# Accurate Garment Prototyping and Simulation

Pascal Volino[1] and Nadia Magnenat-Thalmann[2]

MIRALab, University of Geneva
[1] pascal@miralab.unige.ch
[2] thalmann@miralab.unige.ch

**ABSTRACT**
Despite numerous methods available for cloth simulation, virtual garment prototyping has yet to find its way toward the garment industry, the main issues being simulation accuracy and the potentiality for reproducing the complex behavior of complex garment models. These goals can only be reached through an optimal combination of modeling techniques and numerical methods that combines high computation efficiency with the versatility required for simulating intricate garment designs. We here describe optimal choices illustrated by their integration in a design and simulation tool that allow interactive prototyping of garments along drape motion and comfortability tests on animated postures.

**Keywords:** Mechanical Garment Simulation and Animation, Interactive Design.

## 1. INTRODUCTION

Garment simulation for prototyping application still remains a challenge, because of the high complexity of the simulation context, which goes far beyond the current application fields of the available cloth simulation systems.

Facing the high requirements of garment designers toward not only realism but also quantitative mechanical accuracy on draping and motion, the complexity and diversity of real garment models makes it a real challenge to design a garment simulation system [12]. Among the main difficulties are:

* The size of the garments, which can be meter-long and yet need to be simulated at millimeter accuracy.

* The intricate and highly variable shape of the garments, which interacts through complex contact patterns with the body (which is itself a complex deformable entity), as well as other garments.

* The highly deformable nature of cloth, which translate very subtle mechanical variations into large draping and motion variations which modify completely the visual appearance of garment models.

* The highly intricate anisotropic and nonlinear mechanical behavior of garments, requiring accurate measurement, modeling, and complex numerical methods for their resolution.

Mechanical simulation of cloth is a topic that is currently well explored, and many systems are already able to capture and simulate the mechanical properties of cloth. Among the best known techniques, spring-mass-like particle systems [4] take a large share, because these methods are fast and versatile, but unfortunately quite inaccurate when it comes to modeling complex anisotropic and nonlinear mechanical behaviors. On the other side, accurate methods such as finite elements [6] need huge computational requirements, and are not suited for the simulation of complete garments.

Among existing tools aimed at garment prototyping, most of them are focused only on draping, and are not suited for delivering accurate simulations of garments on animated characters. Animation is however a key issue in garment prototyping, as the motion of garments accounts a lot for the final visual look-and-feel of a dressing style. This however requires the simulation to reach a higher level of accuracy, with simulation of viscous behavior of cloth materials along with numerical integration methods that preserve the actual motion of the cloth surfaces along time. While many cloth simulation applications are available as commercial products, some of them capable of dealing with complete garments, none offers the accuracy necessary for an actual garment prototyping application, and therefore none is currently used in the garment industry.

We here intend to present a system aimed at fulfilling the requirements of a garment designer, which integrates state-of-the-art methods combining both requirements of mechanical accuracy with the power and versatility required for simulating complete dressing styles on animated characters. Among the main features available for the garment designer are pattern design with

interactive garment fitting evaluation, high-quality animation previews on moving characters, along with the possibility of managing dressing styles composed by several complex multilayer garments with many different materials and seamings.

We are putting emphasis on the following aspects:

* In Section 2, we describe a general mechanical model for cloth, which combines the versatility of particle systems with the accuracy of surface-based models, able to simulate the complex anisotropic nonlinear viscoelastic behaviors required for the accurate reproduction of the behavior of cloth, not only for draping applications but also for dynamic simulation of garment motion on animated characters.

* In Section 3, we discuss the particular problem of numerical integration, putting particular emphasis on studying the suitability of the various techniques to dynamical simulation of moving cloth through evaluation of their numerical damping.

* Several additional techniques required for simulating complex garments are also discussed, such as collision detection and processing techniques in Section 4.

* Finally, we show in Section 5 the results and the potentialities of all these methods put together in a system which is actually used to design and simulate complex virtual garments models.

## 2. REPRESENTING THE MECHANICS OF CLOTH

### 2.1. Mechanical Properties of Cloth

Cloth being approximated as a thin surface, its mechanical behavior is decomposed in in-plane deformations (the 2D deformations along the cloth surface plane) and bending deformation (the 3D surface curvature).

The in-plane behavior of cloth is described by relationships relating, for any cloth element, the stress $\boldsymbol{\sigma}$ to the strain $\boldsymbol{\varepsilon}$ (for elasticity) and its speed $\boldsymbol{\varepsilon}'$ (for viscosity) according the laws of viscoelasticity. For cloth materials, strain and stress are described relatively to the weave directions weft and warp following three components: weft and warp elongation (**uu** and **vv**), and shear (**uv**). Thus, the general viscoelastic behavior of a cloth element is described by strain-stress relationships as follows:

$$\begin{aligned}&\sigma_{uu}(\varepsilon_{uu},\varepsilon_{vv},\varepsilon_{uv},\varepsilon_{uu}',\varepsilon_{vv}',\varepsilon_{uv}')\\&\sigma_{vv}(\varepsilon_{uu},\varepsilon_{vv},\varepsilon_{uv},\varepsilon_{uu}',\varepsilon_{vv}',\varepsilon_{uv}')\\&\sigma_{uv}(\varepsilon_{uu},\varepsilon_{vv},\varepsilon_{uv},\varepsilon_{uu}',\varepsilon_{vv}',\varepsilon_{uv}')\end{aligned} \tag{1}$$

Assuming to deal with an orthotropic material (usually resulting from the symmetry of the cloth weave structure relatively to the weave directions), there is no dependency between the elongation components (**uu** and **vv**) and the shear component (**uv**). Assuming null Poisson coefficient as well (a rough approximation), all components are independent, and the fabric elasticity is simply described by three independent elastic strain-stress curves (weft, warp, shear), along with their possible viscosity counterparts.

