N. Gu, S. Watanabe, H. Erhan, M. Hank Haeusler, W. Huang, R. Sosa (eds.), *Rethinking Comprehensive Design: Speculative Counterculture, Proceedings of the 19th International Conference of the Association of Computer-Aided Architectural Design Research in Asia CAADRIA 2014*, 000–000. © 2014, The Association for Computer-Aided Architectural Design Research in Asia (CAADRIA), Hong Kong

TEACHING CAD/CAM WORKFLOWS TO NASCENT DESIGNERS

MADELINE GANNON[1] and ERIC BROCKMEYER[1]
{mgannon, ebrockme}@andrew.cmu.edu

Abstract. The following paper presents a suite of custom software environments that make advanced techniques in digital fabrication accessible to novice, first-year designers. The collective design aides facilitate a number of digital-to-physical workflows, including 3D modeling for CNC milling and 3D printing, 2D patterning for laser cutting, and interactive visualization for projection mapping. Each of the workflows illustrate pedagogical principles for embedding tacit and tactile knowledge into computational frameworks: balancing complexity against functional limits, revealing the underlying abstractions connecting digital geometry to CNC machines, engaging the designer through intuitive and responsive environments, and leveraging generative and interactive digital modeling for serial variation. These digital design and fabrication aides have been used to facilitate formal and material explorations for groups of pre-college and freshmen students, aged 16 to 19. Their resulting tangible artifacts—made from foam, birch plywood, paper, plastic, and light—show that CAD/CAM workflows can be an accessible subject matter for students without prior experience in digital modeling or fabrication.

Keywords. CAD/CAM; computational design education; digital fabrication; design aides; generative design.

1. Introduction

Digital Fabrication—digital design that is integrated with physical production—has become a core component to the education of an architect, artist, and designer. However, the technical skills inherent to standard Computer-Aided Design/Computer-Aided Manufacturing (CAD/CAM) software oftentimes necessitate that it is taught as an advanced topic. As a consequence, a student's first experiences with digital design tend to be *scaleless*, *a-material*, and an *approximated representation* of a space, assembly, or object; the student is not able to evaluate the quality of their digital design through the multi-sensorial feedback that would be facilitated by a physical artifact.

Segregating CAD and digital fabrication into lower-level and upper-level curriculums reinforces the misconception that digital design and physical production are two discrete processes. To overcome this analog-digital dichotomy, CAD/CAM workflows need to become a part of a student's creative skillset from the very beginning of their academic education. Early exposure to CAD/CAM workflows is not meant to supplant the teaching of an-analog craft; there are many valuable lessons to be learned from a splinter (McCullough, 1996; Jacobson-Weaver, 2013). Instead, the scope of these early encounters should be constrained to reinforce an expectation that what can be created digitally can be realized physically. Early exposure presents an opportunity to counteract 'press the button' fabrication (Bechtold, 2007) with a more nuanced understanding of the *messiness* involved in translating an idea constructed in bits into an artifact realized in atoms.

2. Digital-to-Physical Workflows

The following are three custom software environments, tailored to unravel some of the complexities of oscillating between a computer, a fabrication machine, and a material. The first is the *madMeshMaker*, a lightweight program for teaching the tools and techniques of digital fabrication with CNC routers, laser cutters, and 3D printers. Second is *Skwash*, another standalone program for learning about the malleability of digital geometry and 3D printing. Last is *Cityscape*, a multi-layered workflow for transforming physical models through projection mapping.

2.1. MADMESHMAKER

The *madMeshMaker* is a generative modeling environment that enables novices to both CAD and CAM to rapidly gain experience with 3D modeling, 3D printing, laser cutting, and CNC milling. The lightweight application simplifies and streamlines processes in digital design and fabrication through an intuitive and playful interface. Within the application, users dynamically deform a digital surface by clicking and '*splashing*' it around a virtual environment (Figure 1). The topological structure of this digital surface is embedded with the technical knowledge of an experienced fabricator: the software automatically formats the 3D form to export to a series of additive and subtractive processes (e.g., 3D printing, laser cutting, and CNC milling.) By bridging the processes between digital design and fabrication, the *madMeshMaker* facilitates an experience in which anyone, regardless of their technical background, can interact with a form in the virtual realm, and then fabricate it in the physical realm.

