On the role of computational support for designers in action

Marcelo Bernal, Department of Architecture, Universidad Técnica Federico Santa María, Avenida España 1680, Valparaíso, 239-0123, Chile and College of Architecture, Georgia Institute of Technology, Atlanta, GA 30332, USA
John R. Haymaker, College of Architecture, Georgia Institute of Technology, Atlanta, GA 30332, USA
Charles Eastman, College of Architecture, Georgia Institute of Technology, Atlanta, GA 30332, USA and College of Computing, Georgia Institute of Technology, Atlanta, GA 30332, USA

Designers' actions are high-level mechanisms based on heuristics and assumptions learned from professional experience. Significant research has been devoted to understanding these actions as well as finding ways to aid, automate, or augment them with computational support. However, representing and manipulating such tacit knowledge in computational environments remains an open area of research. In this paper, we map designers' actions and relationships to compare them with computational approaches for the generation, evaluation, and selection of design alternatives, and attempt to integrate all of the above. The analysis provides a more thorough understanding of the role of computational approaches in supporting designer actions and identifies challenges and areas of future research.

Keywords: design knowledge, computer aided design, design automation, computer supported design, design technology

Corresponding author:
Marcelo Bernal
marcelo.bernal@usm.cl

Seminal studies on the behavior of designers in action (Schön, 1983), design thinking (Rowe, 1991), the mechanisms of knowing of the design discipline (Cross, 2001), and the notion of design expertise (Akin, 1988) have contributed to the understanding of the nature and complexity of the mental processes of designers. Currently, we acknowledge several well documented distinctive common actions that designers perform in the act of designing, including interpreting design situations (Gero, 1998), co-evolving problems and solutions (Dorst & Cross, 2001; Maher & Poon, 1996), recalling patterns of organization (Lawson, 2004), storing and reusing expert knowledge from specific design domains (Moreno et al., 2014; Popovic, 2004), and dividing tasks in distributed cognitive systems (Hollan, Hutchins, & Kirsh, 2000). All of them are iteratively executed during the design process.

www.elsevier.com/locate/destud
0142-694X *Design Studies* **41** (2015) 163–182
http://dx.doi.org/10.1016/j.destud.2015.08.001

Parallel to these efforts, research in computer-aided design (CAD) has evolved from a focus on computer-aided drafting and modeling to the idea that computers can aid these actions by manipulating abstract symbolic structures similar to those created by the human mind (Kalay, 1985, p 321). Along with observing increased computational power, we have witnessed the development of several computational approaches addressing the entire cycle of the *generation*, *evaluation*, and *selection* of design alternatives. Such approaches range from simply assisting to fully automating or even augmenting the actions of designers and impact the efficiency and effectiveness of design exploration.

When characterizing the role of computers, we need to realize that what is in the mind of the designer and what is represented in the computer are not the same. Eastman (2001, p 6) states that the real structure supporting the design task is an internal representation in the mind of the designer and that external representations are auxiliary structures. The formalization of a design through a model does not necessarily correspond to the complexity of the entire design itself. In fact, computer programs are integrated in more complex cognitive systems. The notion of distributed cognition developed by Hutchins (2000) while he was studying navigation tasks describes these systems, which include the interactions of internal and external representations with team members and even with cultural contexts. Therefore, the challenge from the perspective of computational tools is supporting the behavior of designers within the larger systems of interactions instead of simply reproducing their internal mental mechanisms.

One should also not disregard the threat of potential negative effects derived from the use of computer programs. Robertson and Radcliffe (2009, p 137) documented three such effects: *Circumscribed thinking*, or the limitation of design alternatives to what can be done with the specific tool; *premature fixation*, or the resistance to making design changes resulting from the premature complexity of the structure of the models; and *bounded ideation*, or the distraction from actual creative tasks resulting from technical and software issues derived from the abuse of CAD tools. All three apparently detract from the exploration of design alternatives.

While computers facilitate how designers manipulate information and perform evaluations, limited evidence has also supported their role in improving design quality in the creative process. Computer programs, although powerful tools, also have several limitations that stem from their use of hierarchical data structures for representing geometry and related attributes such as boundary representation (B-Rep) or constructive solid geometry (CSG) (Kalay, 1989), which require explicit declarations that are frequently not available in conceptual design stages, as Gross (1996, p 168) pointed out. In addition, computers are limited at supporting changes in the topology of models that are beyond the scope of dimensional variations, performing rapid evaluations based on heuristics with partial information, and representing the diverse nature of the