A Multi-case Study of Traditional, Parametric, and Generative Design Thinking of Engineering Students

Elisa Koolman[1], John Z. Clay[1], Xingang Li[1], Rundong Jiang[2], Molly H. Goldstein[3], Charles Xie[4], H. Onan Demirel[5], and Zhenghui Sha[1](✉)

[1] Walker Department of Mechanical Engineering, University of Texas at Austin, Austin, USA
zsha@austin.utexas.edu
[2] Association for Energy Affordability, Bronx, USA
[3] Industrial and Enterprise Systems Engineering, University of Illinois Urbana-Champaign, Urbana, USA
[4] Institute for Future Intelligence, Natick, USA
[5] Department of Mechanical Engineering, Oregan State University, Corvallis, USA

Abstract. The recent surge in generative artificial intelligence (AI) applications within engineering design requires equipping future engineers with the skills to effectively utilize these advanced design tools. Understanding how students think about generative AI in design is pivotal in shaping engineering design education. This study aims to explore mechanical engineering students' perceptions and approaches to generative design (GD) compared to parametric design (PD) and traditional design (TD). To achieve this, a comprehensive curriculum encompassing these three design paradigms was developed and administered to seven undergraduate mechanical engineering students. Students engaged with the curriculum activities while their responses and interactions were recorded using a verbal protocol. Three students were selected for an in-depth multi-case study analysis. The qualitative and quantitative findings suggest that students' thought processes and behaviors in GD are influenced and informed by their experiences and understanding of TD and PD.

1 Background and Motivation

With the rise of groundbreaking tools such as ChatGPT, generative artificial intelligence (AI) has attracted significant attention, revolutionizing how AI is applied in various fields, including engineering design [1, 2]. This trend has seen the adoption of AI techniques in numerous design activities such as topology optimization, material design, design synthesis, and product design [3–5]. Within the spectrum of AI-assisted design methods, generative design (GD) stands out for its power to enhance design creativity. GD represents a design paradigm that enables an automatic process of iterative design exploration to find optimal design solutions either via explicit programming or implicit learning methods, such as genetic algorithm and generative AI models [6]. Its utility has been showcased through the integration into commercial computer-aided design (CAD) software, such as Autodesk Fusion 360 and PTC Creo, demonstrating its versatility and

J. S. Gero (Ed.): DCC 2024, *Design Computing and Cognition'24*, pp. 89–104, 2025.
https://doi.org/10.1007/978-3-031-71922-6_6

value to engineering design. We refer to GD as an umbrella term that covers the design methods applying generative AI to iteratively explore the design space and generate a (set of) solution(s) that satisfy human-defined objectives and constraints. These approaches utilize a range of generative techniques, such as genetic algorithms (GAs), variational autoencoders (VAEs), generative adversary networks (GAN), and large language models (LLMs). No matter which techniques are used for GD, the underlying cognitive process involved in design could be similar as they require an inversion design thinking compared to the forward design thinking in TD [11].

Alongside GD, parametric design (PD) is another widely adopted computational design paradigm that existed before the concept of GD. PD utilizes parameters to represent a range of designs. PD approaches typically employ a parametric schema, i.e., a visual interface that computationally articulates the interaction among various design parameters [7]. This schema enables designers to efficiently explore and innovate within the design space by applying logical rules and interrelationships between parameters, thereby facilitating the creation of unique and novel artifacts [7–9]. In contrast to computational design methods, traditional design (TD) [7] is a methodology predominantly guided by human insight, skills, heuristics, and knowledge. TD methods diverge significantly from PD and GD, particularly in terms of the primary drivers of the design process, and this is independent of whether CAD is used or not. In PD and GD, computational algorithms play a crucial role, significantly influencing both the initial creation and the progressive refinement of designs, and human interaction is primarily focused on overseeing and fine-tuning the outcomes generated by these automated processes [10, 11].

Design thinking (sometimes referred to as *design cognition*) generally refers to the collection of mental processes, psychological traits, and previous experiences that a designer utilizes during the design process [12–14]. However, different design paradigms demand specific cognitive approaches, and designers exhibit different behaviors [10, 11]. For example, exploring design possibilities in TD relies on human designers' creativity and experience. This approach often leads to limited solutions and is susceptible to design fixation [15], fatigue, or even the designer's socioeconomic status [16]. Conversely, using computational design tools (e.g., GD tools) in the design process influences both the process itself and designers' thought processes and, thus, their behaviors [17]. GD requires designers to focus on defining design objectives and constraints (e.g., those related to geometry, materials, and manufacturing processes). These constraints and objectives are then used in computational models to automatically search for design concepts that can surpass human cognitive limits.

The increasing use of computational and AI-assisted design tools requires a rethinking and updating of the current engineering education regarding both technical and psychological aspects. This is crucial to prepare the next generation of engineers for these advancements [10, 11, 18]. Therefore, it is essential to understand the relationships and distinctions between students' generative design thinking (GDT), parametric design thinking (PDT), and traditional design thinking (TDT). To that end, this study seeks to answer the following **Research Question (RQ)**: How do engineering students think about and behave during generative design process compared to parametric and traditional design processes?

