ELSEVIER

Computer-Aided Design 37 (2005) 593–608

COMPUTER-AIDED DESIGN

www.elsevier.com/locate/cad

From early virtual garment simulation to interactive fashion design

Pascal Volino*, Frederic Cordier, Nadia Magnenat-Thalmann

MIRALab, C.U.I., University of Geneva, CH-1211 Geneva, Switzerland

Accepted 15 September 2004

Abstract

Virtual garment design and simulation involves a combination of a large range of techniques, involving mechanical simulation, collision detection, and user interface techniques for creating garments. Here, we perform an extensive review of the evolution of these techniques made in the last decade to bring virtual garments to the reach of computer applications not only aimed at graphics, but also at CAD techniques for the garment industry.

As a result of the advances in the developments of virtual garment simulation technologies, we then detail a framework which fits the needs of the garment industry of virtual garment design and prototyping, concentrating on interactive design, simulation and visualization features. The framework integrates innovative tools aimed towards efficiency and quality in the process of garment design and prototyping, taking advantage of state-of-the-art algorithms from the field of mechanical simulation, animation and rendering.

Keywords: Cloth simulation; Virtual garments; Fashion design; Pattern prototyping

1. Introduction

The challenges of virtual garment simulation are numerous, and have attracted research efforts for more than a decade. First dedicated to the realistic simulation of the mechanical behavior of cloth, it soon evolved towards simulation of virtual garments on synthetic characters. While computer graphics gets the most obvious benefits from garment simulation on animated virtual characters, virtual prototyping of garment models is another major application field for the garment industry.

Virtual garment simulation is the result of a large combination of techniques that have also dramatically evolved during the last decade. Unlike the mechanical models used for existing mechanical engineering for simulating deformable structures, a lot of new challenges arise from the highly versatile nature of cloth. The central pillar of garment simulation obviously remains the development of efficient mechanical simulation models, which can accurately reproduce with the specific mechanical properties of cloth. However, cloth is by nature highly deformable, and specific simulation problems arise from this fact. First, the mechanical representation should be accurate enough to deal with the nonlinearities and large deformations occurring at any place in the cloth, such as folds and wrinkles. Moreover, the garment cloth interacts strongly with the body that wears it, as well as with the other garments of the apparel. This requires advanced methods for efficiently detecting the geometrical contacts constraining the behavior of the cloth, and to integrate them in the mechanical model (collision detection and response). All these methods require advanced and complex computational methods where most important key issues remain computation speed and efficiency. For real-time applications however, only specific approximation and simplification methods allow the computation of garment animation, giving up some of the mechanical accuracy of the result in a result rather focused on visual realism. Section 2 reviews the developments of these techniques, from early beginnings (Section 2.1) to the state-of-the-art methods in cloth simulation and collision processing (Section 2.2), leading to the existing current developments in garments.

Cloth simulation has, however, matured enough to introduce its potentials to the garment industry. The main

* Corresponding author.
E-mail address: volino@miralab.unigue.ch (P. Volino).
URL: http://www.miralab.ch

0010-4485/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cad.2004.09.003

needs to be fulfilled are mostly related to virtual garment prototyping, as well as visualization applications related to and virtual fashion and prototyping. Besides the higher level of accuracy needed to simulate the complex features of fashion models, new cloth design systems are now intended to help the garment creator shorten the design process of a garment, using innovative techniques and tools described in Section 3. The creation of a powerful interactive garment design system (Section 3.1) relates to core simulation technologies (Sections 3.2 and 3.3) as well as new dynamic manipulation techniques for the cloth mesh (Section 3.4). Specific methods are aimed at producing real-time garment animations (Section 3.5) to be used in numerous visualization applications, such as virtual try-on (Section 3.6). More than bringing new efficient tools to the garment industry (Section 4), these advances also open the door to new ways of thinking fashion design and garment creation.

2. State-of-the-art in virtual garments

Garment simulation, which started in the late eighties with very simple models such as Weil's approach [54], has taken much benefit from the increasing performance of computer hardware and tools as well as the development of specific simulation technologies which have nowadays lead to impressive applications not only in the field of simulation of virtual worlds, but also as design tools for the garment and fashion industry.

2.1. The beginnings of garment simulation

In the field of computer graphics, the first applications for mechanical cloth simulation appeared in 1987 with the work of Terzopoulos et al. [43,44] in the form of a simulation system relying on the Lagrange equations of motion and elastic surface energy. Solutions were obtained through finite difference schemes on regular grids. This allowed simple scenes involving cloth to be simulated, such as the accurate simulation of a flag or the draping of a rectangular cloth. However, the first applications that really simulated garments started in 1990 (Fig. 1) with the considerations of many other technologies complementing cloth simulation [7,33], such as body modeling and animation, and collision detection and response [55]. These applications innovated by providing the first virtual system allowing virtual garment patterns to be sewed together around a character.

2.2. Recent advances

Since then, most developments were aimed at optimizing the accuracy and efficiency of the methods for simulating cloth accurately and efficiently, along the developments of actual applications and commercial products.

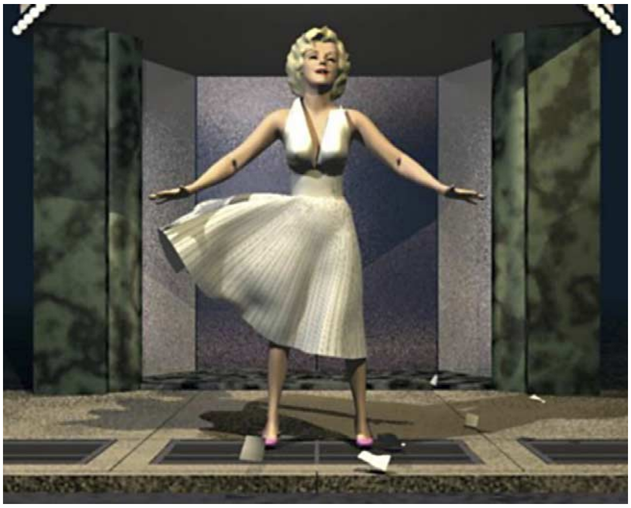


Fig. 1. 'FlashBack': early virtual garments used context-dependent simulation of simplified cloth models.

2.2.1. Mechanical models

The accurate reproduction of the mechanical behavior of cloth has always been a key issue for garment simulation. The mechanical behavior of cloth is usually measured using standardized protocols, such as the Kawabata Evaluation System (KES), or the simpler FAST method, which are based on the experimental measurement of strain-stress curves for elongation, shearing and bending on normalized samples of fabric. Different representations of the cloth surface mechanics then allow the virtual reproduction of the behavior of cloth.

Well known in mechanical engineering, the Finite Element method considers the cloth surface as being discretized in interpolation patches for a given order (bilinear, trilinear, quadrilinear), and an associated set of parameters (degrees of freedom) that give the actual shape to the interpolation surface over the element. From the mechanical properties of the material, the mechanical energy is computed from the deformation of the surface for given values of the interpolation parameters. An equation system based on the energy variation is then constructed with these degrees of freedom. Surface continuity between adjacent elements impose additional constraint relationships. A large sparse linear system is built by assembling successively the contributions of all the elements of the surface, and then solved using optimized iterative techniques, such as the conjugate gradient method.

Finite elements have only had a marginal role in cloth simulation. The main attempts are described in [10,19,25]. Most implementations focus on the accurate reproduction of mechanical properties of fabrics, but restrict the application field to the simulation of simple garment samples under elementary mechanical contexts, mostly because of the huge computational requirements of these models. Furthermore, accurate modeling of highly variable constraints (large nonlinear deformations, highly variable collisions) is difficult to integrate into the formalism of finite elements, and this sharply reduces the ability of the model to cope with the very complicated geometrical contexts which can arise in real-world garment simulation on virtual characters.

An easier and more pragmatic way to perform cloth simulation is the use of particle systems. Particle systems consider the cloth to be represented only by the set of vertices that constitute the polygonal mesh of the surface. These particles are moved through the action of forces that represent the mechanical behavior of the cloth, which are computed from the geometric relationships between the particles that measure the deformation of the virtual cloth. Among the different variations of particle systems, the spring-mass scheme is the simplest and most widely used (Fig. 2). It considers the distance between neighboring particle pairs as the only deformation measurement and interaction source representing the internal elasticity of the cloth.

Particle systems are among the simplest and most efficient ways to define rough models that compute highly deformable mechanical systems such as cloth with computation times small enough to integrate them into systems for simulating complete garments on virtual bodies. Among the main contributions on particle system models, early works considered simple viscoelastic models on regular grids with applications for draping problems with simple numerical integration schemes [41]. Accurate models started with Breen et al. [5] on modeling the microstructure of cloth using parameters derived from KES behavior curves and integration based on energy minimization. However, such accurate models required a lot of computation for solving problems that were restricted to draping. On the other hand, more recent models trade accuracy for speed, such as the grid model detailed by Provot et al. [40] which additionally includes geometric constraints for limiting large deformation of cloth. Additional contributions from Eberhardt et al. [16] with the simulation of KES parameters and comparison of the efficiency of several integration methods. Advanced surface representations were used in [13], where the simulation model and collision detection takes advantage of the hierarchical structure of subdivision surfaces. Modeling animated garments on virtual characters is the specific aim of the work described by Volino et al. [48,49], which investigate improved spring-mass representations for better accuracy of surface elasticity modeling on irregular meshes.

