Parametrically Fabricated Joints: Creating a Digital Workflow

J. Cormack and K.S. Sweet

[1]Victoria University of Wellington, New Zealand

Abstract

Timber joinery for furniture and architectural purpose has always been identified as a skill or craft. The craft is the demonstration of hand machined skill and precision which is passed down or developed through the iteration of creation and refined reflection. Using digital fabrication techniques provides new, typically unexplored ways of creating and designing joints. It is as if these limitations which bind the ratio of complexity and use are stretched. This means that these joints, from a technical standpoint, can be more advanced than historically hand-made joints as digital machines are not bound by the limitations of the human.

The research investigated in this paper explores the ability to create sets of joints in a parametric environment that will be produced with CNC machines, thus redefining the idea of the joint through contemporary tools of creation and fabrication. The research also aims to provide a seamless, digital workflow from the flexible, parametric creation of the joint to the final physical fabrication of it. Traditional joints, more simple in shape and assembly, were first digitally created to ease the educational challenges of learning a computational workflow that entailed the creation and fabrication of geometrically programmed joints. Following the programming and manufacturing of these traditional joints, more advanced and complex joints were created as the understanding of the capabilities of the software and CNC machines developed. The more complex and varied joints were taken from a CAD virtual environment and tested on a 3-axis CNC machine and 3D printer. The transformation from the virtual environment to the physical highlighted areas that required further research and testing. The programmed joint was then refined using the feedback from the digital to physical process creating a more robust joint that was informed by reality.

Introduction

A joint is typically selected based on its ability to provide strength, aesthetic quality and dynamic presence to a situation. This basis of selection is partially defined by the limitations of the craftsman and their tools. Likewise, the skill that the craftsman possesses has its own set of limitations based on their personal experience and expertise. A timber joint which embodies a large amount of complexity can often challenge the skill of the craftsman, compromising attributes of tolerance, application and assembly. These are extremely important qualities which make a successful joint system.

However, if craft is established by automatic machining, joint complexity can increase without affecting these sets of important attributes. This is not to say that human interaction should be excluded, but more so the human and the machine are partners in the collaborative process (Gramazio, Kohler, and Willmann). This benefit of human and machined collaboration is managed efficiently in the parametric quality and scalability of timber joinery. For example, a large beam could be milled with the same, or near similar precision to that of a beam 80 to 90 percent smaller. Of course, in this situation the milling tool would decrease in diameter. However, the attributes of tolerance, application and assembly would not be compromised.

The process in which this parametric environment is set up and developed is as beneficial as the end resulting characteristics of the joint itself. The joint in a project is often not thought of as the centre piece, more so, it is thought of as an alleviation between two members or two parts to a project. Considering this idea, typically opening a parametric file and establishing work around the file is seemingly backward. Being able to insert and instantiate the file into already an established project, is on the other hand, more of a forward thinking process enabling the capabilities of the machine to be programmed into the joint separate from the design.

> Knowing the production capabilities and availability of particular digitally driven fabrication equipment enables designers to design specifically for the capabilities of those machines. The consequence is that designers are becoming much more directly involved in the fabrication processes, as they create the information to be translated by fabricators directly into control data that drives digital fabrication equipment. (Kolarevic, 2008).

It is this forward thinking process of instantiation into a project which was a key driver in the direction of this research. We are no longer designing the form that will ultimately be produced, but the production process itself (Gramazio & Kohler., 2008) If a given joint has a parametric quality, paired with an ability to be inserted into an existing CAD environment, then the work is not restrained by situation. This process of scripting and instantiation from a defined catalogue made the file malleable.

Software Overview

CATIA from D'Assualt Systemes was the parametric software selected for the creation of the joint catalogue due to its graphic interface and ability to generate data from visual representations. This was a key advantage for the researchers who are architects trained in working visually. The software also requires precise geometric relationships to be established between all of the elements of the model: another characteristic that architects excel at in their design process as all parts of an architectural project have a direct relationship with one another. While these characteristics are usually part of software dedicated to the architectural design process, CATIA was selected due to the capabilities of programming complex, user defined relationships and creating machine processes.