2 Methods

The methodological framework of the study, as illustrated in Fig. 1, starts with the development of a design curriculum, which is segmented into three distinct sections, each corresponding to a different design paradigm: traditional design (TD), parametric design (PD), and generative design (GD). This curriculum serves as the primary educational material in our study. We recruited and engaged with students in a three-session experimental setup. In each session, students are involved in a verbal protocol session specific to one of the three design paradigms being studied. Students articulated their design processes and responded to reflective questions. Data collection comprised verbal responses from the verbal protocols, students' responses to curriculum-based practice problems, and sequential process log data capturing participants' interactions within the design software.

To analyze students' verbal protocol and responses to curriculum-based practice problems, we employ the case study analysis [19], which can offer a comprehensive and nuanced understanding of specific instances grounded in real-world contexts. Case studies address complex 'how' and 'why' questions, in contrast to surveys that typically focus on 'where' and 'what' inquiries. A key strength of case studies lies in their integrative approach to evidence analysis with various records. This multifaceted evidence-gathering allows for a more robust and in-depth exploration of the case under study. The case for this study is undergraduate engineering student cognition when using three different design paradigms: TD, PD, and GD. For the analysis of the software data, primarily due to the constraints of small sample size, descriptive statistical methods are used in this pilot study.

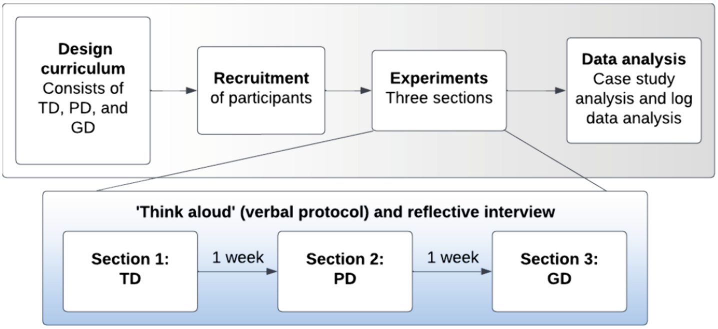

Fig. 1. The methodological framework of the study

2.1 Participants

This study was conducted at an R1, Texas university. Participants were undergraduate mechanical engineering students who were invited through emails and advertisements to their respective design courses at the beginning of the Fall 2023 semester. Students were eligible for participation if they were above eighteen years old, a university affiliate, and were either currently enrolled or had already completed the Introduction to Engineering Design and Graphics course. Figure 2 shows the demographic information of the participants. Seven students participated in the full sequence of three data collection sessions. First-year students made up a majority of the sample set. Most participants had previous experience with CAD. Students were compensated $10, $25, and $25 for the first, second, and third sessions, respectively. A scaffolded payment system was used to incentivize continued participation.

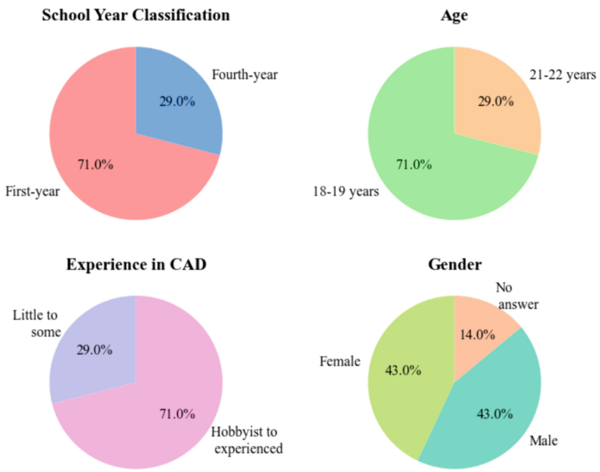

Fig. 2. Participants' demographic information.

2.2 Curriculum Materials

The participants were provided with a forty-seven-page curriculum/workbook covering three design paradigms: traditional design (TD; twenty-three pages), parametric design (PD; fourteen pages), and generative design (GD; twelve pages). The primary goal of the curriculum was to teach GD concepts and provide students with the opportunity to apply these methodologies. TD and PD concepts/methodologies were included in the curriculum to ensure that the reader possessed the relevant knowledge base required to understand and apply GD concepts. The curriculum was developed to be self-contained and accessible to undergraduate students with little to no previous design knowledge. Designing via TD and PD processes framed the differences and possibilities of GD methods in engineering design, i.e., in expediting and expanding human-driven design space exploration [3, 4].

We provided a high-level and domain-general representation of a common design process and described the role of the designer in each of the three paradigms as occurring across four closely related stages: Problem Definition (identifying constraints and objectives), Exploration (of the design space and design of experiments), Evaluation (of artifact performance; considering the Pareto front), and Iteration (design and refinement of experiments). The direction of the design process was emphasized as a key difference between TD and GD throughout the curriculum. TD practices are dependent on human behavior, which generally requires the designer to manually explore the design space to determine how the parameters may be arranged in the objective space to reach the goals, i.e., working from the design space to the objective space (*Forward Design*). However, the GD process generally requires the designer to define the objective space for an AI agent to consider as it expedites design space exploration, i.e., working from the objective space to the design space (*Backward Design*) [10].

