A grammar for joints

How can we design new details when we don't know how they should work?

In his article *Up and Down the Ladder of Abstraction, Bret Victor asks a similar question* to incite designers to deconstruct complex systems into simpler, visually explorable representations that are easier to read and understand (Victor 2011). Among other things, Victor is a programmer, so it is natural for him to break down problems into smaller, manageable challenges. But this well-known problem-solving technique is not exclusive to computer science.

Robert Woodbury wrote to architects and designers in a similar quest to make parametric systems approachable and understandable. In *Elements of Parametric Design*, he hints about the hidden potential of explicitly representing ideas that are usually treated intuitively (Woodbury 2010). For designers, it is easy to think about intuition as a good thing. But what if our intuition isn't nourished enough to solve the problem at hand? To go back to our first question, how can we design new details when the required engineering knowledge is not evident?

Details are compounds of technical aspects and aesthetic desires, easily categorized anywhere between fascination and boredom. They escape universal definitions and are hard to circumscribe. However, a good starting point to identify what they are made of is to look at their conventional drawings. Here, details are typically described statically, silencing intervening ingredients such as the flow of water or the vectors of the structural forces acting on them. The inside detail's machinery can sometimes be hard to reconstruct. So now, in the era of simulation and computational thinking, it might be the right moment to ask: how are the diverse fields of knowledge that the details bring together, such as physics, construction, materials, economy, and aesthetics, combined and explained all at once?

Coming back to Victor's questions, we should be aware that how we describe a phenomenon most surely affects how we understand it. Similarly, choosing a design description is not innocuous as it affects the scope of what the resulting design can be. For us, novice designers, we are interested in learning. And we are interested in finding accurate and flexible descriptions that would let us explore and learn about the space of detail possibilities.

Rule-based systems and grammars

Rule-based systems or grammars are composed of rules applied step-by-step to compute known or new solutions. Introduced by Stiny and Gips in 1972 in a design context as *shape grammars*, they can be used as a production system for emergent design solutions by combining simple geometrical rules (Stiny et al. 1971). Since then, shape grammars have been considered robust generative systems thanks to their simplicity without losing flexibility. This is enabled by two main aspects of shape grammars: their rules are compoundable -they can be freely combined to produce an infinite number of solutions-, and they are extendable -the rule-set can be expanded at any time-. These compoundable and extendable properties can lead to unexpected design spaces by moving beyond parametric and typological solutions (Mueller 2014).

Generative systems like these have been applied to a wide range of architectural problems from spatial configurations to urban design since the '70s and continue to be today a fruitful field of research. In the '80s, Radford and Gero proposed using a rule-based system for detail design (Radford and Gero 1985). They presented a functional wall-roof detail computed based on standard solutions found in catalogs. As catalogs are the primary source for designers to resolve detailing solutions, this approach for automating steps makes a lot of sense. Saving time and helping designers be more explicit about their choice of solutions is already a great advantage. However, could a system like this be used to move beyond the automation of known solutions?

A grammar for joints

In other words, how can we design novel details with grammars? Let's examine the generative grammar for joint design depicted in Figures 1-2. This joint grammar comprises rules frequently used in detailing. The grammar's rules are simple geometric operations, not bound to single manufacturing or material contexts. The rules are described as a set of left side-right side states, which read "*in the presence of* [left side state] *transform it to* [right side state]". For example, rule 1 can be described as: *"in the presence of* a shear force in an interface, *add* a step."

The joint grammar rules can be grouped into different types. The first group is the function-related rules, such as the ones that provide structural support. Examples of this group are the rules marked with an arrow at the interface, representing the type of force applied to it (*rules 1-9*). Let's consider some examples of this group:

- *Rule 1* counteracts a shear force -a force that pushes parts in a parallel direction to the contact interface- by constraining the degrees of freedom of the interface where the rule is applied. Megalithic assemblies would split apart vertically without these steps in between;

- Another example is *rule 4,* which provides support under a compression force -an inwards force, normal to the interface's direction- by extending the contact area where parts meet. Wall thickness can be very thinly manufactured in certain materials; however, this small thickness would potentially limit the capacity of a part to hold compression forces. Adding a reinforcement or ledge at the edge can provide extra support when needed, without changing the overall part's thickness and therefore saving unnecessary material;

- A different take is *rule 6*, which counteracts a tension force -an outward force normal to the interface's direction- by constraining its movement with a bow-tie spline. A common strategy in woodworking, this bow-tie adds a small taper angle in the opposite direction in which the parts tend to fall to the ground, preventing them to do so;

- Alternatively, re-orienting the interface to a normal direction of an axial compression force could be a solution in materials such as stone, as described in *rule 3*.

