

Computational Design Tools and Education: The Smartgeometry Case

Öykü ACICAN

Middle East Technical University
oyku.acican@metu.edu.tr

İpek GÜRSEL DİNO

Middle East Technical University
ipekg@metu.edu.tr

doi: 10.21606/learnxdesign.2019.06048

Abstract: Computational design has brought in novel concepts to architecture and design disciplines. Computational design thinking has evolved due to the potentials of contemporary tools and methods. Experiential learning environments such as computational design workshops offer strategies for a better understanding of the contemporary needs of the computational design education. Smartgeometry (SG) is a computational design organization that operates through workshops of interdisciplinary teams. SG uses and teaches the state-of-the-art computational design tools and methods. Instead of teaching the novel computational design tools in an instructive manner, SG workshops focus on using the potentials of these tools through personal discovery and experimentation. Besides enabling responsive design outputs, tools for sensing, computing and materializing lead to various learning strategies such as learning-by-doing, interdisciplinary collaboration and community building by democratization. This study aims to unravel the impacts of the novel computational design tools and strategies on computational design education through an in-depth qualitative analysis of the SG workshops.

Keywords: *computational design education; tools; interdisciplinarity; smartgeometry*

1 Introduction

Computational design has been shaped by the synthesis of interdisciplinary knowledge, tools and methods. The intellectual foundation for the nature of computational design rests at a particular confluence of domains in fields ranging from mathematics, computer science and systems science to biology and philosophy (Menges & Ahlquist, 2011). Computational design has a critical influence on complex design problems in contemporary architecture. With the development of design tools and methods by the new technologies, architects' and designers' capabilities have been extended immensely (Peters & Peters, 2013).

Scripting is a critical skill for computational design. It is argued that design schools should ensure that all students emerge with sufficient scripting experience as scripting is an essential component of 21st-century design education (Burry, 2013). Scripting and tool-making are becoming essential forms of knowledge that have the potential of becoming the core knowledge, in research, education and practice of design (Oxman, 2017), not only as a technical ability, but also for deeper social aspects such as encouraging participation, supporting interdisciplinary collaboration



This work is licensed under a Creative Commons Attribution-NonCommercial-Share Alike 4.0 International License. <https://creativecommons.org/licenses/by-nc-sa/4.0/>

and communication. Therefore, computational design courses that are realized through coding, designing and fabricating should be placed into primary focus, instead of marginalizing them in the design education curriculum.

Design studio is located at the core of the architectural design education (Schön, 1985). In contrast to instructive learning, which is based on explicit guidance, design studios enable experiential learning, or learning-by-doing. Architecture students bring their knowledge from different areas to work on specific design tasks. The design studio offers the potential to provide a multi-layered and enriching learning experience, where the ideas, positions and artefacts are actively realized rather than simply being described (Nicol & Pilling, 2000; McQuillan, 2005). This paper argues that computational design education can take the studio teaching practices as a model.

Innovative tools and methods are needed for certain strategies of computational design education. Methods such as collaborative working, interdisciplinary knowledge sharing and experimenting, and environments that encourage these methods can help students achieve better computational design skills. Besides, there are pedagogical advantages of using computational tools in project-based learning. Project-based learning is an innovative approach of learning through inquiry and working collaboratively to conduct research and create projects that reflect students' knowledge (Bell, 2010). In project-based learning, authentic explorations are directed with the creative use of technology and tools. These educational strategies can be observed in computational design education as well.

In 2001, the *Smartgeometry* organization was established to integrate the contemporary advancements in technology into architecture through computational design by experimenting with tools and methods of various disciplines. In 2001, the main challenge of computational design in applied architecture was the modelling of complex geometrical forms, which gave its name to the organization. Over the years, Smartgeometry (SG) has evolved into a large community that biannually organizes an event consisting of conferences and workshops. Currently, each SG event hosts 10 workshops of innovative research projects. The workshops are called *clusters* that are organized by the *cluster champions* who are collaborative teams from academia and practice.

