Guided Exploration of Physically Valid Shapes for Furniture Design

Nobuyuki Umetani
University of Tokyo

Takeo Igarashi
Univ. of Tokyo / JST Erato

Niloy J. Mitra
University College London

Abstract

Geometric modeling and the physical validity of shapes are traditionally considered independently. This makes creating aesthetically pleasing yet physically valid models challenging. We propose an interactive design framework for efficient and intuitive exploration of geometrically and physically valid shapes. During any geometric editing operation, the proposed system continuously visualizes the valid range of the parameter being edited. When one or more constraints are violated after an operation, the system generates multiple suggestions involving both discrete and continuous changes to restore validity. Each suggestion also comes with an editing mode that simultaneously adjusts multiple parameters in a coordinated way to maintain validity. Thus, while the user focuses on the aesthetic aspects of the design, our *computational design* framework helps to achieve physical realizability by providing active guidance to the user. We demonstrate our framework on plank-based furniture design with nail-joint and frictional constraints. We use our system to design a range of examples, conduct a user study, and also fabricate a physical prototype to test the validity and usefulness of the system.

Keywords: shape space, guided exploration, shape analysis, physical validity, sensitivity analysis, computational design.

Links: DL PDF WEB VIDEO DATA CODE

1 Introduction

Advances in 3D modeling systems (e.g., Blender, Google SketchUp, etc.) have enabled novice users to design shapes by themselves, thus making content creation widely accessible. However, along with aesthetic appeal of the designed shapes, their physical properties are often very important, especially if the resulting model is to be used in the real world. For example, in the context of do-it-yourself (DIY) furniture design or machine assembly, various physical constraints need to be satisfied — a chair is only useful if it remains stable and does not break under target load distributions. Unfortunately, current modeling systems rarely consider such physical plausibility in the design phase. This makes creating interesting shapes, which also satisfy physical constraints, difficult for those users without domain knowledge and relevant experience.

Traditionally, geometric design and physical functionality are considered independently. In a typical setting, a designer creates a 3D geometric shape that is then validated using a physical simulator, e.g., a finite element method (FEM) solver. If the shape violates one or more physical constraints, it is sent back to the designer who then refines the shape. The process is iterated until a satisfactory design is found. Such a workflow is undesirable: the process (i) is time consuming, essentially amounting to trial-and-error, (ii) provides no guidance to the designer on how to rectify the current constraint violation(s), and (iii) encourages users to opt for standard geometric shapes, thus discouraging novel shape exploration.

For example, Ikea provides a range of design-at-home tools specialized for offices, kitchens, bedrooms allowing the user to prescribe room dimensions, interactively select 3D models from product catalog, place them in the room, and plan a layout. However, such systems allow users only to select from a list of fixed objects. With the growing demand for customization, an ideal system should also allow users to change the shape of the furniture, while still being guaranteed that the objects remain functional, e.g., bookshelves do not collapse under target loads. Thus, our goal is to support *computational design* via real-time exploratory modeling [Talton et al. 2009; Umetani et al. 2011; Yang et al. 2011]. We investigate this in the context of nail-jointed furniture targeted towards producing unusual and artistic shapes with non-standard inclinations, while still ensuring physical validity (see Figure 1).

A few advanced CAD systems (e.g., CATIA) support continuous feedback to check for the validity of designed models. Based on a similar motivation, Umetani et al. [2011] propose an interactive system to give real-time feedback on physical constraints for garment simulations. Such methods, however, only tell whether or not

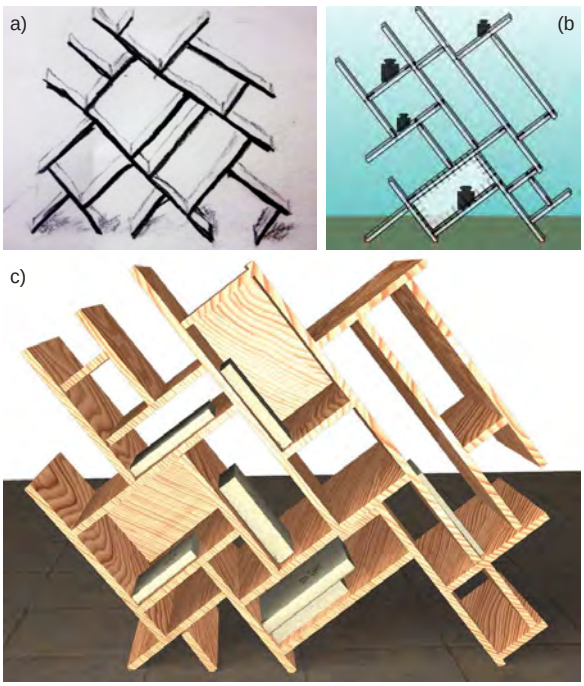

Figure 1: *Modeling a design concept (a) often produces invalid 3D realizations (b) due to model instability (i.e., toppling) or non-durability (i.e., excessive joint force) under target loads. Our interactive computational design framework supports guided shape exploration to help the user reach a valid configuration, which can then be readily manufactured (c).*

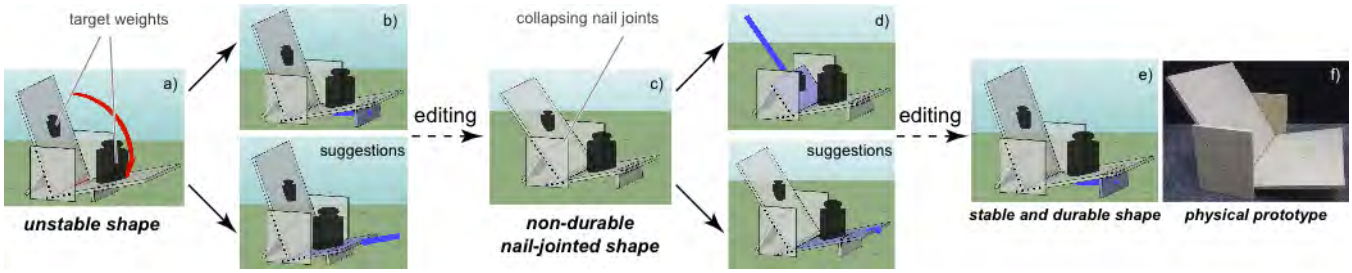

Figure 2: *Starting from a design (a) that is physically invalid due to model instability (i.e., toppling) or non-durability (i.e., excessive joint force), we propose design suggestions (b, d) to restore physical validity. The suggestions provide guided shape space exploration to the user, who can quickly realize valid nail-jointed furniture designs under target weight bearing and practical material specifications (e, f).*

the model is valid; they do not suggest *how* to restore the model's validity. In a notable effort, Whiting et al. [2009] directly optimize procedurally generated buildings over a range of free variables to produce a final model that is structurally stable. However, such an approach is unsatisfactory for exploratory modeling as it neither provides creative support, nor facilitates informed exploration.

Given an initial shape and domain-specific geometric and physical constraints, we propose a computational design framework for efficient and intuitive exploration of valid shapes. Specifically, we actively guide the user to explore those parts of the shape space that satisfy the constraints, thus relieving the user of the burden to ensure realizability via the following modes: (i) We analyze the current shape configuration and indicate the valid range of the parameter being edited. (ii) We also propose both continuous and discrete suggestions with coordinated editing modes to restore validity when the current design is invalid. Note that in contrast to a direct optimization-based solution, we leave the designer in control of form-finding: we provide visualization of valid range and multiple deformation suggestions guiding the designer towards feasible geometric forms, as needed (see Figure 2).

In this work, we enable constrained modeling in the context of a (nail-jointed) furniture design system under geometric and physical constraints. Specifically, we consider three aspects: (a) *connectivity*, i.e., joint connections among planks are geometrically maintained, (b) *durability*, i.e., the object does not break at joints under target load distributions, and (c) *stability*, i.e., the object does not topple or lose contact with the ground. The user interactively designs a shape model using standard modeling operations. In the background, we continuously run simulations of rigid bodies with frictional contact to provide real-time feedback about the structural validity of the design. The system performs sensitivity analysis to understand how design changes affect the *validity* of the design. We use this information to provide both a range of valid parameters being edited and also continuous suggestions to restore validity using novel force-space analysis. Each suggestion comes with a coordinated edit mode that synchronously adjusts multiple components, which is otherwise difficult for users to guess, especially with non-linear constraints. Thus, the user can efficiently navigate the physically valid shape space by following the visualized range and exploring the proposed suggestions (see Figures 1, 17, supplementary video, and demo).

We tested our framework to design a range of furniture under different loads. Our system supports real-time handling of up to 10-20 rigid bodies on a 2.7GHz laptop. Using our system, users can quickly and reliably design valid furniture, often with planks arranged in non-standard configurations. We fabricated a physical prototype and stress-tested the realization to target specifications. We envision that our technique can be easily integrated with a range of modeling tools, enabling novel function-aware form-finding possibilities.