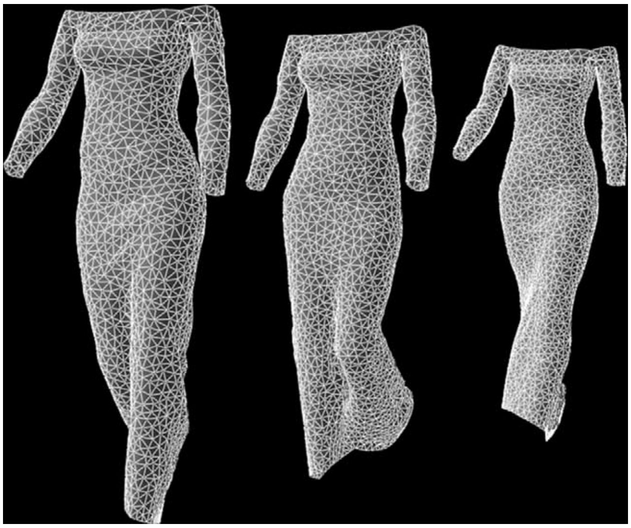


Fig. 2. Particle systems model cloth surfaces as point masses interacting with their neighbors using forces computed from their relative positions.

While various models can be used to compute the force applied on each particle given their position and speed, these forces have then to be integrated along time to obtain the position and speed of the particle for the following timesteps using methods related to the integration of ordinary differential equation systems. Most recent, however, focus on improvements of the numerical integration methods in order to improve efficiency of the simulation.

Explicit integration methods are the simplest methods available for solving first-order ordinary differential systems. They consider the prediction of the future system state directly from the value of the derivatives. The best known techniques are the Runge-Kutta methods. Among them, the fast but unstable and inaccurate first-order Euler method, used in many early implementations, considers the future state as a direct extrapolation from the current state and the derivative. Higher order and more accurate methods also exist, such as the second-order Midpoint method, used for instance in early models by Volino et al. [48], and the very accurate fourth-order Runge-Kutta method, used for instance by Eberhardt et al. [16].

Beside considerations for accuracy, stability and robustness are other key factors to consider. For most situations encountered in cloth simulation, the numerical stiffness of the equations (stiff elastic forces, small surface elements) require the simulation timesteps to be small enough to ensure the stability of the system, and this limits the computation speed much more than accuracy considerations. Adequate timestep control is therefore essential for an optimal simulation. A common solution is to use the fifth-order Runge-Kutta algorithm detailed in [39] which embeds integration error evaluation used for tuning the timestep adaptively [49].

In order to circumvent the problem of instability, implicit numerical methods are being used. For cloth simulation, this was first outlined by Baraff et al. [2]. The most basic implementation of implicit method is the Euler step, which considers finding the future state for which 'backward' Euler computation would return the initial state. It performs the computation not using the derivative at the current timestep, but using the predicted derivative for the next timestep. Besides the inverse Euler method, other, more accurate higher-order implicit methods exist, such as the inverse Midpoint method, which remains quite simple but exhibits some instability problems. A simple solution is to interpolate between the equations of the Euler and Midpoint methods, as proposed by Volino et al. [51,52,53]. Higher-order methods, such as the Rosenbrook method, however, do not exhibit convincing efficiencies in the field of cloth simulation. Multistep methods, which perform a single-step iteration using a linear combination of several previous states, are other good candidates for a good accuracy-stability compromise. Among them, the second-order Backward Differential Formula (BDF-2) has shown some interesting performances, as used by Eberhardt, Hauth et al. [17,22] and Choi et al. [9].

Whatever variation chosen, the major difficulty in using implicit integration methods is that they involve the resolution of a large and sparse linear equation system for each iteration, constructed from the Jacobian matrix of the particle forces against their position and speed. A commonly used simplification involves linearization of the mechanical model so as to obtain a linear approximation of the matrix that does not evolve along time, and on which initial construction and preprocessing allows efficient resolution method to be used, as for example like Kang et al. [29,30,31], or even the matrix inverse to be precomputed as done by Desbrun et al. [15]. A further simplification is to suppress completely the need of computing the matrix using an adapted approximation embedded directly in an explicit iteration. A big drawback of all these methods results from the approximation of the matrix that cannot take into account the nonlinearities of the model (mostly those resulting from the change of orientation of the surface elements during the simulation). While this is acceptable for draping applications, animations behave usually poorly because of excessive numerical damping, which also increases as the timestep becomes large.

The best numerical method for actually resolving the linear system seems to be the Conjugate Gradient method, as suggested by Baraff et al. [2], with various variations and preconditioning schemes depending on how the mechanical model is formulated and geometrical constraints of the cloth integrated.

Most models using implicit integration schemes restrict themselves to using spring-mass systems, as their simple formulation eases the process of defining the linear system to be resolved. However, implicit integration methods can also be used for integrating accurate surface-based particle systems as the one described above, from derivation of the particle force expressions relatively to the particle positions and speeds. This in quite simply integrated into the implicit formulations described by Volino et al. [51,52], and extended toward other advanced methods as by Hauth et al. [22]. These formulations actually blur the line between particle systems and finite element methods, as the described particle system is indeed a first-order finite element method where the implicit resolution scheme corresponds to the energy minimization scheme of finite elements and the build of the linear system matrix to the assembly process of elements into the global system to be resolved. This is a key idea to design a new system which combines the accuracy of finite elements with the efficiency of the techniques used for particle systems, as described in Section 3.

2.2.2. Real-time garment animation

Real-time garment animation poses the challenging problem of how to perform very fast methods for the mechanical computation and collision detection. Accuracy has to be given up in favor of quicker methods that take advantage of geometrical approximations and contextual simplifications.

Implicit integration [2,9] and speed-optimized derivatives [15,36] allow fast simulation of mechanical properties of cloth. However, the computation speed still remains slow for complex garments, and these methods are still limited by the maximum number of polygons they can animate in real-time.

In the specific of area optimizations for mechanical behavior, James et al. [28] have worked on real time simulation. Their paper describes the boundary integral equation formulation of static linear elasticity as well as the related Boundary Element Method discretization technique. Their model is not dynamic, but rather a collection of static postures, limiting its potential applications. Debunne et al. [14] have also recently introduced a technique for animating soft bodies in real time. However, their method works on volumetric meshes and is therefore not applicable to thin objects such as cloth.

Collision detection is another bottleneck in the speed of cloth simulation. Besides the traditional methods, specific optimizations intend to address the problem of real-time simulation. For instance, Vassilev et al. [46] propose to use z-buffer for collision detection in order to generate depth and normal maps. The computation time of their collision detection does not depend on the complexity of the body. However, the maps need to be pre-computed before simulation, also restricting the real-time application.

Some other researchers have used geometrical approaches [1,23,37,54]. Geometrical models do not consider the physical properties of the cloth, therefore providing techniques that produce fast results. However, these techniques are not able to reproduce the dynamics of clothes. Moreover, geometrical techniques require a considerable degree of user intervention. They can be regarded as a form of advanced drawing tools.

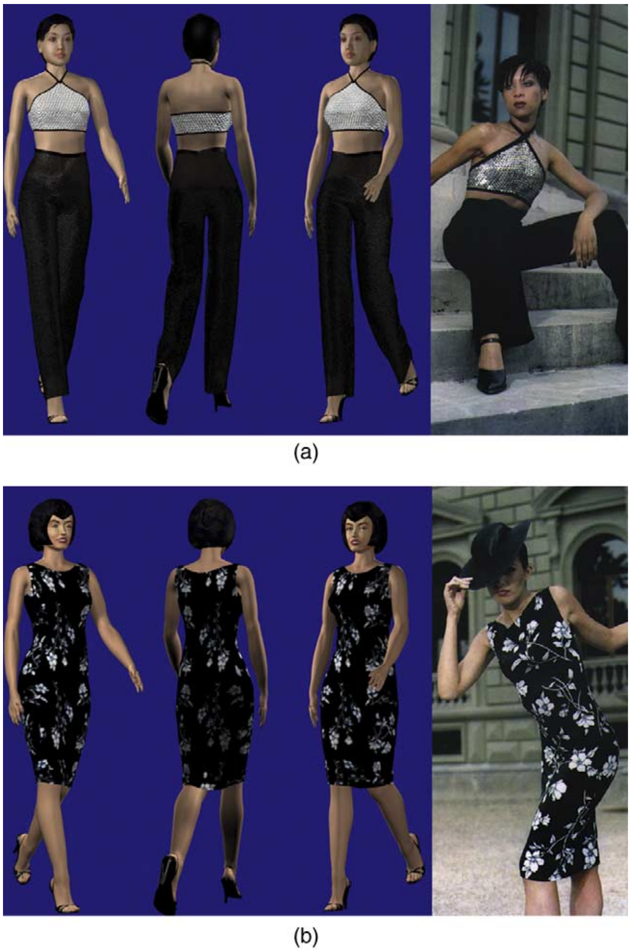
Another approach presented by Grzeszczuk et al. [27] uses a neural network to animate dynamic objects. They replaced physics-based models by a large neural network that automatically learns to simulate similar motions by observing the models in action. The method works in real-time, but it has not been proven that this method can be used for complex simulation, such as cloth.

Hybrid approaches try to combine geometrical and physical deformations [50]. Kang et al. [29,30] improved the visual quality of the garments with a small number of polygons by tessellating the triangles. Their tessellation algorithm, which is based on cubic spline curve, is able to simulate the wrinkles. Oshita et al. [38] use a similar approach. Both these methods are mainly applicable to flat surfaces where physical simulation can be done with a very small number of polygons. However, highly curved surfaces, such as sleeves, need to be simulated with a higher number of polygons.

2.2.3. Garment design and simulation

Since the first developments to produce simulated garments on virtual characters [7,33], cloth simulation and garment animation has made its way not only in computer research (Fig. 3) [49], but also into commercial products aimed both for 3D computer design and the garment industry.