This paper aims to discover the impacts of the computational design tools, methods and strategies on the learning processes through a case study of the SG workshops. SG was selected to be studied because the SG workshops are interactive learning environments that can act as a model for computational design education. SG workshops are conducted by interdisciplinary collaborative teams with the state-of-the-art computational design tools. This workshop environment can help characterize the suitable conditions for creativity and innovation, and therefore has significance for the computational design education.

2 Methodology

For this research, SG workshops were investigated through a case study using the grounded theory method. Grounded theory is used to develop a theoretical understanding of an abstract analytical schema of a process (Creswell, 2007). SG workshops were analysed from multiple data sources, and the impacts of tools and the strategies on the computational design education were searched. Data sources include textual documentation and archival video records of the past SG workshops from the official website and related books and interviews with computational design experts from SG organization and first-person observation of the SG 2018 event in Toronto.

2.1 Data Collection

During data collection, two types of information sources were used. The first is empirical, in that information is gathered through participation to the SG 2018 event in Toronto. Semi-structured interviews with ten computational design experts that are involved in SG as directors, cluster champions and cluster participants were conducted. Interviews lasted 15-30 minutes, and the responses were audio-recorded, transcribed and open-coded. The questions aimed to understand the relationship between interdisciplinarity and innovations. In this research, the observations and the participant responses were interpreted within the framework of the learning practices of SG. Information on the interviewees and their experience on the related fields can be found in Table 1. The questions that were directed are as follows:

1. To what extent does interdisciplinarity have an influence on the SG workshops?
2. What is the relationship between interdisciplinarity and innovation in SG?
3. What are the potentials and impacts of the SG innovation on architecture?
4. Can you give an example of an innovation that SG has brought?
5. What are the critical concepts that are most important for a SG workshop?

The second data source is existing documentation on SG, which includes the textual documents on SG workshops and archival web-based records. Archival records are those compiled by SG, and include textual descriptions and audio-visual sources on the workshop content. Workshop objectives, processes, methods and information about the tool developments are presented in these sources by the cluster champions and participants. All the existing workshop video recordings which include visual and audio data from the website of SG (www.smartgeometry.org) were transcribed for data analysis. The details of the video recordings from the SG 2018: Machine Minds can be found in Table 2.

Table 1. Information about the Interviewees

Code	Role	Specialization	Experience (Years)	Duration of Interview(min)
I-1	SG Director	Computational Design Visual Analytics	37	15.14
I-2	SG Director	Computational Design Mathematics	30	19.39
I-3	Cluster Champion	Architectural Design Computer Science	6	09.46
I-4	Cluster Champion	Robotics Computational Design	10	14.25
I-5	Cluster Champion	Building Performance Chemistry	16	05.45
I-6	Cluster Champion	Parametric Design Collaborative Design	18	10.53
I-7	Cluster Champion	Design Computation Cybernetics	10	09.40
I-8	Cluster Champion	Architectural Design Machine Learning	10	08.58
I-9	Cluster Participant	Interaction Design Spatial Cognition	26	13.36
I-10	Cluster Champion	Computational Design Athletic Footwear	15	60.01

Table 2. Information about the video recordings of the workshops from SG 2018: Machine Minds from SG website

Year	Event Name	Number	Workshop Name	Theme	Duration of Video(min)
2018	Machine Minds	1	AI strategies for space frame design	Structural Exploration	04.45
		2	Behavioral enviro[NN]ments	Adaptive space design	04.25
		3	Data mining the city	Innovative visualization	05.06
		4	Fibrous timber joints	Structural Exploration	05.00
		5	Fresh eyes	Machine Learning	04.39
		6	Inside the black box	Innovative visualization	04.22
		7	Materials as probes	Material research	04.52
		8	Mind ex machina	Robotic exploration	05.16
		9	Soft office	Robotic exploration	05.03
		10	Sound and signal	Acoustic exploration	05.08

2.2 Data Analysis

After data collection, categories and subcategories regarding the computational design processes were identified using an interpretative reading. The textual documents, video recordings, interview data and observations were compared and classified. Data is used to derive new concepts and themes through an explanatory reading.