In the same manner, viscoelastic strain-stress relationships relate the bending momentum to the surface curvature for weft, warp and shear. With the typical approximations used with cloth materials, the elastic laws are only two independent curves along weft and warp directions (shear is neglected), with their possible viscosity counterparts.

Elasticity curves of cloth materials can be evaluated using standardized procedures, such as the Kawabata Evaluation System (KES) of the SiroFAST procedure, which also aim to describe fundamental curve properties using a set of parameters. There are however no standard experimental procedures for measuring the viscosity properties.

### 2.2. Simulation using Particle Systems

The issue is now to define a model for representing these mechanical properties on geometrical surfaces representing the cloth. These curved surfaces are typically represented by polygonal meshes, being either triangular or quadrangular, and regular or irregular.

Continuum mechanics are one of the schemes used for accurate representation of the cloth mechanics. Mechanical equations are expressed along the curved surface, and then discretized for their numerical resolution. Such accurate schemes are however slow and not sufficiently versatile for handling large deformations and complex geometrical constraints (collisions) properly. Finite Element methods express the mechanical equations according to the deformation state the surface within well-defined elements (usually triangular or quadrangular). Their resolution also involves large computational charges. Another option is to construct a model based on the interaction of neighboring discrete points of the surface. Such particle systems allow the implementation of simple and versatile models adapted for efficient computation of highly deformable objects such as cloth.

*2.2.1. Spring-Mass Models*

The simplest particle system one can think of is spring-mass systems. In this scheme, the only interactions are forces exerted between neighboring particle couples, similarly as if they were attached by springs (described by an force/elongation law along its direction, which is actually a rigidity coefficient and a rest length in the case of linear springs). Spring-mass schemes are very popular methods, as they allow simple implementation and fast simulation of cloth objects. There has also been recent interest in this method as it allows quite a simple computation of the Jacobian of the spring forces, which is needed for implementing semi-implicit integration methods (see Section 3).

The simplest approach is to construct the springs along the edges of a triangular mesh describing the surface. This however leads to a very inaccurate model that cannot model accurately the anisotropic strain-stress behavior of the cloth material, and also not the bending. More accurate models are constructed on regular square particle grids describing the surface. While elongation stiffness is modeled by springs along the edges of the grid, shear stiffness is modeled by diagonal springs and bending stiffness is modeled by leapfrog spring along the edges. This model is still fairly inaccurate because of the unavoidable cross-dependencies between the various deformation modes relatively to the corresponding springs. It is also inappropriate for nonlinear elastic models and large deformations. More accurate variations of the model consider angular springs rather than straight springs for representing shear and bending stiffness, but the simplicity of the original spring-mass scheme is then lost.

*2.2.2. An Accurate Particle System Model*

Because of the real need of representing accurately the anisotropic nonlinear mechanical behavior of cloth in garment prototyping applications, spring-mass models are inadequate, and we need to find out a scheme that really simulates the viscoelastic behavior of actual surfaces. For this, we have defined a particle system model that relates this accurately over any arbitrary cloth triangle through simultaneous interaction between the three particles which are the triangle vertices. Such a model integrates directly and accurately the strain-stress model defined in Part 2.1 using polynomial spline approximations of the strain-stress curves, and remains accurate for large deformations.

In this model, a triangle element of cloth is described by 3 2D coordinates (**ua**, **va**), (**ub**, **vb**), (**uc**, **vc**) describing the location of its vertices **A**, **B**, **C** on the weft-warp coordinate system defined by the directions **U** and **V** with an arbitrary origin. They are orthonormal on the undeformed cloth (Fig.1). Out of them, a precomputation process evaluates the following values:

$$\begin{aligned} Rua &= d^{-1}(vb-vc) & Rva &= -d^{-1}(ub-uc) \\ Rub &= -d^{-1}(va-vc) & Rvb &= d^{-1}(ua-uc) \\ Ruc &= d^{-1}(va-vb) & Rvc &= -d^{-1}(ua-ub) \end{aligned} \quad \text{with} \quad d = ua(vb-vc)+ub(vc-va)+uc(va-vb) \tag{2}$$

During the computation process, the current deformation state of the cloth triangle is evaluated using the current 3D direction and length of the deformed weft and warp direction vectors **U** and **V**. They are computed from the current positions **Pa**, **Pb**, **Pc** of its supporting vertices as follows:

$$U = Rua\ Pa + Rub\ Pb + Ruc\ Pc \qquad V = Rva\ Pa + Rvb\ Pb + Rvc\ Pc \tag{3}$$

The current in-plane strains $\boldsymbol{\varepsilon}$ of the cloth triangle is then computed with the following formula:

$$\varepsilon_{uu} = |U|-1 \qquad \varepsilon_{vv} = |V|-1 \qquad \varepsilon_{uv} = \frac{|U+V|}{\sqrt{2}} - \frac{|U-V|}{\sqrt{2}} \tag{4}$$

We have chosen to replace the traditional shear deformation evaluation based on the angle measurement between the thread directions by an evaluation based on the length of the diagonal directions. The main advantage of this is a better accuracy for large deformations (the computation of the behavior of an isotropic material under large deformations remains more axis-independent).

