Legibility of Design in Collaborative Computational Practices

Livanur Erbil Altıntaş[1], Altuğ Kasalı[1], Fehmi Doğan[1]

[1] Izmir Institute of Technology, Turkey
livanurerbil@gmail.com
altugkasali@iyte.edu.tr
fehmidogan@iyte.edu.tr

Abstract. The paper reports key instances from our field observations involving computational design practices to understand the process of generating and assessing alternative design options. Following an ethnographic approach, we investigate the nature of interactions within the team, which we conceptualize as a distributed cognitive system. We have observed a persistent effort –mostly driven by team leaders- to make the design idea -and design process in general- more legible and transparent for individuals within the team as well as others including clients and consultants. Through situated observations of collaborative computational practices, we investigate how design ideas are represented and externalized in a distributed cognitive system with the intention of achieving a legible schema to guide the design process. We report our interpretations concerning the concept of legibility and its various dimensions which predominantly relate to the need to clarify and justify the core design drivers and approaches in form-finding.

Keywords: Legibility, Distributed Cognition, Design Collaboration.

1   Introduction

Computational design involves a form of logic to generate a series of acceptable design solutions (Coates, 2010). All assumptions to underlie formations within developing design need to be explicit to a certain level by sticking to explicit and comprehensible rules (Coates, 2010). Design, however, bears a level of ambiguity which is also claimed to be accommodated within computational practices (Burry, 2011). It is also a key concern that the logic of computation and related morphologies necessitate a level of clarity in parameters and rules to be employed. It is not our intention, at the outset, to contrast the situations of ambiguity and clarity in design representation. Rather,

following the accounts of the participants observed in this study, we frame it as an issue of legibility which has several key dimensions to be further studied.

Through a series of field observations, we focus on how design teams cope with the ambiguity that leads to creative solutions while constructing legible knowledge propagation in a distributed cognitive system. Distributed cognitive system comprises humans, objects, and tools following the work of Ed Hutchins (1995). Distributed cognition assumes that any task can be distributed across different parts of a system (Hutchins, 2014). In this research we have employed the distributed cognition framework in order to account for the interactions occurring between human and non-human components of a system. The paper tries to explore the mechanisms in developing particular solutions which required a level of coordination between participants of the cognitive system. According to the distributed cognitive system framework, 'embedded' individuals interact with artifacts, technologies, and tools to coordinate their internal cognitive tasks with external tools (Kirsh, 2008). Thus, tracking the generation, manipulation, and propagation of representations is critical in understanding the mechanisms within the system.

This study investigates and discusses the design process of two separate architecture offices where three teams of architects were tasked with generating a schema in early phases of architectural design primarily using computational design strategies. The intention of the design team leaders in the cases was to make the design idea -and design process in general- more legible and transparent for individuals within the team as well as others including clients and consultants. Through situated observations of these collaborative computational practices, we investigate how design ideas are represented in a distributed cognitive system with the intention of achieving a legible schema to guide the design process.

2   Method and the Cases

The research is a qualitative study which consists of ethnographic observations of professional architectural teams. Data collection included the following stages. First, we conducted in-situ observations of three competition projects from beginning to end for a month for each one. Ethnographic observations were employed to understand groups and people in their everyday professional lives (Emerson et al., 1995). Second, we conducted semi-structured interviews with significant team participants. The semi-structured interviews were conducted face-to-face to provide a way to explore feelings, opinions and behaviors (Sommer & Sommer, 1997). The interviews helped us develop an understanding of the teams' collaborative processes, the communication

strategies and knowledge representation techniques, and participants' original descriptions of certain situations within their operation.

Data analysis included the following three phases: description, analysis, and interpretation of the qualitative data offered by the culture-sharing group (Creswell, 2007). In the first phase, the data was indexed in a timeline to understand the phases of the team's design process. Data types were coded in the timeline as sketch, photograph, field notes, meeting minutes, video records, audio records, screenshots, and e-mails. In analysis, we have adopted the coding strategies offered by the Grounded Theory (Strauss & Corbin, 1990).