Table 1. The CAD activity for the solar farm design in the design software.

	TD	PD	GD
Problem Definition	Plot designs along a design space based on tilt angle and spacing	Describe how to quantify the solar energy variables	Define the formula for the yearly profit
Exploration	Manually create designs with different design variables	Re-create a pictured solar farm and create two new designs	Use generative AI to generate six designs
Evaluation	Evaluate and plot the manually created designs	Evaluate and plot the designs from the parametric schema	Evaluate and plot the AI-generated designs
Iteration	Manually optimize designs via additional exploration and evaluation	Manually optimize designs via additional exploration and evaluation	Generate, evaluate, and plot six additional designs

Each section contained text detailing the design process and a CAD practice problem to allow students the opportunity to apply the material. The descriptions of each section in the chapters were accompanied by two examples: a text-only car wheel design example and a hands-on, CAD-based solar farm design example that was completed by participants in the software Aladdin [20]. Aladdin is a cloud-based computer-aided design software with traditional, parametric, and generative design capabilities, described in more detail in the following section. The design task given to students for TD, PD, and GD is to design a solar farm using Aladdin on a university-owned space while maximizing profit and efficiency. See Table 1 for a brief description of each of the solar farm design CAD activities completed by participants in Aladdin and see below for additional information on the design concepts that students were taught in the design curriculum. The second, third, and fourth authors wrote the curriculum and created the practice design activities.

The TD chapter was prefaced by a four-page introduction that compared the characteristics of the design and art-making processes and defined three paradigms for solving design problems (TD, PD, and GD). The participant was first introduced to TD, i.e., human-centric traditional engineering principles. The TD: Problem Definition section emphasized the importance of identifying and clearly stating the objectives and constraints during the early design stages, and TD: Exploration explained how one must consider the design decisions made in the previous stages and iterations. The participant was then introduced to diverging, a common exploration strategy that prompts the designer to consider how to achieve multiple antithetical goals. Dominated (i.e., those with worse performance than other designs) and non-dominated designs were then explained (TD: Evaluation). Finally, TD: Iteration emphasized that design is an iterative, non-linear process.

Participants were then introduced to PD as a method that utilizes "algorithms and mathematical equations… based on set rules and parameters" throughout the design process to create highly complex, yet easily modified artifacts [21]. PD: Problem Definition introduced the concept of the parametric schema, i.e., the digital interface that models the relationships and may be manipulated by the designer to create/edit an artifact (Fig. 3, top image), and encouraged the participant to draw comparisons between TD and PD. The PD: Exploration section prompted the participant to explore the design space using the parametric schema. PD: Evaluation explained that the evaluation of computationally generated designs often requires the designer to consider and compare many different artifacts, which may then be improved via further iteration (PD: Iteration).

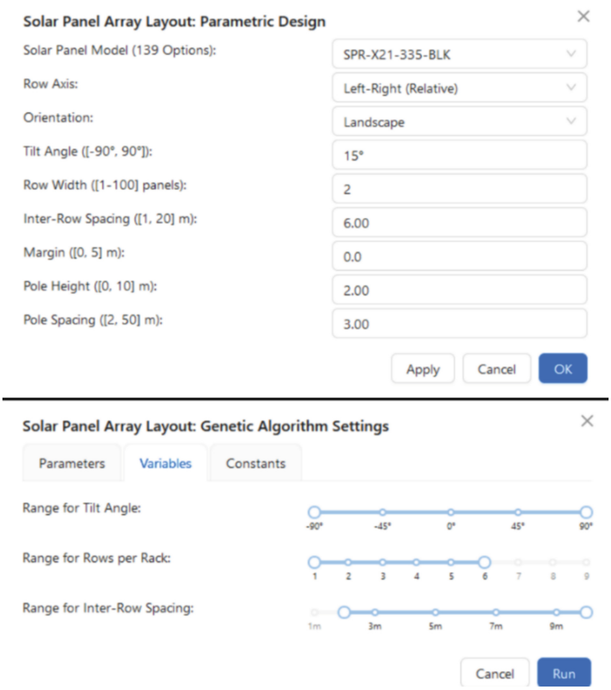

Fig. 3. Solar farm design parameters in Aladdin. Top: the parametric schema in PD. Bottom: The variable ranges in GD.