For applications that model internal in-plane viscosity of the material, the "evolution speeds" of the weave direction vectors are needed as well. They are computed from the current triangle vertex speeds **Pa'**, **Pb'**, **Pc'** as follows:

$$U' = Rua\ Pa' + Rub\ Pb' + Ruc\ Pc' \qquad V' = Rva\ Pa' + Rvb\ Pb' + Rvc\ Pc' \tag{5}$$

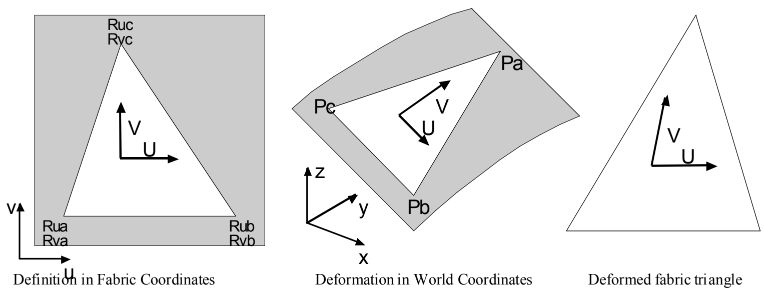
Then, the current in-plane strain speeds $\boldsymbol{\varepsilon}'$ of the triangle is computed:

$$\varepsilon_{uu}' = \frac{U \cdot U'}{|U|} \qquad \varepsilon_{vv}' = \frac{V \cdot V'}{|V|} \qquad \varepsilon_{uv}' = \frac{(U+V)\cdot(U'+V')}{|U+V|\sqrt{2}} - \frac{(U-V)\cdot(U'-V')}{|U-V|\sqrt{2}} \tag{6}$$

At this point, the in-plane mechanical behavior of the material can be expressed for computing the stresses $\boldsymbol{\sigma}$ out of the strains $\boldsymbol{\varepsilon}$ (elasticity) and the strain speeds $\boldsymbol{\varepsilon}'$ (viscosity) using the curves discussed in Part 2.1 Finally, the force contributions of the cloth triangle to its support vertices computed from the stresses $\boldsymbol{\sigma}$ as follows:

$$
\begin{aligned}
Fa &= -\frac{d}{2}\left(\left(Rua\ \sigma_{uu} + Rva\ \sigma_{uv}\right)\frac{U}{|U|} + \left(Rua\ \sigma_{uv} + Rva\ \sigma_{vv}\right)\frac{V}{|V|}\right)\\
Fb &= -\frac{d}{2}\left(\left(Rub\ \sigma_{uu} + Rvb\ \sigma_{uv}\right)\frac{U}{|U|} + \left(Rub\ \sigma_{uv} + Rvb\ \sigma_{vv}\right)\frac{V}{|V|}\right)\\
Fc &= -\frac{d}{2}\left(\left(Ruc\ \sigma_{uu} + Rvc\ \sigma_{uv}\right)\frac{U}{|U|} + \left(Ruc\ \sigma_{uv} + Rvc\ \sigma_{vv}\right)\frac{V}{|V|}\right)
\end{aligned}
\tag{7}
$$

It is important to note that when using semi-implicit integration schemes (see Section 3), the contribution of these forces in the Jacobian $\partial\mathbf{F}/\partial\mathbf{P}$ and $\partial\mathbf{F}/\partial\mathbf{P}'$ can easily be computed out of the curve derivatives $\partial\boldsymbol{\sigma}/\partial\boldsymbol{\varepsilon}$ and the orientation of the vectors **U** and **V**.



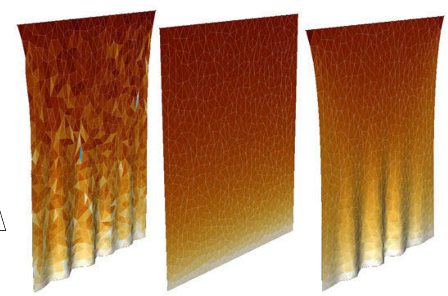



Fig.1 (left): A triangle of cloth element defined on the 2D cloth surface (left) is deformed in 3D space (center) and its deformation state is computer from the deformation of its weft-warp coordinate system (right).

Fig.2 (right): Drape accuracy between a simple spring-mass system along the edges of the triangle mesh (left) and the proposed accurate particle system model (center). Color scale shows deformation. The spring-mass model exhibits inaccurate local deformations, along with an excessive "Poisson" behavior. This is not the case with the accurate model, which may still model the "Poisson" effect if needed (right, with a Poisson coefficient 0.5). The spring-mass model is also unable to simulate anisotropic or nonlinear models accurately.

### *2.2.3. Validation of the Model*

We have tested the accuracy of the model by reproducing a "virtual" tensile test using an isotropic material featuring nonlinear metric behavior $\boldsymbol{\sigma_{uu}}(\boldsymbol{\varepsilon_{uu}}) = \boldsymbol{\sigma_{vv}}(\boldsymbol{\varepsilon_{vv}}) = 2\ \boldsymbol{\sigma_{uv}}(\boldsymbol{\varepsilon_{uv}})$ described with various piece-wise polynomial curves. With elongations up to 50%, the matching accuracy between the original curve and the measured curve did not exceed 0.1% when measured along thread directions, and 1% when measured along arbitrary directions. The error amount was mainly related to the nonlinearity of the curve and the roughness of the mesh.