Contributions. In summary, we propose

- an interactive modeling framework to design valid shapes under geometric and physical constraints;
- a design environment for nail-jointed, plank-based furniture modeling with frictional contact, and implicit simulation of rigid body motion; and
- a force space sensitivity analysis to generate design suggestions with continuous and discrete modifications to restore geometric and physical validity.

2 Related Work

Suggestive modeling. Advances in geometric modeling have resulted in well-established CAD modeling tools. Exploratory design, however, remains challenging. This is mainly because mapping a partially formed design concept to a final 3D shape is ambiguous. Hence, researchers have proposed various frameworks for suggesting possible shapes to inspire and guide the user. Based on user-specified geometric relations across 3D components, Igarashi et al. [2001] generate a gallery of possible subsequent modeling operations to facilitate quick and intuitive modeling. In another influential work, Funkhouser et al. [2004] propose a data-driven modeling system in which the user provides the conceptual design using sketches, while the system suggests plausible geometric realizations by searching through a database of 3D models. Inspired by this philosophy, Chaudhuri and Koltun [2010] propose a data-driven system to compute and present components that can be added to the current design shape. Later, they extend the idea to a probabilistic suggestion system for part-level assembly-based 3D modeling [Chaudhuri et al. 2011]. The Insitu system [2011] provides spatial context by fusing data from multiple sources and combines them with image-billboarding to provide light-weight 3D environments for professional conceptual designs. In the context of appearance modeling, Kerr et al. [2010] perform a user study to test the effectiveness of suggestive interfaces. They conclude that such an interface is well suited for artistic exploration even for novice users, but they also mention that interactivity of such a system is critical. In this work, we introduce a novel suggestive system for exploring only those shapes that are physically valid.

Physical simulation. Given a 3D shape, state-of-the-art techniques can efficiently and accurately evaluate its physical validity. However, the inverse problem is less understood. In the context of animation, even small perturbations in initial parameters can lead to large changes in final configurations. Hence, Popovic et al. [2000]

propose a system to optimize parameters to satisfy user annotated keyframe specifications. In another interesting formulation, Twigg and James [2008] introduce backward steps to generate animations of rigid bodies with desired goal configurations. In this work, instead of motion parameters, we explore how shape changes affect the physical validity of a shape. We then use the information to interactively propose smart shape deformation modes.

The discontinuous nature of friction makes its correct handling challenging [Kaufman et al. 2008]. Rigid body motion is often solved using impulse-based methods or the Linear Complementarity Problem (LCP) formulation (see [Baraff 1994; Stewart and Trinkle 2000] and references therein). Such methods assume that the direction of frictional forces depends on the tangent velocity, thus making estimation of frictional forces acting at the contact points challenging. Since sensitivity analysis of contact forces to design changes plays a crucial role in our framework, we propose a simple way to handle such changes accurately. Earlier, Chenney and Forsyth [2000] extend traditional simulation models to include plausible sources of uncertainty and use a Markov chain Monte Carlo algorithm to sample multiple animations while satisfying a set of constraints to ensure the physical plausibility of animations.

Interactive shape exploration. Immediate and meaningful feedback is essential in any design setting, especially in artistic exploration (see also [Kerr and Pellacini 2010]). Although such design spaces are often high dimensional, only low-dimensional subspaces are typically useful to intuitive exploration. In a data-driven setting, researchers have extracted low-dimensional embeddings (e.g., using mixture of Gaussian models) of desirable design spaces for appearance [Shapira et al. 2009] and for geometric modeling [Talton et al. 2009]. Recently, Ovsjanikov et al. [2011] study variation patterns directly in appropriate descriptor spaces to extract low-dimensional deformation models on a representative template for exploration and navigation of collections of 3D models. In another approach, the deformation framework iWires [2009] demonstrates that direct preservation of inter- and intra-part relations using junction curves is effective for manipulating man-made models. In a related attempt, Yang et al. [2011] propose a geometric framework to identify constrained modeling spaces where appropriate geometric properties (e.g., planarity of quad faces) are preserved in course of deformations and edits. Such exploration approaches, however, mostly focus on geometry, and physical validity considerations have so far been ignored.

Design optimization. Different optimization strategies have been proposed for various design problems: voxel selection and placements to create multiple target shadows [Mitra and Pauly 2009], relief optimization for prescribed shadow footprints [Alexa and Matusik 2010], furniture layout while increasing functional considerations such as accessibility, etc. [Merrell et al. 2011; Yu et al. 2011], or optimizing combinations of materials to reach target deformation behavior [Bickel et al. 2010]. In the context of buildings, Smith et al. [2002] model truss structures by structural optimization, while Whiting et al. [2009] optimize free variables in the context of procedural modeling with regards to structural feasibility by ensuring non-negative force between brick elements. These approaches propose final *optimized* shapes, which are not beneficial in initial exploratory stages. Instead, we introduce shape space investigation to understand the effect of geometric changes on physical validity and use the findings to expose the valid and useful parts of the shape space as suggestion modes (see also [Yang et al. 2011]).

In the context of design rationalization, researchers have worked on minimally changing input designs while maximizing repetitions across molds or triangular patches, thus enabling economic construction of free-form surfaces [Eigensatz et al. 2010; Singh and Schaefer 2010]. These methods, however, are neither integrated with the design phase, nor do they consider any physical durability constraints of shapes.

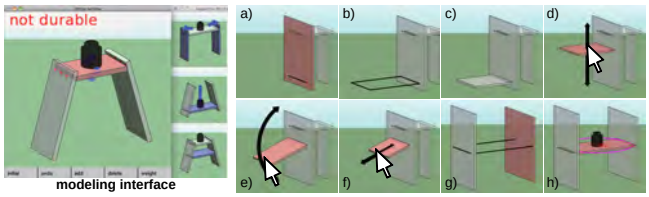

Figure 3: *(Left) The modeling interface consists of the modeling and the suggestion panels. (Right) The modeling interface with typical stages shown: creation, connection, translation, scaling, and rotation of a plank, along with placing a weight.*

3 System Overview

Overview. Figure 3-left shows our modeling interface, which consists of a modeling panel and a suggestion panel. The modeling panel basically works as a standard modeling system (e.g., Google SketchUp), although it is specialized for models consisting of multiple planks connected by nail joints. Our system continuously checks for validity in the background and shows whether or not the configuration satisfies geometric and physical requirements. Specifically, the system examines connectivity, durability, and stability. Note that we do not check for self-intersections at runtime. The result of the analysis appears as an annotation in the main panel during mouse dragging. Further, we provide suggestions in the suggestion panel after mouse release if the current shape is invalid. Suggestions, when selected, appear in the modeling panel.

Modeling user interface. Figure 3-right shows the basic modeling operations supported by our system. The user draws two 2D lines on the screen to specify a new rectangular plank (a-c) of predefined thickness (12 mm in our setting). The first line is drawn by mouse dragging and is placed on a selected plank. The end point of the first line becomes the starting point of the second line and its end point is indicated by a mouse click. The second line is either projected to an existing plank or aligned to the canonical xyz-axis. We automatically generate a joint between the newly created plank and the existing planks on which the first and second line are placed. The user can translate, rotate, and scale a plank using 3D widgets (d-f). When an edge of a plank is placed near another plank, these planks are automatically connected (g). Finally, the user places a weight by clicking on a plank in the weight mode (h). Note that the exact placement of the weight on the selected plank is not important.

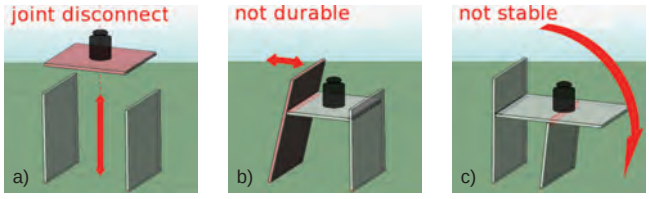

Figure 4: *Warnings flagged for invalid configurations: Joints get disconnected (a), a model becomes non-durable due to excessive force on the nails (b), or it becomes unstable, i.e., topples (c).*

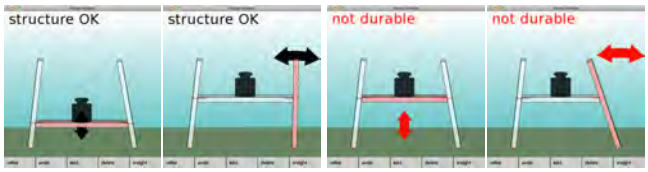

Figure 5: *Range indicators. Range is shown in black when the current configuration is valid and in red when invalid.*

Validity visualization and suggestions. In Figure 4, we show the different scenarios when the current configuration becomes invalid. (a) When a joint becomes disconnected, the system highlights the joint in red. (b) When the model breaks at a joint, the system also highlights the joint in red. (c) When the model falls down, the system shows a red arrow. These warnings automatically appear and are continuously updated as the user interacts with the design, so that the user can move back to a valid state by direct manipulation based on the feedback.

