

Article

Real-Time Numerical Simulation for Accurate Soft Tissues Modeling during Haptic Interaction

Paolo Tripicchio ^{1,*} , Salvatore D'Avella ¹  and Emanuele Ruffaldi ² 

¹ Mechanical Intelligence Institute, Department of Excellence in Robotics & AI, Scuola Superiore Sant'Anna, 56010 Pisa, Italy; salvatore.davella@santannapisa.it

² Medical Micro Instruments (MMI s.p.a.), 56011 Pisa, Italy; emanuele.ruffaldi@gmail.com

* Correspondence: paolo.tripicchio@santannapisa.it

Abstract: The simulation of fabrics physics and its interaction with the human body has been largely studied in recent years to provide realistic-looking garments and wears specifically in the entertainment business. When the purpose of the simulation is to obtain scientific measures and detailed mechanical properties of the interaction, the underlying physical models should be enhanced to obtain better simulation accuracy increasing the modeling complexity and relaxing the simulation timing constraints to properly solve the set of equations under analysis. However, in the specific field of haptic interaction, the desiderata are to have both physical consistency and high frame rate to display stable and coherent stimuli as feedback to the user requiring a tradeoff between accuracy and real-time interaction. This work introduces a haptic system for the evaluation of the fabric hand of specific garments either existing or yet to be produced in a virtual reality simulation. The modeling is based on the co-rotational Finite Element approach that allows for large displacements but the small deformation of the elements. The proposed system can be beneficial for the fabrics industry both in the design phase or in the presentation phase, where a virtual fabric portfolio can be shown to customers around the world. Results exhibit the feasibility of high-frequency real-time simulation for haptic interaction with virtual garments employing realistic mechanical properties of the fabric materials.

Keywords: interactive systems; human-computer interaction; haptic interfaces; numerical simulation



Citation: Tripicchio, P.; D'Avella, S.; Ruffaldi, E. Real-Time Numerical Simulation for Accurate Soft Tissues Modeling during Haptic Interaction. *Actuators* **2022**, *11*, 17. <https://doi.org/10.3390/act11010017>

Academic Editor: Sang-Youn Kim

Received: 4 December 2021

Accepted: 5 January 2022

Published: 8 January 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Many robotics applications of deformable objects manipulations have spread out in the last years [1] in diverse fields such as the manufacturing industry [2], service robotics [3], and others. However, it is a fact that the robotic manipulation of deformable objects has not been studied as its rigid counterpart due to the complexity in modeling, perception, and control. The recent advances in machine learning techniques and computer graphics helped to give effective modeling techniques and data-driven paradigms for solving some of the issues in traditional methods [4–6]. However, it is a common opinion [7] that the interaction with deformable objects is a critical open problem in robotics. It also has relevant consequences in the industrial sector, where it focuses mainly on the manipulation of ropes [8] and clothing items. A great part of the works on deformable objects in robotics is application specific: for example, industrial applications concern problems such as the insertion of electric cables into switchgear [9]. The proposed work presents a soft tissue simulator framework that accurately models the interaction of the soft object during haptic interaction. The main contribution of the essay is the accuracy in the touch sensation of tissues through the haptic interaction, thanks to the underlying physics.

All the mechanical properties of textiles of a specific fabric taken as a whole are known as the “fabric hand” [10]. During the haptic sensation process, the brain combines several data coming from the mechanoreceptors of the skin, the kinesthetic sensors, and

the muscles stimuli in a complex elaboration signal. Such a process allows the appreciating and distinguishing of the different fabric properties of tissues by the haptic sensation. In the textile industry, the evaluation of fabric materials for commerce or production is usually performed manually by stroking the index and thumb fingertips of the hand on small fabric samples of the fabric surface to stimulate the sensing inputs on the evaluator. Thanks to the haptic system [11] and the tissues simulator presented in the manuscript, the fabric hand of specific garments, either existing or yet to be produced, can be perceived virtually with a high degree of fidelity. The potentiality of the system is twofold: it can be used to present and render the fabric hand to customers around the world without the need to carry a heavy set of fabric samples; and, it can be used in the design stage to test new fabrics before production. The main interest of the proposed method resides in the haptic fidelity during the interaction of the cloth simulation through the haptic feedback. Therefore, one crucial part is the high rate and the low latency of the computation, which is an added value with respect to many other software simulations that concentrate only on the graphic part of the problem.

The document is organized as follows: in Section 2, a brief introduction of existing projects on the topic is presented; Section 3 discusses in detail possible modeling approaches and the implemented techniques; Section 4 presents the implemented system for the simulation of and interaction with virtual clothes; Section 5 presents the experimental tests to validate the system; and, Section 6 draws the conclusions of the work.

2. Related Works

The study of textile behavior during hand manipulation is dated back to 1975, with a seminal study on the mechanical properties of the fabric performed by Kawabata [12]. The evaluation system presented in [12] is a standard method employed for measuring the properties of textile fabrics and for predicting the aesthetic qualities perceived by human touch. It includes five instrumental measurements to evaluate fabric bending, shearing, tensile, and compressive stiffness, as well as the smoothness and frictional properties of a fabric surface. Such an evaluation is also useful to analyze how fiber, yarn, fabric construction, and finishing contribute to the perception of softness. The results obtained from [12] allowed the modeling of the elasticity, plasticity, and viscosity of deformable surfaces. Starting from that point, many researchers developed computer models able to reproduce and simulate the correct behavior of several fabric materials. From the early work of Terzopoulos [13], the problem has been approached as a physically based simulation of deformable surfaces applying Lagrange equations of motion and finite element analysis (FEA) [14]. Later other research groups tried different approaches, for instance, using d'Alembert's principle and elasticity properties [15], and in the last years, a new trend has been to exploit particle systems connected by springs. Such an approach discretizes the material itself as a set of point masses, reducing the number of equations and the complexity of the continuum mechanics solution. In the following, starting from the general equation, a brief discussion of the most relevant modeling approaches is presented. The principal methods can be categorized into two main approaches: physically based and data-driven simulations.

The physically based modeling approach of deformable objects exploits the geometric representations of the item employing meshes or particles [16]. The motion of the particles or the vertex is expressed by the time derivative of Newton's second law: $M\dot{x} = F$, where M is the mass, x is the state vector, and F represents all forces acting on it. In the discrete case, the state x is a finite number of points, while in the continuous case, it is treated as a displacement function. Therefore, the motion of the body is obtained integrating starting from an initial state x^0 . For each particle, the motion is:

$$\begin{aligned}x_i^{t+1} &= x_i^t + v_i^t \Delta t \\v_i^{t+1} &= v_i^t + \frac{\Delta t}{m_i} (f_i^{int} + f_i^{ext}),\end{aligned}\tag{1}$$

where f_i^{ext} is the sum of all the external forces such as gravity, input forces, or external constraint forces, and f_i^{int} represents internal constraints forces that are subject to deformation parameters such as configuration x , \dot{x} , and the material properties. Several integration techniques can be used such as explicit Euler, implicit integration and Runge–Kutta method, backward Euler, and others. It is worth noticing that to guarantee a stable simulation, a small time step is required.