Two kinds of products ace currently available: those oriented for general cloth simulation and animation, and those specialized for draping and fitting garment models on virtual mannequins. The first category offer tools for simulating any kind of deformable surface mechanically. They usually offer a simple mechanical model containing only the basic mechanical parameters of cloth (stiffness, viscosity, bending, gravity) modeled as a spring-mass particle system and simulated using state-of-the-art integration techniques. They allow the computation of realistic cloth animation, but do not provide any tool for designing garments. The also offer general collision detection schemes for interaction with any other objects. These tools are usually integrated as plugins into 3D design and animation frameworks. Among the main products, there is MayaCloth integrated into Maya [57], Reactor [58], Stitch [59], SimCloth [60] and ClothReyes [61] for 3DStudio Max, Dynamics integrated into Cinema 4D [62].


(a)

(b)

Fig. 3. Virtual fashion: real models and their virtual counterparts.

The second category focuses on garment draping on virtual mannequins for visualization (virtual fashion, web applications) and prototyping purposes (garment design applications). The CAD applications specialize the simulation on pattern assembly and garment draping using accurate mechanical models of fabrics [8], while the visualization application take advantage of geometric techniques for generating quickly realistic dressed mannequins out of design choices. Both use pattern models imported from professional pattern design tools (Gerber, Lectra, Investronica). These tools also usually provide a standalone environment for setting up the simulation and visualizing the results. Among them, there is Toyobo DressingSim [63], Optitex [64], Browzwear V-Stitcher [65], or web applications such as FitMe.com [66] and MIRALab’s Virtual Try-On [67] (Fig. 4).

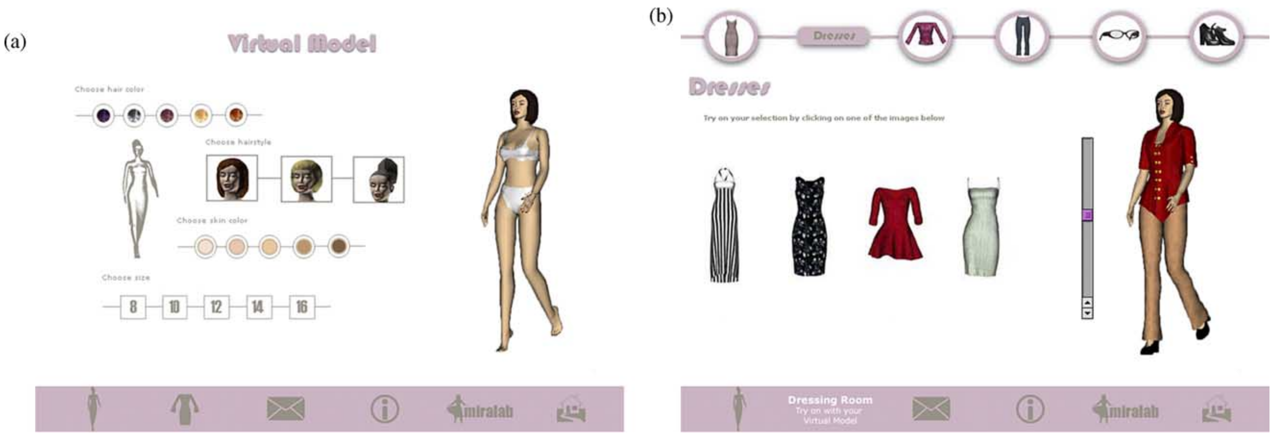
3. A new interactive design system

The garment industry is still attached to the traditional way of designing garments, which is based on designing patterns (shapes of fabric) which are then seamed together on a mannequin. While CAD tools (Lectra, Gerber, Investronica) are now highly involved in the 2D design process of the patterns, the actual prototyping is still performed using real cloth pieces and mannequins. Some attempts were made to bring this validation phase in the virtual world as well, using cloth simulation techniques. This approach is still quite inefficient because of the lack of integration between the 2D pattern design process and the 3D simulation.

Interactivity between design and simulation is the key idea for solving this bottleneck. While current systems require the re-assembly and draping of the garment over the body for any design change on the garment patterns, we propose a new approach offer a smart integration of the 2D pattern shape editor along with the 3D garment shape view to assess interactively the effect of any pattern shape edit or posture and measurement change on the mannequin (Fig. 5).

3.1. Interactive design of patterns

Using patterns imported from industry CAD tools, the proposed framework obviously addresses the issues of placing the patterns around a mannequin, seaming



Try on your selection by clicking on one of the images below

Fig. 4. MIRALab's virtual Try-On on the Web.

the patterns together, and performing the draping using mechanical simulation. However, the main interest of the framework is rather related to the possibility of refining and altering the design of the patterns in real-time, with interactive preview of the effects on the garment draped on the mannequin. These edit possibilities include resizing, reshaping, adding darts, holes, seams, pleats, attachments (zips, buttons), changing the fabric material (mechanical properties and texture). The mannequin can be interactively changed as well, both for measurements and for postures and motion sequences.

The system immediately react to these changes by updating the draped garment interactively. The new garment shape is automatically recomputed using accurate mechanical properties in order to exhibit the features that have been altered by the edits. Geometrical accuracy of the simulation may be changed at any time to provide the optimal compromise between accuracy and interactivity at any stage of the design process. Advanced visualization tools can also display at any time the local mechanical properties of cloth (mechanical deformations, contact forces) for a good evaluation of the fitting and the comfortability, for any chosen size and posture of the mannequin which can also be customized (measurements, posture and appearance) and animated interactively.

Using an innovative design tool for garment design and simulation, we intend to bring new advances not only in the field of garment simulation for computer graphics characters, but also as a valuable design and prototyping tool for the actual needs of the garment industry.

Our system addresses the following issues:

- Provide an interactive pattern design environment that combines in real-time the 2D view of the shape of the pattern on the fabric with the 3D view of the garment draped on a customizable body, and allows interaction to be performed either in 2D or in 3D from any view with interactive visualization of the result.

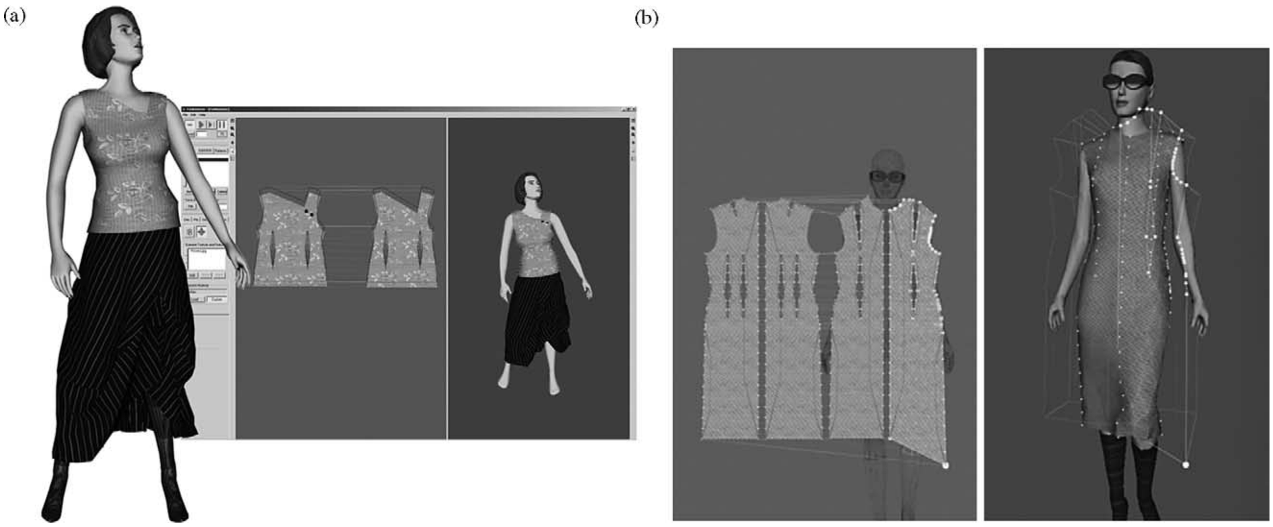


Fig. 5. User interface for interactive garment design (left): edition of the patterns can be performed any time in 2D or 3D with immediate preview of the results (right).

- Offer a large range of simulation techniques fulfilling optimally all the needs of the garment design process, such as fast garment assembly, draping on posture, interactive editing, accurate simulation and animation.
- Provide high-quality garment simulation, not only by simulating accurately the elastic behavior of deformable cloth through a model taking into account elastic strain-stress curves, but also by providing accurate simulation of the dissipative behavior of cloth for producing high-quality animations.
- Offer fast real-time animated preview of the designed garment on a moving virtual character through a dedicated visualization module. The main purpose of the real-time simulation is to provide to the designer a pre-view of the garments on virtual humans. The rendering of a simulation sequence of a dressed virtual human with the accurate approach requires several minutes of computation while the real-time method makes the rendering of the same clothes almost instantaneous. In addition, this module offers the possibility to export the garments and the body for a real-time visualization on the Web. The system not only provides a tool for the design of the garments, but also it makes possible the visualization of these garments on a Web server for on-line shopping, after they have been created.
- Offer sufficient versatility for producing complex garments made of numerous patterns of intricate shape and seams.

These features require the use of improved state-of-the-art techniques for cloth simulation (part 3.1) and collision handling (part 3.2), associated to new techniques for managing dynamically the generation and update of the cloth mesh (part 3.3) for bringing interactivity in the garment design under the tips of the garment designer.

3.2. The mechanical model

3.2.1. Accurate model

This mechanical model is based on an accurate representation of the surface deformation of a triangular surface element equivalent to a first-order finite element description. Weft, warp and shear deformation and deformation speed are accurately measured on each element and translated into forces using the strain-stress curves defined by the mechanical properties. These forces are then propagated onto the mesh vertices, and any kind of integration scheme traditionally used for particle systems may then be used. Bending viscoelasticity is modeled as forces between adjacent mesh elements derived from the bending measured between them.