Another group to consider is the ease of assembly rules (*rules 12-14).* For example, *rule 13* can provide protection of sharp edges to avoid chipping - a common rule applied in any context where products are made out of aggregate materials such as stone. The parallel in concrete would be a chamfer, a very common detail on reinforced concrete exposed columns.

A set of fabrication-related rules are also considered in this grammar (*rules 15-16*). These rules are in place to provide a clean manufacturing product. Both *rules 15 and 16* contemplate the difficulty of producing clean concave corners when using a round tool for cutting by subtracting the corner tip in advance to allow for a tight fit of a peg. *Rule 15*'s fillet is commonly known as a dogbone operation in CNC detailing.

Rules can also have combined functions. For example, *Rule 13*, the fillet rule for smoothing corners, has an important safety function. On everyday tech products, these fillets protect users from being constantly in contact with sharp edges. Another multifunctional rule is *rule 11*, replace material. When applied, it can preserve weak edges, add more grip or flexibility, change the part's expansion properties, and even add an accent on where the piece ends at the same time.

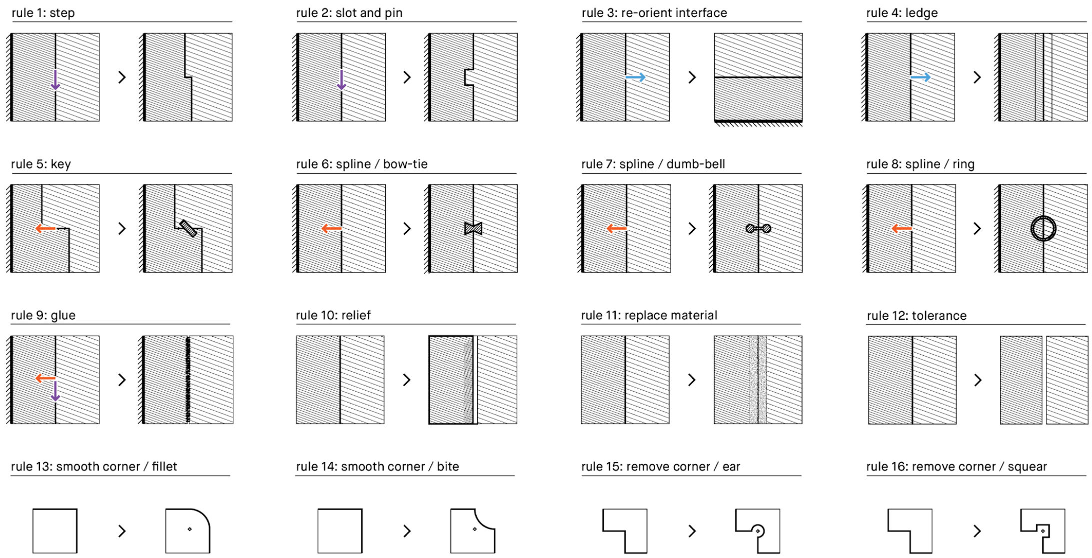

Figure 1. Joint grammar rule-set. Joint grammar rule-set with shear (purple), tension (red), and compression (blue) forces.