CATIA is an engineering platform typically used in the aeronautical or manufacturing industry for its powerful abilities in collaboration, parameterisation and product creation. It is used as an all in one platform from initial design to manufacturing for complex projects and was easily adapted to explorations in joint manufacturing for architecture. The creators of CATIA understand that a design process may require many diverse specialists who have different tasks that require different working environments. Because of this, the software is modular and allows for diversity in work through what they call workbenches. Each workbench is based on this idea of speciality in a collaborative environment. The workbenches have many different and incredibly useful modules that include one for an initial creation of design geometry, one for the programming of intelligence into the digital model through parameters and rules, one for creating machine toolpaths, and one for simulating and outputting machine data – all of which remain continuously linked when any data or parameters change in any module.

The platform really excels is in the ability to programme design rules within the parametric model. Using these programmed design rules in the CAD environment provides a seamless process from design to manufacturing. This establishes a workflow where design rules can be developed, deployed and instantly, changes are made in the accompanied machining and simulation workbenches. Shifting the information to the machine level combined with differentiated, flexible tool guidance enables more complex geometrical relationships to be embraced and produced (Jeska, Pascha, & Hascher). When these files change, they are available to the machining workbench for production simulation. This decreases the amount of back and forth, inefficiencies and human error which is usually encountered in the design to production workflow.

Simple joints

Investigation began by looking at simply constructed joints with known behaviours, specifically the mortise and tenon joint. Simplicity provided an easement in learning the software by providing less variables in the design. The mortise and tenon and other simple, traditional joints are a known condition and are predictable (See fig 1.). They have evolved very little

throughout their use, so this provided the least number of surprises or unknowns in the behaviour of the parametric model. Not having to focus on unknown characteristics in the design meant that focus could be directed to setting up proper, well developed parametric conditions within the software. With a predictable outcome, it was easier to identify successful results.

Parametric software relies on the user to create relationships based on lines, points, planes or some other geometry or user defined variable to drive the "parameters" of the design. With this research project the use of planes as the primary framework for parametric conditions was selected as the mechanism to define all of the joints. The plane allowed the most flexibility and analogy to built conditions. They also remain invisible in that they are not "solid" geometry or parts in the model.

> Modelling with feature - based solid modelling tools requires the user to specify relations and constraints at two different levels: internal relations, which are necessary for computational consistency of the model; and external relations, which are required for keeping the consistency of the model with real world laws. (Valdes, Gentry, & Cavieres, 2013)
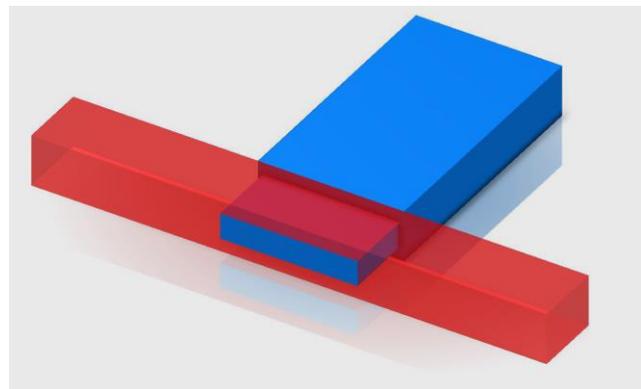
Figure 1: The simple mortise and tenon joint.

The file was originally constructed based on user defined variables or explicit parameters. This proved unfavourable in later stages when trying to create a User Defined Feature (UDF) that would allow generic instantiation. A UDF is essentially a complex copy and paste feature for 3D geometry within CATIA and allows for further scripting of geometry into any model by storing the UDF in a universally accessible database. The complexity is in the ability to define separate parts, features and geometry as either, inputs or outputs that allow the UDF, or part, to flex or morph to the constraints in which it is being instantiated. The UDF was not successful in these early iterations based on the earlier decision to create the parametric relationships that relied on explicit parameters. This limited the flexibility of the instantiation by relying on the user to explicitly define the parameter that drove the part.

This example shows the extent of the influence programmers of CAD software like CATIA, have on the way files are

constructed and indirectly, how the model might look or perform. The bias of each software can be quite substantial and have an abstruse effect on the geometric workflow. In CATIA's case, the software is very prone to problems and errors if the file is not set up correctly – a simple misstep in the creation of the initial 3D model can have a profound effect later when rules and parameters are altered or enforced. The decision to abandon the explicit parameter for another option helped alleviate some of the software bias and provide a more stable construction.