At the crossroad of particle systems and first-order finite elements, this model offers a very good compromise between speed and accuracy. Weft, warp, shear and Poisson viscoelasticity behavior is accurately represented by this scheme and, unlike representations based on spring-mass models, the accuracy is quite well maintained with large deformations.

Another benefit of this model is that it does not rely on regular particle arrays, but satisfies itself from any triangle mesh for describing the cloth surface. This allows complete freedom in the design of the pattern shape, as well as optimal surface representations using adapted or dynamic local mesh refinements.

The numerical integration of the particle forces is carried out using advanced implicit schemes. While the Implicit Euler scheme remains a good choice for ensuring fast and robust computation, our system also implements Inverse Midpoint and BDF-2 methods for providing more accuracy. The system actually allows any the choice of any intermediate formula to be selected between these three schemes, for providing an optimal selection between accuracy, robustness and computation speed.

While draping applications satisfy themselves with any available implicit schemes, the accurate simulation of cloth motion requires a higher accuracy in the model for taking into account the nonlinearities of the numerical system (not only for taking into account the nonlinearities of the strain-stress curves, but also those resulting from the change of orientation of the cloth elements). These nonlinearities alter the force derivative matrix used in the linear system resolution of any implicit scheme. Linearizing the equations for obtaining a constant matrix as performed in most implementations for simplifying the resolution is done at the expense of accuracy which usually translates into excessive numerical damping of the mechanical system, thus becoming unsuitable for animation. Our system implements a modified version of the Conjugate Gradient algorithm for the linear system resolution, which embeds dynamic evaluation of the force derivatives corresponding to the current state of the mechanical system, without explicit construction of the matrix [51,52]. This greatly enhances the quality of cloth animation. The computation time and accuracy is further enhanced by a suitable preconditioning of the equations of the linear system which ensures that the residual error produced my the iterative resolution process in any case comply with mechanical momentum conservation laws. This allows the animation to remain realistic even with if very few numbers of iterations are used in the resolution, which is particularly important for interactive applications.

If even more accuracy is needed for representing cloth animation, the fifth-order Runge-Kutta scheme with error evaluation for adaptive timestep is still available in our system [49]. At the price of a computation which highly increases with the stiffness of the forces and the smallness of the mesh elements, it ensures a guaranteed accuracy in both the position and the speed of the cloth, and thus still a good choice for testing the effect of mechanical parameters on the motion of the cloth.

For time-critical interactive simulations where accuracy is not needed (for instance, the pattern assembly process), a fast simplified mechanical simulation scheme is also available. It is based on a spring-mass mechanical representation following the topology of the mesh, which uses springs of precomputed constant linear viscoelastic stiffness that roughly represents the mechanical parameters of the cloth. The system can transparently switch from one model to another depending on the context.

3.3. Collision detection and response

While much emphasis is put an the efficiency and accuracy of the mechanical simulation, processing the collisions which occur between the cloth and the body, as well as between two different parts of the cloth, remains a compulsory issue to address efficiently. This problem relates both to detection (how to find out efficiently the geometrical contacts between complex moving surfaces) and response (how these geometric contacts alter the mechanical behavior of the cloth realistically, and how this integrates efficiently to the simulation scheme).

3.3.1. Collision detection

The main issue is to master the complexity of the problem due to the discretization. The cloth surface, as well as the body surface, are represented by polygonal meshes that can have several thousands polygons each. Testing each couple of polygons for potential collisions is an unrealistic task. Many optimized algorithms have been developed for that purpose, either based on space subdivision or hierarchisation (voxelisation, octree), object hierarchisation (volume-box hierarchies), spatial projection and ordering.

In the case of cloth simulation, bounding-volume hierarchies are adequate algorithms for collision detection, since the topology of the animated meshes mostly remains constant, and therefore a constant precomputed hierarchy could be used. For this purpose, we have used the Discrete Orientation Polyhedra (DOPs) described by Klosowsky et al. [32] which provide a better alternative to axis-aligned bounding boxes, as they bound more tightly the flat arbitrary-oriented surface regions of cloth. Many optimizations of the hierarchical bounding volume scheme are available, as for example in [4,18,26,24,34,35,56].

Designing a very general cloth simulation framework requires, however, the detection of self-collisions in the cloth surface, which may bend, wrinkle and crumple in very complex patterns. This kind of detection is, however, very inefficient, as every adjacent elements of the mesh are seen as 'colliding' by the detection algorithm. In [47] is detailed an adapted version of a hierarchical algorithm which deals with this issue. It is based on the consideration that self-collisions only occur within surfaces that are bent enough to produce a 'loop'. Hence, self-collisions should only be detected within surface regions that are curved enough to exhibit them.

Hence, in addition to bounding volumes, curvature information has been added in the elements of the hierarchy for performing this test. Furthermore, the algorithm takes advantage of element adjacencies by replacing the bounding-box test by a curvature test it that case. The implementation of this scheme provides a very efficient framework where collision detection is not the bottleneck of simulation performance anymore, and where self-collision detection has only a very minor performance impact.

3.3.2. Collision response

As soon as the distance separating two objects is small enough to consider that the they are interacting, a 'feedback' should be performed on their behavior. Collision response aims to reproduce this interaction in an accurate way, primarily to avoid unrealistic interpenetration of the objects, and secondary to simulate realistic bouncing and friction effects.

Traditional collision response techniques use potential repulsion fields to model the reaction forces, which are intense and highly discontinuous. Though being a 'physical' response easily integrated in the mechanical model, this solution is, however, impractical, because of the high and nonlinear penalty forces that are hard to simulate numerically, and which cannot render precisely the discontinuous reaction of a solid contact.

In an approach introduced in [48], collisions were handled geometrical constraints, using cinematic correction on the constrained elements: positions and speeds were corrected according to the mechanical conservation laws to fit the constraints precisely. This approach allowed to skip the potential walls used to enforce the constraints. More recently, Eberhardt et al. [16] used a similar way to handle friction effects. This approach has been extended in [49] by a more general framework performing corrections not only on positions, but also on speeds and mostly on accelerations according to mechanical momentum conservation laws, ensuring reduction of the mechanical energy and stability. Such approach, which does not rely heavily on position correction, can be integrated in the mechanical model in a better way, and ensures good response stability. The equivalence between collision forces and acceleration corrections resulting from Newton's law also allows the efficient integration of collision response using implicit integration schemes [51,52]. Other variations are aimed toward robustness [6] and solving collision inconsistencies [3].

Our new system combines all these features for handling collisions efficiently, with enough versatility for allowing the simulation of complex multilayer garments (Fig. 6).

3.4. Simulation of dynamic meshes

An interactive design framework obviously needs efficient mechanical simulation and collision techniques

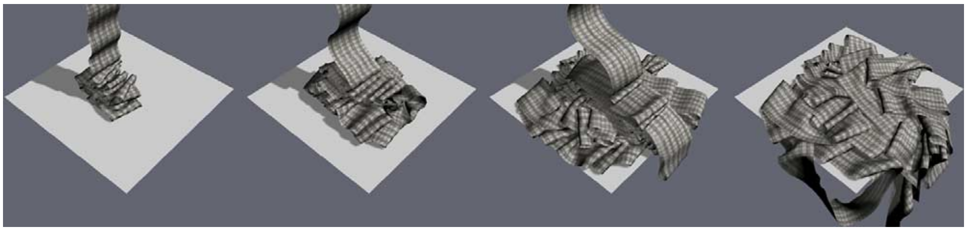


Fig. 6. Efficient and robust collision processing is important for simulating complex cloth.

for animating mechanical objects quickly enough for an immediate feedback of the design actions. However, user interaction will only be useful if an adequate management of the pattern design give immediate feedback of the changes to the mechanical simulation system.

The key idea of the system is to take advantage of a dynamically constructed mesh that automatically adapts to the shape and feature changes of the garment patterns during their design. Hence, any action altering the 2D shape of the patterns would have a direct mechanical effect on the 3D shape of the garment surface.

3.4.1. Altering the shape of garment patterns

Most pattern adjustments deal with resizing and enhancing the shape of the patterns for better fit to the body. It is quite often useless to perform a complete reconstruction of the garment mesh is these cases. An alteration of the fabric coordinates of the mesh vertices are sufficient for matching the new pattern shape. Such scheme is possible as our simulation system does not rely on regular grids for performing the simulation.

Our system takes advantage of a mapping update algorithm which updates the 2D mesh position on the fabric according to the 2D shape modification of the pattern. For this purpose, we have developed an extended mesh representation where each vertex ‘remembers’ which pattern vertices has originated it. Hence, the fabric coordinates of a mesh vertex is actually obtained as a weighted sum of fabric coordinates of vertices defining the pattern shape. When one of these vertices is displaced, all relevant mesh vertices are updated according to the coefficients of the weighted sum which are stored in the data structure (Fig. 7).

A standard Delaunay triangulation process is used for constructing the mesh out of the pattern contour. First, a constrained Delaunay triangulation algorithm triangulates the pattern shape using the initial vertices of the contour only for building an initial mesh. Then, an incremental Delaunay subdivision algorithm splits the mesh triangles until an adequately refined mesh is obtained. Weighted sum coefficients are trivially computed during this process for each new vertex out of the coefficients of the vertices of the split triangle and the barycentric coordinates of the vertex in the triangle. As the final coefficients of the mesh actually depend on the order the triangles are split, it is important to perform the splitting process as uniformly as possible, for instance by maintaining a size-ordered list of triangles for splitting them the largest first.

The biggest interest of this scheme is that any change of the pattern shape in 2D will not affect the cloth surface shape in 3D at all. This allows any pattern shape or size adjustment to be performed by the designer to be carried out whatever the simulation state of the garment, relying only on mechanics to finally bring the garment to its new shape (Fig. 8A–C). This process can also be used for updating in real-time a worn garment as the designed changes the measurements of the patterns, for instance for finding which garment size a the mannequin the best.

