Digitally Fabricating Tilted Holes

Experiences in Tooling and Teaching Design

Tobias Bonwetsch¹, Ralph Bärtschi, Daniel Kobel, Fabio Gramazio, Matthias Kohler
Architecture and Digital Fabrication, ETH Zürich, Switzerland
http://www.dfab.arch.ethz.ch
bonwetsch@arch.ethz.ch

Digital fabrication of building components by means of computer numerically controlled (CNC) machines is of high interest for architects and the building industry.

Common design software does not allow for utilizing the potential that lies within these new technologies. To fully exploit the power of digital fabrication, new design paradigms have to be explored. In our design studios we implement direct scripting, the use of images to control parameters, as well as dynamical and rule based systems, which enables the designer to exploit the possibilities of digital fabrication. This paper provides an overview of the tools we developed. We will present some of the results from these classes and discuss its implications for future tool sets.

It is essential to apply the knowledge of production methods at the starting point of the design process, in order to fully benefit from these new technologies. We believe that novel design strategies emerge out of this shift in production. Our goal is to integrate the principles of digital fabrication into the design process, resulting in a more valuable and sustainable architecture.

Keywords: Digital fabrication; CNC; design tools.

Introduction

The use of procedural design tools has come to the interest of architects over recent years. Identifying the limitations that the predefined tool sets of commercially available CAD-software lay upon the design process more and more architects start to program their own tools tailored to meet specific design tasks (Silver, 2006). Apart from applying information technology to the design process of architecture the advance in technology is also revolutionising the fabrication process. Digital fabrication by the means of computer numerically controlled (CNC) machines allows immediate production from the design data.

Yet, although architects already use the power of scripting their own tools to create sophisticated designs the design data is seldom connected to the fabrication process. We believe that it is essential to incorporate the fabrication logic into the design process in order to fully benefit from the potential of digital fabrication techniques. Even more, novel design strategies emerge out of this shift in concept.
In our research we investigate the possibilities that arise when incorporating digital fabrication logic into the design process.

Until now we have worked with two basic set-ups, investigating the possibilities that arise from a fabrication process that involves a large amount of elements. On the one hand we deployed standard bricks as the basic element, which were layered using an additive fabrication process (Bonwetsch, Kobel, Gramazio and Kohler, 2006). On the other hand we applied a subtractive process investigating holes, which were milled into a surface. Given the possibility by digital fabrication to set the exact position and angles as well as other qualities of the elements that are built into the material we talk about informing the material.

In this project we concentrate on the set up using holes as the basic unit, we will describe several strategies on informing material and incorporating the fabrication logic into the design process and present some of the results gained during several design studios over the last two years.

Test arrangement

Openings or holes are one of the fundamental elements of architecture. We challenged the students to distribute a large amount of holes (1000-2000) on a two by three meters wall and inform each hole with a particular diameter and a certain degree of deflection.

External software tools

As developing surrounding we chose the 3d-modeling software MAYA (www.autodesk.com). It provides a fast Application Interface (API) for plug-in development, scripting support and general data modelling possibilities. Still, it was not possible to build a three-dimensional solid model with thousands of perforations (i.e. tilted holes), as three-dimensional Boolean operations perform at a high cost. As a workaround we generate the solid model of the perforated wall by triangulating both perforated sides with the help of OpenGL/GLU and with a bit of bookkeeping written in C++. Another drawback in using MAYA was the

Figure 1

The robot fabrication facility
inaccuracy of offsetting curves. Therefore, we chose the open-source package gpc for two-dimensional polygon Boolean-operations (Murta, A.: General Polygon Clipper Library, www.cs.man.ac.uk/~toby/alan/software).

Voronoi partitions were built with the open source project qhull (www.qhull.org). Finally, those parts of the tools that were not performance critical were implemented in Python (www.python.org). Python is officially supported in the current MAYA version 8.5 and can also be easily embedded in MAYA 7.0.

Fabrication setup and constraints
As a fabrication tool we applied a six-axes industrial robot equipped with a spindle for milling. An external linear track and a rotating table add another two axis to the fabrication setup. The robot arm can operate a workspace of three by three by eight meters (figure 1).

The control data for the fabrication process is directly generated out of the data model in MAYA using a custom written postprocessor. Common CAM software is not sufficient as it can only handle five axes.

For the material we chose to work with Styrofoam boards, which allow a fast fabrication process. Two prototype walls were completed in concrete. For these a wooden formwork was fabricated with the robot facility and combined with plastic tubes of varying diameter and length for the cut-outs.

A key characteristic of fabrication is that it sets constraints on the design. In our special case of tilted holes four constraints apply:
- a minimal distance of each hole towards the edge
- a deflection of each hole less than 50 degrees out of the vertical axis
- no pair-wise overlapping of holes
- the diameter of each hole has to be contained in a discrete set

While the milling tool defines the first two constraints, the last two are due to the pouring process of concrete.

Design strategies and software tools
The distribution and information of 1000-2000 holes is too laborious to be done manually, therefore we adopted three design strategies:
- direct scripting in Python
- images as information carrier
- dynamical and rule based systems

These can be used separately, as well as in combination. Also, an intervention using the common MAYA GUI is possible at all points.

The perforated wall
An important requirement for the holes is to obey a certain density distribution while preserving the constraint of non-intersection. We solved this requirement in different ways. In the first design studio the holes were diced according to a simplified density function and rejected if they overlapped. In the following design studio, we applied a dynamical model to solve the constraint. A pair-wise repulsing force acts on the holes, while attracting and/or repulsing forces approximate the distribution. By the time of writing the authors are able to distribute objects by a technique called Lloyd's algorithm, which transforms a general Voronoi distribution into a centroidal Voronoi distribution (Hausner, 2001; Deussen et al., 2000).