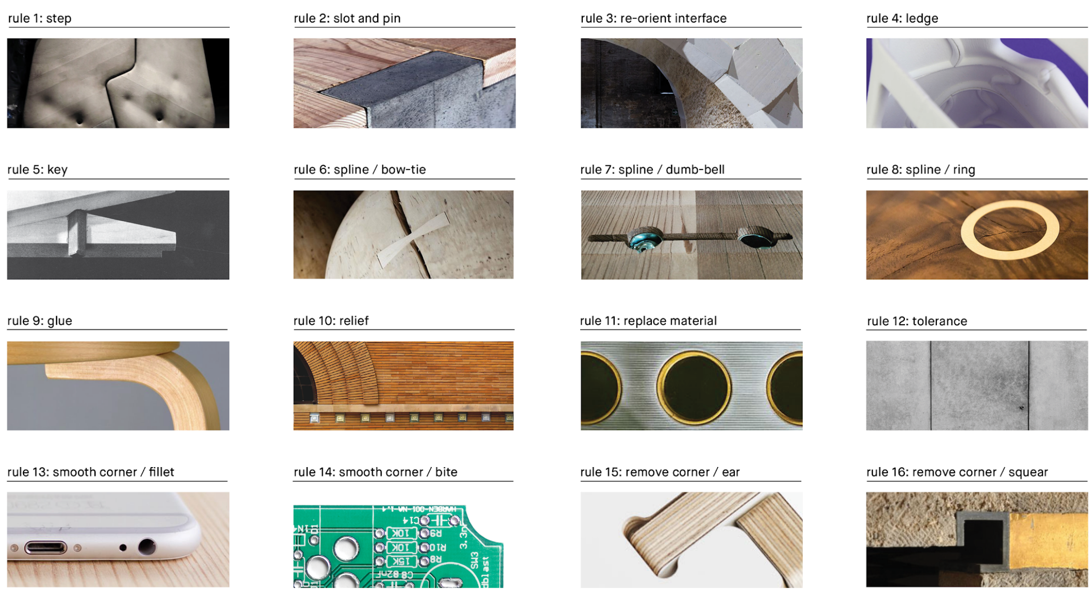

Figure 2. *Joint grammar rule-set applications across different materials and techniques*. First row from left to right: Brandon Clifford, *Patty & Jan*, 2019; Timbermill, *Deus ex machina concrete-wood detail*, 2013; Block Research Group and ODB Engineering, *The Armadillo Vault*, 2016. Photography: Iwan Baan; 4. Larry Sass, Paloma Gonzalez Rojas, *Constructive Design: Rule Discovery for 3D Printing Decomposed Large Objects*, 2015. Second row: Kiyosi Seike, Isuka-tsugi, c.1977; Joshua Vogels, *Vase with butterfly joint*, 2013. Photography: Rose Callahan; *Tite-joint fastener*, 2015; Jeff Martin, *Bronze ring spline*, 2015. Third row: Alvar Aalto, *Stool 60*, 1933. Photography: Maija Holma; Frank Lloyd Wright, *V. C. Morris Gift Shop*, 1948; Jean Prouvé, *Prouvé House*, c.1940; Peter Zumthor, Kunsthaus Bregenz, 1997. Photography: Richard Winter. Fourth row: Apple Inc., *iPhone 6*, 2015; SynthRotek, *Cosmic ECHO Squared*, 2015; Fraaiheid, *+/- Table*, 2013; Carlo Scarpa, *Gavina Showroom*, 1963.

In contrast to the material-independent grammar presented here, a material-based grammar would include only rules related to possible fabrication methods within a specific technique. For example, a wood detail grammar would consist of woodworking rules where specific shaping tools are used. A precast concrete detail grammar would include only rules that can be applied to formwork design. In a stone detail grammar, the way to cut and protect the edges from chipping would be of particular interest. Moreover, other trans-material grammars could generate unexpected translation methods from one technique to another.

As we have seen here, rules are the units, the distinct principles we use to solve specific challenges. The next step is to look at how we can use them in combination to respond to, often contradicting, demands.

Applying rules

Usually, designers tackle the different challenges during the design of a product and its joints with a set of design moves. The rule-based approach enables making these moves more explicit.

Rules can be applied manually or automatically. If rules are applied manually, the designer is in full control of every iteration of the process. For more complex cases, the rules' left and hand sides can be encoded. The result is a generative system to automatically create joint designs. In both manual and automatic implementations, functional rules require extra features or parameters to describe them. In our joint grammar, numerical parameters should include sizes and angles of the right-hand states. In *rule 1*, for example, there are three parameters: step angle, step width, and step height.

Certain aspects concerning the application of rules are relevant to the design of joints. Because rules are self-contained geometric operations applied on geometric units such as lines and vertices, they can be combined freely in different orders to generate alternative joint solutions. In other words, the sequence in which rules are computed defines the final outcome. In principle, we are free to combine them in any order —strongly suggested activity to try out at home. However, after many tries, some sequencing patterns arise. For example, functional rules are usually applied first, followed by ease of assembly rules and fabrication rules. Figure 3 shows a set of material-independent and functionless designs using this set of rules.