In addition to checking whether or not the current configuration is valid, the system computes the valid *range* of the parameter (degrees of freedom, DOF) being manipulated and shows it to the user during direct manipulation (mouse drag). When the current configuration is valid, the system shows the valid range in black. When the current configuration is invalid, the system shows the valid range in red (see Figure 5). Explicitly showing the valid range reduces the need for trial and errors to stay within or return to a valid state during direct manipulation editing.

When necessary, after each mouse release, the system provides suggestions (capped to a maximum of 8 in our setting) on how to resolve an invalid state. When a joint becomes disconnected, the system shows how to reconnect it (Figure 7a). When the model is non-durable or unstable, the system shows how to make it durable and stable (Figure 7b, 7c). Each suggestion consists of a representative configuration and an optional coordinated edit mode. When the user clicks on a suggestion, the representative configuration appears in the modeling panel together with arrow marks indicating the coordinated edits (Figure 6a). The user drags one of these arrows to make coordinated editing, thus allowing the user to control multiple DOFs of a model simultaneously while satisfying the required constraints. These multiple DOFs are coupled together, i.e., the user cannot fix the non-durability or instability moving each DOF individually. For example, in Figure 6, if the user slides the top board of the table toward the left, the angle of the left leg becomes perpendicular to the ground to compensate for the increase of the bending force on the left joint (Figures 6b, 6c).

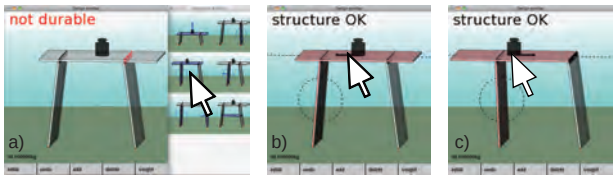

Figure 6: *Example of coordinated editing using suggestions. The table is non-durable and the system gives multiple suggestions (a). The user clicks on a suggestion and it appears in the modeling window (b). The user can change the position of the top board and left leg simultaneously by dragging any of the arrow handles (c).*

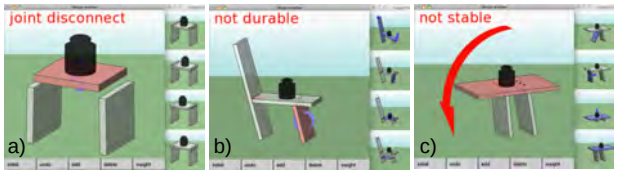

Figure 7: *Example of suggestions. A joint is connected (a), the model is made durable (b), and the model is made stable (c).*

4 Algorithm Overview

As the user edits the model (i.e., adds, removes, translates, rotates, or scales a plank), we first try to the satisfy geometric constraints, i.e., joint connectivity and ground contact, by adjusting the length of the other planks. If we fail to satisfy the geometric constraints, we suggest discrete changes to fix the design. After the model satisfies the geometric constraints, we check the physical validity of the current shape and present the result to the user. We test for durability and stability, which amounts to checking for inequality constraints on joint and contact forces, respectively. In addition to indicating that the design is valid or not, we also analyze how the validity changes with respect to further geometric modifications, i.e., what changes make the invalid model valid, and vice versa. The result of the analysis is used to compute valid ranges and make suggestions. Section 5 describes how we measure and analyze the physical validity, while Section 6 describes how we compute the valid range and make suggestions based on the analysis. Note that frictional contacts with the ground pose a challenge to the sensitivity analysis, and we present a method to address this issue.

5 Physical Validity

In our interactive framework, we continuously analyze the current design to provide feedback to the user about the physical validity of the current shape during the user's editing. Specifically, the system checks two types of physical validity: (i) whether or not the nail joint is durable, and (ii) whether or not the structure is stable. In this section, we first describe how to *measure durability* of a current design by solving constrained rigid body dynamics to obtain forces on the joint. Next, we propose a *sensitivity analysis* to analytically estimate changes in static equilibrium under infinitesimal perturbations of the current design. This analysis helps to generate editing suggestions as well as accelerate the computation of the validity.

5.1 Durability measurement

In any nail-jointed wooden structure, the joints form the weakest links, i.e., such structures primarily break at the joints rather than at other sections [Parker and Ambrose 1997]. Hence, in our framework, we model component planks of wooden furniture as assemblies of unbreakable rigid bodies, while focusing on the joint and the contact forces. We first define joint forces and then explain how to compute joint and contact forces for a given model. Next, we describe how to examine durability based on the obtained joint forces. Although most of the techniques explained in this section are standard in physical simulation, we describe them for completeness. An exception is the treatment of frictional contact. It is not trivial to handle frictional contact within the framework of sensitivity analysis and we present a novel method.

Definition of joint forces. We characterize each nail-joint connection as a constraint between the participating plank pairs. We describe static rigid body equilibrium under joint constraints using standard notation (see Figure 8 and [Geradin and Cardona 2001]).

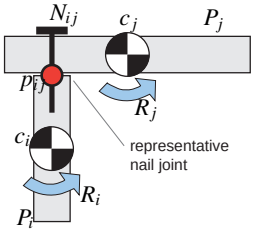

Figure 8: *Typical forces at a nail-joint.*

Let planks P_i and P_j be connected by a nail joint N_{ij}. Further, say that each plank P_i has an initial center position $\mathbf{c}_i \in \mathbb{R}^3$ and we apply a rotation $\mathbf{R}_i \in SO(3)$ followed by a translation $\mathbf{u}_i \in \mathbb{R}^3$. Note that, although plank pairs are connected using several nails at a nail-joint, for simplicity, we represent such nail positions using a single point $\mathbf{p}_{ij}$. The corresponding joint constraints are: (i) a translational part that keeps the participating planks together and (ii) a rotational part that prevents bending. Let,

$$\mathbf{d}^t_{ij} := [\mathbf{R}_i(\mathbf{p}_{ij} - \mathbf{c}_i) + \mathbf{c}_i + \mathbf{u}_i] - [\mathbf{R}_j(\mathbf{p}_{ij} - \mathbf{c}_j) + \mathbf{c}_j + \mathbf{u}_j]$$

$$\mathbf{d}^r_{ij} := vect\left(\mathbf{R}_i^T \mathbf{R}_j\right),$$

where *vect* is an operator that extracts the axial rotation vector of a rotation matrix. Note that since both $\mathbf{R}_i, \mathbf{R}_j \in SO(3)$ are rotation matrices, $\mathbf{R}_i^T \mathbf{R}_j$ is also a rotation matrix. At each nail-joint N_{ij} the joint constraints are:

$$\mathbf{d}^t_{ij} = 0 \quad \text{and} \quad \mathbf{d}^r_{ij} = 0. \tag{1}$$

The set of such constraints for a piece of furniture can be redundant (e.g., if a set of planks is connected in a loop) leading to an over-constrained system. As a solution, we allow for deviations from the exact constraints using a penalty method. Specifically, we measure deformation energy at joint N_{ij} as $E^{joint}(N_{ij}) := 0.5\|\mathbf{d}^t_{ij}\|^2/\varepsilon^t + 0.5\|\mathbf{d}^r_{ij}\|^2/\varepsilon^r$, which we include as the potential energy of the system (see Equation 3). The scalar values ε^t and ε^r are small constants (both set to 10^{-5} in our tests). The derivative of penalty function E^{joint} with respect to $\mathbf{d}^t$ and $\mathbf{d}^r$ are

$$\mathbf{h}^t = \mathbf{d}^t/\varepsilon^t \quad \text{and} \quad \mathbf{h}^r = \mathbf{d}^r/\varepsilon^r \tag{2}$$

and can be seen as constraint forces. We call such forces translation forces and bending forces (in engineering, commonly referred to as the bending moment), respectively. Note that these deviations $\mathbf{d}^t$ and $\mathbf{d}^r$ are influenced by the values of ε^t and ε^r, but $\mathbf{h}^t$ and $\mathbf{h}^r$ are not. The $\mathbf{h}^t$ and $\mathbf{h}^r$ have physical meaning relating to the equilibrium of the forces between planks.

