

Interactive Cloth Editing and Simulation in Virtual Reality Applications for Theater Professionals

D. Koutsonanos¹, K. Moustakas², D. Tzovaras¹ and M. G. Strintzis^{1,2}, Fellow, IEEE

¹ Informatics and Telematics Institute
P.O. Box 361, 57001 Thessaloniki, Greece
Phone: +30 2310 464160, Fax: +30 2310 464164
² Aristotle University of Thessaloniki
54006 Thessaloniki, Greece
e-mail:{koutsona,moustak,tzovaras,strintzi}@iti.gr

Abstract

This paper presents an efficient framework for the interactive simulation and editing of clothing over avatars. Specifically, a very efficient costume designer application is proposed based on new developments in cloth modeling simulation and animation. To animate cloth a hierarchical simulator is used, which speeds up the simulation without losing in accuracy. Enhanced interaction potentiality is provided to the application user, using haptic devices with 6DOF for interaction. Furthermore, the user is capable of editing the cloth mesh, thus generating alternative versions of it, which could be more appropriate for a costume designer application. The mesh editing is performed so as to assure the stability of the simulation and the realism of the application. In order to validate the usability of our application, a number of case studies were conducted, which showed that the present system is efficient, user friendly and realistic.

Keywords: Interactive cloth modeling, cloth editing, virtual reality applications, theater professionals, haptic interaction.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three dimensional graphics and realism, I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling, I.3.8 [Computer Graphics]: Applications

1. Introduction

Interactive virtual simulation is a very challenging research area and admits numerous applications. Applications for educational purposes [NFTS04], for training disabled people [TNF*04] and for entertainment are just some of them. The progress in the area of cloth simulation during recent years regarding the development of fast and stable numerical algorithms, physically correct simulations and visually pleasant results, offers both realism and speed. Thus, applications for interactive cloth simulation are now feasible and can offer to professionals an alternative means of designing cloth.

An interactive cloth editing system should address several 'hot' topics and technological trends in order to develop an enhanced VR tool that provides solutions, targeting the

needs of a wide area of professionals. The application should provide the user with all necessary means for designing, modifying and interacting with 3-D simulations that stimulate his creativity. In [KSFS03] an interactive cloth simulation virtual environment was presented with 6DOF for interaction. The virtual environment is rendered into a stereoscopic display. The user is wearing shutterglasses and interacts with the scene through a virtual panel. In [CC03] an integrated tool for garment designing and fitting to avatars was introduced. The 3D object editing method described in [DHQ04] describes an efficient haptic interface, based on partial differential equations, used for editing static 3D objects.

The interaction with the costume and the realistic simu-

lation of cloth deformations require advanced cloth modeling and simulation techniques. Deformable object modeling and simulation is a very interesting and challenging research area, while it is indispensable in applications that need realistic simulations. Physically-based cloth animation has been under extensive study for the last decades. Particle systems were introduced in cloth simulation with satisfactory results [TW88] and offered to the researchers an alternative point of view. Specific problems like the non-realistic behavior of springs to over-elongate when the stiffness constants are low is addressed in [Pro95], while faster and more efficient collision detection and response algorithms were presented in [Pro97]. An efficient implicit integrator was introduced in [BW98], which was a very promising idea. In fact, implicit integration is much more stable than explicit even if the time step of the simulation is high, because it performs filtering of the force field, which avoids the hazardous effects of instant maxima of the force vector to the simulation. The trade-off for this increment of speed, caused by the increased time step, and stability, is the simulation accuracy. A precise mathematical formulation for the implicit integration based simulators was presented in [VMT00], where researchers tried also to mitigate the low accuracy of implicit integration by using a hybrid simulator, consisting of both explicit and implicit integration. Other issues like collision detection and response have been addressed by many researchers [HB00, BFA02] with relatively good results concerning the simulation accuracy and visual quality.

The main problem in these applications remains the computational effort needed to perform the calculations, which inhibits any attempt of real time simulation, when trying to animate large textiles. In [VB02], adaptive mesh approaches were introduced to emphasize in areas with high detail, like folds and wrinkles. In [DSB99, MDDB00], an alternative integration model was presented, that combines the characteristics of implicit integration without the need to solve large linear systems. Unfortunately, all the above methods decrease visual quality in order to increase the speed of the simulation.