GD was introduced as a method for automating certain design process tasks in the final chapter. The early phases of GD (Problem Definition) require the designer to clearly define and set the goal(s), parameter ranges, and constraints (Fig. 3, bottom image) for AI-driven design space exploration (GD: Exploration; [3, 7, 22]). Despite the reliance on AI to generate potential solutions, human input is still required for subjective decisions which are often related to aesthetic preferences [10]. Students were prompted to further evaluate these designs (GD: Evaluation) by completing an empty two-dimensional scatterplot with X-Y axes of the two solar farm design objectives and identifying which designs were best optimized for multiple objectives. Finally, students were informed that design is an iterative process, even with the use of generative AI *(GD: Iteration).*

2.3 Aladdin and Design Process Behavior

Participants completed twelve practice design activities. Design activities were completed in (or referenced) Aladdin, an open-source, web-based CAD software [20], which enables TD, PD, and GD and evaluation capabilities for solar energy structures (e.g., houses and solar farms) via high-fidelity function simulation and analysis. Aladdin gathers fine-grained design process data by collecting a sequential log of each user action, which allows researchers to analyze low-level design behaviors. Importantly, Aladdin's data collection methods are non-intrusive and unseen by the user to avoid introducing interruptions and/or bias into the design process.

Participant design process behavior took place during the solar energy solution practice activities across three paradigms (Fig. 4). Non-design (e.g., Open Project and Save File) and redundant (e.g., Close Graph) behaviors were removed from the data, and only design-related behaviors were included in the analyses. Design-related behaviors included adding, deleting, editing, and analyzing the performance of solar panels and considering the effect of environmental factors (e.g., the date and time). Several design actions were unique to the paradigm being used. For instance, the designer in TD added a panel via the Add Solar Panel action, but in PD and GD, they respectively made PD Solar Panel Array Layout: Add and Run Genetic Algorithm actions.

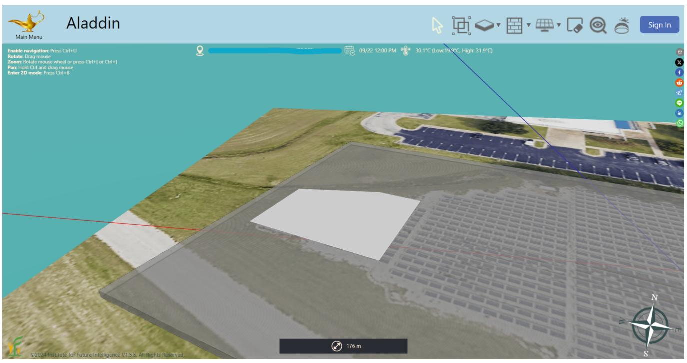

Fig. 4. Solar energy practice activity in Aladdin

2.4 Data Collection

The participant's design thinking process was documented using verbal protocol [23], in which participants were asked to "think aloud" by explaining their thoughts and reasoning for their actions. The verbal protocol is a useful tool for design thinking research as it provides a step-by-step description of a student's design process [24]. Each participant answered three reflective interview questions after the verbal protocol for each chapter of the curriculum.

The students participated in one verbal protocol session for each of the three design curriculum chapters: traditional design (TD), parametric design (PD), and generative design (GD). The participants met for one hour with the same researcher once a week for three weeks. The researcher recorded both the session audio and the student's computer screen. At the beginning of each session, students were provided with a chapter of the curriculum, and as they read the chapter text, they verbalized their thoughts and completed the practice design activities. The students participated in a short reflective interview at the end of each session, and they were asked questions about the chapter they completed in that session. For the final session, they were also asked about the overall curriculum.

We concluded data collection once we reached saturation of data, where additional participants provided little new information. Saturation was based on information regarding the improvement of the curriculum and not on the cognition of undergraduate engineering students while designing.

2.5 Data Analysis Approach

The multi-case study, along with the verbal protocol, is an appropriate data analysis method to answer how students think about GD compared to PD and TD because case studies allow for detailed descriptions of how a phenomenon takes place [19]. We conducted three descriptive case study analyses of the verbal protocol and the reflective interview of students with different design software experience and different academic years: one freshman with little design experience, one freshman with design software experience, and one senior who has taken multiple design courses. Each participant is given a pseudonym for anonymity. Each case study is analyzed independently from the others, focusing on the transcripts from all three design sessions and using the memos the researcher wrote after each session. The structure of the descriptive cases follows the same structure as the curriculum (TD, PD, GD), as well as students' discussions of the differences between each design protocol. These three case studies follow literal replication, meaning the cases are predicted to have similar results [19]. Even though the students have different engineering design skills and years in their program, the participants all have no experience with GD, so the conclusions of this study will describe commonalities among students with different backgrounds.

3 Results and Discussion

3.1 Case Study 1: Sabrina (P102)

Sabrina is a first-year, first-semester student majoring in mechanical engineering. They had some prior experience using CAD software from the class that they are currently enrolled in, Introduction to Engineering Design and Graphics, and they did not have CAD experience in high school. Sabrina have some TD experience and no PD or GD experience. During the TD verbal protocol session, they found the process of manually editing a design to be "testing [their] patience." They tended toward curiosity when investigating the usability of the software. They considered external factors such as the direction of the sunrise and sunset. Though the text in the TD curriculum emphasized design theory, they said that they forgot the theory once they started using the design software: "By the time I was doing the software stuff, I completely forgot about [the theory] and I wasn't really thinking about traditional design anymore."