Computation of joint and contact forces. In this work, we focus on the behavior of shapes under static equilibrium rather than dynamic motion of rigid bodies. We therefore compute forces applied to each joint by directly minimizing the total potential energy of the system with respect to $\mathbf{u}$, $\mathbf{R}$, and $\mathbf{h}$:

$$E^{total}(\mathbf{u}, \mathbf{R}, \mathbf{h}) = -\sum_{i}^{|P_i|} M_i \mathbf{c}_i^T \mathbf{g} + \sum_{ij}^{|N_{ij}|} E^{joint}(N_{ij}) + \sum_{k}^{|N_{contact}|} E_k^{contact}, \tag{3}$$

where M_i is the mass of plank P_i and $\mathbf{g}$ is acceleration due to gravity. The first term captures the gravitational potential energy; the second term models the joint energy; while the last term is due to contact forces as described later (we weigh the terms equally). With the total potential energy E^{total} being nonlinear, we iteratively minimize the potential energy using the Newton-Raphson method. Since the Hessian of the penalty term E^{joint} is ill-conditioned, we treat the constraint forces $\mathbf{h}^r$ and $\mathbf{h}^t$ as independent variables and explicitly solve for them. Specifically, we simultaneously minimize with respect to $\mathbf{h}^t$ and $\mathbf{h}^r$ along with $\mathbf{u}_i$ and $\mathbf{R}_i$ (see Equation 2). Note that since the Hessian of the total potential energy is indefinite, we

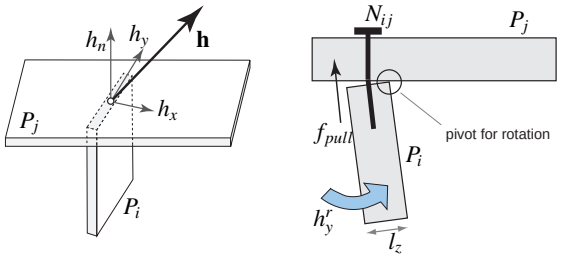

Figure 9: *(Left) Decomposition of a constraint force into components in local coordinates. (Right) The rotation force causes the nail pulling force, which can affect the joint's durability.*

damp the iteration by adding an identity matrix scaled by a small value (we use 10^{-5}) to the diagonal component of the translation and rotation for each plank. We continue the iteration until variables $\mathbf{u}$ and $\mathbf{R}$ stabilize. The solution to this minimization problem yields deformed plank positions $\mathbf{u}$, $\mathbf{R}$, and the joint force $\mathbf{h}$.

Joint durability. Having the translation $\mathbf{h}^t$ and the bending force $\mathbf{h}^r$ at each joint, we check for the durability of the nail-joint under the given forces. Mechanical properties of nails are well understood and have long been standardized with precise specifications on their load-bearing capacities (see [Bergman 2010]). At any joint, the loads on the nails are of two types: (i) a pulling force, which acts along the axis of the nail, and (ii) a shearing force, which acts vertical to the nail axis. We express force $\mathbf{h}$ (i.e., $\mathbf{h}^t$ and $\mathbf{h}^r$) in a local coordinate system: h_n represents the component normal to the joint face for plank P_j, h_x the component in a direction along the normal of P_i, and the remaining component is h_y (see Figure 9). Each component of $\mathbf{h}^r$ denotes the torque to twist the plank P_j with an axis of rotation in each direction. We assume the plank's thickness is smaller than the width of the joint between P_i and P_j. Hence, the bending force h_y dictates the collapse of the joint. In Figure 9, we show how the nail-pulling force arising from bending force h^r_y is modeled. The joint forms a lever with the length of the lever arm equal to $0.5l_z$. Specifically, we model the pulling force as

$$f_{pull} = \frac{1}{N_{nail}}\left(2|h^r_y|/l_z - h^t_n\right), \tag{4}$$

where l_z represents the thickness of the plank (12 mm in our tests) and N_{nail} denotes the number of nails at the nail joint N_{ij}. Then, the shear force is given by

$$f_{shear} = \frac{1}{N_{nail}}\sqrt{{h^t_x}^2 + {h^t_y}^2}. \tag{5}$$

Finally, we mark a joint as durable if both forces are within allowable threshold margins [Bergman 2010].

5.2 Sensitivity analysis

We now investigate how design changes affect the physical validity of a shape as this helps to accelerate the force computations (solving Equation 3) as the user changes the design. More importantly, when needed, the sensitivity analysis helps in generating suggestions for changing the design to restore validity by making the model durable and stable. Specifically, we locally compute a linear approximation to study how forces in equilibrium change with respect to changes to the current design, i.e., we perform a sensitivity analysis [van Keulen et al. 2005].

Let γ represent a shape configuration (see also Section 6). Using implicit rigid body analysis, the static equilibrium can be expressed as a linear system: $\mathbf{A}(\gamma)\mathbf{x}(\gamma) = \mathbf{b}(\gamma)$, where $\mathbf{A}$ is a square matrix and $\mathbf{x}$ is a vector encoding the positions and orientations of all the planks along with the forces $\mathbf{h}$ at the different joints. Vector $\mathbf{b}$ stores

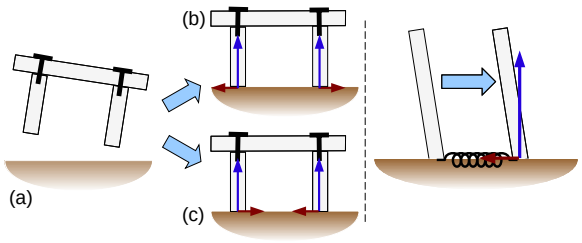

Figure 10: *(Left) Redundancy of the frictional contact forces. The red arrows show the friction forces and the blue arrows show the contact forces. A table with initial configuration (a) falls to the ground and can have multiple possible friction forces like (b) or (c). (Right) The penalty-based frictional force determines a unique friction force as a deviation from the initial position.*

the generalized external forces, i.e., forces due to gravity, contact, and friction acting on the planks. The sensitivity analysis gives

$$\frac{d\mathbf{x}}{d\gamma} = \mathbf{A}^{-1}\frac{d\mathbf{b}}{d\gamma}. \tag{6}$$

Although the configuration $\mathbf{x}$ changes nonlinearly with respect to any initial design change $\delta\gamma$, we found it sufficient to use $\mathbf{x} \rightarrow \mathbf{x} + d\mathbf{x}/d\gamma \cdot \delta\gamma$ as an initial guess to bootstrap the nonlinear iteration and achieve faster convergence.

Frictional contacts. We assume that the design structure is casually placed on the ground and not bolted to it. Hence, friction is essential to prevent sliding under horizontal force. For example, a table depends on friction to resist sliding under horizontal forces, say when we push the table sideways. Although a table with vertical planks as legs can easily support vertical loads, it is fragile even under slight horizontal perturbation, which is undesirable.

Performing an accurate sensitivity analysis with frictional contacts is challenging because frictional forces depend on the direction of the tangent velocities at the contact points. Sensitivity analysis, however, assumes static equilibrium with zero velocity at the contact points and hence cannot be used to determine friction force directions. Further, redundancy among frictional forces poses additional challenges [Klarbring 1990], e.g., even if a chair stands still, the combination of frictional forces is unknown, making it difficult to determine the internal forces (see Figure 10-left).

We propose a simple penalty-based method to address the above problems. In a standard dynamic setting, friction anchors are placed at the impact location and are relocated as the contact points slide with kinetic friction [Erleben et al. 2005]. However, since our setting is static, we assume that (i) all contact points are exactly on the ground and (ii) the contact states do not change during interactions. This allows us to uniquely determine the anchor position with respect to the initial configuration (see Figure 10-right) and analyze frictional force under design changes. Specifically, we place the contact points at the corners of planks that touch the ground. When the user sketches a plank, we detect the plank corner that touches the ground, and mark it as a contact. Note that during design changes we ensure that the contacts touch the ground without penetration or floating in the air (see Section 6). For sliding, we relocate friction anchors so that the (friction) springs do not generate excessive force beyond the limit of Coulomb friction.

6 Exploration of Valid Spaces

In this section, we describe how our framework guides the user towards the *valid* subspace of the configuration space Γ. If the current design is valid, we indicate the range of user manipulations

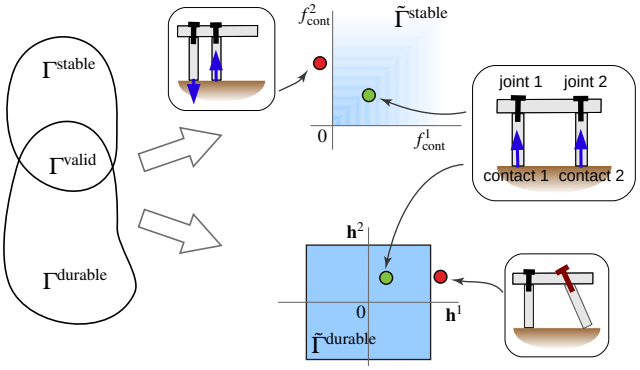

Figure 11: *A shape space point is valid if it is both stable and durable. For invalid shapes, we propose deformation suggestions to return to the valid part of the shape space. We work in force spaces defined by contact forces and bending forces for stability and durability, respectively thus simplifying the problem. Specifically, stability amounts to contact forces being restricted to the first quadrant, while durability amounts to bending forces being restricted to a durability rectangle. Note that although in this example the force spaces are 2D, in general we work in high-dimensional spaces.*

that keeps the design validity. On the other hand, when the current design becomes invalid, we make multiple suggestions to restore validity. Note that even though the (unconstrained) configuration space is high-dimensional, our computational framework only exposes meaningful (i.e., valid) suggestions, thus greatly simplifying the user's task. We make both continuous and discrete suggestions: while continuous suggestions leave the inter-plank joint topology unchanged, discrete suggestions involve adding support materials.