3 Findings

From the data analyses and the open-coding of the interviews, it was found that the SG workshops have various strategies that affect both innovation and the learning process of the participants during the processes of tool making,

using and sharing. As opposed to instructive learning, SG workshops offer an experiential setting, where the learning process is accelerated by close interdisciplinary collaboration, accessibility to various computational design tools and equal opportunity between participants. The qualitative analysis about the SG event and its potentials for a computational design education is discussed around four main topics in this research, which are *tools, discovery and experimentation, interdisciplinary collaboration, and democratization*.

3.1 Tools

Findings from the interviews support that tools and tool making is very critical for the SG workshops and the SG community. Computational design tools for generating, visualizing, optimizing, sensing and materializing play a critical role for design research, experiential learning and interdisciplinary working. Participants in SG learn specific uses of computational tools in an innovative way for the research of their workshop. Tool learning establishes a *scaffolding* for the participants. In education, scaffolding is the assistance or guidance provided by parents or teachers, that aim to extend students' current knowledge and skills towards independent problem-solving (Belland, 2017). Scaffolding during learning requires that the instructors offer students temporary support, increasing their skill acquisition and comprehension, which are essential to complete certain tasks. The high level of tool competency necessary for computational design calls for certain degrees of instructional support that are gradually eliminated as students feel competent and ready to take the responsibility to independently complete creative design tasks. SG workshops, similarly, invest in both instructional learning through explicit guidance for tool usage and exploratory learning by allowing the participants to apply their computational skills and knowledge on creative design problems, further enhancing learning-by-doing. Although the exact share between instruction and exploration varies from one workshop to another, it can be argued that these two learning approaches critically complement each other especially for novice designers.

The tools explored and developed in SG vary with respect to the theme of the workshop or cluster. A wide range of topics have been covered, from structural form-finding (i.e. gridshell structures, agent-based structural design, design with physics engines, adaptive structural skins, form-active structures), material experimentation (composites, ceramics, bricks), parametric design optimization, data-informed design strategies (i.e. acoustical, urban, thermal, humidity data), data/form visualization, robotic fabrication/assembly, machine learning methods, and computer vision. Tools have a considerable part in *experimentation, discovery, interdisciplinary collaboration and democratization*. Firstly, tools initiate the process of discovery. Tools and the development of further tools are enforced in SG because they give better assets for exploring design in different ways (Interviewee 2). Moreover, building the right tools is the first step for discoveries, inventions and innovation (Interviewee 9). Secondly, these discoveries must be achieved by interdisciplinary collaborations, adapting tools and techniques from each other. Interdisciplinarity engages individuals with tools from other fields, where synergies are formed between techniques, methods, materials, borrowed from other disciplines (Interviewee 5). Thirdly, producing a community of tool-makers accelerates the communication which results in a democratized learning environment. Interviewee 2 supports that democratization has been achieved by the accessibility of computational design tools. From these findings, it can be stated that tools act as catalysers that bring computational design learning strategies together.

3.1.1 Responsiveness

Over the years, design products that interact with their environment have gained importance for SG (Interviewees 2&10). Sensing tools such as thermal cameras, heat sensors, light sensors and tracking tools for data collection have become more accessible, and are increasingly being used by the SG workshops. These tools allow the participants to be informed by their surroundings and learn different ways in which the collected data can be integrated into the design processes. The SG workshops that place responsiveness and interactivity to their centre provide new ways of interaction with the design product, which is achieved by data collection, data-based design and feedback loops.

Interviewees 2 and 10 argued that the significance of *context*, which means integrating the surrounding environmental and social aspects to the design, has increased in SG over time. Interviewee 2 explained that the dominant agenda of SG during its early years was to manipulate geometry, whereas a critical shift of focus took place recently, which involved the integration of environmental data, such as air, light, sound, atmosphere, humidity, behavior of inhabitants and urban flows. This can be attributed to a certain level of maturity that computational design has reached, which expanded its attention from mere form-finding to other factors that can inform design for both synthesis and analysis. For instance, in the workshop *Micro Synergetics* from the year 2012, when a user touches the responsive modules, light sensors sense the proximity of the users and through feedback, kinetic modules move (Micro Synergetics, 2012). In addition, the objective of the workshop titled *Sensory Detectives* from SG 2016 is to form a thermal environment that physically simulate the dynamics of heat, moisture and air within a modular pavilion that involves electronic sensors and augmented reality (Sensory Detectives, 2016). Another reason for the focus on

responsiveness has been the increased accessibility of the sensing tools and the ease of interoperability between data collectors and data processors, according to interviewee 10.