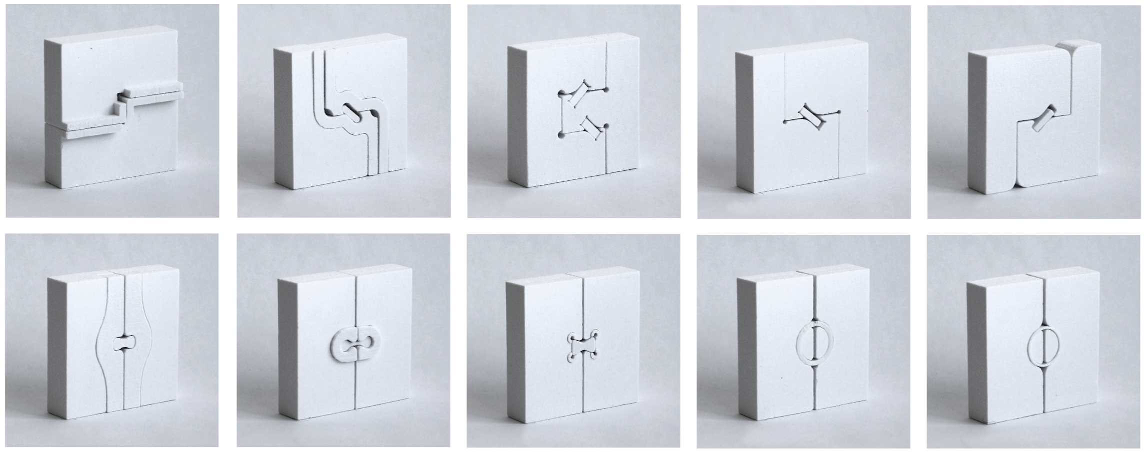

Figure 3. Rule-based designs of joints, Zprints.

It is also important to inspect the left-hand descriptions to know when it is possible to apply rules. Some rules are only used in the corners, and some only on parallel lines. Labels can as well communicate the need for more information. For example, this grammar contains labels in the shape of arrows that represent intervening forces between parts.

Joint rules in context

Joint grammars get more interesting when applied in the context of assemblies. This task includes preliminary evaluation steps to identify the design requirements for joint design, such as finding edge conditions and interpreting the stability and assembly requirements.

The case study of a discretized funicular funnel shell with non-standard edge conditions shown in Figure 4 served as a testing ground for applying a similar detail grammar (Ariza 2016). Here, the goal was to eliminate the use of scaffolding during assembly by using only joint details. To do so, we performed a stability analysis piece by piece with finite element (FE) analysis software (Preisinger and Heimrath 2014). This helped us understand the forces acting on each piece at each stage of the assembly. The reaction forces taken from the FE analysis were then categorized according to their type, made visible to the designer, and matched one by one (first shear, then tension) with our grammar rules' left sides. Finally, we consider the rule's parameters. In this case, size parameters are proportional to the force required to stabilize the structure, and angle parameters depend on the intervening forces' vectors' direction.

At each iteration, we ended up with different rules to apply to solve the distinct stability constraints. An example can be seen in Figure 4 (left), where different tension rules were used to keep the parts in place. A conflict appears when both shear and tension constraints are needed in the same location, a condition that could be solved with our grammar by *rule 9*, glue. But this solution was in opposition to the designer's intention of allowing the structure for later disassembly, so it was not used. This highlights an essential characteristic of grammars: our solutions' design space depends heavily on how nourished our grammar is regarding the design requirements. If we look at joints as collision points between elements and functions, the larger the number of rules or design moves available to us, the larger the choice for resolving their conflicts will be. In the end, an elegant joint is no more than a successful negotiation between physics, budget, and aesthetics.

The type, distribution, and size of joints might change completely depending on several factors. In our case, the order of assembly of a structure had an important role. In this respect, an explicit and accessible joint grammar could allow users to read and interpret building information, such as the assembly sequence of a structure or the type of intervening forces. If we are equipped with a joint grammar, the final object can be readable even after the design phase has ended in a sort of assisted reverse engineering approach.