6.1 Geometric constraints

Aside from the physical validity of the shape, i.e., its durability and stability, shapes designed in our system are geometrically restricted by two constraints: (i) geometrical joint constraints and (ii) contact constraints (see Section 5). We first restrict the design space where the shape satisfies these geometrical constraints and then investigate the physical validity. Each plank has 8 degrees of design freedom: 3 for translation, 3 for rotation, and 2 for edge lengths around the plank faces (the plank thickness is fixed). For each degree of design freedom of the planks, we ensure that the contact constraints and joint constraints are satisfied by adjusting the length of the planks (Figure 12-left). Further, some degrees of freedom are invalid, e.g., if both sides of a plank are nailed, the plank length cannot be adjusted (Figure 12-right). We identify and remove such invalid degrees of freedom from the design space. Note that if there are C number of plank components and $\#\text{DOF}_{invalid}$ number of invalid design degrees of freedom, the constrained design space Γ has dimensions of $N_\gamma := 8C - \#\text{DOF}_{invalid}$. Each basis corresponds to one plank's translation, rotation, or length change and the adjacent planks' length change. We scale the translation and length change basis with the inverse of the size of the maximum bounding box edge length to make the translation and length change DOFs dimensionless, like that of rotational DOFs. Next, we enable exploration in a physically valid subspace of a constrained design space Γ.

6.2 Valid space

Recall that a shape is physically valid if two conditions are satisfied: (i) the shape is *durable*, which amounts to each joint having both

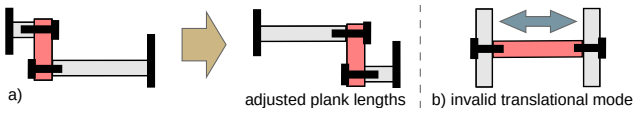

Figure 12: *Constrained design modes: (a) the lengths of neighboring planks of the edited planks are adjusted so that joints stay connected; (b) a translation mode is invalid if both sides of the planks are jointed.*

pulling and shear forces below allowed thresholds, written as

$$|f_{pull}| \leq f_{pull\ max} \text{ and } |f_{shear}| \leq f_{shear\ max} \quad \forall N_{ij}, \tag{7}$$

and (ii) the shape is *stable* (i.e., it does not topple), which amounts to each contact point having a non-negative contact force f_{cont} in the direction normal to the ground, written as

$$f^l_{cont} \geq 0 \ \ \forall \text{ contact points } l. \tag{8}$$

Let the corresponding subspaces of the configuration space Γ be $\Gamma^{durable}$ and Γ^{stable}, respectively. Thus, the valid shape space is $\Gamma^{valid} := \Gamma^{durable} \cap \Gamma^{stable}$. When the current design becomes invalid, the goal is to provide multiple suggestions to return back to the valid shape space (see Figure 11).

The valid space typically has a complex boundary since it is characterized by non-linear inequality constraints. Further, since the configuration space is high-dimensional, computing the exact boundary is difficult and time consuming. Also, it is nearly impossible to arbitrarily pick a valid shape directly from the high-dimensional space Γ^{valid}. Instead, we first pick several meaningful search directions to pursue, i.e., directions such that the invalid shape becomes valid under small manipulations. For each such direction, we use line search to identify configuration intervals where all the validity conditions are satisfied.

Since the boundaries of $\Gamma^{durable}$ and Γ^{stable} are characterized by force inequalities, we consider the valid shape space boundary in the *force space*, i.e., a coordinate space with the forces as the axes. This simplifies the problem as the boundary is then geometrically prescribed by the corresponding inequality. For example, with two contact points, the stable region is 2D and Equation 8 simply indicates that the first quadrant is the stable region (see Figure 11).

To efficiently characterize the joint durability force space, we make two approximations: (i) the translation force $\mathbf{h}^t$ in Equation 4 remains constant with respect to small design changes and only the bending force $\mathbf{h}^r$ varies and (ii) the shearing force in Equation 5 does not change under small design changes. These approximations are true when the bending force $\mathbf{h}^r$ is dominant and more sensitive than $\mathbf{h}^t$ under design changes. Thus, Equation 7 becomes

$$|h^r_y| \leq 0.5 l_z \left(f_{pull\ max} N_{nail} + h^t_n\right) = \Lambda_{max}. \tag{9}$$

Geometrically, the stable region Γ^{stable} is approximated as a high-dimensional, axis-aligned cuboid with edge lengths of Λ_{max} and centered at the origin in the joint bending force space (see Figure 11).

Note that the dimensions of the contact force space and the bending force space are lower than the configuration space dimension ($|\Gamma| \approx 8C$). Specifically, the contact force space has a dimension of the number of contact points, while the joint bending space has a dimension of the number of joints N_{ij}. Next, we describe how to efficiently search for directions in this simplified representation. We denote the boundaries of the stable and durable force space as $\tilde{\Gamma}^{stable}$ and $\tilde{\Gamma}^{durable}$, respectively.

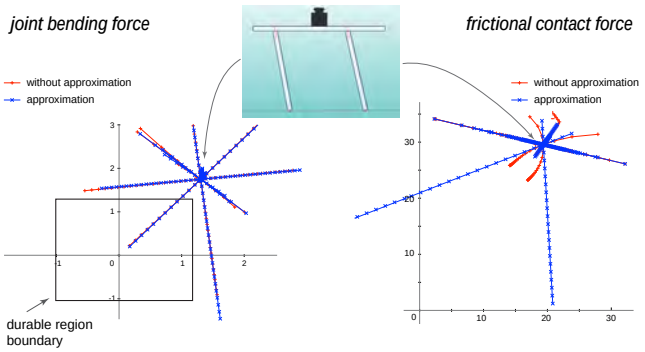

Figure 13: *Comparison of a stable shape and a durable space with and without approximation.*

6.3 Visualization of the valid range

During direct editing, we display the valid range of the parameter being manipulated. To do this, we evaluate the validity by changing the parameter. When the current configuration is already valid, the search proceeds in both directions until it becomes invalid to identify the bounds. When the current configuration is invalid, we first select the direction of the search using the result of the sensitivity analysis and then run a bisection search along that direction to identify the valid range, as explained next.

6.4 Continuous shape suggestions

When the current configuration becomes invalid, we compute several suggestions: (i) if only stability is violated, the system finds search directions to restore stability by analyzing the boundary of $\tilde{\Gamma}^{durable}$; (ii) if only durability is violated, the system finds search directions to restore durability by analyzing the boundary of $\tilde{\Gamma}^{stable}$; and (iii) if both stability and durability are violated, the system first proposes directions to restore stability and then restore durability.

Under local changes, we assume forces to vary linearly according to the design variations. Note that we use linearization only to select good directions (see Yang et al. [2011] for use of higher-order derivatives). After selecting a direction, we run a bisection search along the direction using all the nonlinear constraints without any approximation for actually computing valid designs. Figure 13 shows a sample comparison with and without linear approximation. If we cannot find valid shapes in the direction, we simply omit it from the suggestions (see Algorithm 1).

We found that simultaneously exploring the full design space involves searching over $|\Gamma| \approx 8C$ dimensions, which is impractical for real-time performance. Also, users can find suggestions involving variations across many parts to be confusing. Instead, we focus on suggestions involving at most M degrees of freedom for any suggestion (3 in our examples). We try all the possible combinations of selecting m design DOF-s ($m \leq M$), denoted by $\{\gamma_1, \ldots, \gamma_m\}$. We parameterize a search direction as a unit vector $\mathbf{s} \in \mathbb{R}^m$ with coordinates s_i such that $\sum_{i=1}^m s_i^2 = 1$ and the direction is $\mathbf{s} := \sum_{i=1}^m s_i \gamma_i$.