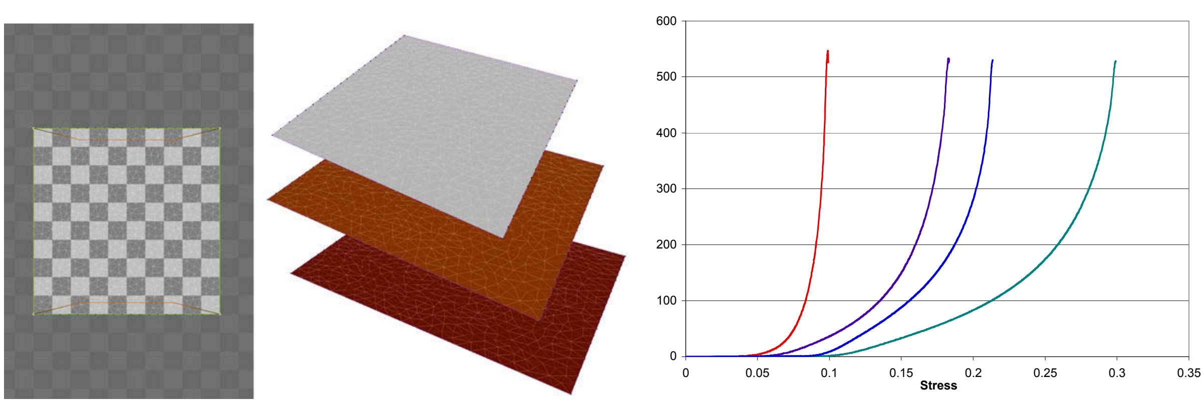



Fig.3: Virtual elongation test: A square sample of fabric, attached along its opposite edges, is extended. Its mechanical properties are defined using polynomial spline approximations of real material (curves). The forces measured on the edges against elongation are then compared to the initial curves.

The model has been extended to handle anisotropic curvature stiffness through bending momentums applied along edges, for which the angle between the adjacent elements give an evaluation of the local surface curvature along the orthogonal direction. Additional mechanical features are integrated in the model as well for simulating the behavior of complex garment features (Fig.4).

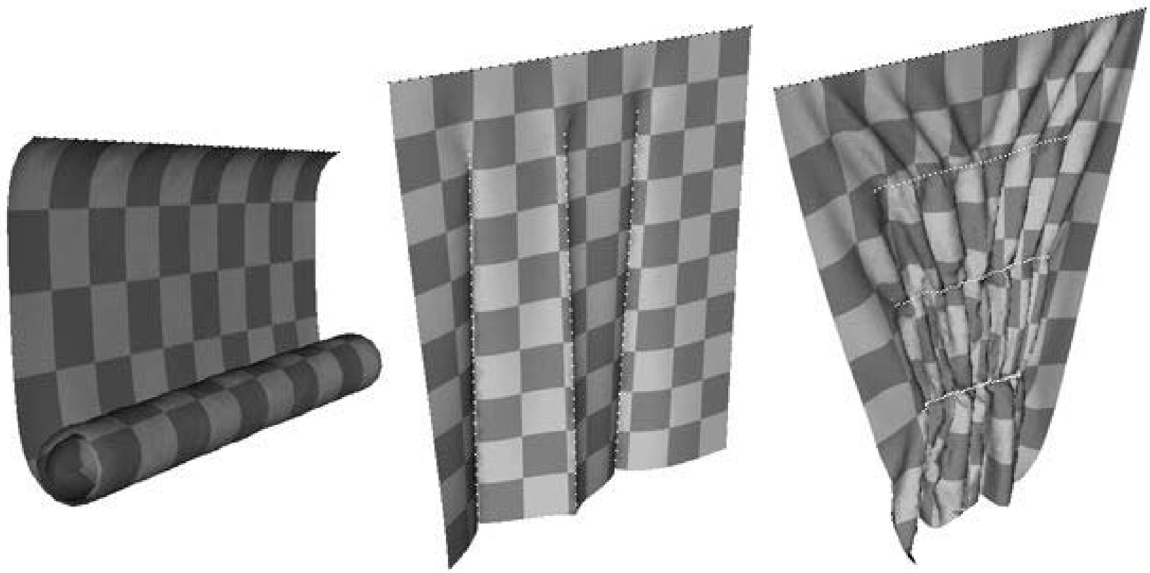


Fig.4: The proposed model simulates accurately anisotropic bending stiffness, with possible rest curvature defined on the surface (left). Rest curvature may also be defined along precise lines (center) . Lines may also carry additional stiffness with their own custom rest length (right). All these features bring lot of potentialities for designing complex garment models.

## 3. NUMERICAL INTEGRATION

The equations resulting from the mechanical formulation of particle systems do usually express particle forces **F** depending on the state of the system (particle positions **P** and speeds **P**'). In turn, particle accelerations **P**'' is related to particle forces **F** and masses **M** by Newton's 2nd law of dynamics. This leads to a second-order ordinary differential equation system, which is turned to first-order by concatenation of particle position **P** and speed **P**' into a state vector **Q**. A vast range of numerical methods has been studied for solving this kind of equations.