During the PD session, the form of playing around took shape in changing the solar panel tilt angle, inter-row spacing, and row width. The student changed the variables until it looked like "the space is actually being put to use," suggesting that their objective was visual. Once the curriculum discussed how to analyze energy output, the student used the efficiency objective to determine the design that is the "winner." When rationalizing the yearly efficiency, they referred to external factors such as seasons. Sabrina referred to previous designs' variables, such as low and high tilt angles, to inform future designs, such as choosing a tilt angle between the low and high angles.

During the GD session, Sabrina followed the curriculum instructions on how to choose an objective, such as designing for the highest yearly profit. Aladdin's genetic algorithm displayed each design that the AI considered as it explored the design space. As Sabrina saw this, they said, "Oh, it's, like, actually changing! That's nice." Sabrina expected the AI-generated designs to be identical for the same objective: "If I'm putting the same exact [objective], why would I get different results?" Sabrina later noticed that the generative algorithm only went through ten iterations per initiation, and the curriculum stated that each design iteration was random. They only understood why each design was different once the curriculum explained why the designs would be different. Aladdin's generative features allow the user to choose between several objectives to optimize while exploring the design space and define the range for each of the relevant variables. Once Sabrina saw that they could edit the ranges, they started narrowing the tilt angle range, so that "the randomness won't be so random." Their idea of the "most ideal" ranges was based on what they had learned about PD as opposed to TD, as PD emphasized the manipulation of variables and their ranges.

Sabrina directly compared GD and PD by saying that GD was, "hands-on enough for me to feel like I'm interacting with the software, but not too hands-on like… the last one [PD] was where I was… getting humbled with my lack of skills." They said that GD "was more just… directly in my face," which allowed them to see "the different possibilities of engineering more widely." PD helped them "learn more about… the real-life application of solar panels." In future projects they would prefer to use GD mixed with PD; GD to generate random designs, and PD to, "tweak them up." They explicitly state that they would not like to use TD because GD is "most time efficient."

Their reasoning for mixing PD and GD is that they wouldn't want a randomly generated design, so they would use the parametric design to personalize the design.

3.2 Case Study 2: Suyash (P103)

Suyash is a first-year mechanical engineering student with four years of CAD experience and is currently taking the design-based course Introduction to Engineering Design and Graphics. Suyash has TD experience and no PD or GD experience. Suyash explored the possibilities of the software during the TD verbal protocol session, such as the effect of environmental factors on solar farm design (e.g., the direction of the sunrise). Suyash also explored the design options offered in the CAD software (e.g., automatic tilting solar panels). They described their first experience with the TD practice problem as "tedious" but "a simple process." Their goal for the first design was the efficiency of power output, and they stated that they neglected cost. When they discovered that their solar farm design had a negative profit margin, they rationalized by saying that solar power "is not close to… the profit margin of like oil or something like coal."

Suyash picked up the software skills of PD readily. They quickly began to detect correlations between design variables, such as the relationship between solar panel row width and the inter-row spacing of panels within a solar farm. During evaluation, they noticed that the profit of their design was still negative, and commented, "Is that because… I designed it wrong, or—there's no way they all lose that much money." They said the PD process "is just more challenging" than the TD process because "it kind of takes time to break [the parameters] down." Finally, they compared PD to TD by saying that the former focuses on "tinkering with the software side of it, instead of doing… manual constraints and stuff like that."

Suyash contemplated GD as they watched the generative algorithm explore the design space, and commented, "You know what it's looking for, but you don't really know how it's doing that." After their second design with the same objective, they questioned why the same objective "spat out different designs." Suyash chose to make designs using the yearly average energy output in design iterations because they're more interested in performance than the cost and revenue, stating that "I'm not… in business, I'm just… a designer." They continued working towards this objective and methodically adjusted the variable ranges one at a time. The changes were based on intuition they gained from the parametric design session. They changed the tilt angle because "based on… what I know from the other [design processes], the tilt angle does matter for efficiency." Finally, they said that GD is "the most modern [design process], definitely how the future is going be," so though they don't understand how AI generates design, they still believe it will be widely adopted.

Suyash said that they would prefer to use GD as opposed to TD or PD in future engineering projects, e.g., their capstone project. However, they specified that TD and PD may still be useful if GD "isn't giving you what you want."

3.3 Case Study 3: Ricky (P108)

Ricky is a fourth-year mechanical engineering student with intermediate knowledge of CAD software and intermediate-advanced knowledge of engineering design. Ricky has experience with TD, and no experience with parametric or generative design. They took the same introductory graphics course Sabrina and Suyash are currently taking in their first year. Ricky is currently in an engineering design methodology course with a group project where they are leading the computer-aided design portion of the project. In the TD session, Ricky considered design outcomes before designing a solar farm, and they considered the effect the angle of the sun would have on average energy output. After their first design, they evaluated the average yearly power output and cost, and they adjusted each design afterward. They compared the two designs' yearly total power output and found that the solar panels with a 20-degree tilt angle performed better than the 0-degree tilt angle, so they decided to design a solar farm with a 45-degree tilt angle. When asked about the approachability of the TD session, they said, "Obviously, it helps that I'm in a major that is pretty focused on traditional design."