Three categories of physically based models can be individuated in the literature: mass-spring systems (MSSs), position-based dynamics (PBD), and continuum mechanics (FEMs). In MSSs, each vertex has a mass, and vertices are connected through spring edges, forming a network. The internal force f_{i-j}^{int} between the pair i - j is expressed by a spring-damping relation along the displacement direction. MSS has been used for diverse objects such as rope [17] or cloth [18] because it is easy to implement and fast to simulations. However, it is not suitable for complex elastic effects, and, therefore, increasing the mesh resolution, the quality of the fidelity decreases. In addition, since the model does not have any physical meaning related to the parameters, it needs a huge tuning effort to obtain the desired performance. PBD models materials as discrete particles without meshes. The simulation is based on an implicit integration deriving the internal forces from holonomic constraints. The peculiarity of such an approach is that it assigns a stiffness level k_j to each constraints and derives v_i^{t+1} from $(x_i^{t+1} - x_i^t) / \Delta t$, where x_i^{t+1} is obtained as an optimization problem, projecting iteratively candidate position to the constraint manifolds with a Gauss–Siedel method. Position-based dynamic approaches are fast with high stability and can models several constraints. They have been used to simulate many objects [19]. However, they are not effective for highly stiff systems. In particular, the stiffness of the system relies on the integration step size, and it does not simulate force effects accurately. In conclusion, PBD gives simulations that are visually plausible but not always physically realistic. In addition, it does not allow for a physical interpretation for some of the parameters. FEM can model material deformation with a physically accurate description for a continuous domain. The state x^t is a displacement function over the coordinates of the material relative to an initial configuration x_0 , also known as rest shape. The gradient of such displacement reveals the distortion of the original geometry. Equation (1) holds for each element and becomes:

$$\rho \ddot{x} = f^{int} + f^{ext} = \nabla \cdot \sigma + f^{ext} \quad (2)$$

where ρ is the density, σ represents the stress and is a symmetric matrix, and the last term of the equation expresses the internal elastic effects. Often, the Hookean materials assumption, which is a linear dependency between the stress and the measurement of the material deformation, is used for modeling small deformation around the rest shape configuration. The linear coefficient matrix for isotropic materials is derived from the Young modulus (stiffness) and the Poisson ratio (lateral and longitudinal strain). For more complex modeling, usually, an energy term is introduced to penalize the amount of deformation, defining hyperelastic relations that are nonlinear, i.e., the Neo-Hookean model [20]. It is worth noticing that the mathematical differential Equation (2) can only be solved for 1D string problems. In the other cases, FEM is used. Such modeling has high physical fidelity and is fast in the linear case, and model parameters have a clear physical interpretation. However, it is computation expensive in nonlinear cases, where it cannot reach real-time performances.

Data-driven methods are quite recent approaches and employ machine learning algorithms to simulate the deformations of soft tissues. They are faster than their physically based counterpart, but even though they are highly realistic, they cannot generate dynamic deformation. Data-driven methods are widely used for face [21] and body [22] animations but are not interactive.

Many simulations based on the aforementioned theoretical models exist and are employed in graphics, robotics, and computer vision. SOFA (Simulation Open-Framework Architecture) [23] models tissue and surgical interaction. It gives built-in mass-spring

models and diverse FEM models, i.e., Neo-Hookean, linear, and corotational linear models. It has been used for vision-based tip force estimation [24,25]. NVIDIA PhysX models cloth using PBD and has been used for haptics and force prediction [26,27] in human dressing tasks. MuJoCo [28] uses a constraint-based method for solving interaction forces that is similar to PBD. It has been applied in learning rope manipulation planning [6] and folding primitives [29]. Several works have investigated the usage of many simulators for deformable objects [28,30]. However, none of such simulators were selected as a benchmark for textile object manipulation, as Mujoco is for rigid objects.

The solutions presented in the literature are usually conceived for usage in a graphical simulation; thus, they need an update rate of just 30–60 Hz to generate a smooth visual transition. All recent advancements focus on realism but are not tailored for haptic interaction or high update rates. Indeed, whenever the main goal is to develop a simulation that involves a haptic interactive system, it is a rule of thumb to adopt a feedback rate above 1 kHz to maintain good stability properties of the system for rigid body contact. For a collision with a deformable surface, this requirement can be lowered, continuing to yield good stability issues and correct perception of touch but still needing an update rate of an order of magnitude greater than the one used in graphical simulations. The finite element implementation, even being computationally expensive, can be accelerated using an implicit numerical integration method to solve the well-known partial differential equation

$$\ddot{x} = M^{-1} \left(-\frac{\partial E}{\partial x} + F^{ext} \right) \quad (3)$$

where the vector x is the geometrical state of the cloth, M its mass distribution, E the cloth's internal energy, and F^{ext} represents all external forces acting on the cloth surface.

The computational time for such an implementation can be even lowered, reducing the number of calculations steps taking into consideration that the linear system is in sparse form and adopting a modified conjugate gradient (CG) iterative method [31]. However, the time needed to solve a CG step increases with the number of finite elements considered. The preconditioning of the system matrix and multidomain approaches can be introduced to obtain an extra speedup in the computation time.

The proposed work employs FEM in simulation and addresses the aspect of the sliding of the textile between the fingers. The target device used for the experimentation is a wearable haptic interface for the fingertips.

3. Discrete Modeling

Starting in a one-dimensional domain (1D), the simplest deformable element to model is a string. The continuous (distributed) system that represents the string is given by the following wave partial differential equation:

$$\rho \frac{\partial^2 U}{\partial t^2} = E \frac{\partial^2 U}{\partial x^2} + F^{ext}, \quad (4)$$

where ρ is the material density, E is the elastic module, F^{ext} is the contribution of external forces to the system, and U is the surface domain. Equation (4) is a hyperbolic equation and needs both the initial and boundary conditions to be solved. By starting from a calculus problem, there is the need to derive a simpler representation and finally obtain an algebraic solution to treat the problem with numerical algorithms. This final step involves discretization and allows the use of the computer for the solution of the initial problem.

A large number of publications on the topic employ the so-called 'particle' model in which a mass-spring representation is used to model the deformable characteristics of objects. In order to obtain this simplified representation, there is the need to substitute the complete and exact continuous system with a lumped description of the string mechanical system, thus passing from a partial differential equation (PDE) to an ordinary differential equation (ODE) system. In fact, from one side, the distributed system gives exact results and is the ideal system to be used in scientific simulations; on the other side, its computation

is expensive and sometimes even impossible because there is no closed solution to the problem. Given these premises, the goal of the modeling is to preserve the accuracy of the system by adopting a discrete and tractable algebraic approximation to the original calculus problem. Finite difference (FD) methods involve the substitution of spatial derivatives with polynomial approximations. In the case of uniform spaced mesh, it is possible to obtain the lumped equation of the string given by:

$$M_i \frac{d^2 U_i}{dt^2} = K_{i+1}(U_{i+1} - U_i) - K_i(U_i - U_{i-1}) + F^{ext} \quad (5)$$

where M_i is the mass of the i -th particle, K_i , K_{i+1} are elastic coefficients (spring constants), and U_i is the value of the function at the sampling point i .

By giving a cross-section A of the string, it can be obtained that the K_i spring constant has a physical meaning, and it is given by $\frac{A E}{l^2}$. Usually, this physical property is not taken into consideration, and the K_i parameter is chosen through a minimization procedure or empirically for stability issues.

Considering a bidimensional (2D) domain for the nodal points U_{ij} and choosing the FD approach, the simplest idea is to use several 1D string elements and connect each particle on each string with the corresponding particle of the neighboring string as in Figure 1.

