

Sit & Relax: Interactive Design of Body-Supporting Surfaces

Kurt Leimer Michael Birsak Florian Rist Przemyslaw Musialski

GCD / TU Wien

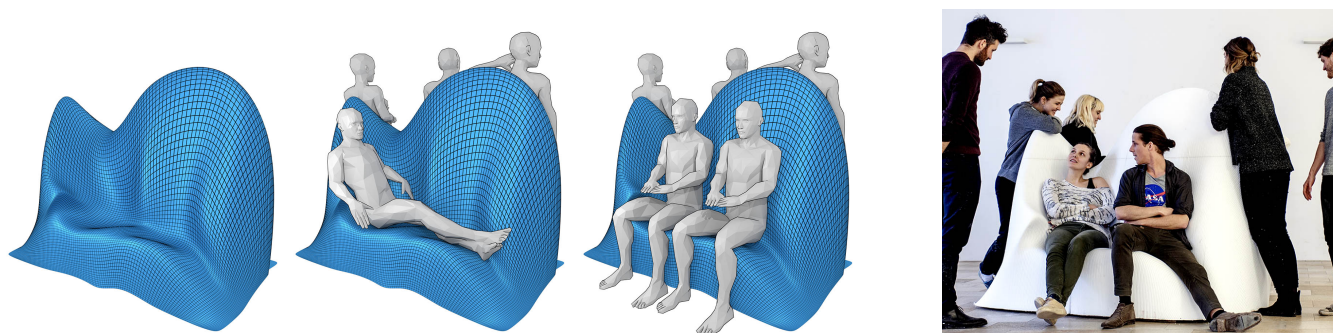


Figure 1: Multi-purpose sitting surface designed for various types of sitting and leaning using our method. Left: three renderings. Right: fabricated result inspected by design students.

Abstract

We propose a novel method for interactive design of well-fitting body-supporting surfaces that is driven by the pressure distribution on the body's surface. Our main contribution is an interactive modeling system that utilizes captured body poses and computes an importance field that is proportional to the pressure distribution on the body for a given pose. This distribution indicates where the body should be supported in order to easily hold a particular pose, which is one of the measures of comfortable sitting. Using our approximation, we propose the entire workflow for interactive design of C^2 smooth surfaces which serve as seats, or generally, as body supporting furniture for comfortable sitting. Finally, we also provide a design tool for RHINOCEROS/GRASSHOPPER that allows for interactive creation of single designs or entire multi-person sitting scenarios. We also test the tool with design students and present several results. Our method aims at interactive design in order to help designers to create appropriate surfaces digitally without additional empirical design passes.

1 Introduction

Product design of human body-supporting objects—for instance sitting furniture—is a difficult design problem. Usually designers try to combine two major aspects: the actual function of the product as well as the aesthetics of its shape. One of the very well known maxims of this process is the *form follows function* principle, which dictates that the form of an object is at least in part determined by its function. However, a unique aesthetic is often desired to make the product stand out from similar products. Therefore the goal is to create an appealingly looking shape and at the same time to make this shape fulfill the functional requirements.

Traditionally, this is achieved by employing an iterative process where a number of prototypes needs to be produced in one-to-one scale in order to figure out what is actually comfortable. The main

reason for this is that it is extremely difficult to judge in advance—especially if only a digital model is created—how comfortable and functional the final product will eventually be.

Our goal is to provide designers a novel way to create free-form surfaces that automatically adapt to human postures. Hence, we propose a new approach for the interactive design of body-supporting surfaces which automatically nestle to the shape of the body in order to make the human feel comfortable in the current pose. To achieve this goal, we propose a measure that gives an indication of how a human in a given posture should be supported in order to distribute the pressure uniformly on an as large an area as possible. Pressure distribution has been identified as one of the objective indicators of comfort [Lue83, Sta95, DKEV03].

Technically, we accomplish it by computing an importance map

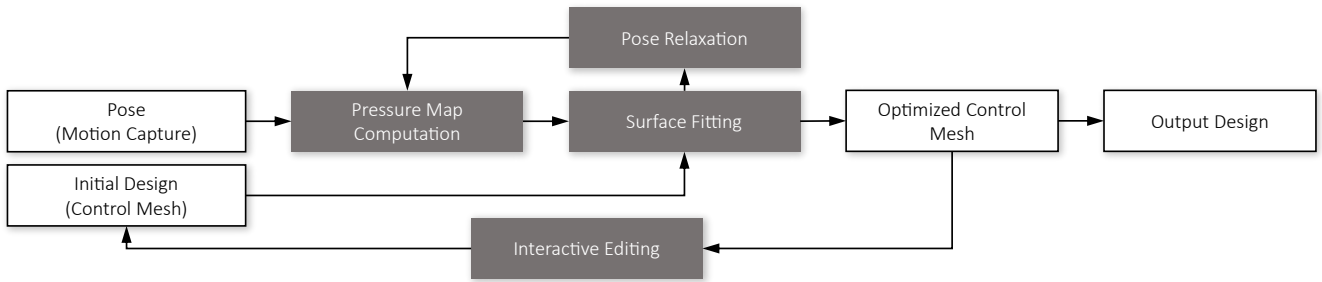


Figure 2: Overview of the interactive design system. Given initial design and a set of poses captured by a motion capture device, our system estimates a pressure distribution on the bodies in the given poses. The artist can then create a social scenario using the given poses and provide a initial control mesh for a surface. Our system then computes an optimized smooth subdivision surface and its control mesh using our surface fitting algorithm. In further design steps, the computed control mesh can be edited interactively and used as input again to generate a new design. Our pose relaxation algorithm also makes it possible to adapt the input poses to the computed subdivision surface if necessary to ensure that all poses can be supported well.

on the human body which is proportional to the physical pressure which the body is exposed to if resting in a given pose. Further, this importance field on the surface of the body is used for fitting of optimally supporting Catmull-Clark subdivision surfaces in an interactive design application.

Our contributions are the following: (i) we cast the problem of interactive design of body-supporting surfaces driven by the pressure distribution acting on the body, where we propose an approximate, physically validated method for an efficient computation of body pressure in Section 4. (ii) Further, we provide an interactive design system for free-form surface fitting in Section 5 as well as for pose relaxation in Section 6. (iii) Finally, we utilize our solution in collaboration with designers in order to design a real-world example and we fabricate a functional product, which we document in Section 7.

2 Related Work

The question of the comfort of sitting is not novel and has been addressed in the field of ergonomics already quite a while ago. For instance, Lueder [Lue83] provided a survey of the assessment of comfort that is relevant to the design of office furniture. Also, [ZHD96] analyzed the multidimensional factors of sitting comfort and discomfort. Other surveys studied the comfort of sitting in vehicle seats [DKEV03], and especially in wheelchairs [Sta95]. These works conclude that the pressure distribution is an objective measure with the most clear association with the subjective rankings [DKEV03].

The application of our method is interactive design driven by physically derived information, as is often the case in computer graphics. For example, Saul et al. [SLM11] introduced an application for easy design of chairs for novice users using an easy-to-use 2d sketching interface. These furniture could be than fabricated from planar sheet materials by cutting them out and putting them together. Umetani et al. [UIM12] proposed a system for computational design of shelves using a physical model which supported the users during the design such that only structurally stable models where created. Further, Zheng et al. [ZLDM16] proposed a method for ergonomics inspired reshaping and exploration of collections of models in order to create novel shapes. Fu et al. [FCSF17] proposed a method for the synthesis of models driven by given postures.

Another field is fabrication-aware design where certain physical properties are taken into account. Pioneering approaches for interactive design and fabrication of plush toys [MI07], furniture [SLM11], or burr puzzles [XLF*11] were presented. Several author also focused on systems for interactive garment editing [UKIG11, JHR*15, BSK*16], and design of physically valid furniture [UIM12].

Our method incorporates fitting of Catmull-Clark subdivision surfaces [CC78] to a given human pose in order to support it optimally, which we have chosen due to their C^2 -smoothness that result in very elegant shapes. Therefore we utilize methods for optimization of the control mesh for the fitting of subdivision surfaces [MK05, CWH*07]. However, in the literature there have also been a number of methods which propose interactive design of free-form surfaces, e.g., [IMT99, NISA07, JHR*15, LPL*17].

The field of vision and machine learning has also utilized poses for the analysis and classification of objects like sitting furniture. For instance, [GGV11] introduced a system that uses an affordance detector in order to determine the functionality of objects by the way humans could interact with them. A similar idea has been pursued by [KCGF14] who use supervised learning on a set of poses in order to further generate static poses of how an functional objects could be potentially used. Furthermore, [KL14] also proposed a method for the generation of poses to a given geometry.