Understanding the initial restrictions of the software was facilitated by the choice to use simple, preliminary joints. The direct feedback provided by the virtual environment, compared to a known condition, aided in identifying limitations and problems in the parametric model. From the initial discovery of the impediments of using explicit parameters, it was decided to further develop the model using generic planes. User defined variables lack the ability to be separated into individual UDF files. They have interdependencies with the part feature which is driven from the explicit variable inside the sketcher workbench. This means that parameters cannot be easily defined upon instantiation. The use of planes provided a normal direction and a point in space. Two qualities which are essential to create an accurate parametric profile. When developed with plane geometries, the relationships of the parametric file could be manipulated after UDF instantiation had taken place making the model responsive and flexible.

The initial plane created had to be created as a datum input, meaning that it had no relationship to the file's coordinate system or any other geometry in the model. The datum plane provided an input for the UDF which meant that it could be created in free space, in any direction and orientation, as long as the input plane was defined. The process was then designed as a three stage integration. For instantiation of a UDF to occur, there had to be an input plane created in the file that would receive the joint, a plane geometry UDF file an external catalogue introduced to the environment and then a final joint UDF file from a catalogue.

Once the model was created with planes as the reference for the geometry and successfully used in a UDF in any condition, the project was able to proceed to more complex models and joint geometry.

More Complex Joints

The research continued to develop utilising more complex, known joints. The progression was redefined by the feedback process within the software and from the work of the previous exploration. As more complex joints were introduced, more variables had to be parametrically assigned and an advanced set of relationships had to be established. For example, in the traditional finger joint, a larger tree structure (programmed relationships in CATIA) consisting of increasing complex geometrical sets was needed to achieve a parametrically flexible version of the joint. These geometrical sets had the upmost importance in the overall function of the joint but had absolutely no relevance to the 3D representation of the

project. These sets act as parametric drivers to the joint without affecting the actual geometry. This did not "primarily define a final geometric form", but provided "a complex and refined generative process of digital materialisation" (Willmann et al., 2016). Line and point functions were introduced into the establishment of the geometrical set allowing angular functions to be constructed in relationship to the part features. This facilitated a parametric finger joint that could be set at stages between 20 and 160 degrees (See fig 2.). While the software would allow the creation of a full 180-degree rotation, limitations were programmed in the geometry so that the joint would not become unreasonable to assemble in the context of reality.
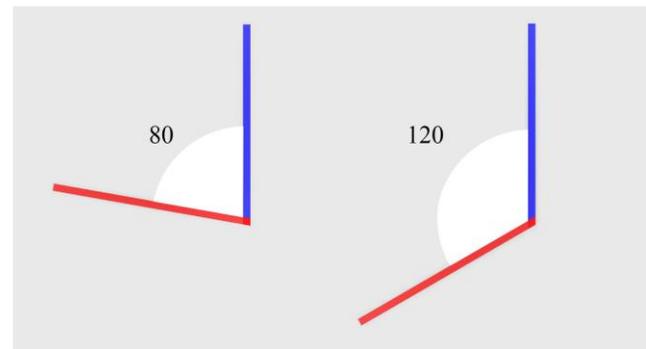
Figure 2: Parametric adjustment of the angular sketch driven plane geometry in the finger joint.

The finger joint had many more complexities other than simply an angular parametric feature. Variables for the number of fingers, finger spacing and finger length were all considered (See fig 3.). These relationships were developed through functions which included the analysis of material length and thickness. This meant that when a new thickness or board length was assigned, the relative amount of fingers adjusted accordingly. Adjusting these values provides an equal strength ratio within the range of possibilities. This benefit could be utilized when creating structural conditions, in particular in later design stages when assembly simulation has begun. The relativity of the relationship means that changes can be made to thickness of material and reductions in strength are not compromised. The power and potential of parametric programming in fabrication lies in the ability to create these parameters and possibly more. Rules of materiality and constructability can be programmed in the models so that geometry is altered based on these inputs. Parameters could be derived from material performance, structural performance or anything else that might affect how the model is constructed in reality.
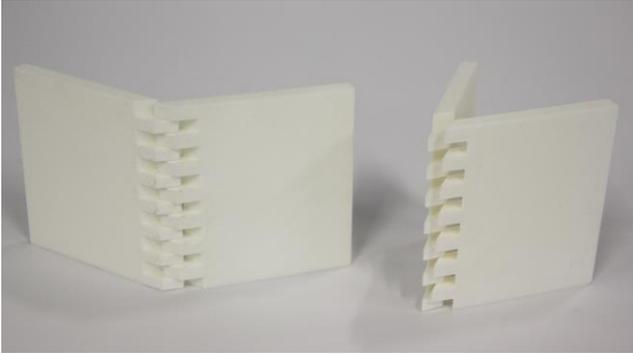
Figure 3: 3D printed prototype of angular finger joint.