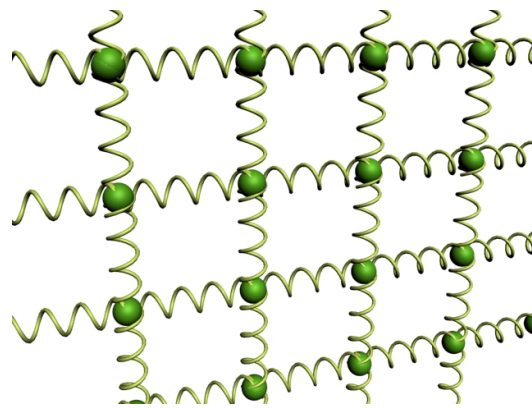


Figure 1. Particles connection in a 2D grid finite difference approximation.

In this way, it is possible to obtain a simple approximation of soft tissue. The approximation introduced here is given by summing up the errors in the two principal directions leading to perceivable errors in four neighbor modeling approaches.

However, this model is designed for equally distributed rectangular meshes. Hence, it provides poor results if employed for simulating generic triangular meshes. A second drawback is that it is challenging to obtain correct simulation parameters for an actual cloth material because the model parameters have no direct relationship with the physical parameters that are commonly employed in the industry.

For such a reason, when there is the need for accurate physical simulations, a Finite Element dynamical model can be used by employing, for instance, 2D triangular elements organized in an unstructured mesh for the modeling of a garment. In this case, the material parameters are considered by the FEM formulation itself that is obtained from the elasticity theory. The FEM model considers the material as a continuous geometry that is decomposed on discrete elements and solves for a finite set of points associated with them. The solution is finally obtained interpolating on the whole domain. A triangle mesh can be used to represent the surface of the cloth in a discrete domain. FEM elements are the mesh triangles themselves, and FEM nodes are their vertices. Thus, the quality of the simulation strongly depends on the resolution of the triangulation.

3.1. Corotational FEM Modeling

Among the possible Lagrangian kinematic descriptions, the Corotational (CR) [32] one is based on the assumption that displacements and rotations can be arbitrarily large, but deformations must be small [33]. The CR description introduces a decomposition of the motion tracking into two components (Figure 2). Starting from a base configuration that represents the origin of displacements, the final configuration is decomposed into a corotated configuration given by a rigid body motion of the base configuration and a deformed configuration where element deformations are obtained concerning the corotated configuration. Assuming that only small deformations occur between the corotated and deformed configuration, the CR approach allows using a linear formulation to compute the internal forces. Under this assumption, it is possible to represent stiffness by using a symmetric matrix K .

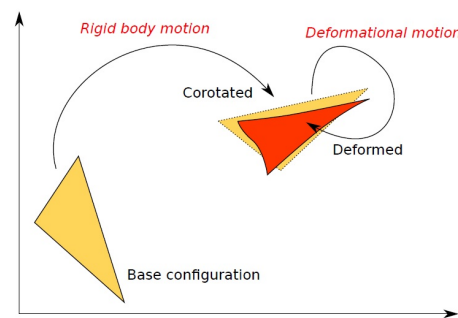


Figure 2. Corotational description of mesh deformation.

Calling e_1 and e_2 two adjacent edges of a triangular element in base configuration, it is possible to compute $\bar{e}_2 = e_2 - (e_1 e_2) e_1$ and $e_3 = \bar{e}_2 \times e_1$ and obtain a base of the 3D space with the matrix $N = [e_1, \bar{e}_2, e_3]$. Computing in the same way a basis \bar{N} for the deformed configuration, the rotational part is obtained as in [34]:

$$R = N\bar{N}$$

The rotation R is used to transform the deformed triangle in the plane of the base configuration, compute the elastic force linearly in that configuration, and then rotate back the obtained forces in the 3D space.

3.2. Fabric Modeling

In work proposed in [14], the formulation of a linearized stiffness matrix directly characterized by physical elastic parameters taken from Kawabata measurements can be found. In particular, in this implementation, the behavior of the simulated fabric takes into account membrane, bend, and viscosity forces.

Membrane forces are due to stretch, compression, and shear contributions. Using the CR formulation, any deformation occurs in the z axis, and the analysis can be carried out on a two-dimensional domain. The force acting on the triangle t can be determined by considering, for each vertex, a contribution produced by the deformation of the connected edges. For a vertex a and edge ab , the force component results as:

$$F = R^t K_{ab}^t \left((R^t)^T x_a - \bar{x}_a \right)$$

where x_a is the position of the vertex a in the deformed state, \bar{x}_a is its position in the base configuration, and R^t is composed by the first two columns of the corotational matrix R . The stiffness matrix is computed as follows:

$$[K_{ab}^t]_{ij} = \left(\sum_{k,l=1}^2 \frac{\partial N_a}{\partial u_i} C_{ikjl} \frac{\partial N_b}{\partial u_j} \right) \Omega^t$$

where C is the elastic tensor containing the material properties, Ω^t is the area of the triangle in the rest state, and $\frac{\partial N_x}{\partial u_y}$ are partial derivatives that can be easily computed for a 2D triangle as:

$$\begin{bmatrix} \frac{\partial N_1}{\partial u_1} & \frac{\partial N_2}{\partial u_1} & \frac{\partial N_3}{\partial u_1} \\ \frac{\partial N_1}{\partial u_2} & \frac{\partial N_2}{\partial u_2} & \frac{\partial N_3}{\partial u_2} \end{bmatrix} = \frac{1}{2\Omega} \begin{bmatrix} y_2 - y_3 & y_3 - y_1 & y_1 - y_2 \\ x_3 - x_2 & x_1 - x_3 & x_2 - x_1 \end{bmatrix}$$

The global stiffness matrices can be obtained as:

$$[K_m]_{ab} = \left(\sum_{t \in \Gamma(a,b)} R^t K_{ab}^t (R^t)^T \right)$$

where $\Gamma(a, b)$ is the set of all triangles containing the vertices a and b . The final membrane force is computed as:

$$F_m(X) = K_m X - \bar{K}_m \bar{X}$$

where X is the coordinate vector of the deformed vertices and \bar{X} is the coordinate vector of the base vertices.

In order to consider bending forces, it is necessary to drop the two-dimensional approach and employ finite elements of a higher order. Such contributions can be generated by projecting the Laplacian operator onto the normal surface direction. By calling $P = nn^t$, the projection operator relative to the normal n , it is possible to proceed to calculate the Laplacian operator as:

$$K_{ab}^t = \begin{pmatrix} \frac{\partial N_a}{\partial u_1} B_1 \frac{\partial N_b}{\partial u_1} & 0 \\ 0 & \frac{\partial N_a}{\partial u_2} B_2 \frac{\partial N_b}{\partial u_2} \end{pmatrix} \Omega^t$$

where B_1 and B_2 are the bending moduli in the weft and warp directions. To estimate the surface curvature, the Laplacian is projected onto all vertex normals n_a^t , n_b^t , and n_c^t of the a , b , and c triangle vertices. A local stiffness matrix can then be calculated as:

$$Q_{ab}^t = \frac{1}{3} \sum_{i \in \{a,b,c\}} P_i^t R^t K_{ab}^t (R^t)^T P_i^t$$

From the local stiffness matrices, a global matrix K_b can be assembled and proceed to solve for the bending forces:

$$F_b(x) = K_b X$$

Internal friction and viscosity by contributions are obtained calculating the viscous matrix K_v in the same way as the elastic matrix, this time inserting the viscosity tensor in replacement of the elastic tensor. The viscosity forces can then be computed as:

$$F_v(v) = K_v v$$

Given the above formulation and time discretization with the backward Euler step, a linear system that results in a sparse symmetric linear problem is obtained, and it can be solved with the iterative conjugate gradient algorithm [35] through preconditioning of the system matrix (PCG).