We have conducted extensive tests for benchmarking numerous integration methods, using performance, accuracy, stability and robustness as criteria. We have selected three candidates, each of which performs best in its own context:

* *1st-order semi-implicit Backward Euler*, which seems to be the best robust general-purpose method for any relaxation task (garment assembly and draping) [1] [14].
* *2nd-order semi-implicit Backward Differential Formula*, which offers increased dynamic accuracy along time (garment simulation on animated characters), at the expense of robustness (unsuited for draping during interactive design) [7].
* *5th-order explicit Runge-Kutta with timestep control*, which offers very high non-dissipative dynamic accuracy (accurate simulation of viscous and dissipative parameters in animated garments), at the expense of computation time (requires small time steps depending on the numerical stiffness, unsuited for stiff materials and refined discretizations) [4].

Our implementation integrates these three methods, and dynamically switches between them depending on the simulation context.

### 3.1. Discussing Integration Methods

*3.1.1. Implicit Integration Methods*

The most widely-used method for cloth simulation is currently the semi-implicit Backward Euler method, which was first used by Baraff et al [1] in the context of cloth simulation. As any implicit method, it alleviates the need of high accuracy for the simulation of stiff differential equations, offering convergence for large timesteps rather than numerical instability (a step of the semi-implicit Euler method with "infinite" timestep is actually equivalent to an iteration of the Newton resolution method) [15].

The formulation of a generalized implicit Euler integration is the following:

$$Q_{(t+dt)} - Q_{(t)} = Q'_{(t+\alpha\, dt)}\, dt \tag{8}$$

The derivative value is not known at a moment after **t**, and is then extrapolated from the value at moment **t** using the Jacobian, leading to the semi-implicit expression which requires the resolution of a linear system:

$$Q_{(t+dt)} - Q_{(t)} = \left( I - \alpha \left.\frac{\partial Q'}{\partial Q}\right._{(t)} dt \right)^{-1} Q'_{(t)}\, dt \tag{9}$$

We have introduced the coefficient **α** so as to modulate the "implicitiсy" of the formula. Hence, **α = 1** is the regular implicit Backward Euler step (stable), whereas **α = 0** is the explicit Forward Euler step (unstable), and **α = 1/2** is the 2nd-order implicit Midpoint step (most accurate, at the threshold of stability).

The **α** parameter is a good handle for adjusting the compromise between stability and accuracy. While maximum robustness is obviously observed for large values, reducing its value increases accuracy (reduces numerical damping) at the expense of stability, and speeds up the computation as well (better conditioning of the linear system to be resolved).

Better accuracy can also be obtained through the use of the 2nd-order Backward Differential Formula (BDF-2), as described by Hauth et al [7]. This uses the previous state of the system for enhancing accuracy up to 2nd-order, with a minimal impact on the computation charge. Its generalized implicit expression is:

$$Q_{(t+dt)} - Q_{(t)} = \beta \left( Q_{(t)} - Q_{(t-dt)} \right) + Q'_{(t+\alpha\, dt)}\, \delta t \qquad \text{with} \qquad \beta = \frac{2\,\alpha - 1}{2\,\alpha + 1} \qquad \text{and} \qquad \delta t = \frac{2}{2\,\alpha + 1}\, dt \tag{10}$$

And its semi-implicit expression is:

$$Q_{(t+dt)} - Q_{(t)} = \left( I - \alpha \left.\frac{\partial Q'}{\partial Q}\right._{(t)} \delta t \right)^{-1} \left( \beta \left( Q_{(t)} - Q_{(t-dt)} \right) + Q'_{(t)}\, \delta t \right) \tag{11}$$

While **α = 1** is the regular implicit BDF-2 step, **α = 0** is the explicit Leapfrog method, and **α = 1/2** is again the implicit Midpoint method. Best accuracy is offered for **α = 1/√3**, where the method is 3rd-order (moderately stable).

Compared to Backward Euler, the main interest of the BDF-2 method is that it exhibits better accuracy for dynamic simulation over time (less numerical damping) for moderately stiff numerical contexts (at the expense of reduced robustness for nonlinear situations) (Fig.5). For very stiff contexts however, this benefit disappears.

While it is possible to implement higher-order BDF methods, their interest is reduced by their lack of stability, and high accuracy could be more efficiently reached using high-order explicit methods. Stability of implicit methods is also affected by the nonlinearities of the mechanical model.

*3.1.2. Explicit Integration Methods*

Unlike implicit methods, explicit methods do not offer convergence to equilibrium if the timestep is too large compared to the numerical stiffness of the equations. On the other hand, they are very simple to implement, and much compute much faster than their implicit counterpart for reaching a given accuracy. This is particularly true for high-order methods, which offer very high accuracy if the timestep is small enough, but diverge abruptly if it exceeds a threshold (related to the stiffness of the equations). This is why an efficient timestep control scheme is essential for the implementation of these methods.

