana-Champaign*
[4]*Institute for Future Intelligence*
[5]*Department of Psychological Science, University of Arkansas*

Introduction

Generative design (GD) is a design method in which computer algorithms computationally consider human-defined objectives, parameter ranges, and constraints to generate designs [1], [2], [3], [4]. Generative systems may take several forms (e.g., grammar-based techniques or self-assembling systems) [3], and evolutionary systems have received much interest in engineering design contexts [4]. These methods resemble biological evolution / natural selection by iteratively generating sets of design artifacts and evaluating them based on a *fitness function*, i.e., how well each fits the pre-defined objectives. The typical GD process begins as the designer translates real-life design parameters by defining the objective space and parameter interactions for an artificial intelligence (AI) agent to understand [2], [3]. Following this, AI explores all the possible permutations in the design space to search for alternatives that fit the pre-defined objective space. Optimization occurs as designs are computationally generated and evaluated by the AI, and those with high fitness are promoted to the next iteration. Artifacts along the *Pareto front* are then presented to the human designer, who must then evaluate both the AI-generated designs and the human set objective space/rule sets that led to their generation [3]. Further iteration may then occur as the designer chooses one or more AI designs to optimize or returns to a previous task in the design process. Iteration offers the designer a deeper understanding of the design and solution spaces, which may guide further design behavior [5].

Generative systems have only begun to receive significant academic attention in the previous two decades [4]. Thus, GD methodologies are relatively new in engineering contexts. The approach and technologies underlying GD have evolved from previous design methods, most notably those developed for parametric design (PD) within the field of architecture [1], [4]. PD is a computational method in which the designer uses a visual interface, or *parametric schema*, to represent and exploit relationships between parameters to explore the design space [4], [6], [7]. PD generally begins as the designer codes the parametric schema to represent the design problem by defining the logical relationships/rule sets between relevant design parameters to guide design space exploration [6]. The human designer must then evaluate both the rule sets and computationally generated designs. As in GD, iteration is human-driven via either 1) manual

* Corresponding author: zsha@austn.utexas.edu

optimization of a subset of designs or 2) by returning to a previous design phase, such as re-defining the rule sets to explore new design spaces.
As generative systems developed from recent technological developments, PD also evolved from traditional design (TD) due to advancements in computational methods and research in design cognition [4], [8]. The key difference between TD and GD / PD methodologies is that the TD process depends predominantly on the human designer and features limited use of computational methods, such as AI optimization or parametric modeling [1]. The TD process may generally be divided between the early/*conceptual* stage and the late/*detailed* stage [9]. Designers in the conceptual stage explore the design space at a high level, often through idea generation via fast and intuitive sketching. These ideas are then realized, tested, and evaluated by the human designer, who drives iteration until stopping criteria are achieved [1], [9].

The neurocognition of a designer working on designing an artifact may be generally referenced as *design thinking* [8], [10], [11], but different paradigms (TD, PD, GD) each engage the designer's cognition in unique ways. For example, the conceptual stage in TD is heavily dependent on cognitive idea generation (i.e., "brainstorming") to explore the design space [14]. However, human ideation provides a limited number of potential solutions and is vulnerable to external factors such as design fixation [13] or fatigue. GD differs from TD in that exploration does *not* consist of cognitive ideation. Instead, human designers must generate new objective spaces for AI to explore and discover logical relationships between parameters that achieve the objectives. In this way, GD requires *inverse thinking* from the objective space to the parameter space, while in TD, designers are required to cognitively explore the parameter space to optimize towards the objective(s).

Aims and Significance

Design paradigms (e.g., TD / PD / GD) each require the human to carry out a unique set of tasks [1], [3], [6], [7], [9] which in turn define *design thinking* [8], [10], [11]. Thus, each paradigm is accompanied by a unique *design thinking* concept. TD requires a designer to engage in *traditional design thinking* (TDT), PD activates *parametric design thinking* (PDT) and *thinking* concepts such as *systems thinking* (ST) [14], [15] and *computational thinking* (CT) [16], [17] are relevant to a wide range of problem domains. Thus, it may be assumed that GD requires *generative design thinking* (GDT). However, there is a dearth of research towards defining GDT and the relevant cognitive processes [18], [19]. The goal of our research is to explore the cognitive processes underlying GD to synthesize a definition of GDT.