3.3. Implemented Algorithm

The system proposed in the following section implements the fabric model thanks to the CR formulation introducing a preconditioned sparse symmetric linear problem to be solved by the iterative conjugate gradient algorithm. The fabric mesh is composed of discrete elements, each constituted by nodes (vertices). The number of nodes depends on the representation topology (3 for triangles, 4 for tetrahedra, etc.). The simulation data elements can be static/precomputed or dynamic. In particular, starting from a rest position,

the simulation computes x_i for all the mesh nodes in the subsequent time steps according to the dynamical and mechanical properties of the fabric. Figure 3 shows the dependency graph in the computation. For each triangle, the area, the surface normal, and its rotation matrix are calculated. The stiffness matrices (K_m , K_{vm} , and K_{bm}) are computed starting from the elastic, bending, and viscous tensors (C , B , D). Such elements are employed together to obtain the stiffness contributions at the local level (Q_*). Gathering the contribution of each triangle into sparse matrices A_* and considering the mass of the fabric associated with each node, the forces accumulated by each element over adjacent nodes and node accumulated forces can be obtained. The external forces F_i and the nodes velocities v_i contribute to the set of equations of the linear system. Finally, the PCG algorithm computes the change in the velocity of the nodes that enables the update of the position of the nodes for the next iteration.

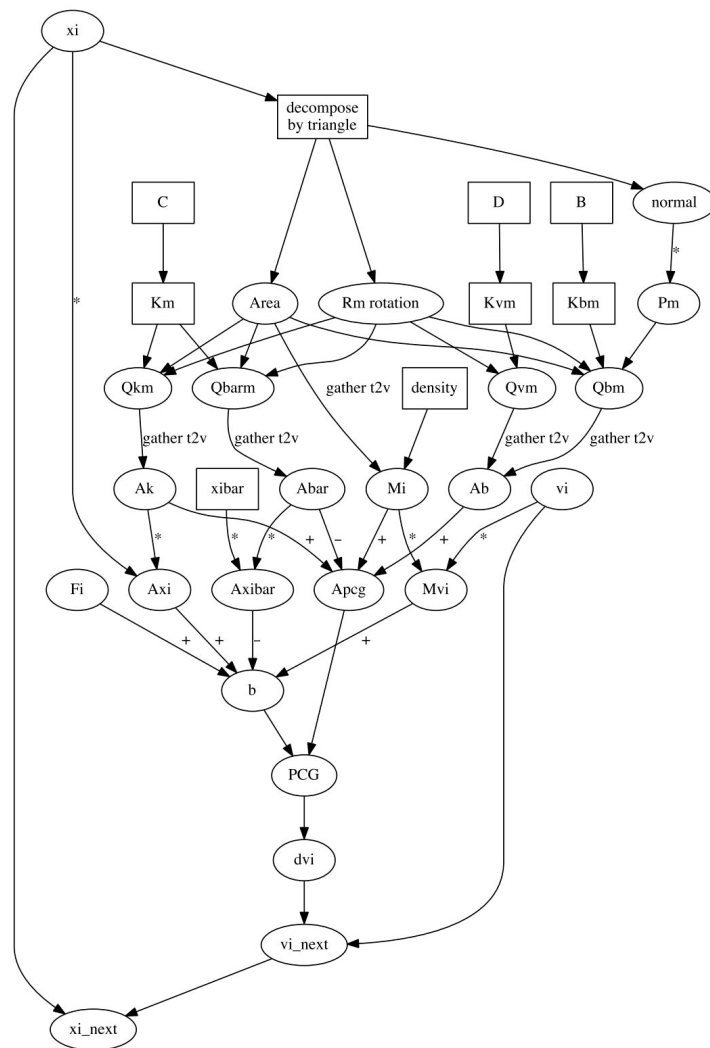


Figure 3. Dependency graph in the computation. Starting from a rest state x_i of the shell and employing the elastic (C), bending (B), and viscous (D) tensors, the algorithm decomposes the shell in triangles computing the area (volume), surface normal, and rotation matrix for each triangle element. It computes the stiffness matrices (K_m , K_{vm} , and K_{bm}) and their contribution at the local level (Q_m , Q_{vm} , and Q_{bm}). Once the information from the triangles is gathered, the computation passes at the vertex level formulating a sparse system with stiffness matrix A . Once integrated, the mass contribution M of each shell node, forces are accumulated over adjacent nodes, and the internal forces contribution can be computed. External forces F_i and nodes velocities contribute to the computation of the updated state. A preconditioned conjugate gradient (PCG) algorithm is implemented to update the position of the nodes by computing the velocity change and update the state for the next iteration.

4. Interaction with the Fabric

Since the user's interaction with the fabric happens mainly through the use of the index and thumb fingertips, the haptic device acts exclusively on such fingers. The developed prototype is composed of a kinesthetic haptic device and a tactile display element, each embedded with its own controllers. The fabric simulation is run on a dedicated computer connected with gigabit ethernet to the embedded controller. The functional components of the system are depicted in Figure 4.

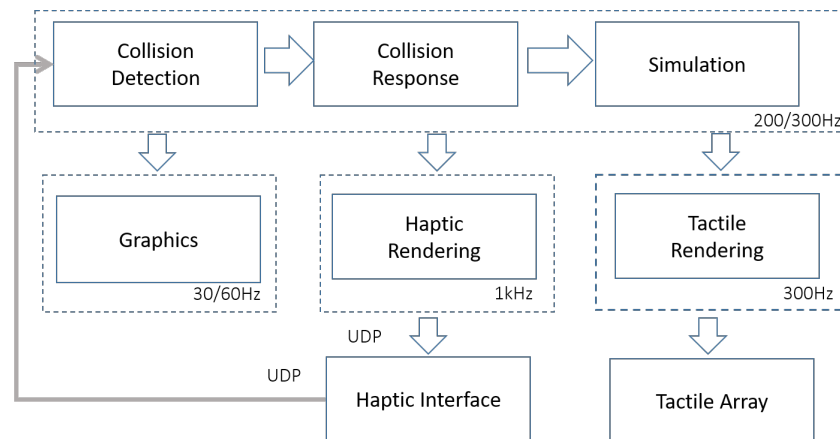


Figure 4. Diagram of the functional elements of the system annotated with the data rates. The communication between the haptic interfaces and the haptic rendering computer makes use of The UDP network protocol to achieve a high communication rate needed for realistic rendering of the interaction.

The kinesthetic feedback is produced by the hand exoskeleton introduced by Fontana et al. [36]. This device allows exerting controlled forces on the index and thumb fingertips of the operator while assuring that the finger motion degrees of freedom are not constrained. Concerning the original assembly, the mechanical plate that comes in contact with the operator's fingertips (i.e., the end-effector) is substituted with a tactile array device to deliver to the user skin spatial and temporal patterns. The tactile transducer is a 5×6 array based on a custom electromechanical minisolenoïd designed and optimized for reduced encumbrance and the weight [37]. The actuator has a diameter of 2.4 mm and is capable of delivering a maximum force of 73 mN. Figure 5 shows the employed hardware interfaces: the tactile array setup and the hand exoskeleton.