Responsiveness has advantages for the learning process, such as increasing the participants' awareness of the environment and their feeling of responsibility towards the context. Computational design has been long charged for being self-indulgent and stylistically driven, due to its initial fascination with complex geometries during the early 2000's (Agkathidis, 2015). While computational form-finding continues to be a fundamental issue for architecture and design, responsiveness in design has great potential for performance-based architectural design in the way it considers environmental data as an inseparable part of building performative requirements such as the design's environmental footprint, costs or occupant comfort.

3.1.2 Materialization

The tools for materializing in SG are usually technologies adapted from the industry and used during design and design research. Materializing design alternatives rapidly in small scales is made possible by the rapid fabrication tools provided by SG. The innovative use of tools for materializing has multiple advantages, from producing complex forms rapidly to grasping complex concepts of computational design. Materializing is supported by experiential education, constructionism, and critical pedagogy (Blikstein, 2013). Materializing tools and digital fabrication allow *learning by doing* in computational design education.

The creative use of materialization tools in the workshops enable abstract computational design concepts to be better learned by the participants. For instance, for the workshop titled *Non-Linear Systems Biology and Design* in SG 2010, the aim was to mimic the tissue formation by materializing complex and custom shaped modules and connecting them with cables. From similar past SG workshops, it was observed that participants had the chance to understand and internalize complex concepts such as emergence, automation and stigmergy by mimicking them with the creative uses of materialization tools.

Learning by doing is enabled in common practice in design studios. Materializing is a core aspect of computational design, therefore tools and methods for innovative processes of digital fabrication and rapid prototyping should be used by students for a better understanding of contemporary concepts and the translation from the virtual to the material by computational means. Creating such understanding is critical for computational design education and it can be achieved by involving projects where materialization is prioritized in a studio environment as SG.

3.2 Discovery and Experimentation

It has been widely argued that the pace of adopting new technologies for the building industry is slower than other industries. The large scale of the designed products, the high-risk associated with complex buildings and their permanence in time are considered as determinant factors in the avoidance of design experimentation. On the other hand, technological advancements show an instant impact on the small scaled design products. For example, a discovery in material science can be directly used in the design of a phone case or a sportswear product (Interviewee 10).

The avoidance of high-risk experiments in the practice of architecture does not necessarily hold true on architectural design education. Design education, as argued by Callicott and Sheil (2000) seeks out the unfamiliar, the unconventional and the methods of other disciplines. According to interviewee 7, SG carries low risk professionally and is a suitable environment for experimentations that lead to new discoveries. For instance, the objective of the workshop titled *Mind ex machina* from SG 2018 is to experiment with robots through customized tasks to discover methods for improving human-robot interaction during the design processes (*Mind ex machina*, 2018).

SG is a laboratory for exposing novel research problems with a small audience and experimenting collaboratively without knowing the outcome in advance (Interviewee 8). There is always the factor of curiosity of the un-known, although the tools, methods and research questions have been present before the workshops. Experimenting enables students to acquire their own experience and to synthesize this experience creatively during the design process (Willey, 2005). These findings indicate that design laboratories as SG are suitable environments for experimentation and discoveries in computational design education.

3.3 Democratization

SG is an environment where interdisciplinary groups are required to creatively solve complex design problems. Therefore, collaboration between all workshop constituents, including the participants, champions and even tools is

crucial. Equality between the participants and their accessibility to people and tools are important factors for the SG workshops. Equality enables respect of others and the confidence to contribute to the group work (Potts, 2000). In SG, because the participants are considered as equals, an environment based on respect and confidence can be achieved (Interviewee 8).

In the context of higher education, it is argued that the misdistribution of power between students and professors should be balanced in order to democratize the studio, which can encourage students to take on the primary responsibility (Dutton, 1987). SG workshops offer equal opportunity to its participants to contribute to the group work without a social or disciplinary hierarchy, and this improves communication. Since true dialog takes place only among equals, hierarchy in design education precludes the possibility of true dialog (Dutton, 1987). During the event, it was observed that the collaborative champion-participant relationship of the SG workshops has less hierarchy than the professor-student relationship of a design studio. This provides a better dialog between the contributors to the workshop, increases the responsibility of the participants, and improves the learning process (Interviewee 4).