Finally, we also utilize inverse kinematics [Bus04] for additional relaxation of the given posture in order to improve the contacts surface [SZGP05, DSP06].

3 Overview

Figure 2 shows the workflow of our interactive design system. The input is one or more human body poses given by triangular meshes and an initial design of the actual surface. In practice, since we are using Catmull-Clark subdivision, the initial design is given by its control mesh, and can be in the most trivial case just a flat patch.

For the bodies of humans we use the BLENDER plugin provided by Manuel Bastioni [Bas18], which allows the generation of body meshes with varying attributes, like gender, mass, size, stature, etc. Since the meshes are skinned and rigged to a skeleton, the user can adjust the poses manually, or alternatively, the poses can be created with a motion capturing device, for instance using the PERCEPTION

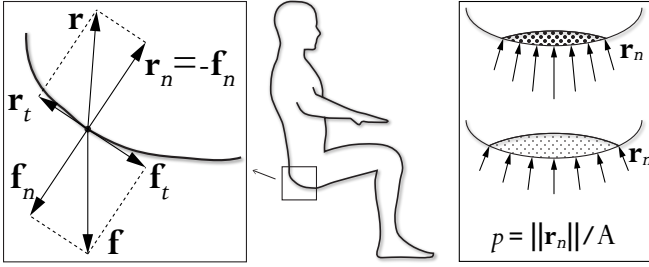


Figure 3: Illustration of the forces acting on a rigid body. Left: reaction forces split into the normal and tangential components. Right: definition of pressure as normal force acting per unit area. If the same forces are acting on a larger area, the pressure is lower.

NEURON system [Neu18]. We have utilized the latter for most of our designs.

In the next step the user can compose a scenario of sitting humans by selecting the captured poses and placing them on the modeling canvas with respect to the control mesh. Our system then computes the pressure distribution necessary to support each human in the given pose using the method further described in Section 4 and uses this information for fitting a subdivision surface as described in Section 5. Additionally, the user can decide to adjust selected poses, either manually, or using inverse kinematics in order to relax the pose with respect to the already computed surface, which is described in Section 6. Finally, the new control mesh can also be edited by either moving or fixing its vertices. This process can be repeated iteratively and the design can be explored until a satisfactory result is achieved.

4 Pressure Field Computation

Our goal is now to find a physically plausible distribution of pressure to which the body is exposed to if resting in a given pose. It is important to support the body where the relative pressure is high. At the same time, we want to make the contact area as large as possible to keep absolute pressure peaks low, so the pose can be considered as comfortable [DKEV03], as shown in Figure 3, right. Usually, the pressure distribution on the body would be found using a sophisticated physical simulation which is time consuming. Since our goal is to achieve interactive rates, we propose a simplified model where we assume the human is a rigid body.

4.1 Pressure Model

In order to compute an approximated distribution of the pressure acting on the human body, we build on top of the static Coulomb friction model (cf. Figure 3). In that model, the force \mathbf{f} acting at each contact point can be split into its normal component \mathbf{f}_n and its tangential components \mathbf{f}_{t_1} and \mathbf{f}_{t_2} . In order to keep the body in static equilibrium, we need to counteract these forces by reaction forces $\mathbf{r} = \mathbf{r}_n + \mathbf{r}_{t_1} + \mathbf{r}_{t_2}$, such that $\int_{\Omega} \mathbf{r} dA = m_b \mathbf{g}$, where m_b is the mass of the body and Ω is the supported contact area.

The actual pressure $p = \frac{1}{A} \|\mathbf{r}_n\|$ is the magnitude of the normal component of the reaction force \mathbf{r}_n divided by the area A the force is acting on (cf. Fig 3, right). The tangential components \mathbf{r}_t are the particular friction forces, whose magnitudes—according to the linear friction model—must be smaller than the magnitudes of the respective tangential components of the body force \mathbf{f}_t . Otherwise,

the body would slip away from the support. This is usually ensured by setting the reaction force components to

$$\mathbf{r}_n = -\mathbf{f}_n, \quad \text{and} \quad \|\mathbf{r}_t\| \leq \mu \|\mathbf{r}_n\|$$

where μ is the friction coefficient that depends on the roughness of the contacting surfaces. This ensures that the friction forces stay within the so-called friction cone.

4.2 Pressure Approximation

In practice, the relative distribution of the pressure on the body surface does not differ greatly when including or disregarding the tangential components of the reaction forces. This is also evident from the physical simulation we have performed with a default value of $\mu = 0.5$ (cf. Section 4.3 and Figure 4b). Moreover, considering the friction coefficients of common materials, humans usually do not run the danger to slide down from the seat. For these reasons, in our strongly simplified model we ignore the tangential components and utilize only the normal directions.

We assume there exists one reaction force \mathbf{r} for each vertex \mathbf{v} of the body mesh whose normal is pointing sideways or downwards (cf. Figure 3, right box). Vertices with normals pointing upwards are excluded from the computation, since they cannot be supported. In order to find out how the normal forces \mathbf{r}_n and the pressure p are distributed on the surface, we consider the projection of the gravity direction vector on the unit surface normal \mathbf{n} scaled by the Voronoi area of each vertex $\mathbf{f} = (0, -A, 0)$, such that

$$\mathbf{r}_n = -(\mathbf{n} \cdot \mathbf{f}) \mathbf{n} \quad \text{and} \quad p = \frac{1}{A} \|\mathbf{r}_n\|.$$

In other words, we assume a good reaction force distribution can be approximated by considering the body as fully surrounded (e.g., enveloped) by a perfectly fitting support, where we ignore the friction forces and the body weight.

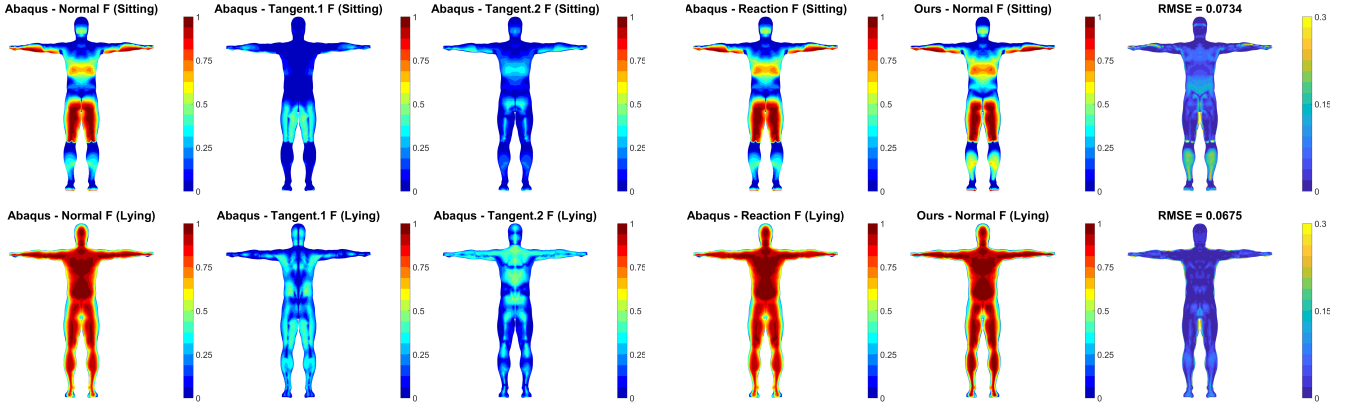
Note that the local vertex area A is factored out for the pressure computation and that in fact, the sum over the reaction forces on the body does not equal the body weight. Nonetheless, there is a linear relationship, and since we normalize the pressure in the range $[0..1]$ and give up the physical units—which is still sufficient for our application (cf. Section 5)—, this issue can be ignored.

While this model might appear too oversimplified, in comparisons to professional FEM simulations we show (cf. Section 4.3), that this approximation has an error (RMSE) in the range of 0.07 on a $[0..1]$ scale and it is still well-suitable for the purpose of an importance map for surface fitting.

4.3 Comparison to Physical Simulation

In order to justify our simplified model, we perform a FEM simulation using the professional physical simulation software ABAQUS [Smi09]. For this purpose we simulate two selected poses—a lying pose and a sitting pose—in a setup as proposed in our model in order to compute ground truth values. Note that here we also include the tangential forces.