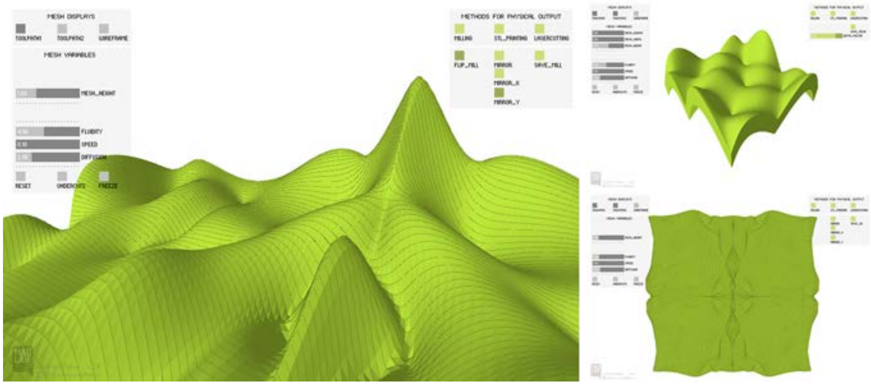

Figure 1. The generative interface enables novices to design and fabricate artifacts for CNC milling, 3D printing, and laser cutting.

2.1.1. Background

The modeling environment was inspired by one of the earliest examples of digital sculpture, *Ridges Over Time (1968)* by Charles Csuri (Figure 2). *Ridges Over Time* was created by programming a 3-axis numerically-controlled router to carve a computational form from a block of wood. The resulting physical artifact represented a profound new relationship between artist and machine, in which the artist could bind their creative desire to the underlying logic of a machinic process. As Csuri and Shaffer (1968) write, “As an extension of man's senses, computer technology can provide an exciting new potential for the creation of art.”

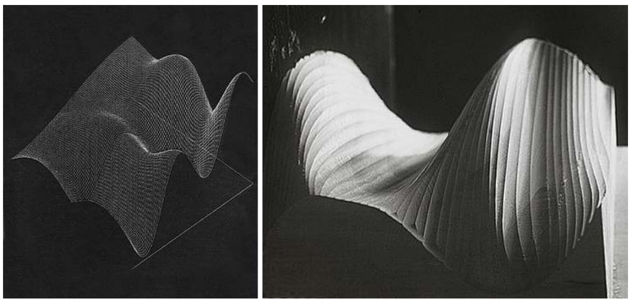

Figure 2. Plotter drawing (left) and physical sculpture (right) of Ridges Over Time (1968).

2.1.2. Interface Design

The articulated tooling marks along *Ridges Over Time* are more than a signature of its unique means of fabrication. They embody the three layers of abstraction necessary to move between digital bytes and physical matter: the logic of a digital geometry, the logic of a machine, and translation from a geometric to machine logic. While Csuri was able to synthesize these layers in his mind, the *madMeshMaker* encapsulates these abstractions within its graphic interface.

The digital geometry in the *madMeshMaker* approaches 3D modeling from a different perspective than conventional CAD software. Rather than *constructing* new geometry (from points, curves, faces, or solids) the virtual environment contains a pre-existing form that is *stimulated* by a user's mouse-clicks. These clicks cause splashes that transform the flat surface into a fluid form that would otherwise be difficult or time consuming for a novice to create using standard CAD tools. In addition, the underlying algorithm used for manipulating the surface (Stam, 2003) ensures that a user's unique interactions generates near infinite formal variation within a single session of using the application.

The logic of various fabrication machines are also embedded within the digital geometry. For example, tool paths for driving a CNC router are dynamically visualized as the user manipulates the digital surface. In addition, the geometry can be formatted to output to laser cutters or 3D printers: exporting to a laser cutter flattens the surface into 2D tool paths for laser etching, and exporting to a 3D printer transforms the surface into a solid, closed mesh (Figure 3). In effect, the application is embedded with some of the technical knowledge of an experienced digital fabricator.

Figure 3. A laser-etched canvas was made to test the mirroring and density functions of the application. The subtle patterns and sub-patterns created by the lighter and darker etch marks emerged from the fabrication process, not the digital geometry.

The graphic interface provides access to a number of quantitative and qualitative parameters for effecting how a fabrication machine interprets the geometry. Quantitative parameters, like *scale*, *mirroring*, or *resolution*, directly impact the size and fidelity of the digital surface when translating to a physical material. Qualitative parameters, such as *fluidity*, *speed*, and *density*, influence formal affects of the materialized geometry. In addition, machine-specific variables, such as *undercut checking* for CNC milling or *surface offset* for 3D printing and double-sided milling, enable more intermediate explorations into these digital-to-physical workflows (Figure 4).

Figure 4. The double-sided milling output from the application was tested by fabricating a free-standing bookstand from birch plywood.