While the explicit 1st-order Euler and 2nd-order Midpoint methods should be restricted to simple applications (beside their simplicity, they have no benefits compared to their implicit counterparts), a popular choice is the 5th-order Runge-Kutta scheme with embedded error evaluation [11]. It is a six-stage iteration process where the computed error magnitude can be used for controlling the adequate timestep very accurately, depending on accuracy and stability requirements. Unlike implicit methods, this method yields a very good guaranteed accuracy (resulting from the high-order, but which may require very small timesteps), which is particularly important for problems where energy conservation is a key issue (for example, evaluating the effect of viscous parameters in the motion of fabrics) (Fig.5). On the other hand, explicit methods are quite unsuited for the fast relaxation of the static cloth draping applications.

### 3.2. Implementation Issues

While there are no particular issues related to the implementation of explicit integration methods, semi-implicit methods require the resolution of large sparse linear equations systems, which are mainly constructed from the Jacobian of the mechanical law **∂F/∂P** and **∂F/∂P**' (their sparse structure relates the mechanical dependency between the particles). Among possible speed-up approximations, the Implicit-Explicit method described in [5] neglects the Jacobian terms generated by the non-stiff forces (which are then explicitly integrated).

A choice candidate for resolving this linear system is the Conjugate Gradient method, which is iterative and thus offers compromise between computation charge and symmetric accuracy, and which also allows efficient implementation for sparse systems.

Among possible optimizations are linearization schemes aimed at performing the computation using a constant approximation of the Jacobian, so as to implement preprocessing optimizations in the resolution. While giving reasonable benefits for draping applications, these approximations however generate large "numerical damping" that slow down convergence and alter highly the motion of the cloth along time [2] [3] [8] [10].

The only solution for simulating the accurate motion of cloth was indeed to use real value of the Jacobian corresponding to the current state of the system. We have taken advantage of the Conjugate Gradient method which only needs the Jacobian matrix products with given vectors to compute these products "on the fly" directly from the system state, skipping the sparse explicit storage of the matrix for each frame. Our system actually allows performing partial linearization of the Jacobian, so as to use the linearization ratio offering the best tradeoff between motion accuracy and stability, depending on the simulation context.

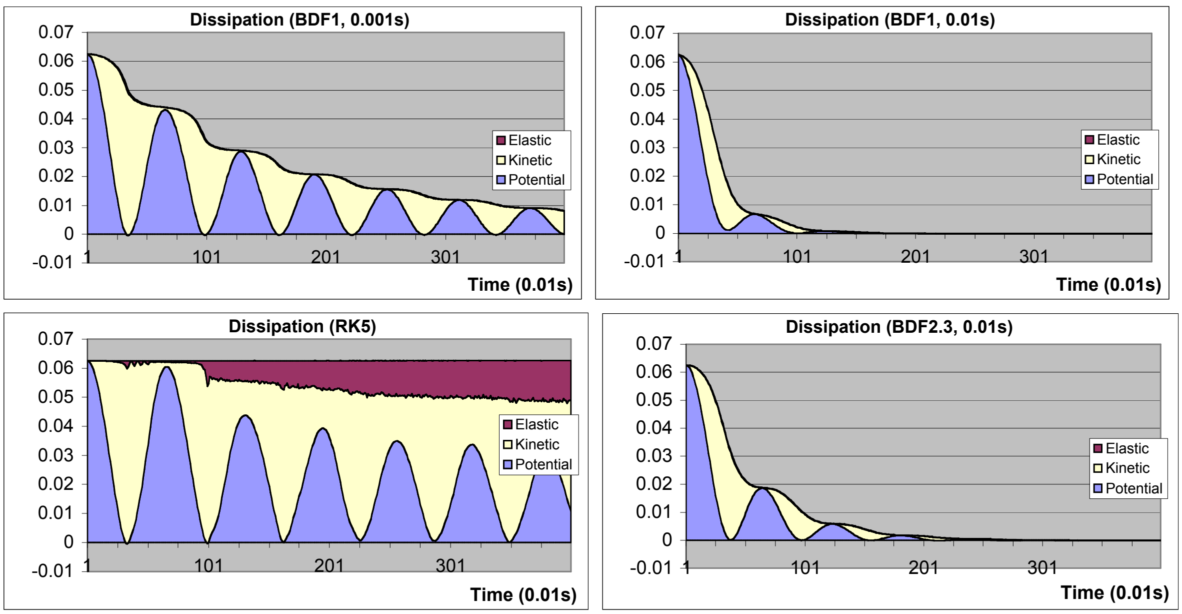



Fig.5: Evaluation of numerical damping of various integration methods using energy dissipation plots along time(50cm x 50cm square cloth, initially horizontal, attached along one edge, linear isotropic 100N/m, 100g/m$^2$, 2cm$^2$ elements, no dissipative parameters). 5th-order Runge-Kutta (botton left) accurately preserves the total energy along time, a good amount of it being transferred to elastic energy through small-scale mesh jittering (timesteps between 0.0001s and 0.00001s). Implicit methods such as Inverse Euler (upper) damp small-scale motion (elastic energy quickly reaches its rest value) the more as timestep increases (from left to right) and the 3rd-order BDF2 variation (bitton right) preserves energy significantly better than Inverse Euler (upper right).

## 4. COLLISION PROCESSING

Collision detection is indeed one of the most time-consuming tasks when it comes to simulate virtual characters wearing complete garments [9]. This task is performed through an adapted bounding-volume hierarchy algorithm, which uses a constant Discrete-Orientation-Polytope hierarchy constructed on the mesh, and optimization for self-collision detection using curvature evaluation on the surface hierarchy. This algorithm is fast enough for allowing full collision and self-collision detection between all objects of the scene with acceptable impact on the processing time (rarely exceeds 20% of the total time). Thus, body and cloth meshes are handled totally symetrically by the collision detection process, ensuring perfect versatility of the collision handling between the body and the several layers of garments [13].