In the PD session, Ricky was eager to evaluate their designs. They used physical/environmental (e.g., shadow mapping) and numerical (e.g., total yearly power output and profit) analysis methods. Ricky defined ratios between parameters when designing. For example, they made their row width 2.5 times the length of their row spacing. Even when their ratios did not perform as well as they expected, they said it gave them "a decent starting point." Halfway through the session, they started using profit as their objective. Even though Ricky has not used PD before, they said it helps that they have CAD experience.

In the GD session, Ricky tested the limits of generative AI by attempting to give the software design parameters that had the potential to result in unrealistic designs. Their expectations of the AI-generated designs with certain objectives were informed by PD, which led them to expect a dense inter-row spacing when the objective was yearly profit. They realized that each AI generation would produce different designs, though they "wouldn't expect it to be completely different just because AI gets a little crafty sometimes and you can't really predict what it will do." Ricky adjusted the parameters of inter-row spacing to have a tight range to "see what it does when given… less space to work with" and to see "if there is a design that it will come up with that might not be… practically feasible." They did this because "exploring how generative design can fail can give… insight into what expectations might be reasonable for a project."

Ricky said that they would prefer to use PD in the future, as "it kind of hits… the sweet spot between traditional and generative design… parametric design is a great way of incorporating new technology into traditionally proven design methods." Additionally, they comment that they would enjoy working on a design project more than just telling an AI to "design it all for me."

3.4 Cross-Case Analysis and Discussion

Sabrina, Suyash, and Ricky used TD and PD as building blocks for GD. Their direct comparisons are shown in Table 2. Each student discussed PD during their GD session, specifically that the PD session informed their expectations of GD or their design decisions during the AI-based activities. The students also adjusted the variable ranges during the GD sessions, indicating they were considering the effects of parameters on their designs. Students no longer considered the effects of environmental variables (e.g., weather, solar heatmap) or the location of the solar panels for GD. Though students discussed a broad range of design parameters in the TD sessions, the first-year students found it tedious. This is most likely the reason that no student said that they would prefer to use TD in future design projects over PD or GD. Though, no student preferred GD on its own either. They were cautious about putting their sole reliance on GD and instead chose to either mix it or only use it as their first attempt and then subsequently use other design processes.

Table 2. Comparison of TD, PD, GD discussion for three case studies

Discussion	Sabrina	Suyash	Ricky
TD	(1) Found TD tedious, (2) Explored broadly, (3) Considered environmental factors	(1) Found TD tedious, (2) Power output objective, (3) Considered environmental factors	(1) Cost output objective, (2) Power output objective, (3) Considered environmental factors
PD	(1) Visual objective, (2) Energy output objective	(1) Defined variable ratios, (2) Cost objective	(1) Defined variable ratios, (2) Visual objective, (3) Cost objective, (4) Energy output objective
GD	(1) Expected identical designs generated by the same objective, (2) Defined variable ranges, (3) Variable ranges based on PD	(1) Expected identical designs generated by the same objective, (2) Defined variable ranges, (3) Variable ranges based on PD	(1) Realized the same objective gave different designs, (2) Defined variable ranges, (3) GD expectations based on PD
TD vs. PD vs. GD	PD is more hands-on than GD	(1) PD is more challenging than TD, (2) GD is the most modern	PD is a "sweet spot" between TD and GD
Preference	GD mixed with PD	GD with TD and PD as secondary options	PD

The case studies suggest that the designer's intuition for the GD boundary conditions stems from their experience using PD. It may be helpful for the designer to leverage experience from PD when they first experience GD, considering that many students come with preconceived notions of AI. For example, Sabrina and Suyash had the preconception that generative AI would always produce the same design when given the same objective. However, this may have been related to PD because the same parameters produced the same design in the PD session. Alternatively, students may have used PD to inform GD purely because that was the most similar design process to GD they had seen, and it was just a week prior. While they may have had the language to compare GD to PD because of the previous session, their actions of bounding parameters in their GD session show that they were using PD techniques in their GD sessions. Design thinking is notoriously difficult to teach, and project-based learning serves as a positive pathway [25]. The solar panel project models effective design teaching and provides information on GD teaching structure.

3.5 Quantitative Results and Discussion

Participants could execute twenty-five unique design actions which generally fell within one of nine action types (Fig. 5). *Add* and *Edit Solar Panel* were the most frequent design behaviors; *Check Environment* and adding solar panels via the parametric schema were the least frequent. TD actions (e.g., *Add*, *Edit*, *Delete Solar Panel*) were observed during PD as participants were instructed to also use the former paradigm and compare it to PD. *Run Genetic Algorithm* was a GD-exclusive action that made up the majority (81%) of the actions in the GD session (*Solar Panel Simulation*: 19%). Participants used *Check Environment* (e.g., *Heliodon*, *Change Date and Time*) to evaluate the effect of environmental factors on their designs during TD and *Generate Solar Heatmap* during PD. However, none of the participants considered environmental factors during the GD session. They may have assumed that AI included these variables during design generation and evaluation and that there would be no use in re-consideration. A non-exclusive explanation is that designers appear to have taken a more radical approach to design iteration in GD than in TD/PD. If the participant felt that a design could be improved, they opted to re-initiate the generative algorithm instead of fine-tuning the algorithm's previous design. The designers may have diverted their attention away from the lower-level, individual parameters, i.e., shadowing on solar panels, and opted to adopt a high-level focus on the ways that AI explored/exploited the parameter ranges.