The simulation is run by synchronizing the actual time with the simulation time. Time synchronization is a delicate element that must be satisfied to maintain simulation and interaction stability. The collision detection and response modules determine if a collision occurs between the virtual representation of the operator's fingertips and the simulated cloth and eventually render a force into the simulation. In the system, undeformable spheres were chosen as the virtual representation for the operator's fingertips. Hence, the collision detection could be run either per vertex or triangle based. Since the sphere is rigid, the collision response may induce shifting motion on the cloth vertices. The haptic rendering is based on a virtual coupling between the interaction spheres and the index and thumb end-effectors acting as a low-pass filter for the different execution rates. The tactile rendering drives solenoids to produce specific patterns on the operator's skin. A graphic element is used to visualize the simulation every time a new computation is ready with a stereographic setup (projector and projection screen or 3D monitor). The stereographic setup allows displaying a proper depth sensation to the operator during the manual evaluation. To further reduce the stress on the operator's hand due to the weight of the haptic device, the latter can be attached from the wrist on a weight-compensating device.

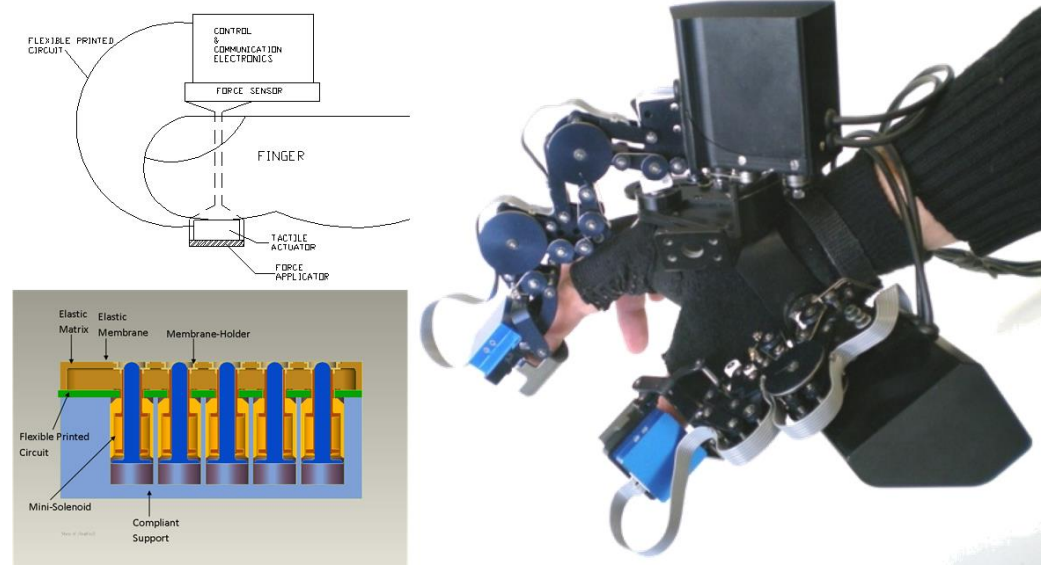


Figure 5. Hardware components of the system. On the **left**, the design of the tactile array placed under the operator's fingertips. On the **right**, the haptic exoskeleton that renders the kinesthetic feedback.

Solving Collisions

The collision check between the cloth and the operator's fingertips is evaluated by computing the distance between implicit sphere surfaces and textile vertices. An alternative option could be the one of checking the collisions between the spheres and each triangle composing the textile. However, the computational cost of this operation is larger and would require propagating the forces exerted on a specific point of the triangle surface back to the triangle vertices. Given the dimension of the sphere radius (fingertip width) compared to the distance between nodes in the simulated textile (dimension of the triangles), the per vertex collision check is a sufficient approximation that gives a boost to the computational complexity of the algorithm. The cost of this operation is linear to the number of vertices.

For each vertex colliding with the sphere (see Figure 6) in the contact phase, a normal vector \vec{n} directed from the origin of the sphere to the vertex is computed. From the intersection of the normal vector with the sphere surface, a point P is selected. The corresponding vertex is then moved instantly on the location of point P before the simulation phase, and it is constrained to move along the tangent space of the normal vector \vec{n} . The velocity delta of motion along the normal direction is computed following point impact mechanics considering the bounciness factor. By inserting into the equations a friction model between the spheres and the cloth, a sliding relative velocity is computed considering the friction coefficients. In the case of static friction, the vertex is constrained exactly on the contact point P .

In the case in which both the spheres are colliding with the same vertex, the response must take into account both the constraint directions from which the tangent space must be computed. If the directions of the normals are orthogonal, the tangent space is a line, and the possible direction of motion is unique.

The constrained solution is fundamental to obtain the forces exerted from the cloth to the fingers and then back to the haptic interface. Concerning penalty-based models, the force feedback is computed based on the penetration depth of textile vertices on the sphere. Conversely, in the constrained solution, the force feedback is obtained from the work performed by the constraint reaction. Following the work proposed in [31], the constraint is inserted in the dynamic equation by projecting the problem on a subspace where such a constraint is always satisfied. This effect can be obtained by filtering the result of the PCG. Figure 7 shows the effect of constraint reactions on the simulation.

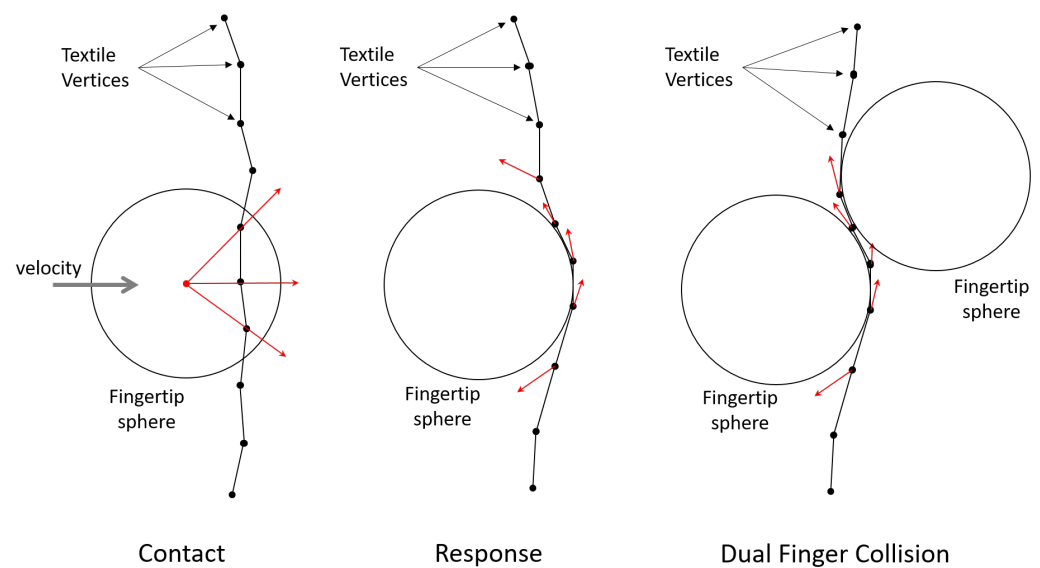


Figure 6. Diagram of the collision response with single- and double-contact spheres on a single filament of a textile. At left, during the contact phase, the normals with the colliding vertices are computed. In the middle, the response phase result comprising both the collision and the update step is shown. On the right, an example of a double-sphere collision is presented.