Collision response is handled using a geometrical scheme based on correction of mesh position, speed and acceleration. This scheme ensures good accuracy and stability without the need of large nonlinear forces that alter the numerical resolution of the mechanical model. Our model simulates contact forces through a perfectly damped reaction model, associated to a Coulombian (solid) friction model.

The implemented collision model ensures full mesh-to-mesh collision response, which can deal with very complex multilayer collisions configurations involving several surfaces. The collision processing is therefore general enough for handling contacts between the several garments of a complex dress style, as well as the interactions between complex fold patterns when animating ample gestures. The model is also accurate enough for reproducing accurately friction behavior, allowing for example pants to hold to the waist with friction alone during character motion, without "cheating" using geometrical attachments. Good stability allows the simulation of complete multilayer garments with millimeter collision thickness despite large cloth speed and tension produced by complex character motion.

## 5. AN INTERACTIVE SYSTEM FOR GARMENT PROTOTYPING

While essential, computational techniques alone are not sufficient for producing a powerful tool allowing accurate and convenient creation and prototyping of complex garments. We have integrated all these techniques into a garment design and simulation tool aimed at prototyping and virtual visualization, and allowing fashion designers to experiment virtually new collections with high-quality preview animations, as well as pattern makers to adjust precisely the shape and measurements of the patterns to fit the body optimally for best comfort.

The high level of interactivity required by these features necessitates simultaneous computation of the 3D garment updated immediately to each design modification done to the patterns. Our design and simulation tool provides a dual view of the garment, featuring both the 2D view of the pattern shapes cut on the fabric and the 3D view of the garment worn by a virtual character, with tight synchronization (Fig.6). Any editing task carried out in one view is directly displayed in the other view.

Fig.6: Between real and virtual: Our system offers high-quality garment simulation, along with highly interactive pattern 2D-3D design and preview tools allowing complex garment models to be designed efficiently with many features such as seams, buttons, pockets, belts...

### 5.1. Interactive Garment Editing Tools

The system features a fast Constrained Delaunay triangulation scheme that allows the discretization of complex patterns described as polygonal lines of control points (2D locations on the fabric). The system allows variable discretization densities over the mesh, as well as size anisotropy (elements elongated in a given direction), for representing adaptively complete garments from large surfaces to intricate details.

The interactivity of the system is based on two main features:

* *Mesh mapping update*: The 2D displacement of any control point of the pattern shape on the cloth surface immediately updates the mesh of that pattern on the cloth, while leaving the 3D drape position of the cloth constant. For obtaining this, each vertex of the mesh keeps track of a weighted sum of the pattern control points, which is computed during the triangulation process. This allows any measurement or shape editing to be directly taken into account by the mechanical simulation without any heavy recomputation, for immediate feedback of any pattern sizing adjustment.

* *Mesh topology reconstruction*: When the topology of the pattern mesh is changed (rediscretization, new features, etc), the 3D drape position of the new mesh is automatically recomputed from the drape position of the old one. During this process, advanced algorithms compute, for each mesh vertex of the new mesh, the location of the surface of the old mesh having identical 2D coordinates on the fabric. Extrapolation methods are used for computing the location of vertices which are located outside the old surface. This allows pattern design changes (new features, darts, seams...) to be added and modified without needing re-assembling and re-draping the garment on the virtual body.

### 5.2. Interactive Garment Prototyping

Put together, these techniques greatly enhance the workflow of garment prototyping. For instance, an initial garment could be quickly draped over a character using a rough mesh. Then, the designer could enhance the pattern shapes, while mesh mapping update automatically alter the mechanical state of the draped garment, changing the draping shape. Once the garment design is ready, a high-accuracy drape is automatically produced using topology reconstruction with a refined mesh.

Combined with the accuracy and speed of the proposed mechanical simulation engine, tasks such as comfortability evaluations are open to the garment designer, through the addition of several visualization tools, such as (Fig.7):

* Preview of fabric deformations and tensions along any weave orientation.
* Preview of pressure forces of the garment on the body skin.
* Immediate update of these evaluations according to pattern reshaping and sizing, fabric material change, and body measurements and posture change.


Fig.7. Interactive garment prototyping and comfort evaluation: More than the garment shape and appearance design (left), the system also allows mechanical comfortability data to be evaluated directly with any change of the pattern design and sizing (right).

Dynamic surface remeshing allows the best compromise between accuracy and computation speed to be selected adaptively according to the needs of the garment designer. For instance, while the garment assembly process can be carried out in a matter of seconds using an approximate mechanical model on a rough garment surface mesh, the garment designer may then switch to a more accurate model for tasks such as accurate draping and comfort evaluation. The model is still efficient enough to react interactively to design changes with garments made of ten thousands polygons, an accurate draping being obtained in a few minutes. Practical geometric accuracy is roughly limited by using 5 millimeters elements (Fig.8). Using time-accurate computation on animated characters, a high-quality catwalk is computed in a matter of a few hours.

