Lesson 1 in Design Computing Does not Have to be with Computers

Basic design exercises, exercises in visual computing

Mine Özkar

Middle East Technical University, Turkey
http://www.archweb.metu.edu.tr

Abstract. This paper proposes basic design instruction as a possible setting for introducing the concept of design computing to architecture students in the first year of their education. The argument is based on two viewpoints. The first extends the positivistic understanding of computation to visual reasoning in general. The second recognizes visual design activity as a reasoning process. The author suggests that, in order for computation to be embedded in the ways of design thinking, an understanding of it needs to be built-in, both as concept and as practice, into learning about the design process. The study draws attention particularly to basic design instruction as a setting for design and computation to be learnt concurrently. Examples from the first year curriculum at Middle East Technical University are given to illustrate the discussion.

Keywords. Basic design: computation; design reasoning; design thinking.

Introduction: the “computation” in design reasoning

Currently in many architecture schools around the world, the integration of computation with design curricula is shaped by how able the educational institutions are in providing the infrastructure for it, and how much the educators incorporate it within their visions of architectural design. The educators’ interests are growingly in redesigning curricula, and with an understanding to integrate computational tools to the general discourse of architecture and its education instead of adding them on in separate courses. (Kvan et al., 2004)

This study too emerges with an interest in redesigning a curriculum, and focuses on integrating computation in the fundamental first year of architecture education. It grows from a particular motivation to uphold the “reflective practice,” (Schön, 1991) and from Nigel Cross’s call for a discipline of design that is separate from the “positivist, technical-rationality basis of The Sciences of the Artificial.” (Cross, 2001) It also concurs with the view that computational tools will be developed to adapt to designers in a “slow and incremental revolution” instead of designers going through a radical shift in their thought processes and roles. (Johnson, 2002) These stances seem especially valid if design computing is to be carefully and thoughtfully integrated with long-lasting pedagogical traditions.
of architecture education. More than to the application of the tools, this study thus gives weight to inquiring, integrating and adapting their underlying concepts to design.

In order to maintain the reflective practice in the integration with computing, it is necessary to reconsider the concepts and the definition of computation in terms of design reasoning. Stiny (2001) recognizes computing as a way of thought, reasoning, beyond its mechanical usability in technological tools. More importantly, he extends its definition from the positivistic understanding of computation to visual calculating, which then can relate to non-mechanical thought processes such as design.

Stiny’s standpoint attempts to abridge a fundamental detachment at the structural level between computing and design reasoning. Design reasoning utilizes the senses just as much as learnt abstractions, and demands a more ambiguous space, where definitions are conditional, and the structuring is possible when necessary. Stiny suggests that we compute accordingly. Such a conception of computing is significantly easing its integration into the culture of design. It should be introduced to the culture as early as in the first year of architecture education, and concurrently with design reasoning.

“Computation” for first year architecture students

There is already interest in proposing new curricula to introduce digital design media to the first year curriculum. (Mark et al., 2001) Many scholars have already initiated discussions on fundamental issues ranging from how to handle the varying backgrounds of first year students to how to adapt architectural computing to early stages of education. (Mark et al., 2003) Now we can also take steps beyond teaching CAAD, and towards infusing computation in the broader conceptual sense discussed above. An introductory curriculum known as basic design instruction seems to provide the needed environment for exposing the “computation” within design reasoning.

Basic design instruction

Basic design education is practiced today as an introductory and formative course in a number of architecture schools around the world. It is primarily based on developing the student’s understanding of how one reasons in design. The student is encouraged to understand, question, and create relations between various objects. In the exercises, the creative process comes across as reasoning that explores uncertainties and redefines constraints with continuous interaction with materials. As Stiny recognizes it, and has shown to an extent in Kindergarten exercises (1980), this is design computing. The educators’ efforts should be in making this perceptible to the student.

Within the advantages of working with abstract forms, visual design rules can be isolated, and thus easily observed, while their variations are tried. (Özkar, 2004) Basic design assignments are petite design exercises with abstract forms that challenge the students with a small number of variables. Devices such as visual rules help recognize these variables a bit more consciously. Figures 1 and 2 illustrate two visual schemas employed in examples by two educators. (Dow, 1900; Sausmarez, 1964). The four alternatives in the first set emerge by the application of the schema shown to the right. The schema divides the square frame perpendicularly into two parts. The four alternatives in the second set emerge from the other schema, which divides the square frame perpendicularly into two equal parts. Representing and comparing the two schemas helps us see that the second is included in the first one. Hence, we understand that one is a specified version of the other despite the great differences in the results.