Durability-restoring suggestions. A desirable search direction $\mathbf{s}$ should quickly make the design durable, i.e., reach the boundary $\tilde{\Gamma}^{durable}$. Thus, for any direction $\mathbf{s}$ we look for

$$t^\star := \arg\min_t t\mathbf{K}_0\mathbf{s} + \mathbf{h}^r_{y0} \in \tilde{\Gamma}^{durable} \tag{10}$$

where matrix $\mathbf{K}_0 \in \mathbb{R}^{N_{ij} \times m}$ defines sensitivities of joint forces with respect to design changes $\mathbf{K}_0 := \nabla \mathbf{h}^r_y = \left[\partial \mathbf{h}^r_y / \partial \gamma_1 \ \ \ldots \ \ \partial \mathbf{h}^r_y / \partial \gamma_m\right]$

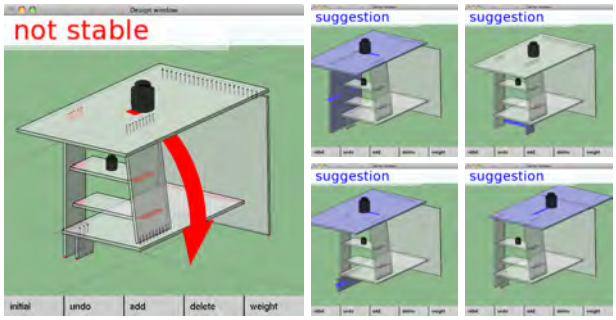

Figure 14: *Stability-restoring suggestions.*

evaluated at the current joint bending force $\mathbf{h}^r_{y_0}$. For interactive performance, instead of finding the minimum step t along direction $\mathbf{s}$, we compute the search direction $\mathbf{s}$ that takes us closest to the origin. Specifically, we choose a direction such that

$$\mathbf{y}^\star := \arg\min_{\mathbf{y}} \left\| \mathbf{K}_0\mathbf{y} + \mathbf{h}^r_{y_0} \right\|, \quad \mathbf{y} \in \mathbb{R}^m, \tag{11}$$

and use $\mathbf{s} = \mathbf{y}^\star / \|\mathbf{y}^\star\|$ to compute t using Equation 10. We use a brute force method and try all the possible $M!$ combinations in the configuration space taking advantage of the simple axis-aligned cuboid approximation of the durable region. Specifically, finding the search direction can be seen as detecting collisions of rays with the durability cuboid where sensitivity of the bending force $\mathbf{K}_0\mathbf{s}$ acts as a ray with its source at $\mathbf{h}^r_{y_0}$. Hence, we cull a direction if either the norm of the sensitivity $\mathbf{K}_0\mathbf{s}$ is small (≤ 1 in our tests) or the ray faces away from the cuboid.

Stability-restoring suggestions. We compute stable shape suggestions similar to the durability case. Specifically,

$$t^\star := \arg\min_t t\mathbf{L}_0\mathbf{s} + \mathbf{f}_{cont\,0} \in \tilde{\Gamma}^{stable}, \tag{12}$$

where $\mathbf{L}_0 \in \mathbb{R}^{N_{contact} \times m}$ is a sensitivity matrix of contact forces with respect to design changes $\mathbf{L}_0 = \nabla \mathbf{f}_{cont} = [\partial \mathbf{f}_{cont}/\partial\gamma_1 \ \ldots \ \partial \mathbf{f}_{cont}/\partial\gamma_m]$ evaluated at the position of the current contact force $\mathbf{f}_{cont\,0}$. We choose a direction $\mathbf{s}$ such that the shape quickly becomes stable. First, we project the current contact force vector $\mathbf{f}_{cont\,0}$ on the stable region to obtain $\mathbf{f}^*_{cont\,0}$ (i.e., clamped to zero) and choose the direction that gets us closest to $\mathbf{f}^*_{cont\,0}$ using a least squares minimization, i.e., $\mathbf{s} = \mathbf{y}^\star / \|\mathbf{y}^\star\|$ such that

$$\mathbf{y}^\star := \arg\min_{\mathbf{y}} \|\mathbf{L}_0\mathbf{y} + \mathbf{f}_{cont\,0} - \mathbf{f}^*_{cont\,0}\|, \quad \mathbf{y} \in \mathbb{R}^m. \tag{13}$$

Algorithm 1 Generating durability-restoring suggestions

```
Generate design modes {γ_0, ..., γ_{N_γ}}
Compute A^{-1} /* In Equation 6 */
Generate sensitivity of h, f_contact against all Dofs in Γ
C : set of combination of integer value
for m = 1 to M do
  for m number combination of modes c = {i_1, ..., i_m} do
    if all subset of c is not in C then
      Compute K_0 and y*        /* Equation 11 */
      Compute t*                /* Equation 10 */
      if t < 1 then
        C ← C ∪ c
      end if
    end if
  end for
end for
For c ∈ C, find range of durable shapes using bisection method
Order the suggestions based on the computed range
```

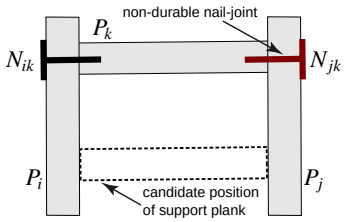

Figure 15: *Heuristic to add a support plank.*

6.5 Discrete shape suggestions

When the structure is not durable, we try to make it durable by adding a support plank as a reinforcement around a joint that is under excessive force. Typically, nail joints connect two planks nearly at a right angle, making it difficult to attach any support material between the planks connected by the non-durable joint. Instead, we try to connect two planks that are parallel to each other and put the supporting plank orthogonal to the planks. We use a greedy strategy. First, we choose a combination of two planks P_i, P_j such that (i) they are nearly parallel (we use $|\mathbf{n}_i \cdot \mathbf{n}_i| < 0.5$ where, $\mathbf{n}_i$ is the face normal of plank P_i, and the same for $\mathbf{n}_j$), (ii) between the two planks there is a third plank P_k connected to P_i and P_j by joints, and (iii) one or both joints N_{ik} and N_{jk} are non-durable. We suggest adding support material between P_i and P_j at a location chosen from several (rule-based) candidate positions so that the support material does not intersect with other planks (see Figure 15). We check for durability of the joint by running a physical simulation to make sure that the support plank is effective. The system tries many combinations of planks until it finds effective supporting planks based on standard rules used in woodwork [Bergman 2010]. Smarter strategies should be investigated in the future.

7 Results

In our system we consider furniture designs using 12 mm medium density fiberboard (MDF) with 32 mm nails, spaced at interval of 20 mm. Such a placement can take a maximum shear force of $f_{shear\,max} = 190$N and maximum pull force of $f_{pull\,max} = 35$KN/m [Parker and Ambrose 1997]. We set the coefficient of static friction to 0.5 in our tests. In our current implementation, we can regularly handle up to 10-15 plank designs at interactive speed. In each exploration session, the user progressively adds planks and proposes an initial configuration with the target load-bearing capacity. For example, in Figure 2, we put 50 kg weight on the horizontal plank and 15 kg on the supporting back plank. The final design was found after several iterations of suggestions and design explorations. We built a physical prototype (the construction took around 4 hours) and found it to behave satisfactorily under the target load (see video).

In Figure 16, we use our system to design non-conventional bookshelves. The computational support is critical as we have little intuition in such unusual situations and cannot benefit from prior experience. Guided exploration helps the user to explore the design limits while not having to worry about physical validity.

Figure 17 shows additional design sessions with our system. Note that we show only a few representative suggestions, while we refer the readers to the supplementary video and demo for details. The user is provided with corrective suggestions *only* when the design becomes invalid. Further, each suggestion comes with a range where the shape remains valid. Thus, even when the suggestion modes involve multiple planks, the user simply has to adjust a *single* parameter along the suggested deformation direction. For