For the simulation we create a rigid shell in the shape of a negative mold of the body by taking all body-mesh faces pointing downwards or sideways. This shell serves as the contact surface which fully supports the body. For the human body, we create a single volume domain as a rigid body. The shell is completely locked in place by boundary constraints, while the body is moved downward



(a) Results of the forces computed by the physical simulation using ABAQUS (cf. Section 4.3). From left for right: magnitude of the normal forces followed by both tangential forces.

(b) Comparisons of our results to ABAQUS. Left: reaction forces (magnitude of the vector sum of normal and tangential forces). Middle: our result (magnitude of normal forces). Right: absolute difference of left and middle.

Figure 4: Results of physical simulation and comparison to ours. Top row: sitting pose—please note that for visualization purpose we render the results on a T-pose. Bottom row: lying T-pose. Please notice the different range for the error image.

by a force of 735N (roughly equivalent to a body weight of 75kg) applied on the center of mass. This procedure is standard for the simulation of a single rigid body in ABAQUS.

To model the contact between body and shell we use a linear pressure-overclosure relationship with contact stiffness of $8e+12$. The tangential contact behavior is modeled with an isotropic friction coefficient of $\mu = 0.5$ and an elastic slip of $1e-10$. Furthermore, we select the set of vertices at which reaction forces are computed in our method by choosing those vertices of the body mesh that are also included in the corresponding contact surface shell to make sure that the contact surface is the same in both methods. The computation time in ABAQUS took a total of 10 seconds for the simulation plus preprocessing, while our system takes 0.03 seconds on average.

In Figure 4a we show the magnitudes of the particular normal forces $\hat{\mathbf{r}}_n$ as well as the tangential forces $\hat{\mathbf{r}}_{t_1}$ and $\hat{\mathbf{r}}_{t_2}$ computed in ABAQUS. Please note that all force magnitudes are taken absolute and normalized to the range of $[0..1]$ to the maximum value of $\hat{\mathbf{r}}_n$.

For the comparison to our results we measure the root-mean-squared error (RMSE) between our estimated normal forces \mathbf{r}_n and the reaction forces $\hat{\mathbf{r}}$ computed in ABAQUS. Note that the reaction force is the magnitude of the sum of normal and tangential forces. We have chosen this comparison (cf. Figure 4b) in order to emphasize why we can omit tangential components without a large error. In fact, the RMSE is about 0.06 – 0.07 for values scaled in a range of $[0..1]$. We also compare the normal forces of both methods, where the error is in the same range. Please refer to supplemental material for this comparison. The results can be seen in Figure 4b.

5 Surface Fitting

In this section we present how the approximated pressure distribution can be utilized for fitting of optimal support surfaces in an interactive modeling application. Our goal is to generate a surface that allows one or multiple persons in a specified posture to sit or lean on it. To do this, we take the distribution of reaction forces of each given posture as input, and then use a Catmull-Clark subdivi-

sion algorithm to fit a smooth surface to the vertices on the body that need to be supported.

Using a quad-mesh $M^0 = \{\mathbf{V}^0, \mathbf{F}^0\}$ as a control mesh, we want to compute the optimal control mesh $M^* = \{\mathbf{V}^*, \mathbf{F}^*\}$ such that the subdivided mesh $M^s = \{\mathbf{V}^s, \mathbf{F}^s\}$ (of some chosen subdivision level s) has minimal distance to the target body mesh $M^b = \{\mathbf{V}^b, \mathbf{F}^b\}$. To do this, we first compute for each sample point $\mathbf{v}_k^b \in \mathbf{V}^b$ the closest vertex point $\mathbf{v}_k^s \in \mathbf{V}^s$ of the subdivided mesh M^s . Please note that two different points $\mathbf{v}_j^b \neq \mathbf{v}_k^b$ may share the same closest vertex $\mathbf{v}_j^s = \mathbf{v}_k^s$. As a distance metric we use a linear blend of the point-to-point distance and the tangential distance, which is the distance from to the tangent plane of the closest point. The reasoning for this is that we would like regions of the surface to still be able to move tangentially once they are close enough to the target body.

Our aim is to find the control mesh vertices $\mathbf{v}^* \in \mathbf{V}^*$ that minimize

$$E^{sds} = \sum_k (1 - \tau) \rho_k \|\mathbf{v}_k^b - \mathbf{v}_k^s\|^2 + \tau \rho_k \|(\mathbf{v}_k^b - \mathbf{v}_k^s) \mathbf{n}_k^s\|^2 + \sum_j \sigma \|L(\mathbf{v}_j^*)\|^2 + \gamma_j \|\mathbf{v}_j^* - \mathbf{v}_j^0\|^2,$$

with the first two terms being the point-to-point and tangential distances, the third being a smoothing term, and the last term being a penalty that prevents select control vertices from moving too far. The value ρ_k is the importance of the vertex \mathbf{v}_k^b based on the computed reaction forces and is given by

$$\rho_k = \frac{\|\mathbf{r}_k\|}{\max_i \|\mathbf{r}_i\|}.$$

\mathbf{n}_k^s is the normal vector at \mathbf{v}_k^s . The parameter τ blends between the two distance metrics and is initialized as 0 and is increased with each iteration step, up to a user defined maximum $\tau_{max} \leq 1$. The operator $L(\cdot)$ is the discrete cotangents Laplace-Beltrami operator [BKP*10] and serves as a smoothing term, regulated using the weight parameter σ . The effect of this smoothing term is demonstrated in Figure 7. Furthermore, to give the user a control over the look of the resulting design, we can soft-constrain any control

mesh vertices $\mathbf{v}_j^* \in \mathbf{V}^*$ to their initial locations using the weight parameter γ_j .

To prevent the surface from intersecting with the target body, we use the following intersection constraint: For every previously found point $\mathbf{v}_k^s \in \mathbf{V}^s$ on the surface we search for the closest point $\hat{\mathbf{v}}_k^b \in \mathbf{V}^b$ on the target body. Then the constraint $(\mathbf{v}_k^s - \hat{\mathbf{v}}_k^b) \cdot \hat{\mathbf{n}}_k^b \geq 0$ ensures that the point \mathbf{v}^s stays of the positive side of the tangent plane of $\hat{\mathbf{v}}_k^b$ with normal vector $\hat{\mathbf{n}}_k^b$.

Minimizing the energy E^{sds} leads to a non-linear system of equations since the point \mathbf{v}_k^s on the surface closest to \mathbf{v}_k^b on the body changes as the surface deforms. We instead linearize this problem and solve the resulting linear system of equations iteratively:

$$\begin{aligned} \min_{\mathbf{V}^*} \quad & \| \mathbf{A}\mathbf{V}^* - \mathbf{b} \|^2 \\ \text{s. t.} \quad & \mathbf{C}\mathbf{V}^* \leq \mathbf{0} \end{aligned} \quad (1)$$

with \mathbf{A} and \mathbf{b} containing the terms of the energy E^{sds} and \mathbf{C} containing the non-penetration constraints. For a more detailed description of these terms, please refer to Appendix A.

To summarize, we perform the following algorithm iteratively until the solution does not improve any more. In each iteration i :

1. Subdivide M_i^0 to compute M_i^s and its normals \mathbf{N}_i^s .
2. Find the closest vertices $\mathbf{v}_k^s \in \mathbf{V}_i^s$ for each sample point $\mathbf{v}_k^b \in \mathbf{V}^b$.
3. Solve the linear system in Eq. 1 to compute \mathbf{V}_{i+1}^0 .

6 Pose Relaxation

A further step we introduce for the convenience of the designer is pose relaxation. It allows to register the given pose to a given surface and also to further relax it in order to better adapt to the surface. Since our input meshes come from a BLENDER plugin [Bas18] and are rigged to a skeleton, we can also use this skeleton to perform inverse kinematics [Bus04] on the pose and propagate the deformation of the skeleton to the surface mesh using linear blend skinning [MTLT89, BP07].

Our goal is to minimize the distance between parts of the pose that need support and the corresponding regions on the computed surface while at the same time avoiding penetrations between the pose and the surface. Further, we do not want the pose to change significantly and therefore also want to penalize large changes of joint angles.

Thus, we parametrize the rigged pose by the joint angles θ of the skeleton, and we cast the problem as an optimization task

$$\min_{\theta} \lambda_{d_p} \| E^{d_p} \|^2 + \lambda_{d_t} \| E^{d_t} \|^2 + \lambda_p \| E^p \|^2 + \lambda_s \| E^s \|^2,$$

where E^{d_p} , E^{d_t} , E^p and E^s are vectorial energy terms for the point-to-point and point-to-tangent-plane distance between pose and surface, the penetration between the pose and the surface, and the similarity between the input θ^0 and the output θ joint angle vector respectively.