In this research, we observed two architectural design offices with three different teams (Team A, Team B and Team C) participating in three different architectural design competitions (C-A, C-B and C-C). Team A consisted of one architect team leader (TL1), one architect job captain who also operated as the coder to lead computational practices (JC1) and seven intern architects (IA#). Team B consisted of one architect team leader (TL1), one architect job captain (JC2) and one coder (C1). Team C consisted one team leader (TL2), and two coders (C2, C3) (Table 1).

Table 1. The observed projects, teams, and participants.

	Office 1		Office 2
Project	Competition A	Competition B	Competition C
Team	Team A	Team B	Team C
Team Leader	TL1	TL1	TL2
Job Captain	JC1	JC1	JC2
Coder	JC1	C1	C2, C3
Intern Arch.	IA1		

Source: Author, 2021

3   Legibility

Lionel March (2015) framed architecture as concretization of abstract mathematic statements. Expressing a design idea is possible in oral or written representations (Simon, 1969). Simon (1969) proposed a taxonomy of representing a design idea that have mathematical, spatial or procedural. Developing parametric design tools makes it possible to create representations of design ideas by all these methods as independently or as combined. Possibility of various representation options makes the design process more efficient and transforms the process into a more explorative one. In this explorative and participative setting, legibility is one of the main issues to create effective coordination among the members in the system.

As a reflection of this rule-based approach in design, we have observed our participants to come up with a legible continuum linking design intentions to an end-product. For instance, in a coordination meeting, the leader in team A (TL1) complained about the absence of a design rule yet to be legible for all:

00:10:44 TL1: …so what's the rule? The thing that bothers me is that I can't read it right away. Let me say, I can't read the distribution rule of these [elements]. Yesterday, everything was more legible; I mean in the design proposals...

To have a better understanding and coordination in a design process, the teams we observed needed to enhance their awareness within their systems. Moreover, to support collaborative work in such complex systems, transparency of tasks and operations are necessary. Each of the participants' high level of awareness is important for dynamic coordination of the tasks during the design process. Particularly, in distributed work environments, tasks are dynamically coordinated to monitor the progress. Within observed practices, the need for legibility was continuously emphasized as a key factor to sustain the dynamic decision-making tasks in design. To provide dynamism in a system, the participants' knowledge domains should be 'overlapped' (Hutchins, 1995) thus the collaborating team could easily follow the design process. In Team A and B, the team leader (TL1) has knowledge of computational design approaches so the team leader (TL1) and the coders (JC1 and C1) coordinated the design process dynamically. The design process was carried out by producing sketches to conduct communication among the team. In Team B, we have observed the team leader (TL1) and the coder (C1) kept the transparency in the sketching phases with clearly identified prescriptions (Fig.1). The team leader (TL1) was the one to produce sketches which establish the rule concerning a particular quality of a wall design.

On the sketches, we can identify the percentage values in numerical format to describe the effect of the openings on the surface of the wall. Following the recipe (Fig.1), then, it was the coder's (C2) turn to process the sketch by re-sketching the idea to come up with a formula on a parametric design software; namely Grasshopper (Fig.2). In the following stages, the team leader was periodically checking the progress on screen to make sure that the initial idea was getting transformed into an architectural component through mathematics and drawings. Thus, both the paper and the software were used simultaneously as the venues to translate a certain form of architectural knowledge (the wall) into mathematically represented one (percentage for gaps), then back into an architectural visual (perforated wall surface). The team leader (TL1) and the coder (C1) have overlapped knowledge domains, such as Mathematics so, the team leader could easily follow and coordinate the design process in Grasshopper.
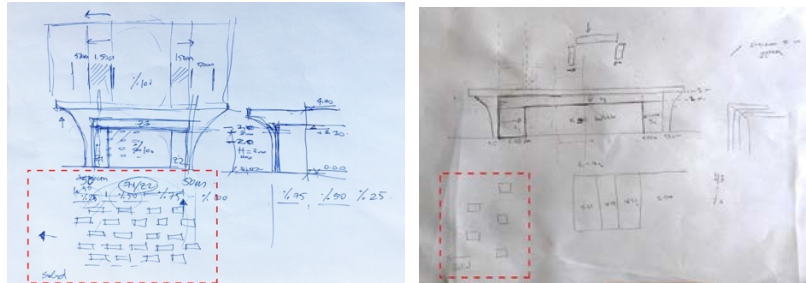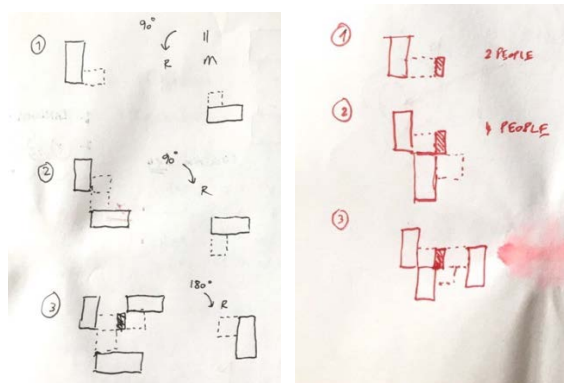
Figure 1. (left) Descriptive sketches developed by the team leader; (right) resketching the idea before parametric definition by the coder (C1).