The present paper proposes an enhanced costume designer application, which makes use of haptic interaction and haptic feedback in order to increase the realism of the interaction. The application displays the results of fitting specific e-garments to a specific avatar and is capable of generating avatar-costume animations, based on efficient cloth modeling and simulation techniques [MTS04, VMT00]. It provides the user with a number of features like browsing the costume and avatar databases, fitting garments to avatars (basic cloth friction, cloth collision, self-collision and cloth draping options are available to the user), modifying the avatar's body posture or orientation and loading a specific animation sequence to create an animation of the avatar and the costume. The most important feature of the costume designer application is that it allows the user to modify the cloth interactively using haptic techniques while the avatar

wears the costume. Thus, the costume designer can adjust and refine the costume and store the updated e-garment back to the database. The developed system utilizes a hierarchical representation of the object using quincunx sampling [MTS04]. Simulation of the mesh is performed in each level using as initial value an efficient prediction of the mesh state, generated optimally, based on the results of the simulation in the upper level, thus speeding up the simulation without losing in accuracy.

The rest of this paper is organized as follows. In Section 2 an overview of the costume designer application is presented. The hierarchical cloth modeling and simulation methods are briefly described in Section 3, while Section 4 presents the collision detection scheme. The haptic interaction and cloth editing procedures are discussed in Section 5. In Section 6 the animation system is described. Finally Sections 7 and 8 exhibit demonstrations of the application and conclusions, respectively.

2. System Overview

The overall block diagram of the system is depicted in Figure 1. As it was described above, the user has to select an avatar, a costume and an animation from the database. The costume and the animation sequence will be applied to the avatar while it wears the e-garment. In the sequel, after the user has selected the multimedia, the initialization procedure is invoked.

The initialization step involves mainly two operations: a) the hierarchical modeling of the selected cloth mesh and b) the fitting of the modelled cloth to the selected avatar. After the modeling and fitting operations are completed, the application enters its continuous operation loop. During execution time, the application invokes the appropriate modules according to the user input. Specifically, the application allows the user to select from two discrete operational modes: (a) the animation mode (Mode A in the figure), where the user can evaluate the animation of the avatar while it wears the e-garment or (b) the cloth editing mode (Mode B in the figure), where the user can edit the cloth while the avatar wears it. The application, depending on the active operation mode, performs the required steps:

- In mode A, the animation module is invoked, which updates the avatar mesh according to the selected animation. After the avatar mesh is updated, the cloth simulator updates the cloth mesh responding to the new collisions that occur between the cloth mesh and the newly moved avatar.
- In mode B, the main application module invokes the haptic interaction module which communicates with the haptic devices (either CyberGrasp or Phantom Desktop) and retrieves the haptic information. This information is used by the haptic interaction module so it can provide feedback to the haptic devices and is also sent to the model update module. The latter, using the haptic information,

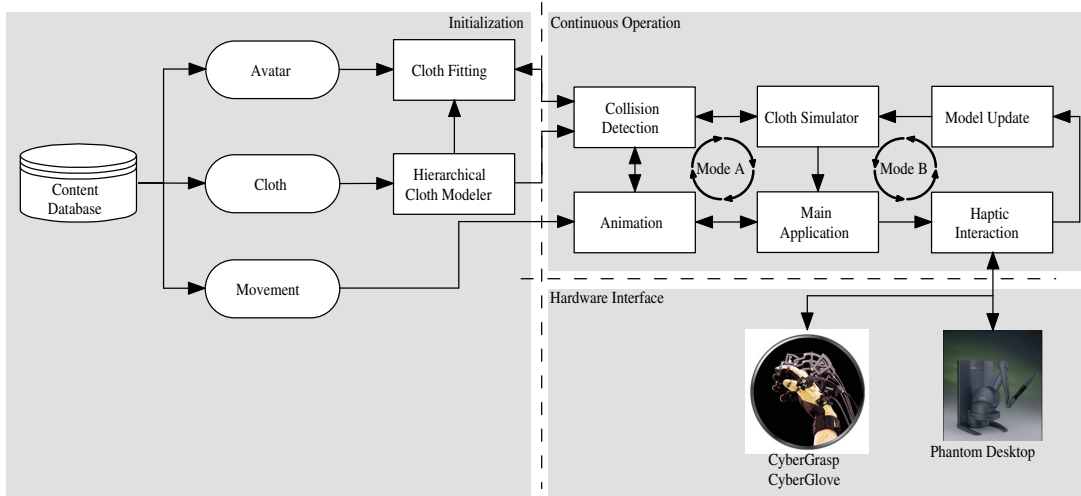


Figure 1: System Diagram.

calculates the applied force that the user exerts and updates the mesh of the cloth. Next, the cloth simulator module after retrieving the updated cloth mesh, checks for any collision detection with the avatar and renders the output.