The terms E^{d_p} , E^{d_t} and E^p are not evaluated on the whole body mesh but only on a subset of vertices that belong to regions that need support. Therefore, we start the inverse kinematics procedure with an evaluation of the reaction forces that are needed for an optimal support of the pose (cf. Section 4) and denote the subset of m supportable body vertices with \mathbf{V}^b and the magnitude of the re-

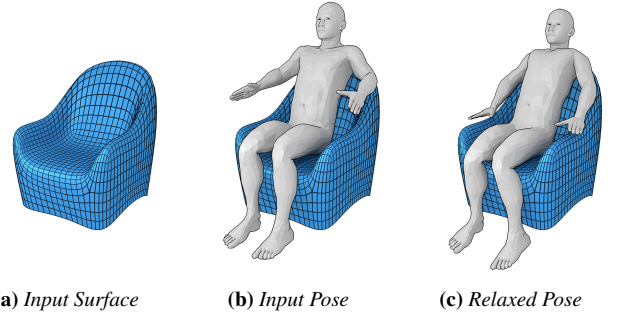


Figure 5: Pose relaxation using inverse kinematics. The designer can further relax the input poses in order to let them to adapt to the current surface (please refer to Section 6)

action force for each $\mathbf{v}_i^b \in \mathbf{V}^b$ with $\|\mathbf{r}_i\| = r_i$. The subset of corresponding surface vertices \mathbf{V}^s is identified by a nearest neighbor search. For the evaluation of the point-to-tangent-plane distance E^{d_t} , we additionally compute the tangent planes at all $\mathbf{v}_i^s \in \mathbf{V}^s$ which are defined by the surface normals \mathbf{N}^s and the distance values \mathbf{D}^s such that $\mathbf{n}_i^s \cdot \mathbf{v}_i^s - d_i^s = 0$.

In detail, we define the energy terms as

$$E_i^{d_p}(\theta) = r_i \left\| \mathbf{v}_i^b(\theta) - \mathbf{v}_i^s \right\|,$$

$$E_i^{d_t}(\theta) = r_i \left(\mathbf{n}_i^s \cdot \mathbf{v}_i^b(\theta) - d_i^s \right),$$

$$E_i^p(\theta) = r_i \left(\frac{1}{2} - \frac{1}{\pi} \arctan \left(a \left(\mathbf{n}_i^s \cdot \mathbf{v}_i^b(\theta) - d_i^s \right) + b \right) \right),$$

and

$$E^s(\theta) = \theta - \theta^0.$$

The $\arctan(\cdot)$ function in E_i^p is used as a differentiable discrimination between penetration (values close to 1) and no penetration (values close to 0) of body and surface. We usually set the parameter $a = 1000$ to limit the transition area between penetration and no penetration and $b = 10$ to shift the inflection point of the function slightly into the penetrated area. We usually set the values λ_{d_p} , λ_{d_t} , λ_p , and λ_s to 0.2, 0.2, 0.5 and 0.1 respectively.

For an efficient optimization procedure we additionally compute the Jacobian matrices \mathbf{J}^{d_p} , \mathbf{J}^{d_t} , \mathbf{J}^p , and \mathbf{J}^s analytically, which we explain in Appendix B. Further we solve the non-linear optimization problem as proposed by [Bus04].

Figure 5 shows a result of pose relaxation performed on a given pose and a seat model. Please note that in this case the arms of the pose are relaxed to lean on the surface which allows to further explore the design with a new pose.

Note that it is possible for physically implausible poses to occur when the input pose is very different from the optimal pose. To prevent such poses, we use empirically chosen box constraints for the joint angle vector θ .

7 Design Process and Results

In this section we describe the design process of examples we have created in collaboration with design students.

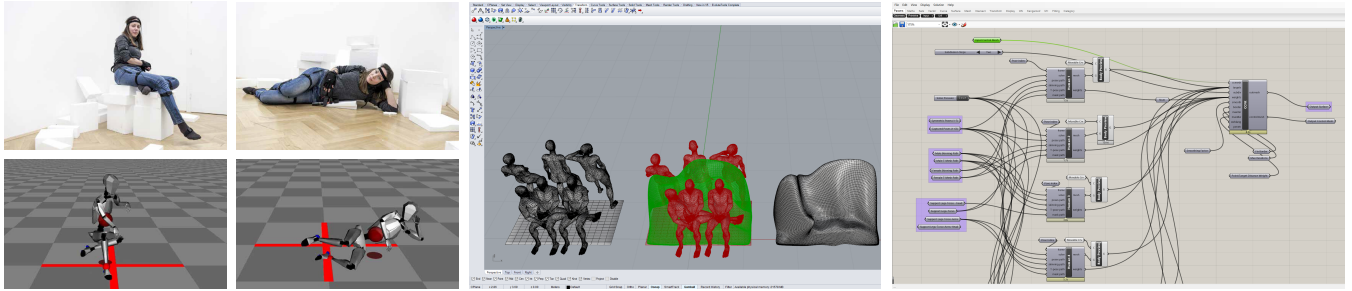


Figure 6: Design process. Left: recording of poses using a motion capture device. Center: interactive design using RHINOCEROS and GRASSHOPPER. Right: GRASSHOPPER canvas.

7.1 Design Process

For the design process, we either allow the user to create own poses from a rigged mesh, or to import captured poses from a motion capture system. For our experiments, we created a database of 62 poses, which were captured by a PERCEPTION NEURON [Neu18] device. In this process, Styrofoam pieces of varying geometries were used as supports to make it easier to adopt different poses (cf. Fig. 6 and video).

In the next step, we use the 3d-modeling software RHINOCEROS in combination with a custom GRASSHOPPER-plugin to design the surface (cf. 6, right). At this point, the designer can use the following operations to further control the process:

- Create a Catmull-Clark control mesh of an initial design or use a geometric primitive as starting point (e.g., flat surface).
- Fix selected vertices of the control polygon.
- Import poses from the database or create own poses.
- Place poses at desired location and orientations with respect to the initial design.
- Choose which body parts should be supported for each particular pose (torso, legs, arms, head, or their mutual combinations).

It is furthermore possible for the placed poses to overlap (like the design shown in Figure 1), although it is necessary to manually align them such that all of them can be supported well. Using these operations, the designer can create a scenario of the desired sitting landscape and let our solver create a new control mesh.

Having the new control mesh, further interactive editing steps are possible:

- Edit the control mesh by moving or fixing vertices or splitting the faces.
- Edit the scenario by moving, adding, or removing the poses.
- Relax the poses using inverse kinematics.

After each editing step (or a series of steps), our solver can generate a new control mesh and the final subdivision surface. The running times depend on the resolution of the control mesh, number of subdivision levels, and the number of poses, and are in practice in the range of one to several seconds.

7.2 Designer Response

We have asked 5 design students to test our modeling paradigm. Each student spent between 2-3 hours working with the tool and was asked to create a sitting scenario of her choice. The results of these sessions are depicted in Figures 9 and 10.

After the session, we asked the participants the following ques-

tions, which could be answered with four answers: poor (1), neutral (2), good (3), and very good (4).

1. How do you judge the general suitability of the system?
2. How do you judge the possibilities to control the outcome?
3. How do you judge the workflow simplification given by the tool?
4. How intuitive is the process?
5. How do you judge the quality of the achieved results?
6. Would you like to use the tool for your own project?

Q	P1	P2	P3	P4	P5	mean	median
1.	3	4	4	3	4	3.6	4
2.	2	3	3	2	3	2.6	3
3.	2	4	3	4	4	3.4	4
4.	3	4	2	3	2	2.8	3
5.	1	4	2	2	3	2.4	2
6.	2	3	4	2	4	3.0	3

Table 1: Results of the questionnaire given to the design students after using our system. Please refer to Section 7.2 for the particular questions.

The results shown in Table 1 allow to conclude that the method has been positively received in general. Please note the probands had only 2-3 hours for experimentation, which is truly a short period of time for creation of a design with a novel and unfamiliar tool.