2.1.4. Workshops

The *madMeshMaker* was featured in two introductory workshops: one introducing digital fabrication to college freshman, and another introducing both digital design and fabrication to high school seniors. The participants of both workshops came in with little to no 3D modeling experience, and no digital fabrication experience. Each workshop began with a brief lecture on CAD/CAM technologies, then a hands-on session generating digital geometry with the *madMeshMaker*. After the modeling session, the instruction team compiled the exported forms into a single file to carve from foam by a 4-axis CNC router (Figure 5).

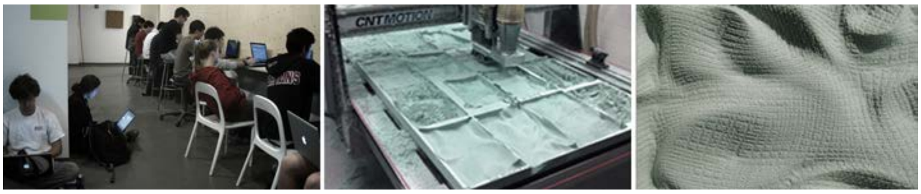

Figure 5. Workshop participants used the madMeshMaker to design and fabricate abstracted topographies.

2.2. SKWASH

Skwash is a sketch-to-fabrication stand-alone application for 3D printing. It enables novices to 3D modeling to rapidly generate simple digital geometry that can be deformed into more complex and expressive forms. The underlying structure of the digital models—no matter how deformed—is automatically optimized for 3D printing. By ensuring a 3D printable mesh, the interface alleviates a major technical obstacle for beginning fabricators. Moreover, *Skwash* demonstrates how a seemingly stable and rigid geometry can be made infinitely malleable in digital environments.

2.2.1. Background

Drawing-based interfaces have been an area of interest in CAD for over fifty years (Sutherland 1963). Not only is drawing a familiar medium for communicating in design-related disciplines, these interfaces have an expressiveness that is difficult to replicate in traditional Window-Icon-Menu-Pointer (WIMP) interfaces. Sketching is a viable interaction model for bridging digital and analog techniques during the design process (Gross 1996; Igarishi 1999). More recently, research in drawing-based interfaces has focused on bridging sketching with CAD/CAM processes (Mori and Igarishi, 2007; Saul et al, 2011; Johnson et al, 2012). These efforts are largely focused on translating loose, gestural sketches into precision information for fabricating with CNC machines. By contrast, *Skwash* teaches how precise digital sketches can be transformed into gestural and expressive physical form through 3D printing.

2.2.2. Interface Design

The *Skwash* interface has two modeling modes, *normative* and *reactive.* Normative mode enables a designer to explicitly model a revolved 3D geometry. With a few mouse-clicks, a designer can quickly create geometrically simple objects, such as cups, bowls, vases, plates, etc. Reactive mode empowers the interface to squash and deform the normative 3D model by bouncing it around a simulated physics environment (Figure 6). By incorporating physical simulation and deformation into the modeling process, the application demonstrates the malleability and playfulness that a design can achieve with digital form.

The mesh of the 3D geometry is encoded as a soft-body spring model: an elastic skeleton that keeps the form from collapsing while bouncing around the physics environment. This prevents self-intersections in the geometry and ensures that it is automatically optimized for 3D printing—no matter the deformation from the reactive mode. Resolving this optimization problem within the virtual environment alleviates some of the technical tasks usually assigned to the designer. Moreover, it eliminates a time consuming step between design and fabrication, and facilitates a smoother transition from digital to physical production. While 3D modeling tends to be taught as a series of precise and controlled geometric operations, *Skwash* demonstrates how playful interaction with digital geometry can inspire novel formal discoveries.

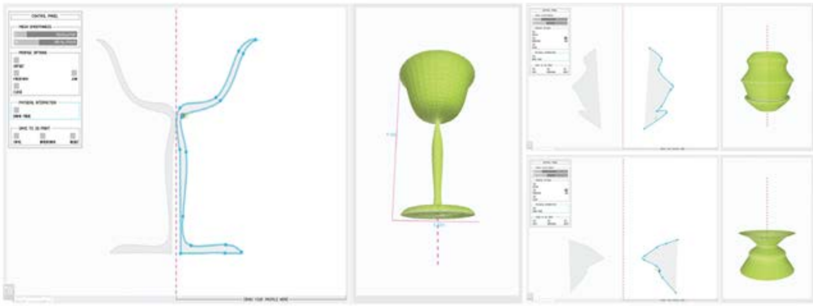

Figure 6. Normative mode (left) and reactive mode (right) in Skwash.