Figure 2. (left) parametric definitions in Grasshopper developed by the coder (C1); (right) Last version of façade design. Drawn by the coder (C1).

We also observed a similar process within the work of Team A. The task, in this case, was to develop a site plan by generating a compositional rule to organize pre-determined lodging units. In the beginning of the design process, the coder (JC1) and the intern architect (IA1) developed alternatively two design solutions. Both alternatives were initiated by a work session with hand-drawn sketches (Fig.3, 5) which were then generated in a software; namely Rhino (Figure 2, 4).

Figure 3. Sketches developed by the coder (JC1) to prescribe the rules. Drawn by JC1.
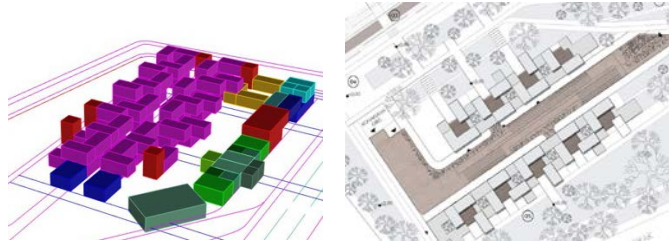
Figure 4. Sketches applied on Rhino Software. Drawn by JC1.
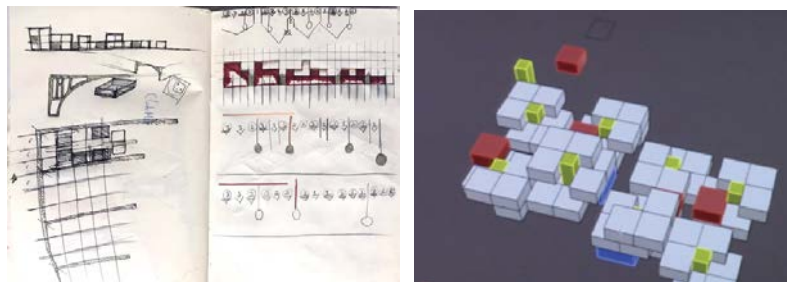
Figure 5. (left) 2D sketch drawings of the rule applied units design; (right)3D drawings of the alternatives by the intern architect. Drawn by IA1.

The coder (JC1) explained the start of the design idea through a 'mathematical logic' represented in hand sketches. Following the logic, the coder (JC1) applies a 'derivation method' and repeats the logic according to the method in Grasshopper. In the following figure (Fig.6), the relation of logic and product is illustrated:
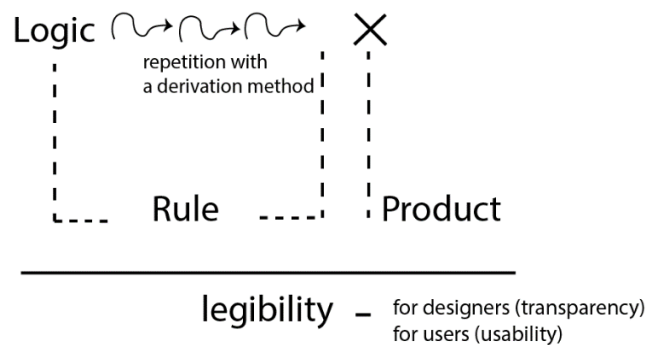
Figure 6. An illustration of legibility in a rule-based design. Drawn by the author.