Besides direct parametric relationships, the rule and check tools within CATIA help to control the user's ability to deconstruct or change the essential principle conditions of the file. The rule scripts allow parameters and constraints to automatically change if a given, stated parameter also changes. In respect to the cross lapped dovetail joint – another complex joint that was explored – the file has a set limitation in relationship to dovetail width and the rule defines that when depth increases past an X statement, then dovetail width increases by 20 percent, providing a more accurate proportion of geometry for joint fabrication.

The check tool is a piece of script which can provide a warning page to appear when a scripted parameter either increases or decreases past a specified value. A check script could be inserted into a joint file and control certain aspects of a design. For example, a finger joint's individual finger thickness could have a check script specifying a value in which structural weakness or machining restrictions could prove problematic if the joint were to be created.

Joint Fabrication

Further work investigated the manufacturing and physical creation of selected joints. Some were chosen to be 3D printed while others were chosen to be created on the 3-axis machine. This was a key element of the research as the physical production provided feedback on how the joint would need further refining or reprogramming. One such parameter that was programmed from the digital to physical feedback loop was the attribute of tolerance. This was over looked in the CAD model as it is easy to disregard physical relationships and limitations of materials in a virtual environment. Once understood in the physical world, the tolerance parameters were then defined within the file. This was a simple parameter based on plane relationships in context to specific parts of the file. Adding tolerance parameters made it possible to control the constructability for either single assembly or successive assembly of physical joints.

The cross lap dovetail was developed with disregard to tolerance. In this case the joint could have been pushed together with a large amount of force. Tolerances are specified to allow for imperfections and inherent variability without compromising performance (Kolarevic 14). However, this joint would not be able to be taken apart without some or all joint

failure occurring (See fig 4.). Defining this parameter is primarily based on the joints operation and purpose. Where a larger tolerance would be needed for architectural purpose and a smaller, more precise tolerance parameter would be needed for product and furniture design conditions.

Figure 4: *Tolerance restrictions did not allow the cross lap dovetail to be fully assembled.*

Both timber beam and plywood sheet joints were developed throughout the research but due to size and machine limitations, apart from the dovetail beam joint, only sheet joints were actually fabricated. The fabrication of the sheet joints provided an opportunity to differentiate them from the timber joints that remained as virtual constructs in the computer. Fabricating these joints illustrated outlier problems which were never consciously understood while in the virtual environment. For example, the fingered mortise and tenon joint had tolerance variables developed within the tree structure. The tolerance, set at 0.25mm was large enough for assembly to occur. However, when the joint was created using a 3-axis machine, the joint was unable to be assembled (See fig 5.). The reason why the assembly was not defined as outlined by the files tolerance directions was due to the inconsistencies of plywood sheet quality. The stock sheets had defective properties within its manufacturing, the repercussion was that sheet thickness deviated + or – 1mm in certain areas. It becomes problematic when material qualities do not align to parametric relationships. The plywood thickness was developed as a main parametric function. The deviation of this plywood thickness broke the parametric relationship of the joint, resulting in miscalculated tool paths and inaccurate cutting profiles.

Figure 5: Fingered mortise and tenon joint.

The precision of the modelling process and now the fabrication process shifts the issue of tolerance from the designer to the manufacturer. Previously material defects could be accounted for in the construction process. Now with more precise modelling and manufacturing techniques, material manufacturers will be held accountable for manufacturing a consistent product that can be used without fear of inaccuracies.

Critical Reflection

Using all of the software's tools for joint creation delivers a powerful method for digital fabrication. Design rules and UDF callouts can produce a parametrically defined and varied result with simple inputs of code. The rule and check scripts come unique to each file and each individual task, permitting complex controls of structural and machining responsibility. This system works well and provides for many opportunities in the design and manufacturing process but it is not without flaws. Like many parametrically linked design files, when undefined variables that are not considered are changed, the relationships become invalid. This is prominent in CATIA due to its parent/child dependencies created in its tree structure. The way this design tree is ordered has major influence on the success of some of these features. When certain parts are reordered or changed slightly, the parametric file breaks and the relationships inherently become invalid.