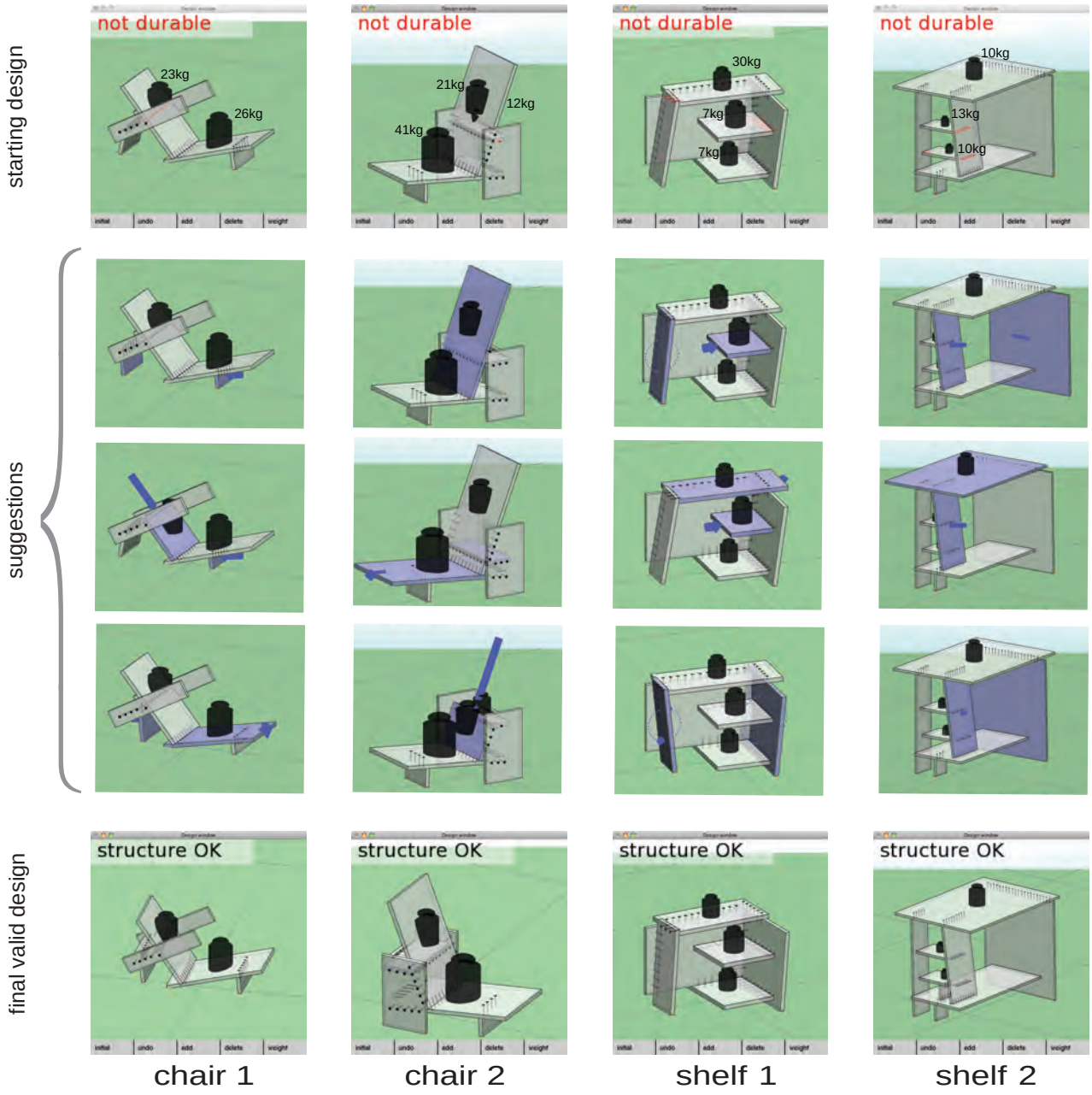

Figure 17: *Typical nail-jointed furniture design sessions in our guided exploration framework. Only a few suggestions are shown in each example. Please refer to the accompanying video and demo for further explanation. Note that suggestions often involve synchronous manipulation of multiple planks, which is difficult to perform without computational support.*

example, in the chair 1, we show 3 different suggestions each involving a pair of planks to be simultaneously manipulated to restore validity. In case of chair 2, the situation is similar, but we have 3 specified weights.

In the case of shelf 1, we note that geometrically the initial and final configurations are not very different. Even then, the validity-restoring path is non-trivial to find by trial and error, especially since there are different interactions involving simultaneous rotation and anisotropic scaling of multiple components. In the case of shelf 2, the top and the big side planks get adjusted over the course of the guided exploration to result in a shape that can withstand the three vertical loads. Note that the complexity of the configuration space rapidly grows with the number of planks, making it increasingly difficult to design valid shapes manually without computational support and guidance. We observe that while it is possible to restore validity by using thicker planks with more weight-bearing capacity (see Figure 18), this unfortunately results in higher cost, lower efficiency, and unnecessarily bulky designs. Table 1 presents typical continuous and discrete suggestion generation times. For generating suggestions, we can explore $\sum_{k=1,2,3}\binom{N_\gamma}{k} = O(N_\gamma{}^3)$ directions in real-time even for 15-20 planks using linear approximation with the line search step taking the majority of the time. We recall that each additional plank increases N_γ by roughly 8 (see Section 6.1).

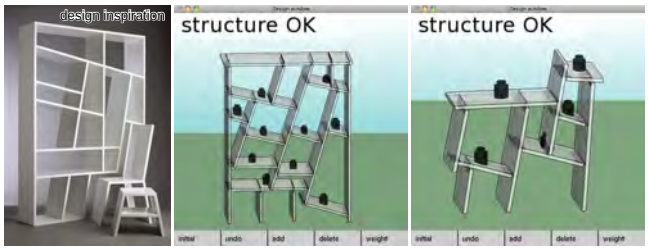

Figure 16: *(Left) Designing non-standard furniture is difficult for novice users. Our guided exploration framework allows users to design strange configurations easily with target load specifications.*

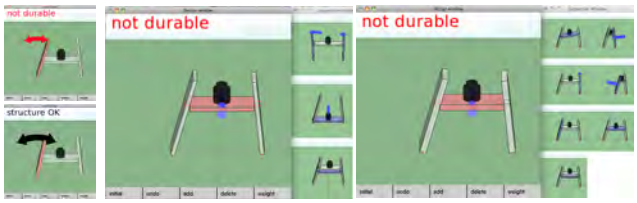

Figure 18: *Effect of plank thickness. Increasing the plank's thickness leads to a larger valid shape space and more suggestions.*

Validation. We validated the durability of our nail-joint model (see Equation 4) in a simple cantilever beam example with the same material assumed in our system. We observe that the maximum weight cantilever beam closely follows our model (see supplemental material). Analogous to yielding, we observed a rapid increase in shape deformation around the maximum predicted weight.

User study. We performed a user study to obtain feedback from users. Recreating a typical design scenario, we asked the user to design a piece of furniture freely following their own concept. Nine test users (novice designers, CS graduates, 1 female) designed furniture with three types of systems: (i) a system without feedback from the physical simulation (i.e., the participants were not informed about which joints were undurable or whether the furniture toppled during design) and without suggestions, (ii) a system with feedback from the simulation (validity check and valid range visualization) but without suggestions, and (iii) a system with feedback and suggestions (our system). While using system (i), the user was allowed to see the result of simulation up to five times whenever she liked, assuming the use of a traditional shape modeling software along with a simulation software. Each participant started by creating a concept design on paper. Then she created 3D furniture models with the three systems to realize their concept design. To counter-balance learning effects, we separated the nine participants into two groups: five participants used the system in the order (i, ii, iii), while the rest used the system in the reversed order (iii, ii, i). On

	Figure 17 right	Figure 16 right	Figure 16 middle	Figure 1 right
#planks	9	10	20	28
#joints	13	13	33	49
#continuous suggestion	8	8	8	6
candidate generation (*ms*)	13.2	22.3	160	758
line search (*ms*)	92.3	83.1	670	1512
#discrete suggestion	1	1	2	1
discrete suggestion (*ms*)	3.8	5.6	48	52
total time (*ms*)	110	123	880	2420

Table 1: *Performance statistics on a laptop computer with an Intel Core^{TM}i7 2.8GHz CPU with 4GB RAM.*

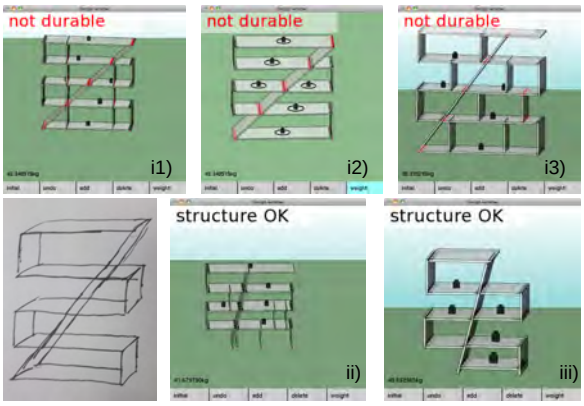

Figure 19: *Starting from a design concept (bottom-left), three failed attempts with no feedback or suggestions (i1-3), using only feedback without suggestions (ii), and results using our system (iii).*

an average, the participants took roughly 30 minutes per successful design. We present session histories in the supplementary materials.

In Figure 19, we show a session where the participant, who was proficient at Google SketchUp, used the systems in order (i, ii, iii). The participant simply failed to design a valid shape using system (i). With system (ii), he managed to design a valid piece of furniture, but he complained that the shape of the furniture was boring and far from his initial design concept. Using system (iii), he successfully designed a valid piece of furniture closely following his initial concept. Other participants had a similar experience (see supplementary materials for the other user sessions). All nine participants successfully created valid pieces of furniture close to their initial concepts with our system. Note that even the participants who used the system in the order (iii, ii, i) mostly failed to recreate the design with system (ii) or (i) although they had seen successful designs while using system (iii). Intuitively a validity-restoring suggestion often involves synchronous editing of multiple parts, which is challenging without suitable computational support. A participant commented that displaying the range of edits was very useful for fine-tuning a design. We note that a more rigorous quantitative comparative study of such creative design support is needed.