See Fig. 6 for the number of overall and unique actions made by each participant in each session. Design process behavior data were not logged for three of the sessions (Participant 102 TD; Participant 109 PD and GD) due to unforeseen issues. Excluding these two participants, the other designers made an average of 230.8 (Standard Error = 14.6) overall actions and showed a range of 11 to 13 unique actions. Finally, participants completed at least 90% of 16 out of the 21 sessions.

4 Limitations and Future Work

The sample for the present study was limited to undergraduate students, which may have biased the data due to the differences between novice and expert designers [26]. Future work should consider including professional designers. Additionally, each participant read the chapters in the following order: TD, PD, GD. However, the order of presentation may have influenced their opinions and behaviors in the subsequent sessions. Future studies will randomize the order of the paradigm presentation to evaluate order effects.

We plan to deploy this study at multiple universities and leverage additional data to explore questions relating to how students make design decisions when using GD,

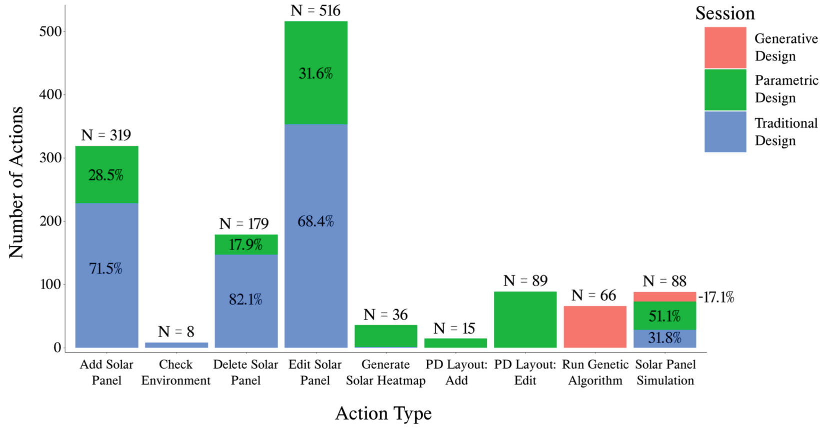

Fig. 5. The relevant design process actions made by students and their frequencies overall for the three design paradigms

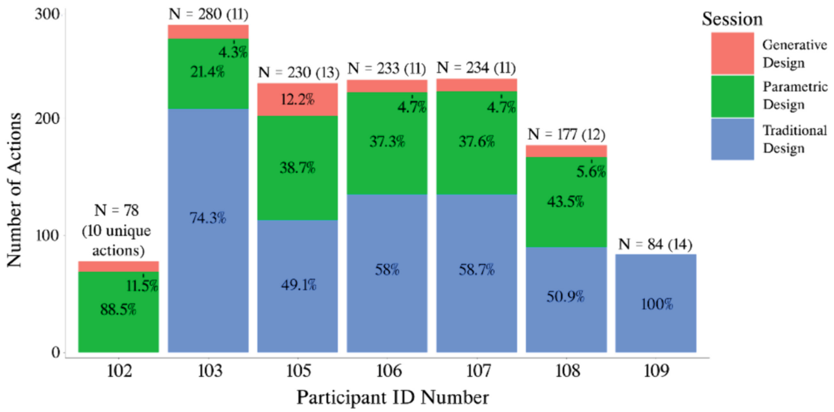

Fig. 6. The number of design process actions made by each participant overall for the three design paradigms. Data was not collected for Participant 102's TD session, and Participant 109's PD and GD sessions.

and how to measure design success in the context of generative AI. We also plan to employ machine learning techniques for sequential CAD data analysis (see [27] for an example of these techniques) for this larger sample. This future analytical strategy is anticipated to yield deeper insights into the students' design activities, enabling a comparative examination across the various design paradigms.

5 Conclusion

This study begins to answer the *RQ* (How do undergraduate students think about and behave during generative design process compared to parametric and traditional design processes?) by analyzing how students think about and behave during GD compared to

PD and TD. The verbal protocol provides an in-depth set of data on students' thought processes, while the screen recording and design process behavior data provide detailed information on behavior. This study can inform future work on the design thinking of students using GD, especially the consideration that students build on previous PD skills. The verbal protocol revealed a nuance of student design thinking that was not reflected in the quantitative data. Students had varied perceptions of the differences between TD, PD, and GD. Though they all found each design process different, the students found that GD is less hands-on and more "modern" than TD and PD.