The illustration (Fig.6) expresses the design process and the legibility through a rule-based logic. The main idea is that a designer creates a logic to

apply to design solution. Then, the logic is repeated according to a derivation method. The design is created following the derivation method. All over the process is called as a 'rule' by the team participants. The aim of having a rule is to create a legible design solution for designers who evaluates in competition jury and for users to create useful spaces.

The common feature in the instances presented above was the intention to make the design evolution legible as the team progressed to develop an end-product. The participants were motivated to be explicit and legible concerning creation of a rule to guide the formal evolution. One of our participants (JC1) explained the reason of creating a rule to have a unity in form:

"Originally the goal is to create all the rules on the smallest building block, and then generate that unit, and get the overall form. Why do we do this? First, to generate a 'regular' form. Randomly generating one form, each different from one another, means that it does not have a base unit. The mass formed at that point is a unit in itself. However, in each of our design work, we originally aimed to produce an overall mass - with a repeating unit- subject to the rules. Second, this is a competition; we have a very short time. Actually, in all projects, time is very short and fully producing and generating a unit has always made our work easier; which is actually a kind of mass customization, which has a counterpart in real life. The whole, made up of units following a rule, relieves the entire process from design to manufacturing." [manuscript by the coder (JC1) on e-mail at Jan 3, 2019, 10:34 am]

In the presented cases, the team leader (TL1) and the coders (JC1, C1) worked in collaboration from the very initial steps of the design process. Each of the participant aimed to express their ideas freely and expressively. The team was motivated to apply computational design methods while creating design idea. Thus, mathematical representations, including geometrical organizations, were simplified complex design ideas. Each of the design idea was legible for the team to criticize and to develop it further. Creating geometric shapes in a rule-based system and sketching design idea as schematic with numerical explanations might improve the legibility in the team.
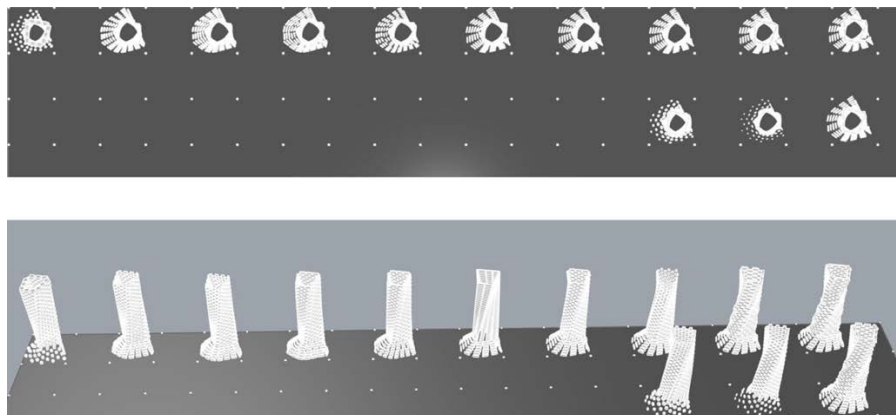
4   Design Evolution

Architectural design is an exploratory process and design itself evolves through the process. Designers try to create descriptions of their intentions with regard to design solutions through different representations. Descriptions of design ideas are necessary for those who formulate the ideas, and for the team participants. Expressing the ideas' evolution was one of the strategies of the observed teams. Evolution in nature is based on two different mechanisms: (i)

transmitting genetic information from one generation to the next; and (ii) natural selection from a proposal pools of population (Hybs & Gero, 1992). After developing multiple but unsuccessful proposals, we have observed the team leader (TL2) emphasizing the idea of design evolution through various images which were the illustrations of an evolutionary processes for a vehicle design evolved from biological forms. Then the team leader explains what is the meaning of evolution in design for them:

00:00:52 TL2: … show the evolution of it! you should say "here it is!", then you jump to this, then you jump to that, then you jump to that, there is no such thing here, so the cause-effect relationship is broken!