8 Implementation and Fabrication

8.1 Implementation

Our algorithms are implemented in MATLAB using its optimization routines `lsqlin` for surface fitting (interior-point) and `lsqnonlin` (trust region reflective) for inverse kinematics. For the processing of the pose data obtained from the PERCEPTION NEURON motion capturing software, we use the MOCAP library [Law18]. We also make use of the geometry processing utilities provided by the GPTOOLBOX library [JO16]. As a front-end we use the 3d-modeling software RHINOCEROS with the node-based algorithmic modeling extension GRASSHOPPER. We developed our own GRASSHOPPER components that take input data from RHINOCEROS and feed it to a running MATLAB instance for use with our algorithms. The result then gets passed back to RHINOCEROS, enabling an interactive design process.

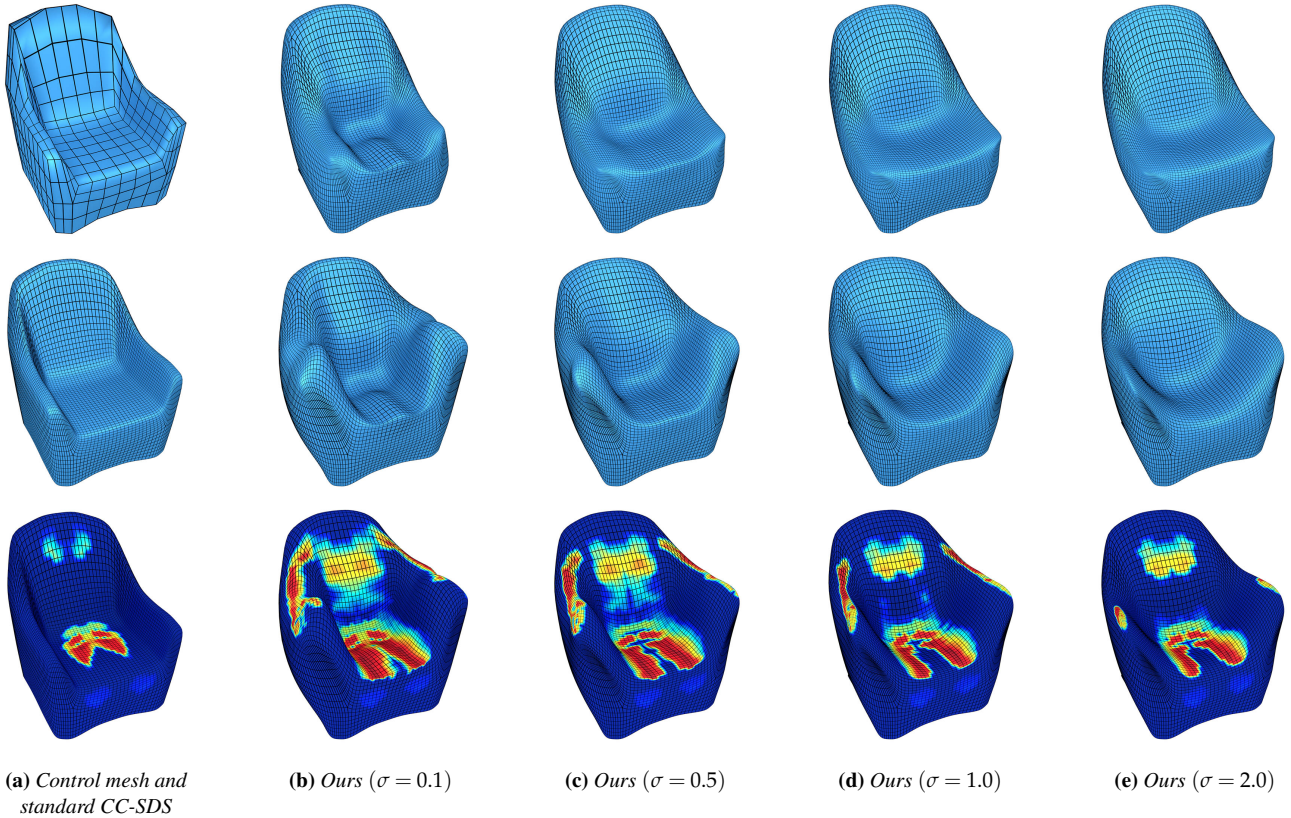


Figure 7: Variations using the control mesh shown in Fig. 7a created with varying values of the smoothing parameter σ which weighs the Laplacian operator. Using this parameter the designer can balance between the importance of the input body map and smoothness of the surface. Top row: arms of the input body have not been considered to be supported. Middle row: arms are supported. Bottom: contact area and pressure on the seat. Refer to Section 5 for more details.

8.2 Fabrication

As a result we have designed a multipurpose surface for three sitting poses (overlapping) and three standing poses (cf. Figure 1). The entire design process took about 6h, where the pose-capture session took about 3h and the following digital design and tuning session with Rhino took another 3h. The by far longest time was needed for the fabrication, where the preprocessing, preparation, and final milling time took about 4 days. For the milling from Styrofoam we have used the software SPRUTCAM 10 for the computation of the milling tool paths and a KUKA KR60 HA industrial robot arm for milling.

9 Discussion and Conclusion

9.1 Discussion

Our method does have a number of limitations. The major limitation is that we assume that the body surface is rigid, which is not the case in practice. However, without this assumption the computation of the pressure map would result in complex non-linear computation, which would be too complex for this kind of application. We think that treating the input poses as articulated bodies consisting of rigid body segments connected by joints could possibly lead to a more accurate pressure distribution, while still maintaining the interactivity of the design tool. We believe that this is an interesting

direction for further research, but for our application, the increased complexity outweighs the benefits as we are already able to generate interesting designs that are well optimized for the given input poses.

For the surface fitting, the assumption of rigidity of the body also poses a limitation, as the actual human body deforms when in contact with a surface. We account for this softness of the body by allowing a certain margin (about 3cm) for the connection between the body and the surface. While we use a type of collision detection (cf. Section 5), it still can happen that the body penetrates the surface if the resolution of the subdivision is lower than of the body (e.g., finger or toes). This, however, does not diminish the results. For inverse kinematics, we resolve it by using an $\arctan(\cdot)$ function as a differentiable step-function in order to distinguish between penetration or not.

In the case of the multipurpose surface, the major technical limitation is that a small control mesh patch does not provide enough degrees of freedom in order to account for all poses. This can be approached with a higher resolution control patch, however, this solution also has limitations and makes the design process more difficult. In the future, it would be interesting to provide a patch per pose and allow for stitching of several patches under maintenance of certain continuity (e.g., C^1), like proposed by Peters [Pet00].