The open plan layout of the physical environment within which SG takes place, points to the importance of interaction between different workshops. The visual and physical interactions between the participants from different workshops help increase their awareness of various design problems, tools and methods as well as design proposals. It was observed that participants are encouraged to visit all the workshops and be informed about other research processes that are happening simultaneously. Therefore, sharing of knowledge does not only take place in the workshops but also between workshops. Interviewee 8 contrasts this situation with certain design firms that do not allow any communication of their inventions with the outside world, which is limiting for the designers. SG is the opposite of such privacy, where sharing and learning are not limited. This accessibility of information contributes to the democratized environment of SG.

Interviewees concur that the computational design tools engaged a community of computational designers, and networking has gained importance. Due to the increased importance placed upon sharing and networking, computational tools have become more accessible. Sharing and community support has become the norm, which has major effects on the new generation of computational designers. When asked about the innovation that SG has brought, Interviewee 8 clearly stated that the SG community created a socialist environment where everybody is at the same level, and everybody can share any kind of tool and information. With this community, learning extends beyond the limits of the classroom towards a network of people exchanging knowledge and sharing their ideas and code. The interviewees agree that the creation of such a community and democratization of the new computational tools is one of the most significant impact that SG has brought to the computational design education.

3.4 Interdisciplinary Collaboration

Interdisciplinary collaboration has been identified as one of the most critical factors for computational design education and for SG. The gathering of people from different educational backgrounds who have different perspectives, ways of thinking and processes for developing methodologies, make them un-learn their existing knowledge, and start learning from each other (Interviewee 10). The interdisciplinary nature of SG enables the participants to learn more by sharing their individual skills and knowledge with the other participants (Interviewee 7). As such, disciplinary and personal competencies, knowledge and skills can be brought together, complementing each other towards interdisciplinary design, which is critical for computational design education.

Interdisciplinary collaboration is widely associated with creativity. The heterogeneous set of skills and experiences of members of interdisciplinary teams give rise to an enhanced capacity for creativity and novelty (Sutton & Hargadon, 1996). Interdisciplinary teams need to integrate knowledge from different disciplines to bring about innovative design outcomes. To integrate knowledge from different disciplines, complex design tasks should be set (Nicol & Pilling, 2000). Such complex design tasks are present in the SG workshops, and those tasks go beyond the limits of architecture. Interviewee 4 stresses that SG is an architectural event, but the workshops work on complex problems beyond the limits of architecture where interdisciplinary knowledge is critical. For instance, the workshop titled *Nano-Gyroids* from SG 2016 requires integrating knowledge from chemical crystallography and design (Nano-Gyroids, 2016).

In SG, participants can learn interdisciplinary collaboration, which can help them acquire a foundation in team working that will be beneficial in learning to communicate with specialists from other fields. Interdisciplinary collaboration accelerates the learning process by rapid sharing of knowledge, recognition of the limits of a single discipline, and approaching a problem from different perspectives (Interviewee 10). As computational design requires multiple perspectives, tools and methods, it is essential for computational design education to be interdisciplinary.

4 Conclusion

Strategies, environments and tools for computational design education is critical for architecture students to adapt to the technological advancements. The computational design processes and the impact of creative computational design tools in SG workshops were discussed in this research as a case for computational design education. The results of this study indicate that experimentations, interdisciplinary collaboration, democratization and tools have a critical role for learning computational design and enhancing computational design thinking. This study on SG has found that the aspects of computational design such as responsiveness and materialization are achieved by adapting, developing and customizing different tools for complex design problems by experimentation in a democratized environment.

This research also shows that SG workshops involve both instructive and experiential learning that support each other. This study concludes that computational design education should involve and integrate project-based learning and scaffolded instruction. Therefore, experimentation through complex design problems should be integrated in computational design education. SG also indicates that computational design cannot be taught around a single discipline as it is constituted with the involvement of different disciplines. The SG workshops, their creative proposals and innovative outcomes show that the education of computational design can be improved in an interdisciplinary and collaborative environment, where the tools are democratized and discovery through experimentations is emphasized.