The coders (C2, C3) created multiple alternatives but were not successful for the team leader (TL2) due to the absence of a legible evolutionary process. Eventually, one of the processes have led to a proposal (Fig.7) that was found as successful in the way of demonstrating the evolutionary process of the form. The team leader was evaluating the evolutionary process as a logical chain that kept the main idea similar to the genetic information transmitted in nature.

Figure 7. Evolutionary process of the proposal. Drawn by C3.

The team leader's (TL2) main emphasis was on a form of "cause-effect" relationship in the design solution. In this "cause-effect" relationship, TL2 was seeking the logic of the idea as a process that was clearly illustrated. Thus, the design solution could be legible for themselves and the other designers.

5   Discussions and Conclusions

Legibility is a term used in urban design and it is defined as the ability of an environment to be organized in such a way that it can form a recognizable and coherent pattern (Lynch, 1960). The main aim of a designing a city as legible is to create a recognizable path for the users who experience the city from entrance to the end as a journey. In a similar manner, we observe how the idea of legibility come into play as the individual or collective design work must be legible in every step of the design journey. The rationale behind being legible in a distributed team dwells on the justification of design intentions for oneself and for the other teammates. Legibility is necessary in every level of architectural design process, such as, understanding existing situation of the design area, understanding of design idea in the design process, clarity among team participants, communication between the team and external parties in design process and in construction process, and legibility of the design idea for the users.

This research establishes the significance of legibility in the sense of knowledge propagation within/among designer/s in the design process. Pursuant to observations and interviews, the findings demonstrate the concept of legibility in design as a pivotal issue in the collaborative computational design practice. As offered by the qualitative analysis of observational notes and interviews, the term legibility emerged mainly around two routes which occasionally overlap in design discussions: (1) Concerning clarity in communication and articulation of design intentions, legibility was valued to make the developing morphology understandable to others; (2) legibility was also equally valued when designers want to explain and represent the logic of computational operations that govern the formal evolution of design. Briefly, the former involves reasoning about the formal qualities of the architectural product, whereas the latter is about making the formal evolution within the process legible to third parties. We suggest that the motivation concerning legibility relates to the need to clarify and justify the core intentions and the protocol followed by designers.

Design is a creative process and the relationship between creativity and idea diversity is emphasized (Guilford, 1973). Diversity of ideas is related with ambiguity. Parametric design creates idea diversity which shows the tolerance of ambiguity on the design idea (Mahmoud et al., 2020). In the observations, the teams' preference was mainly parametric design which supported idea diversity. In design teams, individuals seek and provide legibility on ideas and process to provide communication within an ambiguous setting. Thus, the team participants can carry out the process as legible and transparent. To be legible, the team participants used the sketch tool as idea generator and translator. The team participants aimed to be more cognizable by making sketches either to understand their own thoughts, or to transfer them to the other participants. For