If a file breaks there are significant challenges in identifying the cause or problem area. The degree of the challenge to correct the issue is directly related to the complexity of the joint that has been created. The more complex the joint is, the larger the tree structure and the larger amount of possible issues faced. This problem is resolved however, through software proficiency. Much like learning any other complex skill, the more times it is performed, the basic definitions which make up that skill become inherently natural. Those successive failures in basic file order are reflected upon and benefit starts to appear in the form of mastery. During the course of the research this became apparent. Tree set up and geometry affiliation became efficient and accurate.

Being able to have such control over the process of joint creation allows the designer to develop a richer understanding of relationships between the virtual and the built. They extend the architect's human capabilities; they improve his or her overview and multiply the possibilities for control of the design (Gramazio & Kohler., 2008). The software platform forces the designer to think about the process in its entirety, even when at a fundamental creation stage. The perpetual feedback loop from the relationship of the physical to the virtual constantly re-inform the design process making every stage more robust.

Future Research

There are many paths for future work on this research. The first will be to programme the joints with much more intelligence. While this research proved that creating a catalogue of joints that could be easily instantiated into any

model was possible, it did not add the next level of intelligence to the joints. The structural properties of the joint itself, as well as the material behaviour can be programmed into the next round of research. The designer will be able to choose the material and the structural requirements of the joint and receive feedback from the model as to whether the envisioned design is possible or not. The aesthetic qualities of the joint will be more of the focus at the beginning rather than simplified or altered at the end of the design process.

Another path will be to further build upon the ability of CATIA to generate geometry from scripts. Instead of relying solely on the power of the graphic interface, further research will capitalise on the software's capability to generate new geometry from smaller parts through rules created in a scripted environment. This takes the design creation from a one off situation to one that becomes generative. The designer can describe a desired condition and allow the rules programmed into the parts to generate a geometric model much more efficiently. With the rules of materials and structure built into the parts, the architect can rely on the software to generate a design that can actually be built. The focus of the work can be on design issues rather than problems of constructability.

Finally, as the research gains confidence in the creation of small parts like joints, it can expand to include systems that the joints will just be a small part of. These systems could include walls, roof structures or any other amalgamation of pieces into something larger than itself. It is not expected that the software will generate "Architecture" but will aid in the manufacture of architectural systems that are designed and set in place by the architect – giving more control to the design process and less concern for the construction process.

Conclusion

This research project combined several new ideas as a proof of concept for an innovative digital workflow within the field of architecture. It is rare for an architectural practice to consider the specifics of manufacture due to the technicalities and time constraints involved. By building a catalogue of digital "parts" that parametrically change or adapt to an existing designed environment, the architect will have much more control on the output of the design. Joints once crafted by hand for more function than aesthetics can now be created for both purposes. Design modules can have the intelligence of structure and materials programmed into them and the designer can focus on the design relationships rather than the technicalities of whether it will work or not. The success of this project will lead the way to a new way of thinking about architectural design – one that involves considering the production of the manufactured piece from the beginning rather than altering the design at the end to suit manufacturing limitations. It will bring the title of master builder back to the architect.

References

Gramazio & Kohler. (2008). *Digital materiality in architecture*. Baden: Lars Mèuller Publishers.

Gramazio, F., Kohler, M., & Willmann, J. *Reality Matters: The robotic touch : how robots change architecture*.

Jeska, S., Pascha, K. S., & Hascher, R. *CNC production for timber structures : Emergent timber technologies : materials, structures, engineering, projects*.

Kolarevic, B. (2008). *The Risky Craft of Digital Making In: Manufacturing material effects : rethinking design and making in architecture*. New York: Routledge.

Valdes, F., Gentry, R., & Cavieres, A. (2013). A Process--Centric Approach for Teaching Digital Fabrication. *SiGraDi 2013, 2013*, 404.

Willmann, J., Knauss, M., Bonwetsch, T., Apolinarska, A. A., Gramazio, F., & Kohler, M. (2016). Robotic timber construction - Expanding additive fabrication to new dimensions. *Automation in Construction, 61*, 16-23. doi: 10.1016/j.autcon.2015.09.011