References

- Agkathidis, A. (2015). *Generative Design*. London: Laurence King Publishing.
- Bell, S. (2010). Project-Based Learning for the 21st Century: Skills for the Future. *The Clearing House*, 83(2), 39-43. DOI: 10.1080/00098650903505415
- Belland, B.R. (2017). *Instructional Scaffolding in STEM Education: Strategies and Efficacy Evidence*. Springer Open.
- Blikstein, P. (2013). Digital fabrication and 'making' in education: The democratization of invention. *FabLabs: Of Machines, Makers and Inventors*, 4, 1-21.
- Burry, M. (2011). *Scripting Cultures: Architectural Design and Programming (AD Primers)*. West Sussex: John Wiley & Sons.
- Callicott, N., & Sheil, B. (2000). The Degree Laboratory: The work of Unit Six at the Bartlett School, University College London. In S. Pilling (Ed.), *Changing Architectural Education: Towards a new Professionalism* (pp. 60-71). London: RIBA Publications.
- Creswell, J. W. (2007). *Qualitative Inquiry & Research Design: Choosing among Five Approaches* (2nd edn.). Thousand Oaks, CA: Sage Publications.
- Dutton, T.A. (1987). Design and Studio Pedagogy. *Journal of Architectural Education*, 41(1), 16-25. DOI: 10.1080/10464883.1987.10758461
- Menges, A., & Ahlquist, S. (Eds.). (2011). *Computational Design Thinking (AD Reader)*. London: John Wiley & Sons.
- Micro Synergetics [Video file]. (2012). Retrieved December 18, 2018 from <https://www.smartgeometry.org/sg2012workshops>
- Mind ex machina [Video file]. (2018) Retrieved December 20, 2018 from <https://www.smartgeometry.org/sg2018workshops>
- McQuillan, T. (2005). Informed Architecture: Three Tensions. In E. Harder (Ed.), *Writings in Architectural Education* (Vol. 26). EAAE Prize.
- Nano Gyroids [Video file]. (2016). Retrieved December 25, 2018 from <https://www.smartgeometry.org/sg2016workshops>
- Nicol, D., & Pilling, S. (2000). Introduction. In S. Pilling (Ed.), *Changing Architectural Education: Towards a new Professionalism* (pp. 1-21). London: RIBA Publications.
- Non Linear Systems Biology and Design [Video file]. (2010). Retrieved December 20, 2018 from <https://www.smartgeometry.org/sg2010workshops>
- Oxman, R. (2017). Thinking difference: Theories and models of parametric design thinking. *Design Studies*, 52, 4-39. DOI: 10.1016/j.destud.2017.06.001
- Peters, B., & Peters, T. (2013). Introduction. In B. Peters & T. Peters (Eds.), *Inside Smartgeometry* (Vol. 01, AD Smart). (pp. 8-19). Chichester, UK: John Wiley & Sons.
- Potts, W. (2000). The Design Studio as a Vehicle to Change: The Portsmouth Model. In S. Pilling (Ed.), *Changing Architectural Education: Towards a new Professionalism* (pp. 208-216). London: RIBA Publications.
- Schön, D. (1985). *The Design Studio*. London: RIBA Publications.
- Sensory Detectives [Video file]. (2016). Retrieved December 25, 2018 from <https://www.smartgeometry.org/sg2016workshops>

Sutton, R.I., & Hargadon, A.B. (1996). Brainstorming groups in context: Effectiveness in a product design firm. *Administrative Science Quarterly*, 41, 685-718).

Willey, D.S. (2005). Architectural Education in the Digital Environment. In E. Harder (Ed.), *Writings in Architectural Education* (Vol. 26). EAAE Prize.

About the Authors

Öykü Acican has graduated from METU Department of Architecture in 2016. She is continuing her master's degree on computational design at the same department.

İpek Gürsel Dino is an Associate Professor at Middle East Technical University, Department of Architecture. Her research mainly focuses on computational design, performative architecture, parametric generative systems, virtual reality and BIM.