example, in the sketches produced by the coder, the main idea was conveyed on a paper, and then the parametric design tool was used to produce different alternatives in line with that main idea. Thus, initial sketches become a legible guide for the coder. On the other hand, the team leader developed a sketch to express the idea to the coder. Then, the coder re-sketched all of what he heard during the drawing phase from the team leader and what he saw on the sketch developed by the team leader. Re-sketching was appeared as a legibility tool. Coder performed the re-sketching step as a synthesis phase of the initial sketch, thus he outlined Grasshopper definition. The following illustration (Fig.8) represents the design process accordingly the creation of legibleness space in the distributed work.
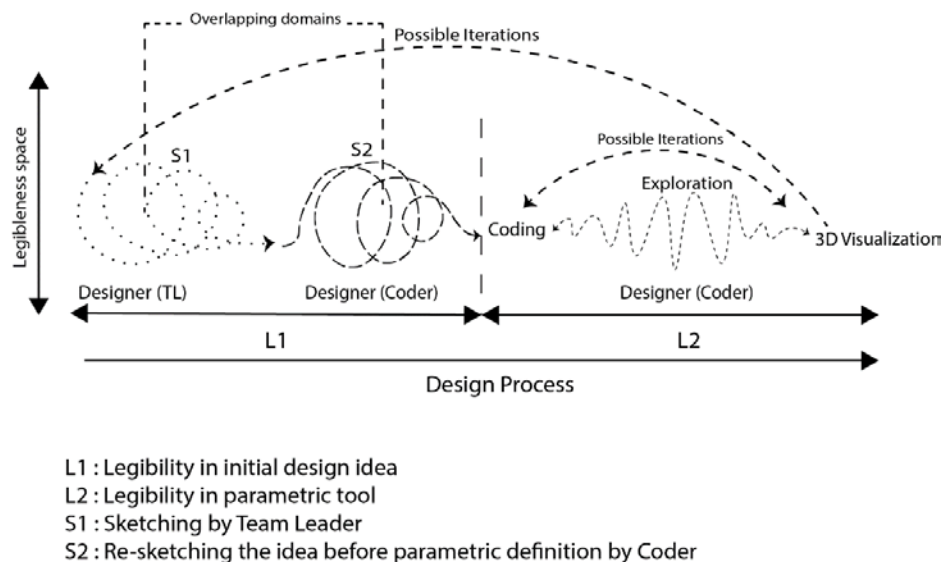
Figure 8: An illustration of legibleness space in design process. Drawn by the author.

The importance of distribution of a cognitive process in a team and the interaction among them are emphasized (Hutchins, 1995). The interaction, by its nature, needs to be explicit as much as possible in the complex nature of the design process. On the other hand, a research emphasizes if a team participants are not motivated to be explicit, the result of the production is more creative than the team which is motivated to be explicit and creative (Sawyer, 2017). Sawyer (2017) emphasize improvisation in the interaction of the teams which leads to unpredictable and unexpected solutions which are innovative. Therefore, in this double-edged sword, design teams need to be both creative and legible in a distributed cognitive system. In an improvised communication, team participants need to be legible to conduct a fluid process. In architectural design process, improvision might be occurring through verbal and visual

representations such as sketching. In computational design, the ability of having idea diversity might support improvision in the team through multiplicity.

References

Creswell, J. (2007). *Qualitative Inquiry and Research Design: Choosing Among Five Approaches.* Sage.

Emerson, R. M., Fretz, R. I., & Shaw, L. L. (1995). *Writing Ethnographic Fieldnotes.* The University Of Chicago Press.

Guilford, J. P. (1973). *Characteristics Of Creativity.* Illinois State Office of the Superintendent of Public Instruction, Gifted Children Section.

Hutchins, E. (1995). *Cognition in the Wild.* MIT press.

Hutchins, E. (2014). The cultural ecosystem of human cognition. *Philosophical Psychology*, *27*(1), 34-49.

Hybs, I., & Gero, J. S. (1992). An evolutionary process model of design. *Design Studies*, *13*(3), 273-290.

Kirsh, D. (2008). Distributed cognition: A methodological note. In I. E. Dror & S. Harnad (Eds.), *Cognition Distributed: How cognitive technology extends our minds.* John Benjamins Publishing.

Mahmoud, N. E., Kamel, S. K., & Hamza, T. S. (2020). The Relationship Between Tolerance Of Ambiguity and Creativity in Architectural Design Studio. *Creativity Studies*, *13*(1), 179-198.

March, L. (2015). Mathematics and Architecture Since 1960. In O. M. Williams K. (Ed.), *Architecture and Mathematics from Antiquity to the Future* (pp. 553-578). Birkhäuser, Cham.

Sawyer, K. (2017). *Group Genius: The Creative Power of Collaboration.* Basic Books.

Simon, H. A. (1969). *The Sciences of the Artificial.* MIT Press.

Sommer, B., & Sommer, R. (1997). Interviews. In B. Sommer & R. Sommer (Eds.), *A Practical Guide to Behavioral Research: Tools and Techniques* (Fourth ed.). Oxford University Press.

Strauss, A., & Corbin, J. (1990). *Basics of qualitative research: Grounded theory procedures and techniques.* Sage.