To steer the deflection of the distributed holes several different concepts were applied. As a geometrical model the concept of target points was introduced, where the holes are forced to align with a connected point in space. Other concepts set the directions of the holes according to a function or according to a two-channel image. The rotation point of each hole can be set in the middle or on either side of the wall. This allows keeping the distribution image regular on one side of the wall. Table 1 illustrates several of the discussed design steps.

The Voronoi wall
In a next step we generalized the circular holes to an arbitrary form. As the calculation of intersection between holes of an arbitrary outline gets
Step 1: Random distribution of holes with two different diameters. The two big circles in the corner represent target points. The left circle is the target for the smaller holes and the right target applies to the larger holes.

Step 2: Pair-wise repulsion acting on the hole set solves intersections.

Step 3: The targets are applied to the deflection. New intersections appear.

Step 4: According to step 2 dynamics solve the intersections. The intersecting holes are moved in each step in the direction of the gradient of the true three-dimensional distance. Since some holes change their position the targets have to be applied again, which can lead to new intersections.

Step 5: Corresponding three-dimensional solid-model. This solid-model acts as a visual control of the design and can be used to fabricate scaled models applying a gypsum based 3d-printer.

Table 1
Screenshots taken from MAYA during a typical design based on the dynamical tool.
quite complicated we started out with a plane polygon partition. To achieve this we decided to apply a Voronoi tessellation, since it can be easily described and provides convex polygons. A Voronoi tessellation in two-dimensional space is a description of a set of points, where the area circumscribing each point is maximised. The third dimension is reintroduced by considering two corresponding Voronoi partition on either side of the wall. In this case corresponding refers to two points on either side of the wall, which define a vector. If there are enough vectors and they are not too deflected, we get two similar Voronoi partitions. In order to acquire the milling paths for each side of the wall, the resulting polygons are downsized by half of the defined border width and the radius of the milling tool.

Still, there are infinite possibilities to connect both corresponding paths with the fabrication tool. As a solution we introduced the attribute *spin* as an additional degree of freedom for each hole. The centroid is calculated for each milling path. Every point on the path has a certain angle in respect to the centroid and its horizontal axis. If the angles on both curves are offset by \( s \), we call \( s \) the spin of the hole. A spin equal to 180 results in a minimum volume for the hole, whereas a spin of 0 maximises the volume.

Although, the Voronoi partition solves the overlapping constraint on the surface, it gives no guarantee that the holes do not intersect in the cross-

![Initial distribution scripted by a student. The holes are arranged as elliptical cells. Their total deflection is governed by their horizontal position while their orientation is dependent on the position on the ellipse.](image1)

![Corresponding 3d solid-model without spin. An export of the cross-section of the wall reveals intersections between neighbouring holes, which could not be considered in building the 3d solid-model.](image2)

![Corresponding 3d solid-model with a maximum spin on all holes. All intersection problems are solved.](image3)
section of the wall. Through the adoption of the spin attribute most of the intersection problems were solved, without the need to implement a complicated dynamical system. Furthermore, it provided a straightforward solution to describe a complex geometry of a void volume, which is derived directly out of the fabrication process.

Maintaining a constant border between each hole is a challenging problem we still have to solve. As the cut of a tilted cylindrical milling tool in a plane results in an ellipse, it is not sufficient to offset the polygons by a constant factor. Rather, at each point of the milling path the tool offset has to be calculated in accordance to an ever-changing ellipse. Until now the changing offset is only an approximation.

Table 2 illustrates the design of a Voronoi wall. The degrees of freedom were the same as for the tilted holes in Table 1. The only difference being that instead of a radius each hole has an attribute border width and spin.

Discussion

A discontent with the standard tool sets of common CAD software in architecture results in the necessity for architects to design their own software tools tailored to their specific needs. Especially when applying the logic of non-standardized digital fabrication (e.g. handling and informing a large amount of elements) standard software proves to be insufficient. It is vital to incorporate the fabrication logic with the design process in order to fully benefit from the potential that lies within digital fabrication techniques. Digital fabrication does not only open up possibilities, but also lays constraints onto the design. Incorporated at the very beginning these constraints can act as guidance and lead to novel design strategies.

In the course of the different design studios the students were provided with three design strategies that ranged from a high degree of freedom (i.e. direct scripting) to a restricted control over the design (i.e. pressing buttons to trigger complex dynamical systems). The more restricted dynamical and rule based systems require an empiric approach in order to fathom the scope of options they provide. Still, the feeling of most students was that the authorship of a design is transferred from the user of such tools towards the hidden logic of a computer program. Although, in transferring complex problems into a system of discrete rules creating computer scripts resembles somewhat the creation of architecture, it cannot be expected that architects take the place of software engineers. In our experience the combination of the different strategies proved to be most promising. Predefined complex and costly algorithms were combined with the possibility of direct scripting, which opened the degree of freedom for the designer and yielded the highest degree of diverse results. Also, expressing transition functions as gradients in images proved to be a

Figure 2
Concrete Prototypes 3m x 2m x 0.12m. Left: The holes are placed on a grid. The deflection and radius are read from a three-channel image. Right: The holes are distributed randomly a dynamical system solves the intersections between the holes. Target points control the deflection of the holes.
more intuitive approach for architects than writing code (figure 2).

Creating specialised design tools is necessary in order to inform a large amount of elements, as it cannot be done manually. Transferring this information into the physical reality is one of the main potentials of digital fabrication. The physical results of the design studios depict the potential of applying the logic of digital fabrication to the design process, as design strategies such as the spin attribute of each hole are derived directly out of the fabrication process (figure 3+4).

Acknowledgements

The authors would like to thank Michael Lyrenmann for accompanying the fabrication process and all of the students that took part in our design studios over the last two years.

References