Another incentive to utilize the concept of visual rules and schemas to study design is coming from past attempts. Visual rules and schemas have
been employed, with specific reference to the application of shape grammars, in upper level design studios at various schools. (Knight, 1999) In studios based on given grammars, G. Celani (2004) observes that students misconceive the grammar as formally rigid. Similarly, students in studios where they are asked to create their own rules and grammars often do not realize that rules are tools at their command; rules come across as idealizations. The pedagogy involved must afford the students deeper inquiry into how spatial relations get established and how rules work. Celani draws attention to the need to “prepare” novice students for the use of shape grammars in studios. Pursuing first year students to think about their processes with visual rules and schemas seems worthwhile as a follow up to these studies also.

A few basic design exercises, recently practiced at Middle East Technical University (METU), the institution where the author is teaching, are described below to illustrate a potential in terms of familiarizing the students with the concept of computing through their own design processes.

Architecture Curriculum in METU

ARCH 101 Basic Design Studio is the foundational course of the curriculum in METU Department of Architecture. It is the only full studio in the first semester, and the precursor to the second semester ARCH 102 Introduction to Architectural Design Studio. The school, which has a strong design tradition, is currently at a position to revise the curriculum, and the integration of computational design technologies to a greater extent is in consideration.

At present, basic CAAD knowledge is taught in two-week workshops over the summer after the first year, and later supplemented in the third and fourth years in elective courses on Computational Literacy in Architecture, Computer Aided Drafting and Design, Mathematics in Architecture (Gonenc-Sorguc, 2005), Design Methods, and Digital Design Studio. Often a small number of second year students manage to take one of these electives, but the integration of computers generally lacks in the early years of education. This year for the first time, a non-credit basic computer course was included in the first semester curriculum. Still being developed, the course introduces general computer logic and algorithmic thinking through basic exercises. In upcoming semesters, the intention is to have it supplement the studio by encouraging students to make the connection in between the processes learnt in the two courses.

Basic design instruction at METU

ARCH 101 is currently taught by a group of five instructors with various levels of experience and varying research interests. The studio meets every other day, three times a week. The exercises described below were given over a two-week period in the beginning of fall in 2004. The exercises are based on certain formal rules and encourage the students to understand and reinterpret these. The
exercises also demand that the students develop their own rules about possible relations between varying forms in a perceptual thinking process. Sixty-seven students took the course, and met as one group for these exercises. The discussion with instructors occurred mostly after each assignment is completed. The series of exercises was as follows:

1. Photographing of various body positions in front of the studio wall. September 27.
2. Arranging six of these standard sized photographs in a two-dimensional composition, September 29.
3. Drawing lines on the photographs to transform them into geometric entities, October 1.
4. Translating the lines of the geometric study to planes in a different medium of black and grey paper, October 1.
5. Making a two-dimensional composition with thirteen elements derived from stage 4; black and grey on white paper, October 4.
6. Making a two-dimensional composition with three of the elements on a square background, October 6.
7. Positioning nine of these squares to make a new square; “In the end, the nine parts should make a whole that is more than what they can achieve by themselves.” October 11.
Each exercise in fact deserves individual discussion. The traditional pedagogical issues aside, many topics related to computation come to mind. Nonetheless, within the scope of the paper, the discussion below will be limited to the overall process of the entire series.

Examples in Figures 3 and 4 are works for stage 2 by two students. The similarities and differences in the reasoning followed by each student are made explicit when the formal properties of the works are compared. We do not openly see the many rules and schemas applied separately, as they are not discretely represented. We also have little clue about the painstaking design process. Still, the works allow for a comparison as in the case for Dow and Sausmarez’s squares.

In both figures, the six frames of photography are arranged in groups of two. Multiple relations are considered between the elements of single photographs. Each work achieves continuity of lines even if with formally different elements. In one there is a repetition of dual relationships, whereas in the other the repetition results in one continuous movement. Both manage to achieve a whole. It is easy to talk about such relations between the elements of design here mostly due to the abstract quality of these exercises. Additionally, constraints are defined; elements are similar.

At stage 2, students have thought of, seen and used some formal properties in the exercises with photographs. Each new exercise in stages 3-7 encouraged them to contemplate on what properties they have already recognized and how they have utilized them in a previous thought process. Figure 6 shows reassessment of the formal properties of the work from stage 1 in stage 3. The student starts identifying some of the formal features she sees by drawing abstract lines. It is as if she is telling us what she sees.

Similarly exercises of stages 5 and 6 in Figures 7 and 8 illustrate that students have reconsidered their perceptions. They have reassessed the formal properties of elements in past exercises as they modified them to fit into the new problem. Some formal features persist whereas some get deleted or modified.