2.3. CITYSCAPE

Cityscape is a multi-layered workflow for transforming physical models through projection mapping. Projection mapping is the process of using dynamic 2D digital imagery to animate 3D physical environments. This digital/physical technique is particularly appropriate for architectural media, as innocuous physical models are a primary tool for representing and communicating proposed visions of the built environment. Rather than use commercially available projection mapping software, *Cityscape* is designed to teach students how data can move between different software environments. Our projection mapping workflow moves geometry data from a conventional CAD tools to a visual programming environment to a java-based programming environment. In this project, sixty first-year architecture students created the responsive audio/visual elements for projecting onto a physical model of an abstracted cityscape. The following describes the workflow for creating and calibrating the digital and physical models for the performance, using Rhinoceros3D, Grasshopper3D, and Processing.

2.3.1. Workflow Design

The workflow begins in a standard CAD environment, where an abstracted cityscape is modeled as a series of three-dimensional boxes. The model is designed so a single projector can light every exposed face of the assembled boxes. The 3D model is then used for two purposes: fabrication and data extraction. The physical assembly is fabricated by unfolding the digital model into flat sheets for laser cutting. Point data from the left, right, and top faces of the digital model are extracted to external text files via a Grasshopper3D script. These text files then feed into a programming template that recreates 3D model as a 2D scene in Processing (Figure 7).

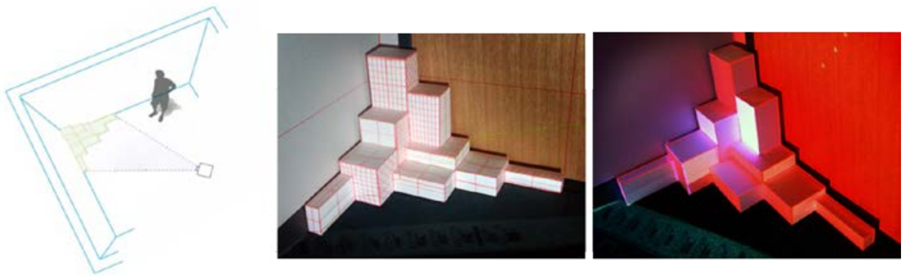

Figure 7. Physical setup and initial calibration and projection tests.

Once the scene is within the programming template, students have complete control over how the faces of the assembly are rendered. Students use the template to program 60 seconds of coordinated audio and visuals. They were taught how to animate faces and edges of the boxes (e.g., blinking, fading, pulsing) and how to structure their code to synchronize visual behaviours to specific moments of their chosen audio component (Figure 8).

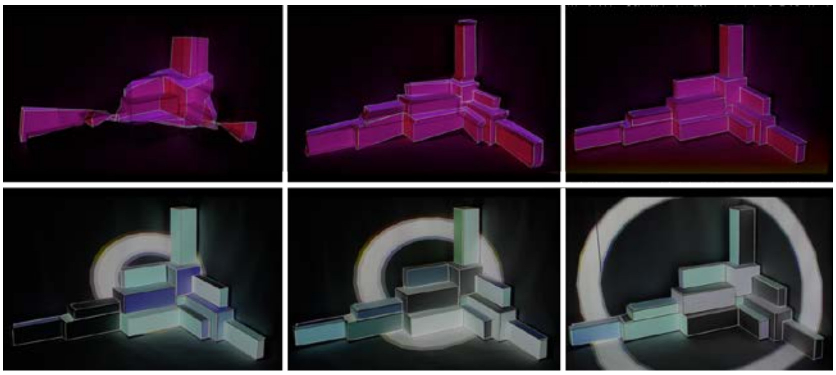

Figure 8. Students create dynamic graphics that animate the underlying physical model.

To calibrate the projector, the authors wrote an interactive program for manually adjusting the 2D digital geometry to align with the faces and edges of the physical model in the performance space. These calibrated points are then re-exported as text files, and distributed to update each of the student’s programs. Building the programming template around these editable text files gave the students optimal flexibility while writing their code, and mitigated complications with aligning digital and physical geometry during the sixty performances.