Limitations. We consider planks to be perfectly rigid and unbreakable. In practice, however, planks deform under heavy loads, influencing their nail-joint behavior and ultimately they can break. This is especially true in the context of shelves or other furniture with long segments without any supporting structures. We also do not consider curved planks or shifting loads in our framework. Further, our linear approximations for computing durability and stability constraints can be violated in highly non-linear regions. Although it is possible to consider higher-order approximations, we decided against such a choice in favor of interactivity. Finally, we restricted $M = 3$, thus limiting the range of design possibilities. In certain cases, it is desirable to explore the range of meaningful suggestions especially in designs with many components, or when the initial design is far from the valid space.

Exploring valid design spaces is difficult, especially when the constraints are non-linear. While characterizing the valid space itself is difficult, exploring high degrees of freedom design spaces is challenging as the valid regions maybe disjoint forming islands or have narrow connection pathways among valid spaces, posing further challenges. In such cases, our technique can fail to find durable configurations, even when they exist. Finally, we do not consider aesthetics in our framework. Ideally, aesthetic considerations should come from designers while our goal is simply to computationally

assist the form-finding process by guiding the designer away from invalid or uninteresting parts of the shape space.

8 Conclusion

In this paper, we presented an interactive computational design framework for guided exploration of physically valid shapes for nail-jointed furniture. Our system provides active real-time guidance to the user to help her avoid invalid designs, either due to stability violations, or due to excessive joint bending forces. We propose a novel force-space analysis for both bending forces and frictional constraints to generate multiple suggestions, along with valid deformation ranges, involving both continuous and discrete geometric changes. We used our system to design a range of furniture and also demonstrated the utility of the system by building a physical prototype.

A lot remains unexplored in this area. In the future, we want to ensure validity for dynamic furniture, e.g., designing a physically valid rocking chair. A possible approach is to treat the problem as a coupled exploration of multiple shapes based on the contact points to the ground and the relative (upright) orientation of the shape. Subsequently, we can simultaneously explore the multiple shapes, while adding a regularity term to favor edits that are consistent across all shapes (since correspondence is known). Finally, we plan to support exploration of shape design involving a large number of components, e.g., designing a building, etc.

Acknowledgements. We thank Lars Hesselgren, Helmut Pottmann, Anthony Steed, and Michael Wand for their comments, and the anonymous reviewers for their thoughtful and useful suggestions. The work was supported in part by a KAUST visiting student grant, the Marie Curie Career Integration Grant 303541 and JSPS. We also thank Christina Amati for the video voiceover and Shuitian Yan for rendering the teaser image.

References

ALEXA, M., AND MATUSIK, W. 2010. Reliefs as images. *ACM TOG (SIGG.) 29*, 60:1–60:7.

BARAFF, D. 1994. Fast contact force computation for nonpenetrating rigid bodies. In *ACM SIGGRAPH*, 23–34.

BERGMAN, R. 2010. *Wood Handbook – Wood as an Engineering Material*. Forest Products Laboratory.

BICKEL, B., BÄCHER, M., OTADUY, M. A., LEE, H. R., PFISTER, H., GROSS, M., AND MATUSIK, W. 2010. Design and fabrication of materials with desired deformation behavior. *ACM TOG (SIGG.) 29*, 63:1–63:10.

CHAUDHURI, S., AND KOLTUN, V. 2010. Data-driven suggestions for creativity support in 3D modeling. *ACM TOG (SIGG. Asia) 29*, 183:1–183:10.

CHAUDHURI, S., KALOGERAKIS, E., GUIBAS, L., AND KOLTUN, V. 2011. Probabilistic reasoning for assembly-based 3d modeling. *ACM TOG (SIGG.) 30*, 35:1–35:10.

CHENNEY, S., AND FORSYTH, D. A. 2000. Sampling plausible solutions to multi-body constraint problems. In *ACM SIGGRAPH*, 219–228.

EIGENSATZ, M., KILIAN, M., SCHIFTNER, A., MITRA, N. J., POTTMANN, H., AND PAULY, M. 2010. Paneling architectural freeform surfaces. *ACM TOG (SIGG.) 29*, 4 (July), 45:1–45:10.

ERLEBEN, K., SPORRING, J., HENRIKSEN, K., AND DOHLMANN, H. 2005. *Physics Based Animation (Graphics Series)*. Charles River Media, 8.

FUNKHOUSER, T., KAZHDAN, M., SHILANE, P., MIN, P., KIEFER, W., TAL, A., RUSINKIEWICZ, S., AND DOBKIN, D. 2004. Modeling by example. In *ACM TOG (SIGG.)*, 652–663.

GAL, R., SORKINE, O., MITRA, N. J., AND COHEN-OR, D. 2009. iWIRES: an analyze-and-edit approach to shape manipulation. *ACM TOG (SIGG.) 28*, 33:1–33:10.

GERADIN, M., AND CARDONA, A. 2001. *Flexible Multibody Dynamics: A Finite Element Approach*, 1 ed. Wiley, 2.

IGARASHI, T., AND HUGHES, J. F. 2001. A suggestive interface for 3D drawing. In *UIST*, 173–181.

KAUFMAN, D. M., SUEDA, S., JAMES, D. L., AND PAI, D. K. 2008. Staggered projections for frictional contact in multibody systems. *ACM TOG (SIGG. Asia) 27*, 5, 164:1–164:11.

KERR, W. B., AND PELLACINI, F. 2010. Toward evaluating material design interface paradigms for novice users. In *ACM TOG (SIGG.)*, 35:1–35:10.

KLARBRING, A. 1990. Examples of non-uniqueness and non-existence of solutions to quasistatic contact problems with friction. *Archive of Applied Mechanics 60*, 529–541.

MERRELL, P., SCHKUFZA, E., LI, Z., AGRAWALA, M., AND KOLTUN, V. 2011. Interactive furniture layout using interior design guidelines. *ACM TOG (SIGG.) 30*, 87:1–87:10.

MITRA, N. J., AND PAULY, M. 2009. Shadow art. In *ACM TOG (SIGG. Asia)*, 156:1–156:7.

OVSJANIKOV, M., LI, W., GUIBAS, L., AND MITRA, N. J. 2011. Exploration of continuous variability in collections of 3D shapes. In *ACM TOG (SIGG.)*, 33:1–33:10.

PACZKOWSKI, P., KIM, M. H., MORVAN, Y., DORSEY, J., RUSHMEIER, H., AND O'SULLIVAN, C. 2011. Insitu: sketching architectural designs in context. *SIGG. Asia 30*, 182:1–182:10.

PARKER, H., AND AMBROSE, J. 1997. *Simplified Design of Wood Structures*. Wiley.

POPOVIĆ, J., SEITZ, S. M., ERDMANN, M., POPOVIĆ, Z., AND WITKIN, A. 2000. Interactive manipulation of rigid body simulations. In *ACM SIGGRAPH*, 209–217.

SHAPIRA, L., SHAMIR, A., AND COHEN-OR, D. 2009. Image appearance exploration by model-based navigation. *CGF (EUROGRAPHICS)*, 629–638.

SINGH, M., AND SCHAEFER, S. 2010. Triangle surfaces with discrete equivalence classes. *ACM TOG (SIGG.) 29*, 46:1–46:7.

SMITH, J., HODGINS, J. K., OPPENHEIM, I., AND WITKIN, A. 2002. Creating models of truss structures with optimization. *ACM SIGGRAPH 21*, 1, 295 – 301.

STEWART, D., AND TRINKLE, J. 2000. An implicit time-stepping scheme for rigid body dynamics with coulomb friction. In *IEEE ICRA*, vol. 1, 162–169.

TALTON, J. O., GIBSON, D., YANG, L., HANRAHAN, P., AND KOLTUN, V. 2009. Exploratory modeling with collaborative design spaces. *ACM TOG (SIGG. Asia) 28*, 167:1–167:10.

TWIGG, C. D., AND JAMES, D. L. 2008. Backwards steps in rigid body simulation. *ACM TOG (SIGG.)*, 25:1–25:10.

UMETANI, N., KAUFMAN, D. M., IGARASHI, T., AND GRINSPUN, E. 2011. Sensitive couture for interactive garment modeling and editing. In *ACM TOG (SIGG.)*, 90:1–90:12.

VAN KEULEN, F., HAFTKA, R., AND KIM, N. 2005. Review of options for structural design sensitivity analysis. part 1: Linear systems. *Proc. CMAME 194*, 30 -33, 3213 – 3243.

WHITING, E., OCHSENDORF, J., AND DURAND, F. 2009. Procedural modeling of structurally-sound masonry buildings. *ACM TOG (SIGG. Asia) 28*, 5, 112.

YANG, Y.-L., YANG, Y.-J., POTTMANN, H., AND MITRA, N. J. 2011. Shape space exploration of constrained meshes. *ACM TOG (SIGG. Asia) 30*, 6.

YU, L.-F., YEUNG, S.-K., TANG, C.-K., TERZOPOULOS, D., CHAN, T. F., AND OSHER, S. J. 2011. Make it home: automatic optimization of furniture arrangement. *ACM TOG (SIGG.) 30*, 86:1–86:12.