Fixed vertices

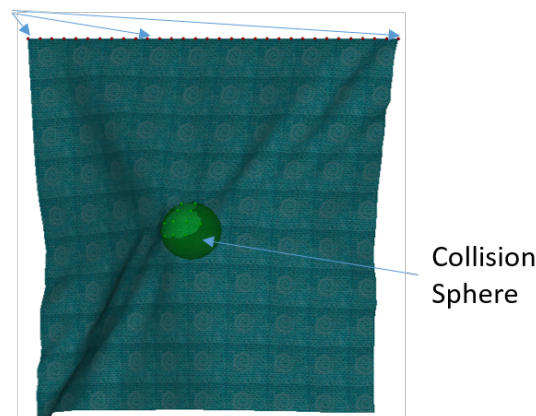


Figure 7. One finger interaction with 32×32 vertices simulated cloth. The top row of vertices is fixed. Green points over the sphere surface are colliding textile vertices.

5. Evaluation

In the following, some assessments of the proposed approach are presented.

5.1. Catenary Configuration

In order to prove the effectiveness of the FEM simulation, several testing configurations were evaluated. In particular, the catenary configuration was assessed by placing a rectangular fabric element perpendicular to the gravity field, fixing a row of vertices at the two extremities of the element, and finally checking the simulation result on the free-to-move vertices. The experimental results show negligible deviations from the exact analytic solution, and the weight force is distributed uniformly on each node of the fabric. Such an experiment proves that the only approximation is given by the dimension of triangular elements. In addition, if properly chosen, the simulation renders a realistic behavior of the cloth during the interaction with the human finger.

5.2. Collision with the Fingertip

Figure 8 shows a snapshot of a time frame of the simulation where a 32×32 elements cloth with dimensions 20×20 cm fixed at two adjacent extremities collides with a fingertip. In the picture, a schematic representation of the fabric vertices, with the classification in free-to-move, fixed, or colliding categories, is shown together with a representation of the collision force acting on the cloth. The direction and magnitude of the collision/reaction forces result as consistent with the fabric interaction and is well within the limits of the rendering capabilities of the system haptic interface.

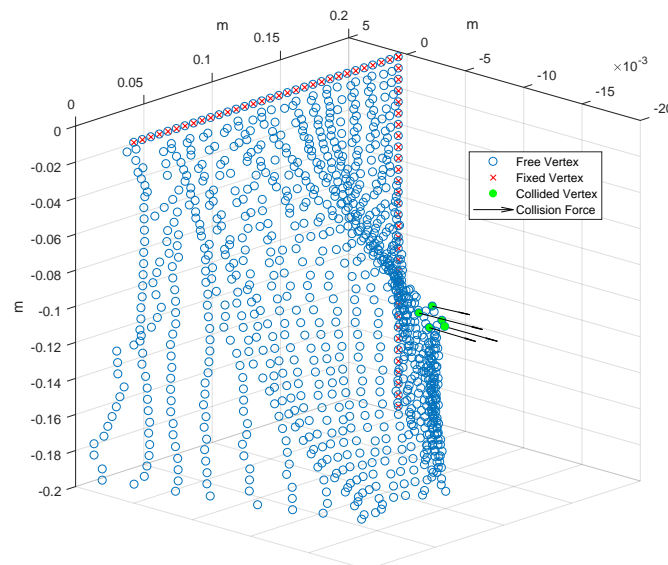


Figure 8. One finger interaction with 32×32 vertices simulated cloth shown as a 3D scatter plot. The top and the right row of vertices are fixed. Green circles are the colliding vertices with the sphere surface and the black arrows represent the collision forces.

5.3. Computational Complexity

An estimation concerning the computational complexity and the memory consumption of the algorithm can be formulated. It is based on the number of vertices in the cloth N_v , the number of collisions during the interaction N_c , and the number of iterations of the PCG algorithm N_i . The memory consumption of the model is estimated as $607N_v$ floating-point numbers. Assuming a 32×32 vertices configuration and single-precision floating-point computation, the occupied memory is about 2.3 MB, while for a 128×128 tissue in single-precision, a 38 MB model is obtained. For such a reason, memory consumption is not considered an issue. The computational complexity can be obtained by the following flops estimation that takes into account all the mathematical operations performed on the presented algorithm:

$$C = 7940N_v + 36N_c + 156N_vN_i + 36N_cN_i$$

As such, by increasing each of the considered factors, the computational load of the algorithm increases proportionally.

In order to improve the computation speed of the simulation steps, the algorithm was implemented for parallel computation employing both multi-CPU and GPU development. However, the two steps that mostly require computation time in the algorithm are related to the update of the FEM model and the PCG. The GPU implementation requires the transfer of data from and to the GPU memory during the update step. Therefore, the speed-up in this operation is limited. The PCG step is mostly iterative, and as such, it cannot be efficiently parallelized. For such a reason, the GPU implementation does not introduce evident improvements regarding the computation speed. The multi-CPU solution instead

performed better. A comparison of mean simulation time for different garment dimensions and different core exploitation is shown in Figure 9. In particular, the figure shows the whole iteration time and the contribution of the update and PCG steps. In all the tests, the computing hardware is a Xeon E560 Quad-Core Intel CPU at 2.4 Ghz mounted on an Intel S5520SC motherboard, and the computation uses double-precision floating-point representation.

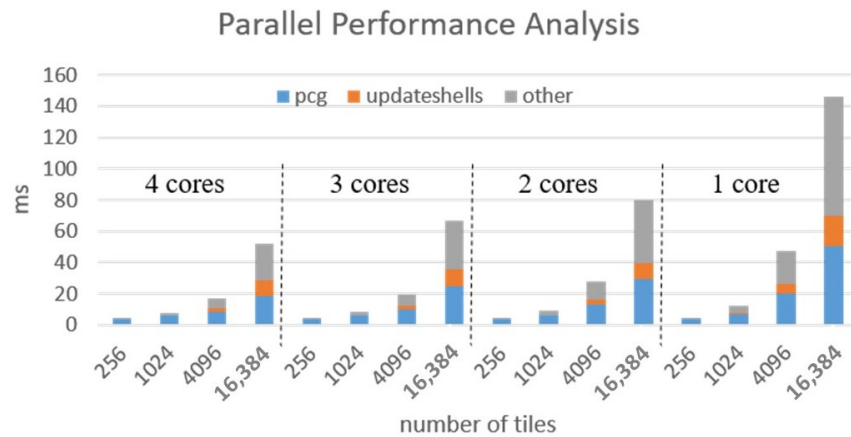


Figure 9. Processing time of the steps of the simulation algorithm. Comparison for different garment sizes (16×16 , 32×32 , 64×64 , and 128×128 elements) and the usage of multiple core configurations (1–4 cores) is shown. The computational steps were grouped showing the most expensive computations: The preconditioned conjugate gradient (PCG), the update of the overall shell (updateshells) and the rest of the operations (others).

5.4. Fabric Hand Fidelity

A qualitative sensing perception experiment was conducted to assess the fabric hand fidelity of the simulated tissues. In the textile industry, the evaluation of fabric materials is usually performed by stroking the index and thumb fingertips of the hand on small fabric samples of the fabric surface. During the haptic sensation process with the hand, the brain combines several data (coming from the mechanoreceptors of the skin, the kinesthetic sensors, and the muscles stimuli) into a complex signal to appreciate and distinguish the different fabric properties of the tissues.