3.4.2. Repositioning new garment surfaces

When large modifications and topological changes are made on the patterns (new features, adding darts and holes) the mesh mapping update process described above cannot be applied, and a new mess taking into account the new pattern features has to be regenerated. For avoiding the process of rebuilding and redraping the garment on the body, an automatic mesh replacement process will use the position of the old mesh to replace the new mesh in a similar way (Fig. 8C–F).

The repositioning algorithm updates the 3D position of a vertex of the new mesh using the 3D position of the polygon of the old mesh which contains the 2D fabric position of the vertex. This polygon is found efficiently in the mesh by taking advantage of the hierarchical structure of the mesh constructed for performing collision detection during simulation. If no polygon of the old mesh contains the vertex, its position is derived from an average extrapolation of the nearest triangles of the old mesh.

The repositioning algorithm performs best when the shape of the new pattern is similar to the shape of the old one. If their shape is significantly different, we can

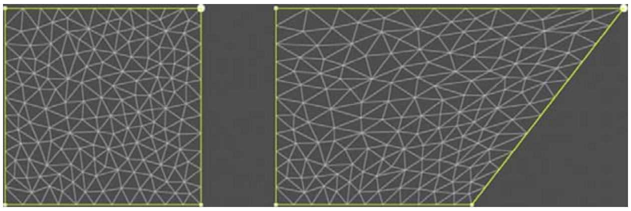


Fig. 7. Interactive edition of patterns: each mesh vertex of the pattern surface is displaced by a weighted amount of the displacement of the pattern contour point.

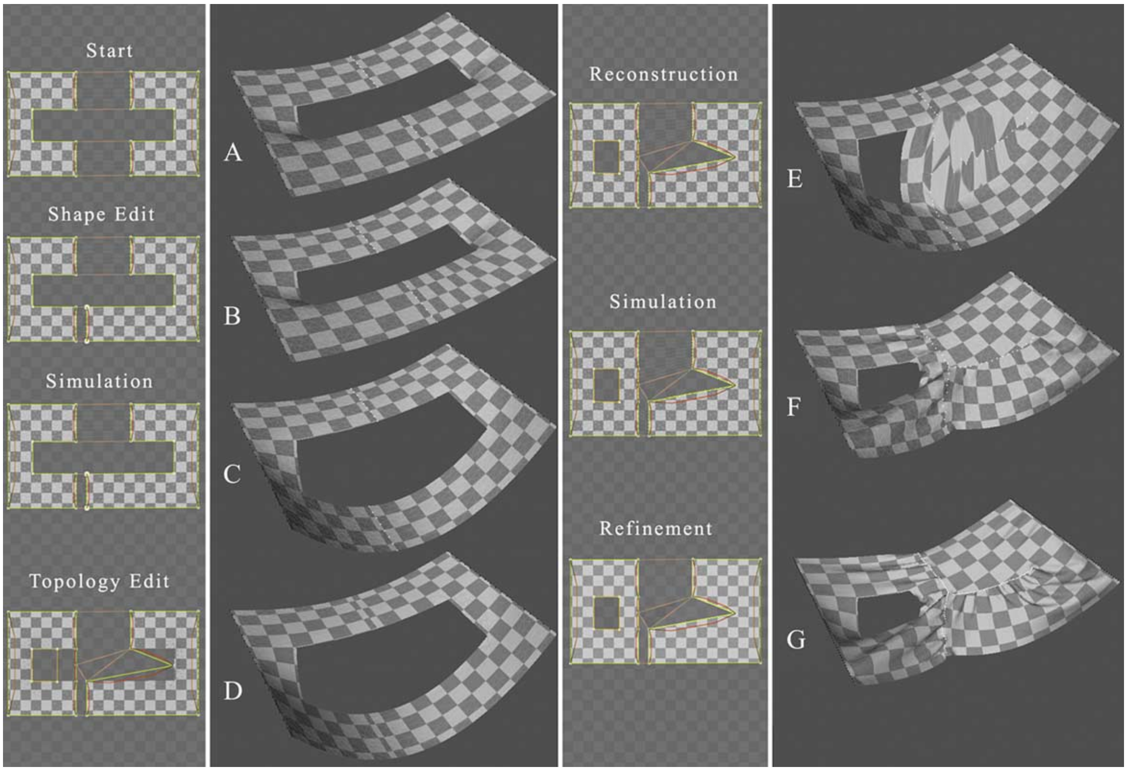


Fig. 8. Interactive garment design: from an initial draped pattern (A), editing the 2D contour of a pattern while keeping its 3D shape constant (B) allows mechanical simulation to smoothly drape the pattern to its new 3D shape (C). If pattern editing involves topology or seaming changes (D), the pattern surface is automatically reconstructed in the same position (E) using interpolation if necessary, and simulation performs the draping (F). The surface may be refined at any time (G) for more accuracy.

efficiently use the mapping update algorithm described above to roughly approximate the old pattern mesh to the new 2D pattern shape before reconstructing and repositioning the new mesh.

Our pattern design system performs automatically and transparently the mapping update and the repositioning of the garment surface according to the design operations applied on the patterns, so as to exhibit automatically the result of the design. The same process is also used for switching dynamically from one mesh resolution to another, depending the amount of interactivity or accuracy needed for a particular design step (Fig. 8F and G).

3.4.3. Techniques for fitting garments on a body

Considering body sizes and postures is important for designing garments that fit the body well. Rather than recreating a new 3D garment for each body to be dressed, we can take advantage of the versatility of the system to provide automatic garment adaptation to any body size and posture change.

Cloth deformation data can easily be extracted from the mechanical model used in the system. This can be displayed directly on the cloth surface in the form of strain or stress diagrams illustrating the deformation of the tension accurately computed on each region of the cloth. The garment designer can by this way point out where the garment is excessively in tension, correct the pattern shapes accordingly and observe the effect of the correction in real-time. Pressure of the garment on the body skin, illustrating the comfortability of the garment, can be evaluated in the same way.

Our system allows a wide customization of body measurements. Any body may be described by a set of measurements, which can either be explicitly specified, either computed from other specified measurements using statistics (Fig. 9) as described by Seo et al. [42]. Input data may for instance be extracted from 3D body scanner data. While the initial draping of a designed garment is performed on a generic body of ‘standard’ measurements, the user may then change any measurement with on-the-fly fitting of the garment on the body, immediately assess fitting and comfortability, and change garment size and measurement accordingly.

Postures can be changed interactively in the same way. While the garment is usually created on a generic body

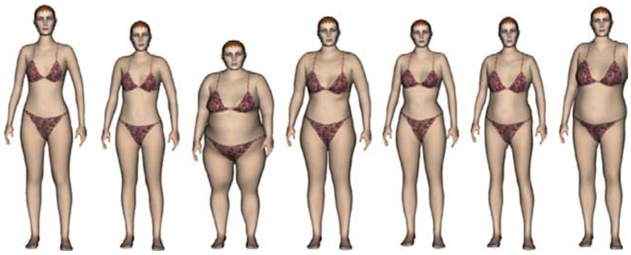


Fig. 9. Bodies of different measurements.

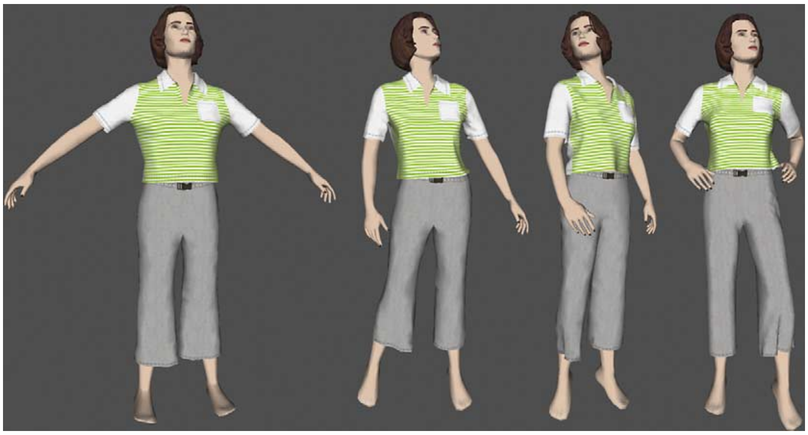


Fig. 10. Animation from one posture to another allows testing the fitness and the comfortability of a garment.

posture that facilitates garment assembly and draping, the user may then switch between any suitable posture. The system automatically generates a small animation of the body from the initial posture to the new one, and mechanical computation will automatically adapt the draped garment to the new posture (Fig. 10). Beside creating artistic poses, this also helps the garment designer to validate the shape of the garment using the comfortability tools.

Future systems may simplify some garment design tasks by testing automatically the created garment for fitting and comfortability against a selected range of sizes and postures. Other applications may be the automatic size and shape selection of a garment model according to the measurements of the customer, obtained for instance from a personal smartcard of measured using a body scanner.

3.5. Real-time animation of garments

The real-time animation has been integrated to the garment design tool to provide a preview of the clothes. Given an animated body and a garment whose rest shape has been computed with the garment design tool, this module provides the real-time visualization of the garments on the animated body. This module can be used at any time during the making of the clothes helping the designers to judge the quality of their clothes.