Finally in stage 7, illustrated in Figures 9 and 10, differently than in the previous exercises, the students were given a design schema. In this particular exercise, they explored geometric transformations of their square units while adapting them
to the new framework through changes in their elements where necessary. It aimed at thinking the relationships of parts in a given whole. The elements of the unit square were altered according to how two units and hence their elements come together, as well as how the various hierarchies of parts unify in the overall composition.

The fast-paced assignments challenge the students to see the continuity in their thought as well as the back and forth process, and that definitions are flexible and contextual. The works illustrated above represent this process and the shifts in visual schemas.

Class discussions at the end of each stage help in observing these shifts. As a natural outcome of this environment, students make comparisons between their works in each stage and between their own stages throughout the series. Still, there is a need to increase the explicitness of their computations to them.

**Discussion: Proposals for an enhanced basic design instruction**

The author proposes that students primarily be encouraged to narrate their process through a formal device of visual rules as proposed by Stiny (2001). Although works illustrated above are still too complex to be mechanically computed on a machine, each represents results of computed schemas of formal relationships. The exercise in stage 7 especially has a straightforward computational potential with visual rules due to the employment of geometric transformations of the unit square. The fundamental difficulty with visual rules in studios is in conveying to the student that a process can be represented through rules. This difficulty is mainly due to the fact that the student does not have a developed sense of rules in a flexible process. Thus the focus in the basic design studio is on that students grasp the changing aspects of a design process and understand the concept of dynamic rules within it. In following their design process through visual rules, students will not only develop a sense of computation as reasoning but also learn about design as reasoning.

Basic design education usually tries to demonstrate that rules vary and that the students learn to develop their own. In the interaction with others, each student apprehends that he or she is doing things differently or similarly. Contexts, rules, and goals are all dynamic. Rules applied by an indi-
One way to perceive all this personally is with the frequently changing short-term exercises described above. With even a faster pace of production, the number of representations also increases. Then the process can be even more explicit because the number of variables and their variation between stages decrease.

An alternative way to perceive the process is that assignments take longer, but the students are deliberately stopped at certain stages in the design process and asked to (non-verbally) record what they are seeing and doing. Instead of contemplating on the past processes, getting overwhelmed with possibilities, or future determinisms, this intends to encourage perceptual variance of the object in the making. Again the records make the process more explicit.

For this explicitness, it is important that each stage is simultaneously reviewed in terms of visual rules and schemas applied. For the second stage in the series of exercises given above, some of the recordings could have been represented as the following visual rules:

That rules are represented as such implies that the student has conscious thoughts about organizing these shapes. With a multiple number of these rules, the student will be able to follow in his or her thought process what schemas persist and which ones change, how schemas translate from one context to another. The shape grammar formalism may not fully be commonplace in the basic design studio. But the dynamic use of shape rules certainly fits the core of basic design education. Future work will attempt at making it perceptible throughout the curriculum. Now in an age of emerging technologies, the intention is to prepare the student not only to design but thinking about design as computation. Basic design studio can incorporate the concept of rules more rigorously. Computation can then be reflective of the design process. This way creative thinking is not compromised in design.

Concluding remarks

The current study has many aspects to be developed. An important question about the format of visual rules, whether they are abstract or exact representations, needs to be considered. The interface with the students deserves careful preparation. Long-term evaluation systems have to be worked out to test the success and failures of the pedagogies applied. Ways to incorporate other elements of the curriculum are to be developed. How to eventually integrate computational tools along with the concepts, and establishing the technical infrastructure needed are some of the other tasks to pursue.

The approach presented in this study generally aims the difficulties faced in integrating computing and design at schools with strong design traditions such as METU.Benefiting from the existing design tradition, while updating its foundational core for contemporary tools, seems to be appropriate and at the same time inevitable. The approach of the study also revises the notion of the modernist ba-
sic design tradition, commonly known as a Bauhaus legacy, in the general sense through describing the design process as reasoning.

Kvan et al. (2004) discuss the Bauhaus curriculum in relation to computer curricula and draw attention to the integration of ‘art and craft’ – content and technique, or design and tools – much relevant today as well. Although in full agreement, the author for the moment points to the basic design studio only as a platform where design methods integrate with the concept of computation, and not the physical tools just yet. This study is thought as an initial phase of a longer research program in which, conveying the understanding of design as computation and computation as reasoning precedes the use of computer tools as a strategy to integrate computation to the studio.

Defining the design process as reasoning with uncertainties and sagacity is important to set models to those who create tools of abstraction for designers. Designers should compute like they design. Showing visual rules as an integral representation of design will not immediately change how computers work. But it would be a way to integrate computation to design education early on. At the least, it would reinstate the designers’ confidence in how they reason.

Acknowledgements

The exercises described have been prepared collectively by Tugyan Aytac-Dural, Selahattin Önur, Nihal Bursa, Mine Özkar and Erkan Gencol in the fall of 2004.

References