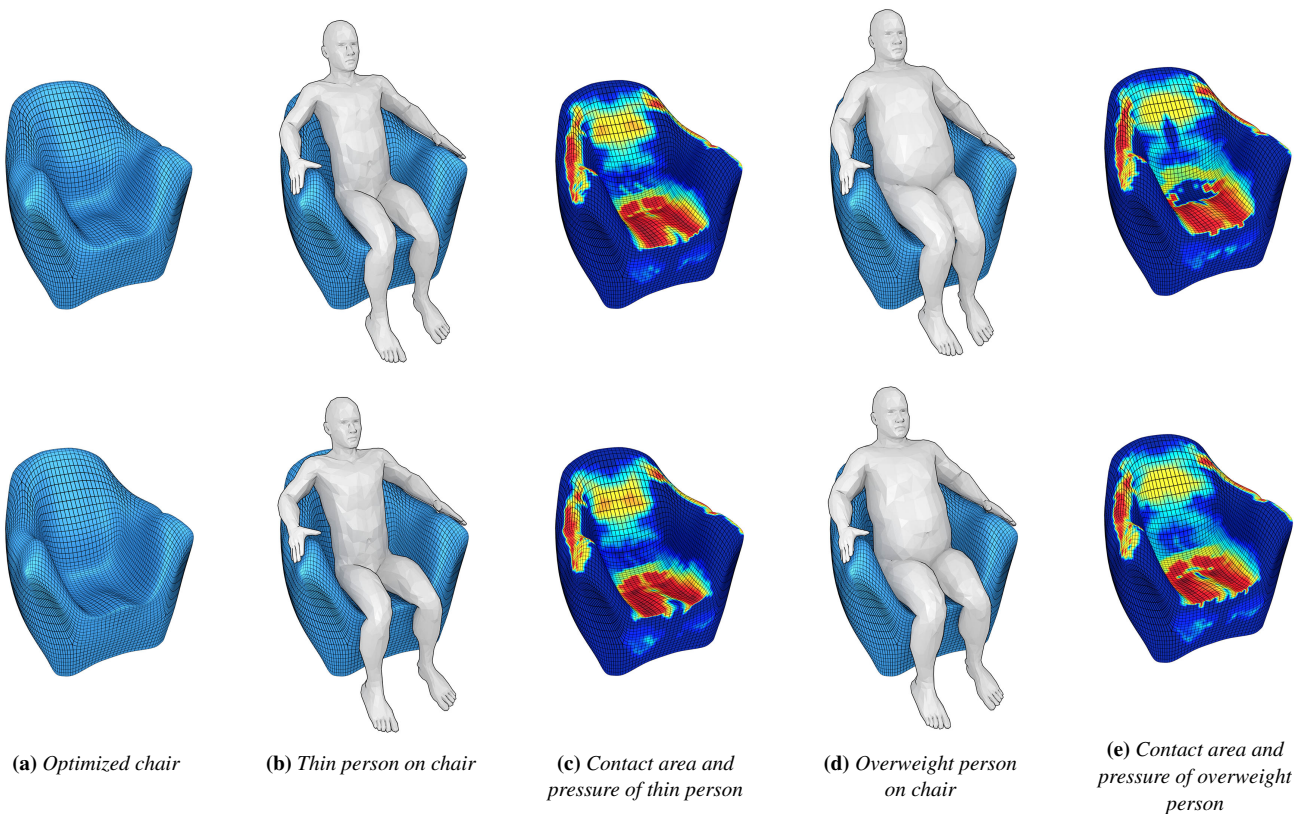


Figure 8: Chair optimized for 2 persons with different body types. Top: optimized for thin person. Bottom: optimized for overweight person.

Further, the incorporation of sharp edges and creases would be interesting as well. Finally, adaptive subdivision to improve fitting in areas where more freedom is needed could also be approached.

Finally, there is also a question of the body size and stature. While we use an average body for all of our designs, the designer is free to use different body types which can be easily created using the plugin of Bastioni [Bas18]. To optimize for various body types, it is possible to use multiple overlapping bodies as input to the surface fitting algorithm, but in order to make the optimized surface as comfortable as possible for all used body types, the body shapes need to be aligned manually. Another option would be to increase the smoothing operator σ in order to make the surface smoother, but at the same time less customized.

Figure 8 shows how different body types influence the result of the optimized surface. As input we use the same chair mesh and pose as in Figure 5b, but with 2 different body types. The two optimized chairs shown on the left (Fig. 8a) may seem very similar at a glance, but greatly differ in how comfortable they are for persons of various body types. The top chair is optimized for a person with a thin build, while the bottom chair is optimized for an overweight person. We use our pose relaxation algorithm to see how a person of a certain body type fit into a chair optimized for the other body type. As can be seen in Fig. 8b and 8c, the thin person has no problem sitting on either chair. However, Fig. 8d shows that the overweight person needs to push the thighs together to fit into the chair optimized for the other person, and even then a large area of

the buttocks does not have any contact with the seat, which can be seen in Fig. 8e.

9.2 Conclusions and Future Work

In this paper we have proposed a method for interactive design of body-supporting surfaces that is driven by the pose of the human body as well as the pressure distribution on the body's surface. Our method is intended to help designers create appropriate surfaces digitally without additional empirical design passes on the one hand, and to ensure physical plausibility on the other hand. Further, it aims at interactive rates in the range of a few seconds.

Our main contribution is an interactive modeling system that utilizes captured body poses and computes an importance field that is proportional to the pressure distribution on the body for a given pose. This distribution indicates where the body should be supported in order to easily hold a particular pose, which is one of the measures of comfortable sitting.

We tested our system with design students and presented a number of results from these sessions. We also demonstrated a fabricated result. In the future, our method could serve as a basis for interactive design of various interesting furniture, for instance inflatable furniture, bean bags, as well as design furniture in general.

References

- [Bas18] BASTIONI M.: Manuel Bastioni official page, 2018. 2, 5, 8
- [BKP*10] BOTSCH M., KOBELT L., PAULY M., ALLIEZ P., LEVY B.: *Polygon Mesh Processing*. Ak Peters Series. Taylor & Francis, 2010. 5

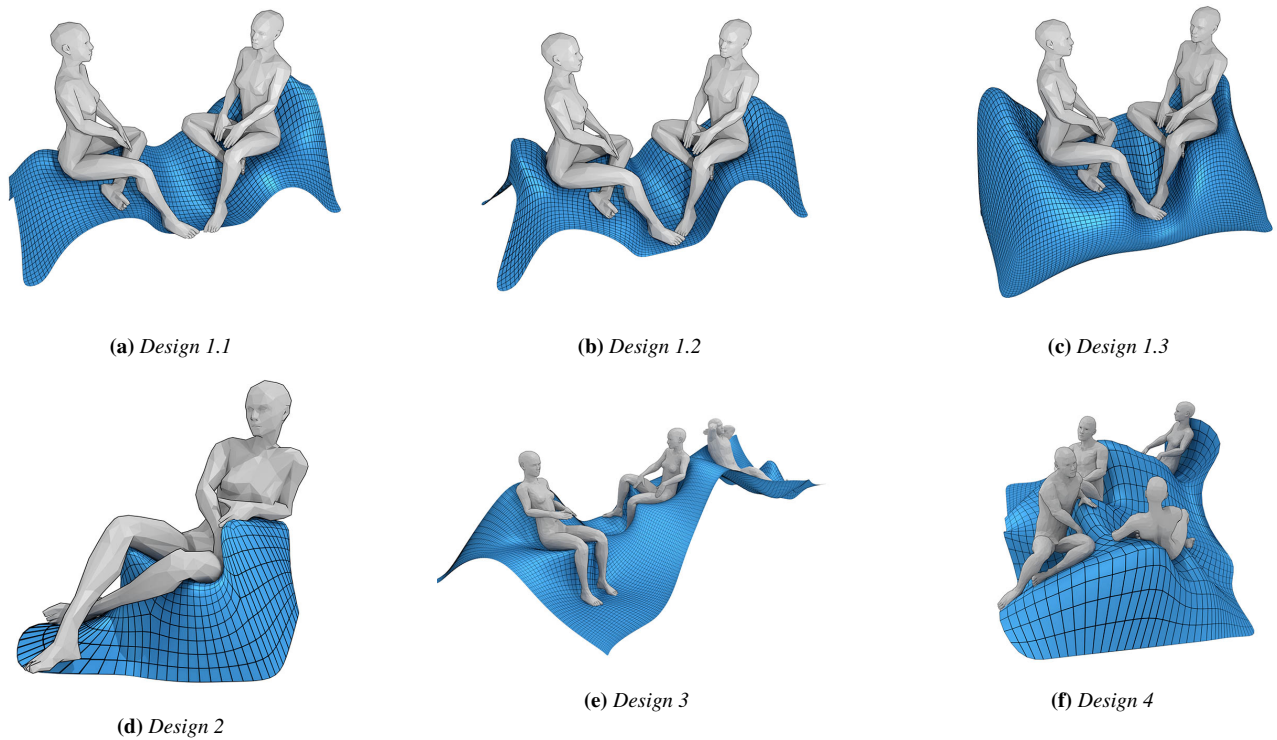
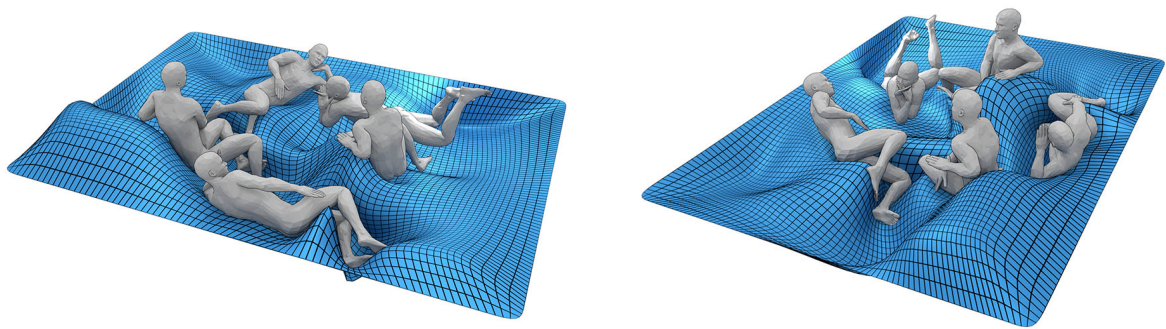
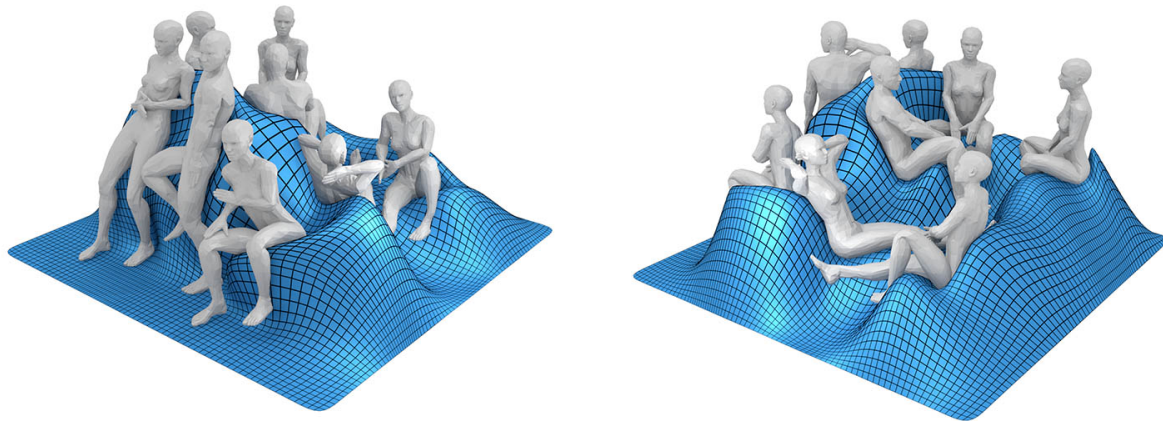