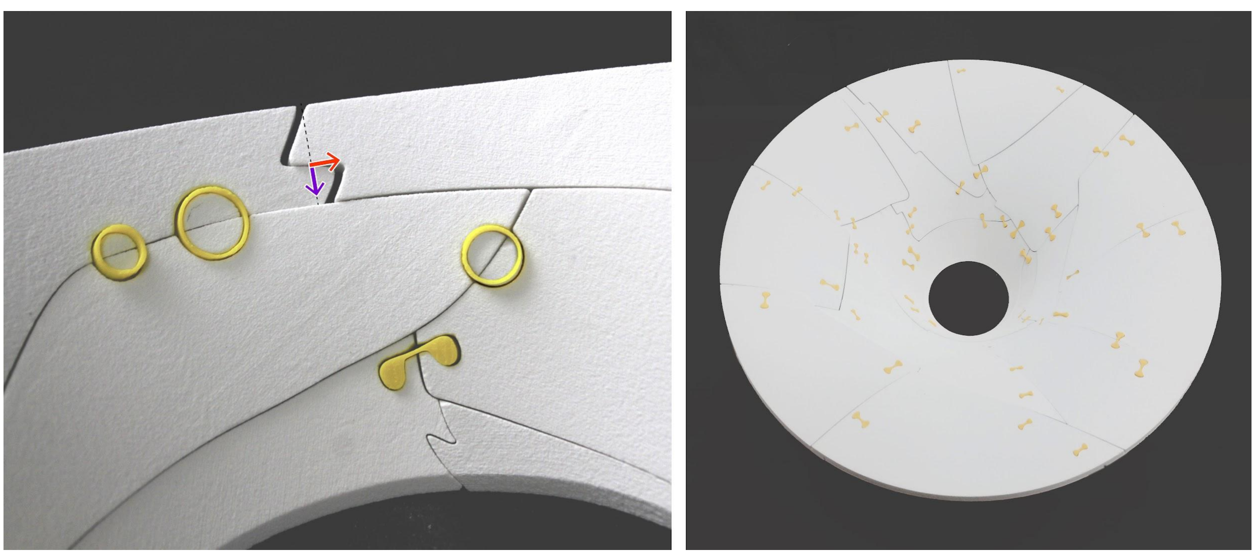

Figure 4. left: Joint grammar prototype showing applications of rule 7 (spline/dumb-bell), rule 8 (spline/ring), and rule 1 (step). Right: structure assembled using only joint grammar details without additional supports.

Unlearning joints

We have seen that grammars are not only useful for designing but also for looking at known topics with fresh eyes. Looking at synthetic designs such as joints equipped with these basic principles or rules can train everyone in building their own detailing knowledge. The process of looking into singular principles of joint design and making them explicit one at a time forces us to discriminate, name, and explicitly engage with aspects that are often too ordinary to be discussed. Setting up a rule-based system can even help novice designers learn about a particular material or manufacturing technique.

Returning to the concept of detail as a compound of diverse motivations and functions, we can now go back to historical examples and see them in a new light. We can learn so much from every object around us by having a method to decode them in simple principles. We can start by asking: *why are the borders of the keyboard keys where I am writing slightly round? How many fillets, chamfers, and splines can I find on the chair I am sitting in? Where are they located? Why are materials cut in this or that direction? Is that anything to do with the direction of forces? or rather a decision based on its manufacturing process?* And so on…

Compelling designs might contain multiple reading levels, and certainly, one of them is their constituent rules. Going back to where we started, we can see computational thinking as an efficient tool that could save us a lot of time, but more importantly, it can give us an enhanced pair of eyes.

Bibliography

Ariza, Inés. 2016. "Decoding Details: Integrating Physics of Assembly in Discrete Element Structures." Masters' thesis, Massachusetts Institute of Technology.

Mueller, Caitlin T. 2014. "Computational Exploration of the Structural Design Space." Doctoral dissertation, Massachusetts Institute of Technology.

Preisinger, Clemens, and Moritz Heimrath. 2014. "Karamba—A Toolkit for Parametric Structural Design." *Structural Engineering International* 24 (2): 217–21.

Radford, A.D., and J.S. Gero. 1985. "Towards Generative Expert Systems for Architectural Detailing." *Computer-Aided Design* 17 (9): 428–35.

Stiny, George, and James Gips, 1971. "Shape Grammars and the Generative Specification of Painting and Sculpture." *IFIP Congress (2)*, vol. 2, no. 3: 125-135.

Victor, Bret. 2011. "Up and Down the Ladder of Abstraction." 2011. http://worrydream.com/LadderOfAbstraction/.

Woodbury, Robert. 2010. *Elements of Parametric Design*. London; New York, N.Y.: Routledge.