Psychometric functions describe the relation between a physical intensity of a stimulus and the intensity perceived by a user. The psychometric procedure involves the observation of subjects' responses to a sequence of stimuli. Different stimulus conditions are usually employed to obtain perception thresholds along the psychometric function stimulus axis [38]. In a forced-choice design of psychometric experiments, a pair of stimuli are presented to the user, with one of them being the reference stimulus to be identified. In particular, this work adopted a two-alternative forced-choice (2-AFC) design choosing, as a reference stimulus, a wool fabric with a bending coefficient of $15 \times 10^{-6} \text{ Nm}^2/\text{m}$ on the warp and weft directions according to the KES-F (Kawabata Evaluation System for Fabrics). The users were asked to haptically explore two different simulated textiles through finger interaction by a circular motion of the finger according to the bending directions and to select, on each step, the one perceived as having a lower bending rigidity. The reference bending coefficient is alternated with one computed at every step using the QUEST algorithm as proposed by Watson and Pelli [39]. QUEST is a Bayesian adaptive method that evaluates thresholds of psychometric functions through sequences of steps. The chosen psychometric function is the Weibull function

$$W_T(x) = 1 - (1 - \gamma)e^{-\frac{\beta}{10}(x-\alpha)},$$

where β represents the slope of the psychometric function depending on the conditions, α is introduced to make zero the ideal test point, and γ expresses the probability of success at zero intensity. In the 2-AFC, γ is assigned a value of 0.5. In particular, QUEST uses

a probability density function representing the initial guess about the location of the threshold, and then it uses the Bayes theorem to update the algorithm after each response, choosing an optimal stimulus to be presented to the user. The algorithm stops based on the confidence interval of the threshold and the number of steps (50 in our case). This work employed the MATLAB implementation by Pelli. Six participants (one female, five males, aged 26–40, right-handed) performed two repetitions that required approximately 40 steps of QUEST. The results of the test session show that the subjects could recognize the presented reference stimuli with an error rate of about 30% (Figure 10).

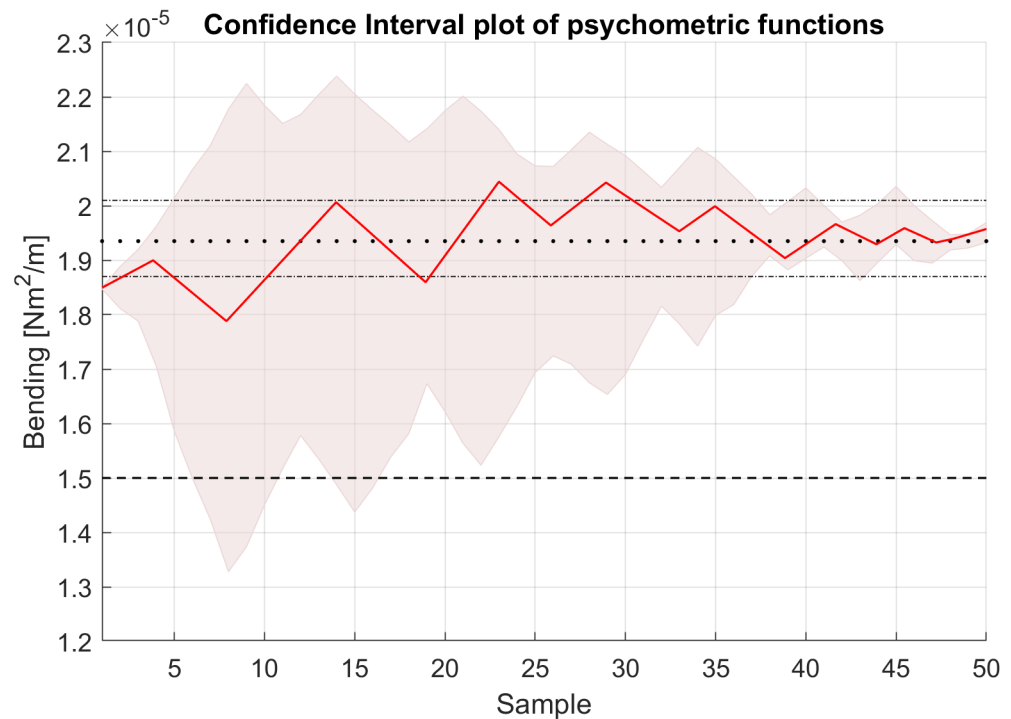


Figure 10. The red curve shows the psychometric function of a single assessment by one of the participants, and the shaded red area represents the confidence interval obtained from the twelve psychometric functions of the test participants. The resulting perceived threshold is found to be equal to $19.35 \times 10^{-6} \text{ Nm}^2/\text{m}$ (close to the initial hypothesis of $15 \times 10^{-6} \text{ Nm}^2/\text{m}$).

Considering typical bending moments in the literature, corresponding to wool fabrics, the range of values is large and depends on several factors such as the yarn waving and interleave among the others. For such reasons, having obtained close thresholds in 12 different psychometric tests confirms that the system produces good fidelity of the simulated mechanical properties and can be used to distinguish between different textiles.

6. Conclusions

A discussion on modeling strategies for deformable bodies, and in particular for fabric simulations, is presented. A complete haptic system for the user interaction with a virtual cloth is introduced, focusing on the simulation of clothes. In particular, a FEM approach provides realistic material mechanics within the simulation. The haptic interaction involves a kinesthetic feedback device and a tactile array. The simulation results prove the capability of real-time simulation for the interaction with virtual clothes. The presented parallel implementation may allow the simulation of large garment sizes without compromising the accuracy and stability of the interaction.