Achieving these goals has many potential benefits, including (but not restricted) to the fields of design education, design research, and human-AI collaboration in GD. Identifying the cognitive processes relevant to GD will facilitate the education of next-generation designers, as teaching is made difficult due to little research into the ways in which successful generative designers think throughout the GD process. Additionally, GDT efficacy may be measured via psychometrics to allow for empirical investigations into GDT. Lastly, a robust definition of GDT and the underlying cognitive processes will facilitate more efficient human-centered GD and human-AI collaboration within the GD process. GD technologies have the potential to augment or discourage human creativity in design, but future research is needed to explore how human factors interact with generative AI [5], [19].

The Evolving Design Thinking Model

To achieve this goal, we review design cognition literature, specifically concepts related to *design thinking*, e.g., *engineering / traditional design thinking, parametric design thinking*, *systems thinking*, and *computational thinking*. For each concept, we highlight the psychological/neuropsychological processes that underlie human design activities, such as creativity, in the conceptual phase of TD [1], [14]. Cognitive concepts identified in our review are evaluated for their relevance to the GD process, and those of high relevance are highlighted and discussed to supply an informed theoretical ground for GDT.

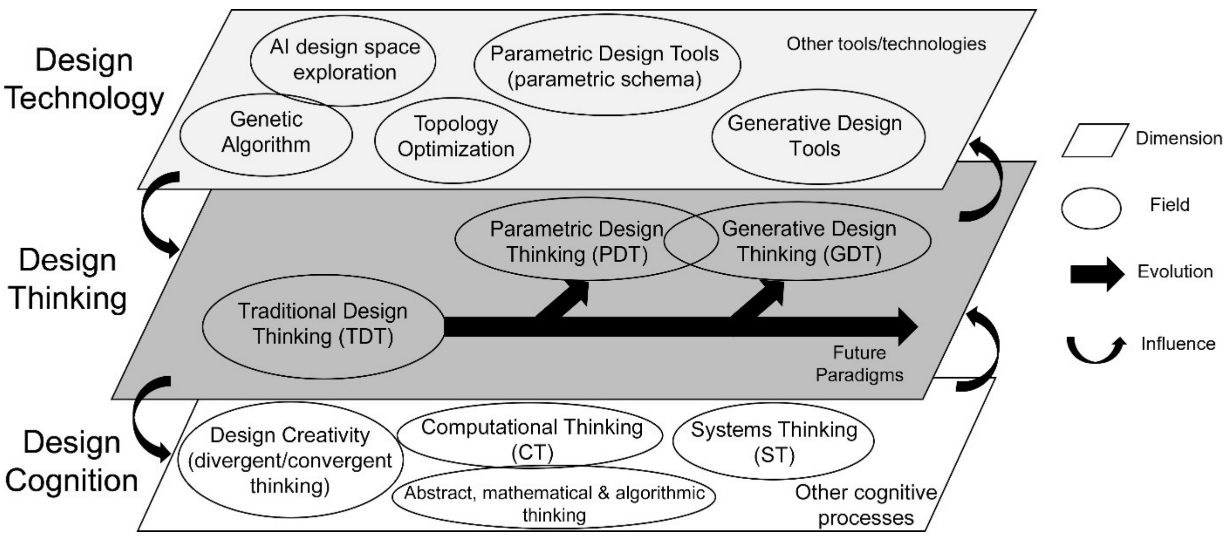

Figure 1: The Evolving Design Thinking (EDT) Model [17]

TDT consists of the cognitive processes activated by the TD process [8], [10], [11], but technological development (e.g., GAs, machine learning, etc.) and knowledge expansion (design cognition research) have evolved TD and TDT. This results in *evolving design thinking*, a dynamic set of task-relevant cognitive processes that influences and is influenced by design paradigms. Technological advancements facilitated the evolution of PD from TD, which motivated research into PD-relevant cognitive processes (i.e., PDT) [3], [6]. Further advancements such as generative software and AI technologies have given rise to GD [2], [4], which opens the need for GDT research. We developed the **Evolving Design Thinking (EDT)** model to guide our review process (Figure 1) [17].

The EDT model is a meta-representation of *design thinking* concepts and their relationships across three levels: *Design Technology* (i.e., technologies and computational techniques that carry out design), *Design Thinking* (i.e., the dynamic, paradigm-specific *thinking* concepts), and *Design Cognition* (i.e., research on the cognitive processes relevant to design). This framework can be used to examine the influences of technological development and advancements in design cognition research on the evolution of GDT from related *thinking* concepts. The cognitive processes discussed in relation to other thinking concepts can then be reviewed and used as a basis for a definition of GDT. The review will contribute new knowledge to the existing literature by providing an in-depth understanding of GDT and a clarification of the obscure boundary between various design thinking concepts. While the review work is in progress, we share the EDT model in this paper in the hope that it could benefit the community and anyone who wants to use it in their own research on design thinking and design cognition.