3.5.1. Garment preprocessing

Simulating garments in real-time requires drastic simplifications of the simulation process to be carried out, possibly at the expense of mechanical and geometrical accuracy. Our approach, described by Cordier et al. [11,12] is based on a hybrid method where the cloth is segmented into various sections where different algorithms are applied. When observing a garment worn on a moving character, we notice that the movement of the garment can be classified into several categories depending on how the garment is laid on, whether it sticks to, or flows on, the body surface. For instance, a tight pair of trousers will mainly follow the movement of the legs, whilst a skirt will flow around the legs. Thus, we segment the cloth into three layers that we define as follows (Fig. 11):

- Layer 1 (Stretch cloth): garment regions that stick to the body with a constant offset. In this case, the cloth follows exactly the movement of the underlying skin surface.
- Layer 2 (Loose cloth): garment regions that move within a certain distance to the body surface are placed in another category. The best examples are shirtsleeves. The assumption in this case is that the cloth surface always collides with the same skin surface and its movement is mainly perpendicular to the body surface.
- Layer 3 (Floating cloth): garment regions that flow around the body. The movement of the cloth does not follow exactly the movement of the body. Collisions are not predictable; for a long skirt, for instance, the left side of the skirt may collide with the right leg during animation.

These three categories are animated using three different cloth layers. The idea behind the proposed method is to avoid the heavy calculation of physical deformation and of collision detection wherever possible, i.e. where collision detection is not necessary. The main interest of our approach is to pre-process the target cloth and body model so that they

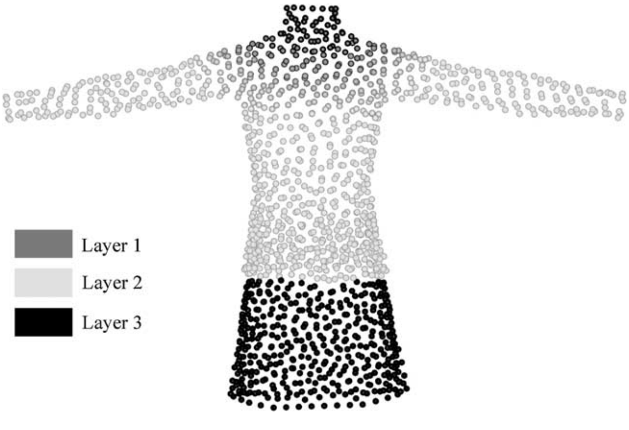


Fig. 11. Segmentation of garments.

are efficiently computable during runtime. The garments are divided into a set of segments and the associated simulation method is defined for each.

The segmentation process dispatches each garment region to its adequate layer. From the garment in its rest shape on the initial body, the distance between the garment and the skin surface is used to determine to which category the cloth triangles belong. Associated with each segment are distances from the skin surface that are used to determine the category. Each segment falls into one of three categories: tight, loose and floating clothes. Cloth vertices that are located closely to the skin surface belong to the first or the second layer. Cloth vertices that do not collide with any skin surface belong to the third layer.

3.5.2. Garment animation

In Refs. [11,12], techniques for real-time clothes simulation are proposed. Garment vertices are animated with three different methods, depending on which layer they belong that is defined during the pre-processing stage.

Tight clothes in Layer 1 follows the deformation of the underlying skin. These deformations are calculated thanks to the mapping of the attachment data of the skin to the garment surface.

For Layer 2 that is composed of loose clothes, the relative movements of clothes to the skin remain relatively small, keeping a certain distance from the skin surface. Consider the movement of sleeve in relation with the arm: for a certain region of the garment, the collision area falls within a fixed region of the skin surface during simulation. With this in mind, the scope of the collision detection can be severely limited. A basic assumption made is that the movement of the garment largely depends on that of the underlying skin and yet it should not follow the skin surface rigidly. It is necessary to simulate the local displacement of the garment from the skin surface.

Two different methods have been developed, one for cloth deformation on the limbs (trousers and sleeves), the other one for the deformation of cloth on the trunk. Cloth vertices on the limbs are enclosed in half spheres that are attached to the skin surface (Fig. 12). Vertices inside these spheres are displaced with the equation of the rigid body motion. A function defines the diameter of the spheres depending on the relative position of the cloth vertex to the normal of the skin surface.

Cloth vertices located on the trunk are animated with a rough mesh. This rough mesh is animated with a physic-based method. The cloth mesh is deformed with the FFD method using the position of the vertices on the rough mesh (Fig. 13).

Layer 3 is composed of vertices that freely float around the body. This will take care of cases, such as a large skirt floating around the legs. Any part on this skirt can collide with any part of the leg. The simulation of this layer uses a classical approach with particle system and collision avoidance.

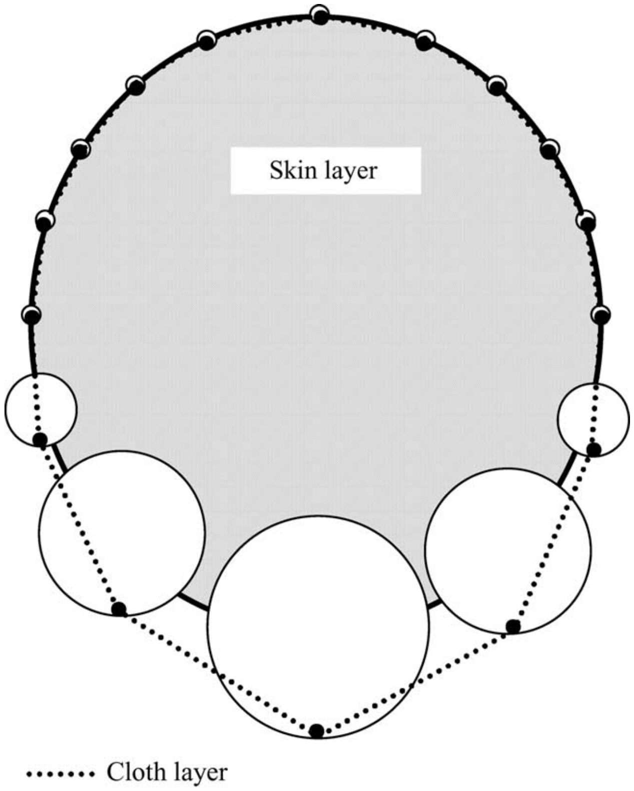


Fig. 12. Cross-section of a limb with a garment.

3.6. The virtual try-on

The garment design tool would not be complete if the visualization of the garments by the customers would not be possible. The web application 'Virtual Try-On' has been developed for this purpose. With the Virtual Try-On [11, 12], customers can choose garments and try on 3D mannequins that are adjusted to their body measurements, and are assisted to conduct proper online purchase of apparels. The Virtual Try-On supports a number of efficient and interactive operations, such as automatic adjustment of the 3D mannequin according to the shopper's body measurement, the selection and trial of different garment items, the online fitting/resizing of the garment to the mannequin and real-time simulation of the garment movement (Figs. 14 and 15).

The Virtual Try-On is composed of two components, namely the Web Server and the Client Application (Fig. 16).

The online clothing store Web server consists of two databases: the body database that contains 3D mannequins and the garment database that is composed of the garments available for purchasing. Upon user selection, the chosen 3D garment model and the body are downloaded to the client. These garments have been designed with the garment design tool and exported to the garment database.

The Web Client is an ActiveX control installed on the customer machine, which main purpose is the simulation and the visualization of the garments and the body

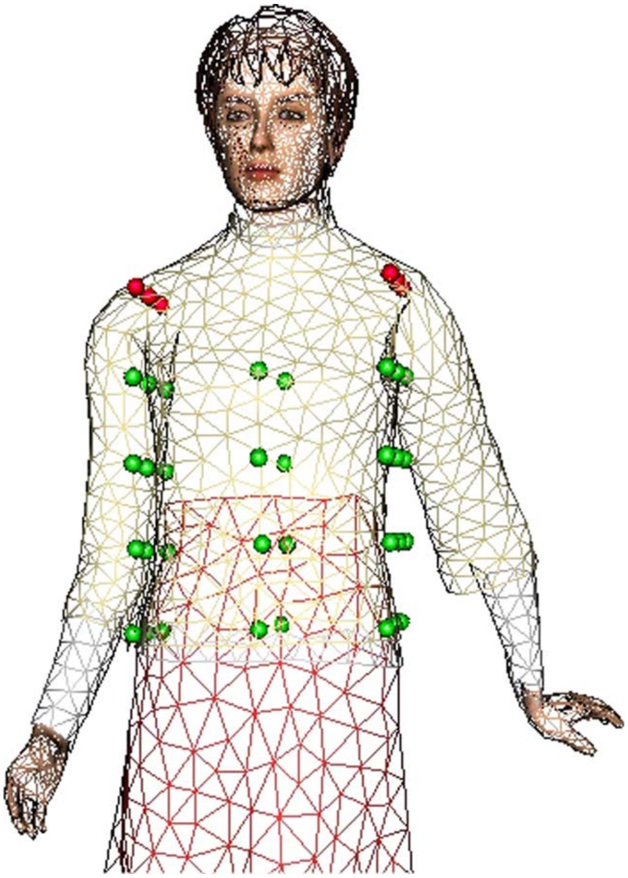
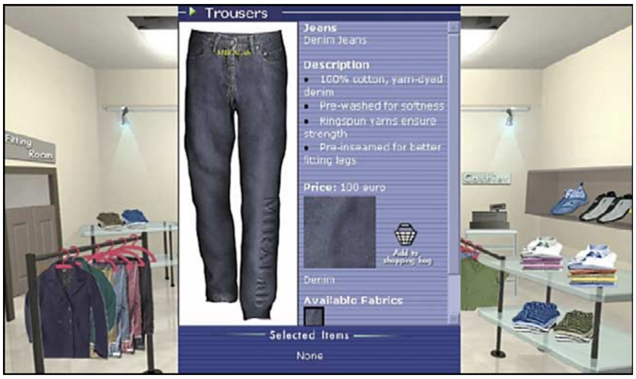


Fig. 13. Control (light) and attach (dark) points for FFD.

downloaded from the Web server. The Web Client application contains the body/garment sizing module and the real-time garment simulation module. The body/garment sizing module provides functionalities to deform the 3D mannequin from the customer's input body size and resize the selected garment accordingly. Once the garments and the mannequin have been adapted to the customer's measurements, the animation of the dressed mannequin is taken care of by the real-time garment simulation module.