Author Contributions: Conceptualization, P.T. and E.R.; methodology, P.T. and E.R.; software, P.T. and E.R.; validation, P.T., S.D. and E.R.; writing, P.T., S.D. and E.R.; project administration, E.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Ethical review and approval were waived for this study, due to the research containing only nonidentifiable data about human beings.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sanchez, J.; Corrales, J.A.; Bouzgarrou, B.C.; Mezouar, Y. Robotic manipulation and sensing of deformable objects in domestic and industrial applications: A survey. *Int. J. Robot. Res.* **2018**, *37*, 688–716. [[CrossRef](#)]
2. Leizea, I.; Mendizabal, A.; Alvarez, H.; Aguinaga, I.; Borro, D.; Sanchez, E. Real-time visual tracking of deformable objects in robot-assisted surgery. *IEEE Comput. Graph. Appl.* **2015**, *37*, 56–68. [[CrossRef](#)] [[PubMed](#)]
3. Kapusta, A.; Erickson, Z.; Clever, H.M.; Yu, W.; Liu, C.K.; Turk, G.; Kemp, C.C. Personalized collaborative plans for robot-assisted dressing via optimization and simulation. *Auton. Robot.* **2019**, *43*, 2183–2207. [[CrossRef](#)]
4. Clegg, A.; Yu, W.; Tan, J.; Liu, C.K.; Turk, G. Learning to dress: Synthesizing human dressing motion via deep reinforcement learning. *ACM Trans. Graph. (TOG)* **2018**, *37*, 1–10. [[CrossRef](#)]
5. Li, Y.; Wu, J.; Tedrake, R.; Tenenbaum, J.B.; Torralba, A. Learning particle dynamics for manipulating rigid bodies, deformable objects, and fluids. *arXiv* **2018**, arXiv:1810.01566.
6. Yan, W.; Vangipuram, A.; Abbeel, P.; Pinto, L. Learning predictive representations for deformable objects using contrastive estimation. *arXiv* **2020**, arXiv:2003.05436.
7. Billard, A.; Kragic, D. Trends and challenges in robot manipulation. *Science* **2019**, *364*, 6446. [[CrossRef](#)]
8. Galassi, K.; Palli, G. Robotic wires manipulation for switchgear cabling and wiring harness manufacturing. In Proceedings of the 2021 4th IEEE International Conference on Industrial Cyber-Physical Systems (ICPS), Victoria, BC, Canada, 10–12 May 2021; pp. 531–536.
9. De Gregorio, D.; Zanella, R.; Palli, G.; Pirozzi, S.; Melchiorri, C. Integration of robotic vision and tactile sensing for wire-terminal insertion tasks. *IEEE Trans. Autom. Sci. Eng.* **2018**, *16*, 585–598. [[CrossRef](#)]
10. Behery, H.M. *Effect of Mechanical and Physical Properties on Fabric Hand*; Woodhead Pub Limited: Sawston, UK, 2005; Volume 42.
11. Bergamasco, M.; Salsedo, F.; Fontana, M.; Tarri, F.; Avizzano, C.A.; Frisoli, A.; Ruffaldi, E.; Marcheschi, S. High performance haptic device for force rendering in textile exploration. *Vis. Comput.* **2007**, *23*, 247–256. [[CrossRef](#)]
12. Kawabata, S.; Niwa, M. Fabric Performance in Clothing and Clothing Manufacture. *J. Text. Inst.* **1989**, *80*, 19–50. [[CrossRef](#)]
13. Terzopoulos, D.; Platt, J.; Barr, A.; Fleischer, K. Elastically Deformable Models. *SIGGRAPH Comput. Graph.* **1987**, *21*, 205–214. [[CrossRef](#)]
14. Eitzmuß, O.; Keckeisen, M.; Straßer, W. A fast finite element solution for cloth modelling. In Proceedings of the 11th Pacific Conference on Computer Graphics and Applications, Canmore, AB, Canada, 8–10 October 2003; pp. 244–251.
15. Aono, M. A Wrinkle Propagation Model for Cloth. In *CG International '90: Computer Graphics around the World*; Springer: Tokyo, Japan, 1990; pp. 95–115.
16. Bender, J.; Müller, M.; Macklin, M. *Position-Based Simulation Methods in Computer Graphics*; Eurographics (Tutorials): Geneva, Switzerland, 2015; p. 8.
17. Schulman, J.; Gupta, A.; Venkatesan, S.; Tayson-Frederick, M.; Abbeel, P. A case study of trajectory transfer through non-rigid registration for a simplified suturing scenario. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 4111–4117.
18. Kita, Y.; Kanehiro, F.; Ueshiba, T.; Kita, N. Clothes handling based on recognition by strategic observation. In Proceedings of the 2011 11th IEEE-RAS International Conference on Humanoid Robots, Bled, Slovenia, 26–28 October 2011; pp. 53–58.
19. Macklin, M.; Müller, M.; Chentanez, N.; Kim, T.Y. Unified particle physics for real-time applications. *ACM Trans. Graph. (TOG)* **2014**, *33*, 153. [[CrossRef](#)]
20. Boonvisut, P.; Çavuşoğlu, M.C. Estimation of soft tissue mechanical parameters from robotic manipulation data. *IEEE/ASME Trans. Mechatron.* **2012**, *18*, 1602–1611. [[CrossRef](#)] [[PubMed](#)]
21. Sibbing, D.; Kobbelt, L. Data Driven 3D Face Tracking Based on a Facial Deformation Model. In *Vision, Modeling & Visualization*; Bommes, D., Ritschel, T., Schultz, T., Eds.; The Eurographics Association: Geneva, Switzerland, 2015. [[CrossRef](#)]
22. Santesteban, I.; Garces, E.; Otaduy, M.A.; Casas, D. SoftSMPL: Data-driven Modeling of Nonlinear Soft-tissue Dynamics for Parametric Humans. *Comput. Graphics Forum* **2020**, *39*, 65–75. [[CrossRef](#)]
23. Faure, F.; Duriez, C.; Delingette, H.; Allard, J.; Gilles, B.; Marchesseau, S.; Talbot, H.; Courtecuisse, H.; Bousquet, G.; Peterlik, I.; et al. Sofa: A multi-model framework for interactive physical simulation. In *Soft Tissue Biomechanical Modeling for Computer Assisted Surgery*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 283–321.
24. Petit, A.; Ficuciello, F.; Fontanelli, G.A.; Villani, L.; Siciliano, B. Using physical modeling and RGB-D registration for contact force sensing on deformable objects. In Proceedings of the ICINCO 2017—14th International Conference on Informatics in Control, Automation and Robotics, Madrid, Spain, 26–28 July 2017; Springer: Berlin/Heidelberg, Germany, 2017; Volume 2, pp. 24–33.

25. Haouchine, N.; Kuang, W.; Cotin, S.; Yip, M. Vision-based force feedback estimation for robot-assisted surgery using instrument-constrained biomechanical three-dimensional maps. *IEEE Robot. Autom. Lett.* **2018**, *3*, 2160–2165. [[CrossRef](#)]
26. Yu, W.; Kapusta, A.; Tan, J.; Kemp, C.C.; Turk, G.; Liu, C.K. Haptic simulation for robot-assisted dressing. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 6044–6051.
27. Erickson, Z.; Clever, H.M.; Turk, G.; Liu, C.K.; Kemp, C.C. Deep haptic model predictive control for robot-assisted dressing. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 4437–4444.
28. Todorov, E.; Erez, T.; Tassa, Y. Mujoco: A physics engine for model-based control. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura, Portugal, 7–12 October 2012; pp. 5026–5033.
29. Petřík, V.; Kyrki, V. Feedback-based fabric strip folding. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 773–778.
30. Coumans, E.; Bai, Y. PyBullet, a Python Module for Physics Simulation for Games, Robotics and Machine Learning. 2016. Available online: <http://pybullet.org> (accessed on 3 December 2021).
31. Ascher, M.U.; Boxerman, E. On the modified conjugate gradient method in cloth simulation. *Vis. Comput.* **2003**, *19*, 526–531. [[CrossRef](#)]
32. Belytschko, T.; Glaum, L.W. Applications of higher order corotational stretch theories to nonlinear finite element analysis. *Comput. Struct.* **1979**, *10*, 175–182. [[CrossRef](#)]
33. Felippa, C.; Haugen, B. A unified formulation of small-strain corotational finite elements: I. Theory. *Comput. Methods Appl. Mech. Eng.* **2005**, *194*, 2285–2335. [[CrossRef](#)]
34. Rodríguez-Navarro, J.; Susín Sánchez, A. Non structured meshes for cloth GPU simulation using FEM. In Proceedings of the 3rd Workshop in Virtual Reality Interactions and Physical Simulation, Ascona, Switzerland, 25–30 June 2006; pp. 1–7.
35. Saad, Y. *Iterative Methods for Sparse Linear Systems*, 2nd ed.; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2003. [[CrossRef](#)]
36. Fontana, M.; Fabio, S.; Marcheschi, S.; Bergamasco, M. Haptic Hand Exoskeleton for Precision Grasp Simulation. *J. Mech. Robot.* **2013**, *5*, 041014. [[CrossRef](#)]
37. Salsedo, F.; Marcheschi, S.; Fontana, M.; Bergamasco, M. Tactile transducer based on electromechanical solenoids. In Proceedings of the 2011 IEEE World Haptics Conference, Istanbul, Turkey, 21–24 June 2011; pp. 581–586. [[CrossRef](#)]
38. Hill, N.J. *Testing Hypotheses about Psychometric Functions*; University of Oxford: Oxford, UK, 2001; Volume 600.
39. Watson, A.B.; Pelli, D.G. QUEST: A Bayesian adaptive psychometric method. *Percept. Psychophys.* **1983**, *33*, 113–120. [[CrossRef](#)]