Acknowledgment

The authors would like to gratefully acknowledge the financial support from the U.S. National Science Foundation (NSF) via grant # 2207408. Any opinions, findings, and conclusions expressed in this publication or presentation are those of the authors and do not necessarily reflect the view of the NSF.

References

[1] J. Mountstephens, and J. Teo, "Progress and challenges in generative product design: a review of systems," Computers, 9(4), 80, 2020. https://doi.org/10.3390/computers9040080

[2] M. McKnight, "Generative Design: What it is? How is it being used? Why it's a game changer," KnE Engineering, International Conference on Design and Technology, 2017, 176-181. https://doi.org/10.18502/keg.v2i2.612

[3] J. McCormack, A. Dorin, and T. Innocent, "Generative Design: A Paradigm for Design Research," in J. Redmond, D. Durling, and A. de Bono, (eds.), Futureground - DRS International Conference 2004, 17-21 November, Melbourne, Australia.

[4] I. Caetano, L. Santos, and A. Leitão, "Computational design in architecture: Defining parametric, generative, and algorithmic design," Frontiers in Architectural Research, vol. 9, no. 2, pp. 287–300, 2020. https://doi.org/10.1016/j.foar.2019.12.008

[5] J. I. Saadi, and M. C. Yang, "Generative Design: Reframing the Role of the Designer in the Early-Stage Design Process," Journal of Mechanical Design, pp. 1-48, 2023. https://doi.org/10.1115/1.4056799

[6] R. Oxman, "Thinking difference: Theories and models of parametric design thinking," Design Studies, vol. 52, pp. 4–39, 2017. https://doi.org/10.1016/j.destud.2017.06.001

[7] R. Oxman, "Theory and design in the first digital age," Design Studies, vol. 27, pp. 229-265, 2005. https://doi.org/10.1016/j.destud.2005.11.002

[8] L. Hay, P. Cash, and S. McKilligan, "The future of design cognition analysis," Design Science, vol. 6, 2020. https://doi.org/10.1017/dsj.2020.20

[9] K. Ulrich, S. Eppinger, and M. C. Yang, "Product Design and Development," 7th ed.; McGraw-Hill: Hightstown, NJ, USA, 2020.

[10] J. S. Gero, and J. Milovanovic, "A framework for studying design thinking through measuring designers' minds, bodies and brains," Design Science, vol. 6, e19, 2020. https://doi.org/10.1017/dsj.2020.15

[11] U. Kannengiesser, and J. S. Gero, “Design thinking fast and slow: a framework for Kahneman’s dual-system theory in design,” Design Science, vol. 5. e10, 2019. https://doi.org/10.1017/dsj.2019.9
[12] J. Alcaide-Marzal, J. A. Diego-Mas, and G. Acosta-Zazueta, “A 3D shape generative method for aesthetic product design,” Design Studies, vol. 66, pp. 144–176, 2020. https://doi.org/10.1016/j.destud.2019.11.003
[13] D. G. Jansson, and S. M. Smith, “Design fixation,” Design Studies, vol. 12, i1, pp. 3-11, 1991. https://doi.org/10.1016/0142-694X(91)90003-F
[14] M. Greene, R. Gonzalez, P. Papalambros, A.-M. McGowan, “Design thinking vs. systems thinking for engineering design: What’s the difference?” in 21st International Conference on Engineering Design, ICED17, 2017, p. 10.
[15] S. Gianelloni, and M. H. Goldstein, “Modelling the Design Systems Thinking Paradigm,” in ASEE Virtual Annual Conference, 2020. https://doi.org/10.18260/1-2--34981
[16] V. J. Shute, C. Sun, and J. Asbell-Clarke, “Demystifying computational thinking,” Educational Research Review, vol. 22, pp. 142–158, 2017. https://doi.org/10.1016/j.edurev.2017.09.003
[17] J. M. Wing, “Computational thinking,” Communications of the ACM 49, no. 3, pp. 33-3, 2006.
[18] X. Li, H. O. Demirel, M. H. Goldstein, and Z. Sha, “Exploring Generative Design Thinking for Engineering Design and Design Education,” in 2021 ASEE Midwest Section Conference, hosted virtually in Fayetteville, AR, Sep. 13-15, 2021.
[19] H. O. Demirel, M. H. Goldstein, X. Li, and Z. Sha, “Human-Centered Generative Design Framework: An Early Design Framework to Support Concept Creation and Evaluation,” International Journal of Human-Computer Interaction, pp. 1.12, 2023. https://doi.org/10.1080/10447318.2023.2171489