Jeans
Denim Jeans

Description
- 100% cotton, yarn-dyed denim
- Pre-washed for softness
- Ringspun yarns ensure strength
- Pre-inseamed for better fitting legs

Price: 100 euro

Denim

Available Fabrics

Fig. 14. Interface for a web-based virtual try-on application.

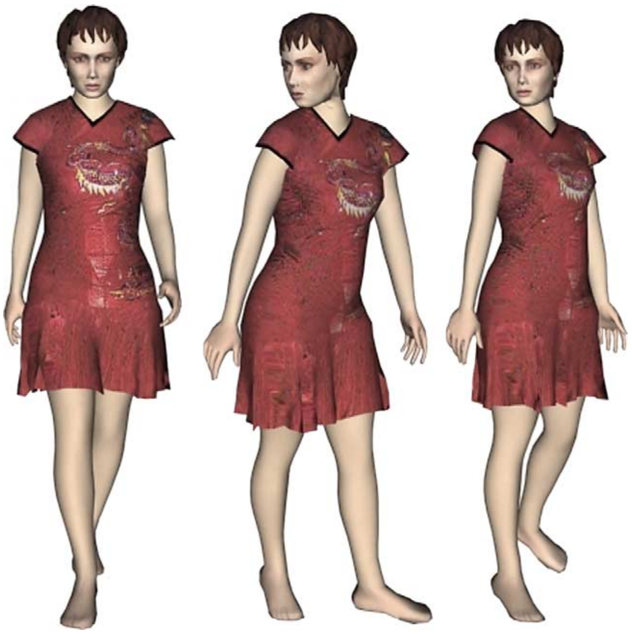


Fig. 15. Real-time animation of a dress.

The simulation is based on the method described in the Section 3.5.2.

4. Perspectives

While virtual prototyping still cannot pretend being a complete substitution to manual prototyping for a matter of accuracy, it nevertheless allows speeding up the design cycle of garments by allowing the garment designer to address quickly the issues related to draping and fitting. Our interactive system enhances these features by drastically shortening the cycle between pattern edition and visualization of the changes on the garment, skipping the process of reconstruction of new virtual garment models for each step of the design. As the measurements and the posture of the mannequin can also be changed interactively, this system is a great tool for validating fitting and comfortability of garments, for instance to assess how the garment will react to measurement changes of the body (aging, pregnancy), and various postures (sitting, walking, or specific postures for sportswear) (Fig. 17).

The benefit of this framework is not only directed to the garment design process. It can also be used as a visualization tool for customers who need a quick preview of customized garments on mannequins that have their own measurements. This could be available in the form of virtual fitting rooms. Web applications would also take advantage of the real-time animation preview for the convenience of the customers.

Blurring the line between 2D pattern shape design and the actual 3D garment wear on a custom mannequin finally opens the door to new potentialities in the field on how

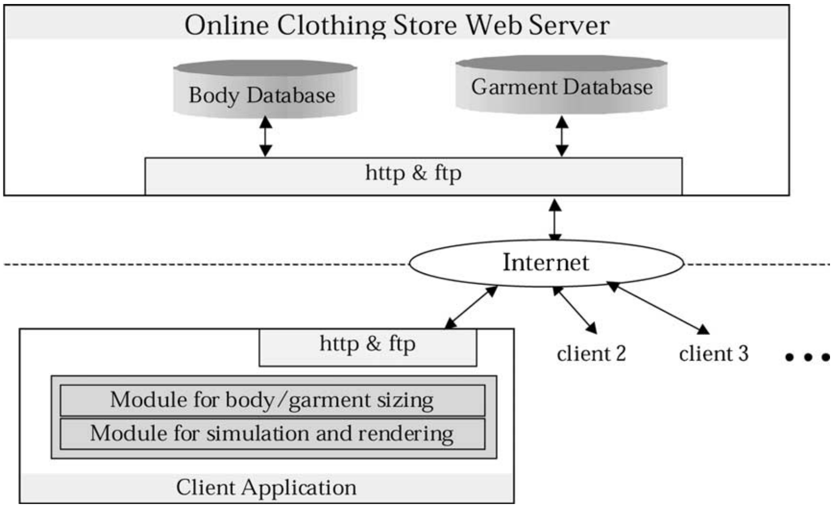

Online Clothing Store Web Server
Body Database
Garment Database
http & ftp
Internet
http & ftp
client 2
client 3
• • •
Module for body/garment sizing
Module for simulation and rendering
Client Application

Fig. 16. Architecture of the web-based virtual try-on application.

garments are being created. In the current design process where pattern makers 'flatten' the sketches drawn by fashion designers in a long and tedious process, future advances could short-circuit the consideration of pattern shape during the design phase by providing advanced 3D garment sketching tools directly to the fashion designer for dressing a mannequin directly in the virtual world.

The priority for 3D virtual garment development is still oriented a fast and exact simulation with fast visualization [45]. One major issue is the accuracy of the simulation for representing accurately wrinkles, folds, and refined design details. For instance, the software still needs the patterns to be designed in a way suitable for simulation, mostly by cleaning up the pattern from all the small feature details that cannot be simulated, also by taking into account how seaming is managed by the software (for instance by removing the overlap of cloth along seaming lines, overlap which should ideally be properly simulated with its specific mechanical behavior).

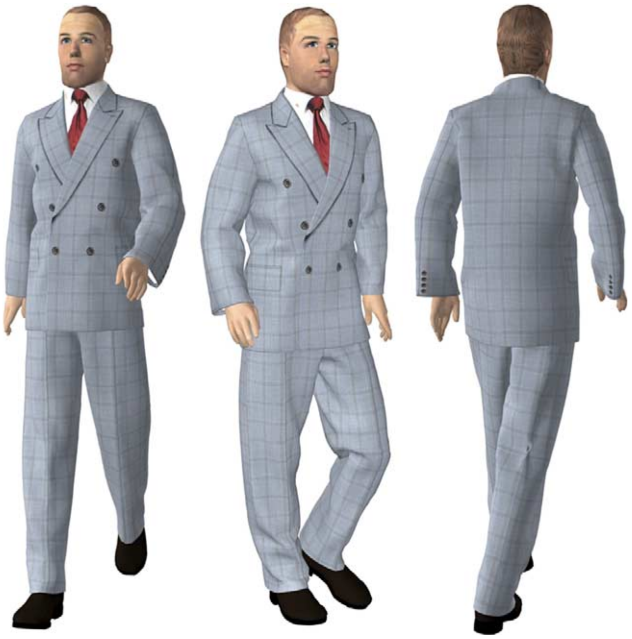


Fig. 17. Accurate animation of virtual garments.

Despite important progress [20,21], some advances are still required for obtaining the ideal system where patterns directly obtained from traditional pattern design frameworks can be accurately simulated with all their relevant features without any user interaction. For instance, this concerns automatic placement and assembly of complex garments around a character (for instance, multilayer garments), ideally for obtaining a system that would automatically generate a virtual garments directly from the output of CAD pattern design systems. Other functions have to be more automatic to speed up the design and simulation process, such as the making of folds, darts, buttons or other features.

However, new virtual simulation may also offer new paths toward innovative design techniques for the garment industry. For instance, the design of the garment could be done directly in 3D, with automatic generation of the pattern shapes as output for the pattern maker. In a further step, new 3D interaction tools and advanced features, such as development in automatic recognition of shapes out of sketches of a fashion designer, is a vast area of new potentialities.

Acknowledgements

This work is supported by the Swiss National Science Foundation (SNSF). Thanks to Marlène Arévalo and Christiane Luible for their design work to the garment models illustrating this paper.

References

[1] Agui T, Nagao Y, Nakajma M. An expression method of cylindrical cloth objects-an expression of folds of a sleeve using computer graphics. Trans Soc Electron, Inf Commun, J73-D-II 1990;7:1095–7.

[2] Baraff D, Witkin A. Large steps in cloth simulation. Comput Graph (SIGGRAPH'98 proceedings) 1998;32:106–17.

[3] Baraff D, Witkin A, Kass M. Untangling cloth. ACM Transactions on Graphics, proceedings of ACM SIGGRAPH 2003: 22(3)862–87.

[4] Bergen G. Efficient collision detection of complex deformable models using AABB trees. J Graph Tools 1997;2(4):1–14.

[5] Breen DE, House DH, Wozny MJ. Predicting the drape of woven cloth using interacting particles. Comput Graph (SIGGRAPH'94 proceedings) 1994;365–72.

[6] Bridson R, Fedkiv R, Anderson J. Robust treatment of collisions, contact and friction for cloth animation. Comput Graph (SIGGRAPH'02 Proceedings) 2002.

[7] Carignan M, Yang Y, Magnenat-Thalmann N, Thalmann D. Dressing animated synthetic actors with complex deformable clothes. Comput Graph (SIGGRAPH'92 Proceedings) 1992;26(2):99–104.

[8] Cugini U, Rizzi C. 3D Design and simulation of men garments. WSCG workshop proceedings 2002.

[9] Choi KJ, Ko HS. Stable but responsive cloth. Comput Graph (SIGGRAPH'02 Proceedings) 2002.

[10] Collier JR, Collier BJ, O'Toole G, Sargand SM. Drape prediction by means of finite-element analysis. J Textile Inst 1991;82(1):96–107.

[11] Cordier F, Magnenat-Thalmann N. Real-time animation of dressed virtual humans. Eurographics 2002 proceedings. Oxford: Blackwell; 2002.

[12] Cordier F, Seo H, Magnenat-Thalmann N. Made-to-measure technologies for online clothing store. IEEE CG & A special issue on web graphics. New York: IEEE Press; 2002 p. 38–48.

[13] DeRose T, Kass M, Truong T. Subdivision surfaces in character animation. Comput Graph (SIGGRAPH'98 Proceedings) 1998;32: 148–57.