Figure 9: Several results created by design students using our method. The top row shows three design variations using the same input poses achieved by fixing different control vertices oder changing design parameters. Please refer to Section 7 for more details.

- [BP07] BARAN I., POPOVIĆ J.: Automatic rigging and animation of 3D characters. *ACM Transactions on Graphics* 26, 99 (jul 2007), 72. [5](#)
- [BSK*16] BARTLE A., SHEFFER A., KIM V. G., KAUFMAN D. M., VINING N., BERTHOUSOZ F.: Physics-driven pattern adjustment for direct 3D garment editing. *ACM Transactions on Graphics* 35, 4 (2016), 1–11. [2](#)
- [Bus04] BUSS S.: Introduction to inverse kinematics with Jacobian transpose, pseudoinverse and damped least squares methods. *IEEE Transactions in Robotics and Automation* 17 (2004). [2](#), [5](#)
- [CC78] CATMULL E., CLARK J.: Recursively generated B-spline surfaces on arbitrary topological meshes. *Computer-Aided Design* 10, 6 (nov 1978), 350–355. [2](#)
- [CWH*07] CHENG K.-S., WENPING WANG, HONG QIN, WONG K.-Y., HUAIPING YANG, YANG LIU: Design and Analysis of Optimization Methods for Subdivision Surface Fitting. *IEEE Transactions on Visualization and Computer Graphics* 13, 5 (sep 2007), 878–890. [2](#)
- [DKEV03] DE LOOZE M. P., KUIJT-EVERS L. F., VAN DIEËN J.: Sitting comfort and discomfort and the relationships with objective measures. *Ergonomics* 46, 10 (2003), 985–997. [2](#), [3](#)
- [DSP06] DER K. G., SUMNER R. W., POPOVIĆ J.: Inverse kinematics for reduced deformable models. *ACM Transactions on Graphics* 25, 3 (jul 2006), 1174. [2](#)
- [FCSF17] FU Q., CHEN X., SU X., FU H.: Pose-Inspired Shape Synthesis and Functional Hybrid. *IEEE Transactions on Visualization and Computer Graphics* 23, 12 (dec 2017), 2574–2585. [2](#)
- [GGV11] GRABNER H., GALL J., VAN GOOL L.: What makes a chair a chair? In *CVPR 2011* (jun 2011), IEEE, IEEE, pp. 1529–1536. [2](#)
- [IMT99] IGARASHI T., MATSUOKA S., TANAKA H.: Teddy. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques - SIGGRAPH '99* (1999), pp. 409–416. [2](#)
- [JHR*15] JUNG A., HAHMANN S., ROHMER D., BEGAULT A., BOISSIEUX L., CANI M.-P.: Sketching Folds. *ACM Transactions on Graphics* 34, 5 (2015), 1–12. [2](#)
- [JO16] JACOBSON A., OTHERS: {gptoolbox}: Geometry Processing Toolbox, 2016. [6](#)
- [KCGF14] KIM V. G., CHAUDHURI S., GUIBAS L., FUNKHOUSER T.: Shape2pose: Human-centric shape analysis. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 120. [2](#)
- [KL14] KANG C., LEE S.-H.: Environment-Adaptive Contact Poses for Virtual Characters. *Computer Graphics Forum* 33, 7 (oct 2014), 1–10. [2](#)
- [Law18] LAWRENCE N. D.: mocap: Matlab Motion Capture Toolbox, 2018. [6](#)
- [LPL*17] LI C., PAN H., LIU Y., TONG X., SHEFFER A., WANG W.: BendSketch: Modeling Freeform Surfaces Through 2D Sketching. *ACM Transactions on Graphics* 36, 4 (2017), 1–14. [2](#)
- [Lue83] LUEDER R. K.: Seat Comfort: A Review of the Construct in the Office Environment. *Human Factors: The Journal of the Human Factors and Ergonomics Society* 25, 6 (dec 1983), 701–711. [2](#)
- [MI07] MORI Y., IGARASHI T.: Plushie: An Interactive Design System for Plush Toys. *ACM Transactions on Graphics* 26, 3 (jul 2007), 45. [2](#)
- [MK05] MARINOV M., KOBELT L.: Optimization methods for scattered data approximation with subdivision surfaces. *Graphical Models* 67, 5 (sep 2005), 452–473. [2](#)
- [MTLT89] MAGNENAT-THALMANN N., LAPERRIÈRE R., THALMANN D.: Joint-dependent local deformations for hand animation and object grasping. In *Proceedings on Graphics interface '88* (dec 1989), pp. 26–33. [5](#)
- [Neu18] NEURON P.: Perception Neuron by Noitom | Perception Neuron motion capture for virtual reality, animation, sports, gaming and film, 2018. [3](#), [6](#)



(a) Design 5



(b) Design 6

Figure 10: Two additional designs of sitting landscapes for multiple persons.

- [NISA07] NEALEN A., IGARASHI T., SORKINE O., ALEXA M.: Fiber-mesh: designing freeform surfaces with 3d curves. *ACM Transactions on Graphics* 26 (2007), 41. 2
- [Pet00] PETERS J.: Patching Catmull-Clark meshes. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques - SIGGRAPH '00* (New York, New York, USA, 2000), ACM Press, pp. 255–258. 7
- [SLMI11] SAUL G., LAU M., MITANI J., IGARASHI T.: SketchChair: an all-in-one chair design system for end users. In *Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction - TEI '11* (New York, New York, USA, jan 2011), ACM Press, p. 73. 2
- [Smi09] SMITH M.: *ABAQUS/Standard User's Manual, Version 6.9*. Simulia, 2009. 3
- [Sta95] STAARINK H. A. M.: Sitting posture, comfort and pressure: assessing the quality of wheelchair cushions. *Journal of Rehabilitation Sciences* (1995). 2
- [SZGP05] SUMNER R. W., ZWICKER M., GOTSMAN C., POPOVIĆ J.: Mesh-based inverse kinematics. *ACM Transactions on Graphics* 24, 3 (jul 2005), 488. 2
- [UIM12] UMETANI N., IGARASHI T., MITRA N. J.: Guided exploration of physically valid shapes for furniture design. *ACM Transactions on Graphics* 31, 4 (jul 2012), 1–11. 2
- [UKIG11] UMETANI N., KAUFMAN D. M., IGARASHI T., GRINSPUN E.: Sensitive couture for interactive garment modeling and editing. *ACM Transactions on Graphics* 30, 4 (jul 2011), 1. 2
- [XLF*11] XIN S., LAI C.-F., FU C.-W., WONG T.-T., HE Y., COHEN-OR D.: Making burr puzzles from 3D models. *ACM Transactions on Graphics* 30, 4 (jul 2011), 1. 2
- [ZHD96] ZHANG L., HELANDER M. G., DRURY C. G.: Identifying Factors of Comfort and Discomfort in Sitting. *Human Factors: The Journal of the Human Factors and Ergonomics Society* 38, 3 (sep 1996), 377–389. 2
- [ZLDM16] ZHENG Y., LIU H., DORSEY J., MITRA N. J.: Ergonomics-Inspired Reshaping and Exploration of Collections of Models. *IEEE Transactions on Visualization and Computer Graphics* 22, 6 (jun 2016), 1732–1744. 2

Acknowledgments

We thank students who participated to the evaluation of the method: Ada Gulyamdzhis, Anna T. Pöll, Jasmin Plaikner, Jasmin Redl, Miriam Bachmann, and Tetyana Vovk. We thank Lukas Gersthofer for help with renderings. This research was funded by the Austrian Science Fund (FWF P27972-N31) and the Vienna Science and Technology Fund (WWTF ICT15-082).