3. Discussion

Nascent architects and designers face many challenges when learning how to translate an intangible idea into a physical reality. By introducing digital fabrication along side of digital design, concepts at the core of design disciplines—such as scale, material, joint, and detail—can be incorporated within a student's creative explorations from the very beginning of their education. Moreover, early exposure to CAD/CAM processes can also teach a number of valuable lessons that are unique to digital media:

- *The value of serial variation* – duplicating a single digital surface over a series of materials or machining operations can be physically demonstrative of how a homogenous digital model can differ when making subtle real-world design decisions.
- *Curating generative designs* – when computational techniques can generate catalogues of 'interesting' form, how does a designer decide what is an *appropriate* form for their design objective?
- *Virtual versus physical constraints* – rapid design-to-physical workflows enable students to discover that while there are no formal constraints in virtual space, each of the machines used to fabricate a design brings its own set of limitations.
- *Understanding abstraction* – raw point data can be fed to any number of physical outputs: to cut and carve with a CNC router or laser cutter, to layer with a 3D printer, or to overlay with a digital projector. Once a designer understands this fundamental abstraction, they have the ability to decide on an appropriate physical output

The workflows discussed above—the *madMeshMaker*, *Skwash*, and *CityScape*—demonstrate three different approaches for balancing computational complexity with the functional limits of a CAD/CAM teaching tool: from the *madMeshMaker*'s low level of digital input for a high amount physical output, to *CityScape*'s high level of digital input for a low amount physical output. These workflows share three pedagogical principles one should consider when designing teaching tools for CAD/CAM: (1) teach the *process*, not the *tool*: mitigate as many complexities that are tool specific, and highlight subtle nuances of the digital-to-physical process; (2) create *intuitive* and *responsive* interfaces: digital environments that dynamically react to a student's input lessens the technical overhead of learning a workflow; (3) use *generative* strategies for digital input: enable a student to rapidly generate multiple digital designs to explore a physical process of fabrication. While these lessons provide a framework for young designers to explore digital fabrication, mastery of these processes require a deep knowledge in analog fabrication, drafting, sketching, and formal design.

4. Conclusions

Teaching CAD/CAM workflows to nascent designers enables beginner students to gain experience and confidence in moving between a computer, machine, and material. The digital-to-physical workflows presented are designed with minimal technical overhead, and allow students to dive right into the act of digital/physical *making*. By hiding the technical complexities associated with CAD/CAM workflows within intuitive interfaces, the software interfaces do not impede the connection between designer and tangible output. The simplicity of the interaction promotes playful repetition and trial-and-error learning. The tacit knowledge appropriated through experience will aid the student in developing a more mature cognitive understanding of the dichotomous relationship between digital design and fabrication.

References

Bechthold, M. (2007). *Teaching Technology: CAD/CAM, Parametric Design and Interactivity*. In Predicting the Future: 25th eCAADe Conference Proceedings, 767-775. eCAADe: Conferences.

Csuri, C. and Shaffer, J. (1968.) Art, computers and mathematics. In *Proceedings of the December 9-11, 1968, fall joint computer conference, part II* (AFIPS '68 (Fall, part II)). ACM, New York, NY, USA, 1293-1298. DOI=10.1145/1476706.1476759

Gross, M. and Do, E. (2004). *The Three R's of Drawing and Design Computation Design.* Computing and Cognition 2004, pp. 613-632.

Igarashi, T., Matsuoka, S. and Tanaka, H. (2007). Teddy: a sketching interface for 3D freeform design. In *ACM SIGGRAPH 2007 courses* (SIGGRAPH '07). ACM, New York, NY, USA, Article 21 . DOI=10.1145/1281500.1281532

Jacobson-Weaver, Z. (2013). Mastery and Apprenticeship in the Digital Divide: Demystifying Code Through Craft. In *[En]Coding Architecture 2013.* Carnegie Mellon University Press.

Johnson, G., Gross, M., Do, E. and Hong, J. (2012.) Sketch it, make it: sketching precise drawings for laser cutting. In *CHI '12 Extended Abstracts on Human Factors in Computing Systems* (CHI EA '12). ACM, New York, NY, USA, 1079-1082. DOI=10.1145/2212776.2212390

McCullough, M. (1996). *Abstracting craft: The practiced digital hand*. Cambridge, Mass.: MIT Press.

Saul, G., Lau, M., Mitani,J. and Igarashi, T. (2010.) SketchChair: an all-in-one chair design system for end users. In *Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction* (TEI '11). ACM, New York, NY, USA, 73-80. DOI=10.1145/1935701.1935717

Stam, J.: 2003. A simple fluid solver based on FFT, *Journal of Graphic Tools.* **6**(2), 43–52. DOI=10.1080/10867651.2001.10487540

Sutherland, I. (1963). Sketchpad: a man-machine graphical communication system. In *Proceedings of the May 21-23, 1963, spring joint computer conference* (AFIPS '63 (Spring)). ACM, New York, NY, USA, 329-346. DOI=10.1145/1461551.1461591