[14] Debunne G, Desbrun M, Cani MP, Barr A. Adaptive simulation of soft bodies in real-time. Computer animation, annual conference series. New York: IEEE Press; 2000.

[15] Desbrun M, Schröder P, Barr A. Interactive animation of structured deformable objects. In: Proceedings of graphics interface.: AK Peters; 1999.

[16] Eberhardt B, Weber A, Strasser W. A fast, flexible, particle-system model for cloth draping. Computer graphics in textiles and apparel (IEEE computer graphics and applications). New York: IEEE Press; 1996 p. 52–9.

[17] Eberhardt B, Etzmuss O, Hauth M. Implicit-explicit schemes for fast animation with particles systems. In: Proceedings of the eurographics workshop on computer animation and simulation. Berlin: Springer; 2000 p. 137–51.

[18] Ehmann SA, Lin MC. Accurate and fast proximity queries between polyhedra using convex surface decomposition. Computer Graphics Forum. vol. 20. Oxford: Blackwell; 2001 p. 500–10.

[19] Eischen JW, Deng S, Clapp TG. Finite-element modeling and control of flexible fabric parts. Computer graphics in textiles and apparel (IEEE computer graphics and applications). New York: IEEE Press; 1996 p. 71–80.

[20] Fuhrmann A, Gross C, Luckas V. Interactive animation of cloth including self-collision detection. J WSCG 2003;11(1):141–8.

[21] Fuhrmann A, Gross C, Luckas V, Weber A. Interaction-free dressing of virtual humans. Comput Graph 2003;27(1):71–82.

[22] Hauth M, Etzmuss O. A high performance solver for the animation of deformable objects using advanced numerical metds. Eurographics 2001 proceedings. Oxford: Blackwell; 2001.

[23] Hind BK, McCartney J. Interactive garment design. Visual computer. vol. 6. Berlin: Springer; 1990 p. 53–61.

[24] Hubbard P. Approximating polyhedra with spheres for time-critical collision detection. ACM Trans Graph 1996;15(3):179–210.

[25] Gan L, et al. Study of fabric deformation using non-linear finite elements. Textile Res J 1995;65(11):660–8.

[26] Gottschalk S, Lin MC, Manosha D. OOBTree: a hierarchical structure for rapid interference detection. SIGGRAPH 96 conference proceedings. Reading, MA: Addison Wesley; 1996 p. 171–80.

[27] Grzeszczuk R, Terzopoulos D, Hinton G. Neuroanimator: fast neural network emulation and control of physics-based models. SIGGRAPH 98 conference proceedings, annual conference series. Reading, MA: Addison Wesley; 1998 p. 9–20.

[28] James D, Pai D. Accurate real-time deformable objects. SIGGRAPH 99 conference proceedings, annual conference series. Reading, MA: Addison Wesley; 1999 p. 65–72.

[29] Kang YM, Choi JH, Cho HG, Lee DH, Park CJ. Real-time animation technique for flexible and thin objects. WSCG' proceedings 2000 p. 322–29.

[30] Kang YM, Choi JH, Cho HG, Lee DH. An efficient animation of wrinkled cloth with approximate implicit integration. The visual computer journal. Berlin: Springer; 2001.

[31] Kang YM, Cho HG. Bilayered approximate integration for rapid and plausible animation of virtual cloth with realistic wrinkles. Computer animation proceedings, IEEE computer society 2002 p. 203–11.

[32] Klosowski JT, Held M, Mitchell JSB. Efficient collision detection using bounding volume Hierarchies of k-dops. IEEE Trans Visualization Comput Graph 1997;4(1).

[33] Lafleur B, Magnenat-Thalmann N, Thalmann D. Cloth animation with self-collision detection. IFIP conference on modeling in computer graphics proceedings. Berlin: Springer; 1991 p. 179–97.

[34] Larsson T, Akinine-Möller T. Collision detection for continuously deformable bodies. In: Proceedings of eurographics, short presentations. Oxford: Blackwell; 2001 p. 325–33.

[35] Metzger J, Kimmerle S, Etzmuss O. Hierarchical techniques in collision detection for cloth animation. J WSCG 2003;11(2): 322–9.

[36] Meyer M, Debunne G, Desbrun M, Barr AH. Interactive animation of cloth-like objects in virtual reality. J Visual Comput Anim 2000.

[37] Ng H, Grimsdale RL. GEOFF-A geometrical editor for fold formation. In: Chin R et al, editor. Lecture notes in computer science. Image analysis applications and computer graphic, vol. 1024. Berlin: Springer; 1995. p. 124–31.

[38] Oshita M, Makinouchi A. Real-time cloth simulation with sparse particles and curved faces. Computer animation. New York: IEEE Press; 2001.

[39] Press WH, Vetterling WT, Teukolsky SA, Flannery BP. Numerical recipes in C. 2nd ed. Cambridge: Cambridge University Press; 1992.

[40] Provot X. Deformation constraints in a mass-spring model to describe rigide cloth behavior. Graphics interface'95 proceedings.: AK Peters Ltd; 1995 p. 147–54.

[41] Sakagushi Y. Minoh m, Ikeda K. PARTY: "Physical Environment of Artificial Reality for Dress SimulAtion: A dynamically Deformable model of Dress". Trans Soc of Electronics, information and Communications, Dec 1991; p.25–32.

[42] Seo H, Magnenat-Thalmann N. ACM SIGGRAPH 2003 symposium on interactive 3D graphics. Monterey, CA: Addison-Wesley; April 2003 [to appear].

[43] Terzopoulos D, Platt JC, Barr H. Elastically deformable models. Comput Graph (SIGGRAPH'97 proceedings) 1987;21:205–14.

[44] Terzopoulos D, Fleischer K. Modeling inelastic deformation: viscoelasticity. Plast, Fract, Comput Graph (SIGGRAPH'88 proceedings) 1988;22:269–78.

[45] Magnenat-Thalmann N, Volino P, Cordier F. Avenues of research in dynamic clothing. Computer animation proceedings, IEEE computer society 2002 p. 193–202.

[46] Vassilev T, Spanlang B. Fast cloth animation on walking avatars. Eurographics proceedings. Oxford: Blackwell; 2001.

[47] Volino P, Magnenat-Thalmann N. Efficient self-collision detection on smoothly discretised surface animation using geometrical shape regularity. Comput Graph Forum (Eurographics'94 proceedings) 1994;13(3):155–66.

[48] Volino P, Courchesne M, Magnenat-Thalmann N. Versatile and efficient techniques for simulating cloth and other deformable objects. Comput Graph (SIGGRAPH'94 proceedings) 1995;95: 137–44.

[49] Volino P, Magnenat-Thalmann N. Developing simulation techniques for an interactive clothing system. Virtual systems and multimedia (VSMM'97 proceedings). Geneva, Switzerland: IEEE Press; 1997 p. 109–18.

[50] Volino P, Magnenat-Thalmann N. Fast geometrical wrinkles on animated surfaces. WSCG'99 proceedings, IEEE computer society 1999.

[51] Volino P, Magnenat-Thalmann N. Implementing fast cloth simulation with collision response. Computer graphics international proceedings, IEEE computer society 2000 p. 257–66.

[52] Volino P, Magnenat-Thalmann N. Accurate collision response on polygonal meshes. Computer animation 2000 proceedings, IEEE computer society 2000.

[53] Volino P, Magnenat-Thalmann N. Comparing efficiency of integration methods for cloth simulation. Computer graphics international proceedings 2001.

[54] Weil J. The synthesis of cloth objects. SIGGRAPH 86 conference proceedings, annual conference series. vol. 20. Reading, MA: Addison Wesley; 1986 p. 49–54.

[55] Yang Y, Magnenat-Thalmann N. An improved algorithm for collision detection in cloth animation with human body. Comput Graph Appl (Pacific graphics'93 proceedings) 1993;1:237–51.

[56] Zachmann G. Minimal hierarchical collision detection. In: Proceedings of the ACM symposium on virtual reality 2002 p. 121–28.

[57] http://www.aliaswavefront.com/

[58] http://www.discreet.com/

[59] http://www.digimation.com/

[60] http://www.chaosgroup.com/

[61] http://www.reyes-infografica.com/

[62] http://www.maxon.de/

[63] http://www.dressingsim.com/

[64] http://www.optitex.com/

[65] http://www.browzwear.com/

[66] http://www.fitme.com/

[67] http://www.miralab.ch/

Pascal Volino is a computer scientist and researcher. He studied engineering sciences in Lyon, France and graduated in 1992. He then joined MIRALab, University of Geneva, and has since then been working on various topics in the field of cloth simulation, including techniques for collision detection, mechanical simulation and user interaction. He obtained his PhD on this topic in 1998. He is currently involved into several European projects involving cloth simulation and virtual garments.

Frédéric Cordier is currently a post-doctoral researcher at the Dept. of Computer Science, at Advanced Institute of Science and Technology, under the direction of Professor Sung Yong Shin, in charge of the development of a sketching interface. After studies at Université Claude Bernard, Lyon, France, he graduated his Ph.D. in 2004 at MIRALab, University of Geneva, on the topic of real-time cloth simulation.

Nadia Magnenat-Thalmann has pioneered research into virtual humans over the last 20 years. She studied at the University of Geneva and obtained several degrees including Psychology, Biology, Chemistry, and a PhD in Quantum Physics at the University of Geneva in 1977. From 1977 to 1989, she was a Professor at the University of Montreal in Canada. In 1989, she founded MIRALab, an interdisciplinary creative research laboratory at the University of Geneva. She has published more than 200 papers, is editor-in-chief of the Visual Computer and the Journal of Visualization and Computer Animation journal. She has received several awards and recently, she has been nominated at the Swiss Academy of Technical Sciences.