Appendix

A Surface Fitting Details

For simplicity we will explain how to solve the optimization problem starting with point-to-point distances only. Given a mesh M^s of subdivision level s , the vertices of the next subdivision level $s + 1$

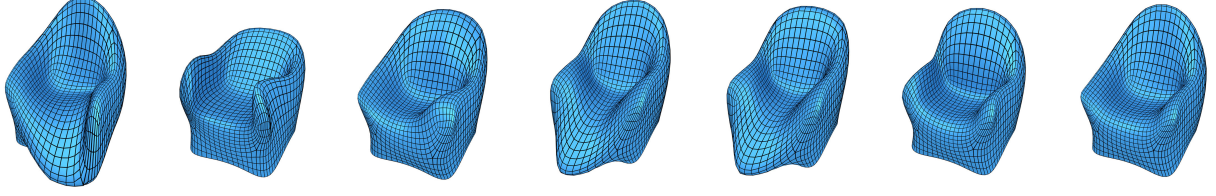


Figure 11: A series of designs using asymmetric input poses created with the control mesh shown in Fig 7a.

can be computed using the subdivision matrix \mathbf{S}^s :

$$\mathbf{V}^{s+1} = \mathbf{S}^s \mathbf{V}^s,$$

or, starting from the initial control mesh M^0 using

$$\mathbf{V}^{s+1} = \mathbf{S}^s \mathbf{S}^{s-1} \dots \mathbf{S}^1 \mathbf{S}^0 \mathbf{V}^0 = \bar{\mathbf{S}}^s \mathbf{V}^0.$$

Using the matrix $\bar{\mathbf{S}}^s$, it is possible to compute the optimal control mesh vertices \mathbf{V}^* by solving a system of linear equations

$$\hat{\mathbf{S}}^s \mathbf{V}^* = \mathbf{V}^b,$$

where $\hat{\mathbf{S}}^s$ is the matrix whose k th row is the row of $\bar{\mathbf{S}}^s$ corresponding to the vertex \mathbf{v}_k^s , and \mathbf{V}^b is the matrix containing the sample point coordinates \mathbf{v}_k^b of the body mesh in each row.

To modify this approach to work with the tangent distance, we need to separate the x -, y - and z -coordinates. Let

$$\mathbf{x}^* = \begin{bmatrix} \mathbf{V}_1^* \\ \mathbf{V}_2^* \\ \mathbf{V}_3^* \end{bmatrix}$$

denote the column vector of concatenated x -, y - and z -coordinates of \mathbf{V}^* and let $\hat{\mathbf{N}}$ denote the matrix whose k th row contains coordinates of normal vector \mathbf{n}_k . Let $\hat{\mathbf{N}}_1$ denote the matrix containing the x -coordinates of the normals \mathbf{n}_k in its diagonal, with $\hat{\mathbf{N}}_2$ and $\hat{\mathbf{N}}_3$ defined analogically. Then we can solve the following system of linear equations to find \mathbf{x}^* :

$$[\hat{\mathbf{N}}_1 \hat{\mathbf{S}}^s \quad \hat{\mathbf{N}}_2 \hat{\mathbf{S}}^s \quad \hat{\mathbf{N}}_3 \hat{\mathbf{S}}^s] \mathbf{x}^* = \hat{\mathbf{N}}^T \mathbf{V}^b.$$

The full system of equations, including the parameter τ to blend between point-to-point and tangential distance, the Laplacian \mathbf{L} as a smoothing term controlled by weighting parameter σ , and the weights γ that allow constraining vertices to their original location is given by

$$\begin{bmatrix} (1-\tau)\mathbf{W}\hat{\mathbf{S}}^s & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & (1-\tau)\mathbf{W}\hat{\mathbf{S}}^s & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & (1-\tau)\mathbf{W}\hat{\mathbf{S}}^s \\ \tau\mathbf{W}\hat{\mathbf{N}}_1\hat{\mathbf{S}}^s & \tau\mathbf{W}\hat{\mathbf{N}}_2\hat{\mathbf{S}}^s & \tau\mathbf{W}\hat{\mathbf{N}}_3\hat{\mathbf{S}}^s \\ \sigma\mathbf{L} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \sigma\mathbf{L} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \sigma\mathbf{L} \\ \gamma\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \gamma\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \gamma\mathbf{I} \end{bmatrix} \mathbf{x}^* = \begin{bmatrix} (1-\tau)\mathbf{W}\mathbf{V}_1^b \\ (1-\tau)\mathbf{W}\mathbf{V}_2^b \\ (1-\tau)\mathbf{W}\mathbf{V}_3^b \\ \tau\mathbf{W}\hat{\mathbf{N}}^T\mathbf{V}^b \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \gamma\mathbf{V}_1^0 \\ \gamma\mathbf{V}_2^0 \\ \gamma\mathbf{V}_3^0 \end{bmatrix},$$

with \mathbf{W} containing the vertex weights ρ_k in its diagonal.

B Jacobians for Inverse Kinematics

In our kinematic model, each physical joint j has three rotational degrees of freedom (Euler angles) for rotations about the local frame axes and therefore three corresponding entries in the joint angle vector $\boldsymbol{\theta}$. Only the hip joint, representing the root of all other joints, has three additional translational degrees of freedom for translations of the whole body in global x , y and z direction.

In order to explain how to compute each entry $J_{i,k}^{d_p}$ expressing the change of the value $E_i^{d_p}$ w.r.t. the parameter θ_k , we have to introduce the function $\delta(j,k)$ that returns 1 if the joint j is influenced by the parameter θ_k and 0 otherwise. Note that a joint j is influenced by a parameter θ_k not only when θ_k directly corresponds to j , but also when j is a direct or indirect child of the joint the parameter θ_k corresponds to. Finally, the function $w(i,j)$ returns the skinning weight from the joint j on the vertex \mathbf{v}_i^b and $\mathbf{u}(i,j)$ returns the contribution point for the vertex \mathbf{v}_i^b from the joint j in the skinning process.

Let \mathbf{a}_k be the axis and \mathbf{p}_j be the position of a rotational joint j corresponding to parameter θ_k . The computation of the derivative of the position of the vertex \mathbf{v}_i^b w.r.t. the parameter θ_k then comes down to

$$\frac{\partial \mathbf{v}_i^b}{\partial \theta_k} = \mathbf{a}_k \times \left(\sum_{j=1}^{n_\theta} \delta(j,k) w(i,j) (\mathbf{u}(i,j) - \mathbf{p}_j) \right).$$

If θ_k corresponds to a translational joint with translation axis \mathbf{a}_k (in our case only the hip joint), the derivatives can be computed significantly easier with

$$\frac{\partial \mathbf{v}_i^b}{\partial \theta_k} = \mathbf{a}_k.$$

Finally, we compute the entries for all Jacobians according to

$$J_{i,k}^{d_p} = \frac{\partial e_i^{d_p}}{\partial \theta_k} = \frac{r_i}{\|\mathbf{v}_i^b - \mathbf{v}_i^s\|} \left((\mathbf{v}_i^b - \mathbf{v}_i^s)^T \frac{\partial \mathbf{v}_i^b}{\partial \theta_k} \right),$$

$$J_{i,k}^{d_t} = \frac{\partial e_i^{d_t}}{\partial \theta_k} = r_i \left(\mathbf{n}_i^s \cdot \frac{\partial \mathbf{v}_i^b}{\partial \theta_k} \right),$$

$$J_{i,k}^p = \frac{\partial e_i^p}{\partial \theta_k} = - \frac{a \cdot r_i \left(\mathbf{n}_i^s \cdot \frac{\partial \mathbf{v}_i^b}{\partial \theta_k} \right)}{\pi \left(\left(a \cdot \left(\mathbf{n}_i^s \cdot \mathbf{v}_i^b - d_i^s \right) + b \right)^2 + 1 \right)}.$$