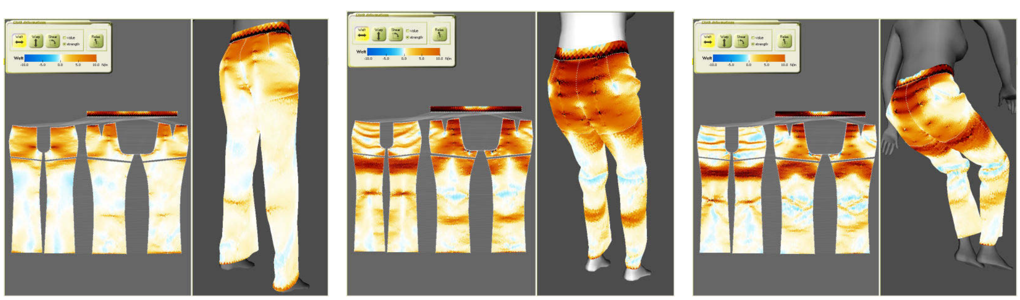


Fig.8: Virtual prototyping: Displaying weft constraints on an animated body (from standing to sitting). Element size is 5mm.

### 5.3. Perspectives

The system described here has already been used for creating fashion models that have enough realism for reproducing accurately the behavior of real garments (Fig.9). More than draping, our system is able to compute realistic animations thanks to good time-accurate numerical techniques applied to viscoelastic mechanical models of cloth. The core technologies of this system are now being adapted to actual needs of the garment industry through collaborative projects, which deal on mechanical characterization of fabrics, virtual prototyping, manufacturing processes, e-commerce. Although some advances are still welcome in the area of efficiency and accuracy of mechanical simulation techniques, the challenge is now to create new tools that will ensure to the garment industry a smooth transition from tradition to novel possibilities offered by virtual simulation.

Fig.9: Real fashion garments, and their animated virtual counterparts, designed and simulated with our system.

***Acknowledgements***

*We are grateful to all the people who contributed for technical help, suggestions, garment design and artwork (Christiane Luible and Nedjma Cadi). This work was funded by the European LeapFrog CA project.*

## BIBLIOGRAPHY

[1] D. Baraff, A. Witkin, Large Steps in Cloth Simulation, *Computer Graphics* (SIGGRAPH'98 proceedings), Addison-Wesley, 32, pp 106-117, 1998.

[2] K.J. Choi, H.S. Ko, Stable but Responsive Cloth, *Computer Graphics (SIGGRAPH'02 proceedings)*, Addison Wesley, 2002.

[3] M. Desbrun, P.Schröder, A. Barr, Interactive Animation of Structured Deformable Objects, *Proceedings of Graphics Interface*, A K Peters, 1999.

[4] B. Eberhardt, A. Weber, W. Strasser, A Fast, Flexible, Particle-System Model for Cloth Draping, *Computer Graphics in Textiles and Apparel (IEEE Computer Graphics and Applications)*, IEEE Press, pp 52-59, Sept. 1996.

[5] B. Eberhardt, O. Etzmuss, M. Hauth, Implicit-Explicit Schemes for Fast Animation with Particles Systems, *Proceedings of the Eurographics workshop on Computer Animation and Simulation*, Springer-Verlag, pp 137-151, 2000.

[6] J.W. Eischen, S. Deng, T.G. Clapp, Finite-Element Modeling and Control of Flexible Fabric Parts, *Computer Graphics in Textiles and Apparel (IEEE Computer Graphics and Applications)*, IEEE Press, pp 71-80, Sept. 1996.

[7] M. Hauth, O. Etzmuss, A High Performance Solver for the Animation of Deformable Objects using Advanced Numerical Methods, *Eurographics 2001 proceedings*, Blackwell, 2001.

[8] Y.M. Kang, H.G. Cho, Bilayered Approximate Integration for Rapid and Plausible Animation of Virtual Cloth with Realistic Wrinkles, *Computer Animation 2000 proceedings*, IEEE Computer Society, pp 203-211, 2002.

[9] J. Metzger, S. Kimmerle, O. Etzmuss, Hierarchical Techniques in Collision Detection for Cloth Animation, *Journal of WSCG*, 11 (2) pp 322-329, 2003.

[10] M. Meyer, G. Debunne, M. Desbrun, A. H. Barr, Interactive Animation of Cloth-like Objects in Virtual Reality, Journal of Visualization and Computer Animation, John Wiley & Sons, 2000.

[11] W.H. Press, W.T. Vetterling, S.A. Teukolsky, B.P. Flannery, *Numerical Recipes in C*, Second edition, Cambridge University Press, 1992.

[12] P. Volino, F,. Cordier, N. Magnenat-Thalmann, From Early Virtual Garment Simulation to Interactive Fashion Design, *Computer-Aided Design*, 37, Elsevier, pp 793-608, 2005.

[13] P. Volino, N. Magnenat-Thalmann, Developing Simulation Techniques for an Interactive Clothing System, *Virtual Systems and Multimedia (VSMM'97 proceedings)*, IEEE Press, Geneva, Switzerland, pp 109-118, 1997.

[14] P. Volino, N. Magnenat-Thalmann, Implementing fast Cloth Simulation with Collision Response, *Computer Graphics International Proceedings*, IEEE Computer Society, pp 257-266, 2000.

[15] P. Volino, N. Magnenat-Thalmann, Comparing Efficiency of Integration Methods for Cloth Simulation, *Computer Graphics International Proceedings*, IEEE Computer Society, 2001.