Acknowledgements. The authors 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. Makatura, L., et al.: How can large language models help humans in design and manufacturing? arXiv. https://doi.org/10.48550/arXiv.2307.14377 (2023)
2. Picard, C., et al.: From concept to manufacturing: evaluating vision-language models for engineering design. arXiv. https://doi.org/10.48550/arXiv.2311.12668 (2023)
3. Regenwetter, L., Nobari, A.H., Ahmed, F.: Deep generative models in engineering design: a review. J. Mech. Des **144**(7), 071704 (2022). https://doi.org/10.1115/1.4053859/1136676
4. Mountstephens, J., Teo, J.: Progress and challenges in generative product design: A review of systems. Computers **9**(4), 80 (2020). https://doi.org/10.3390/computers9040080
5. Li, X., Wang, Y., Sha, Z.: Deep learning methods of cross-modal tasks for conceptual design of product shapes: a review. J. Mech. Des. **145**(4) (2023). https://doi.org/10.1115/1.4056436
6. Demirel, H.O., Goldstein, M.H., Li, X., Sha, Z.: Human-centered generative design framework: an early design framework to support concept creation and evaluation. Int. J. Hum. Comput. Interact. (2023). https://doi.org/10.1080/10447318.2023.2171489
7. Caetano, I., Santos, L., Leitão, A.: Computational design in architecture: defining parametric, generative, and algorithmic design. Front. Architectural Res. **9**(2), 287–300 (2020)
8. Oxman, R.: Thinking difference: theories and models of parametric design thinking. Des. Stud. **52**, 4–39 (2017)
9. Oxman, R.: Theory and design in the first digital age. Des. Stud. **27**(3), 229–265 (2006)
10. Li, X., Demirel, H.O., Goldstein, M.H., Sha, Z.: Exploring Generative Design Thinking for Engineering Design and Design Education. In: 2021 ASEE Midwest Section Conference (2021)
11. Clay, J.Z., et al.: Thinking inversely in engineering design: towards an operational definition of generative design thinking. In: ASEE Annual Conference and Exposition, Conference Proceedings (2023)
12. Foster, D.: Generative deep learning: teaching machines to paint. Write, Compose, and Play (Japanese Version) O'Reilly Media Incorporated, pp. 139–140 (2019)
13. Hay, L., Cash, P., McKilligan, S.: The future of design cognition analysis. Des. Sci. **6**, e20 (2020)
14. Gero, J.S., Milovanovic, J.: A framework for studying design thinking through measuring designers' minds, bodies and brains. Des. Sci. **6**, e19 (2020)
15. Jansson, D.G., Smith, S.M.: Design fixation. Des. Stud. **12**(1), 3–11 (1991)

16. Castillo-Vergara, M., Galleguillos, N.B., Cuello, L.J., Alvarez-Marin, A., Acuña-Opazo, C.: Does socioeconomic status influence student creativity? Think Skills Creat **29**, 142–152 (2018)
17. Saadi, J.I., Yang, M.C.: Generative design: reframing the role of the designer in early-stage design process. J. Mech. Des. **145**(4), 041411 (2023)
18. Brown, A., Goldstein, M.H., Clay, J., Demirel, H.O., Li, X., Sha, Z.: A Study on generative design reasoning and students' divergent and convergent thinking. J. Mech. Des. **146**(3), 031405 (2024)
19. Yin, R.K.: Case Study Research and Applications: Design and Methods, 6th edn. SAGE Publications, Incorporated, Thousand Oaks (2017)
20. Jiang, R., Ding, X., Xie, C.: Solarize your world: addressing climate change through renewable energy engineering. Phys. Teach. **61**(8), 694–698 (2023)
21. Jabi, W.: Parametric Design for Architecture. Hachette, UK (2013)
22. McKnight, M. Generative design: what it is? How is it being used? Why it's a game changer. KnE Eng. **2**(2), 176–181 (2017)
23. Atman, C.J., Bursic, K.M.: Verbal protocol analysis as a method to document engineering student design processes. J. Eng. Educ. **87**(2), 121–132 (1998). https://doi.org/10.1002/j.2168-9830.1998.tb00332.x
24. Murray, J.K., Studer, J.A., Daly, S.R., McKilligan, S., Seifert, C.M.: Design by taking perspectives: how engineers explore problems. J. Eng. Educ. **108**(2), 248–275 (2019)
25. Dym, C.L., Agogino, A.M., Eris, O., Frey, D.D., Leifer, L.J.: Engineering design thinking, teaching, and learning. J. Eng. Educ. **94**(1), 103–120 (2005)
26. Luo, Y., Becker, K., Gero, J., Alarcon, I.V., Lawanto, O.: Systems thinking in engineering design: differences in expert vs. novice. Int. J. Eng. Educ. **37**(5), 1398–1413 (2021)
27. Rahman, M.H., Xie, C., Sha, Z.: Predicting sequential design decisions using the function-behavior-structure design process model and recurrent neural networks. J. Mech. Des. **143**(8), 081706 (2021)