The application can not be operated under both modes at the same time; when the user is playing the animation, the cloth editing feature is inactive and vice versa. It should be clarified though, that after any cloth editing operations, the edited cloth mesh is also used in the animation phase. This way, the user can edit the cloth that he/she needs and evaluate the behavior of the edited cloth while the avatar that wears it performs various animations. In the following all modules shown in Figure 2 constituting the costume designer application will be presented in detail.

3. Hierarchical representation and simulation of cloth

Different approaches have been presented in the past for the modeling of cloth. They can be divided into two major categories. Those using particle models [TW88] and those using continuous models [MDM*02]. The present framework utilizes particle models to represent cloth.

3.1. Particle Model

In the last decade the mass-spring model has been used excessively from the researchers to represent the cloth's surface. This model is assumed to be the most promising model in terms of computational complexity compared to continuous models also presented in the past. The mass-spring model consists of a lattice of mass particles linked with massless linear springs. Particles react not only to the spring forces but also to physical forces applied to them such as gravity, viscosity, damping, wind resistance, friction etc.

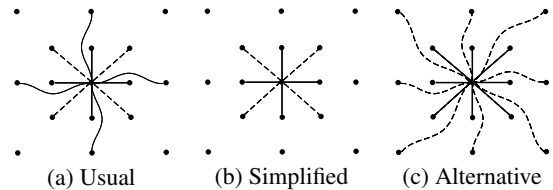


Figure 2: Mass-spring models

In most applications, the model illustrated in Figure 2a is assumed, where each particle is linked to its 12 nearest neighbors. Structure, shearing and bending springs are the straight, dashed and curved lines, respectively. However, some variations of this approach have been also presented. The model in Figure 2b does not include bending springs. Figure 2c illustrates an alternative model, which uses eight springs (straight) to react in shrinking and eight (dashed) to react in stretching. The non-linear nature of this model may cause problems of convergence and stability. Methods utilizing non-linear mass-spring models have also been presented in the past [MDM*02]. The proposed method assumes the general linear particle model illustrated in Figure 2a.

3.2. Hierarchical representation of the mesh

The cloth mesh is modelled hierarchically in the present framework. In particular, quincunx sampling is utilized to build the pyramid, due to its isotropic nature [MTS04]. Figure 3 illustrates three consecutive levels of the generated pyramid.

Initially the simulation variables, position, velocity and force vectors, of each pyramid level, are derived from the

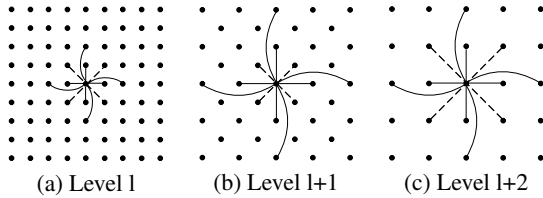


Figure 3: Dynamic model

base level, by simply assigning to them the values of the corresponding mass points. This would be sufficient if explicit integration was used. However, the present framework uses implicit integration, where the force derivatives are mandatory for the simulation. Thus, in each level, a full mass-spring model has to be implemented. There are two ways of generating them, dependent on the objective of the hierarchical implementation.

If the sampled model should have the same behavior with the original in terms of their rest state, then a static equivalent model has to be developed for the sampled model. Its parameters (stiffness coefficients, etc.) have to be determined in order to minimize the error at the rest position of the models.

On the other hand, if the sampled model should have the same behavior with the original in terms of their state at the next time step, then a dynamic equivalent model has to be developed. Its parameters should be selected to minimize the error between the two models at the next time step. Thus the derivative matrices, have to be as similar as possible.

In the present case, the sampled meshes are not simulated as independent entities and there is no concern about their rest state. What is important is their dynamical behavior at each time instant to be as close as possible to the state of the lower level. Therefore a dynamical equivalent model is adopted for the sampled meshes. In Figure 3 the adopted equivalent dynamic model is illustrated, where same springs (straight, dashed, curved lines) represent the same spring coefficient values.

3.3. Hierarchical simulation of cloth

A schematic description of the proposed simulation algorithm (for a N-level pyramid) is presented in Figure 4.

The system update is obtained using implicit integration [BW98]. The mesh is initially simulated at the top level of the pyramid. The simulation output, consisting of vertex positions, velocities etc., is filtered [MTS04, TS00, KTS01], and used as initial value for the simulation of the lower level, thus speeding up its convergence. The procedure is repeated iteratively until the bottom level is reached. The overall procedure tends to produce about 23% faster simulations [MTS04] when compared to non-hierarchical implicit integration based methods [VMT00, BW98].

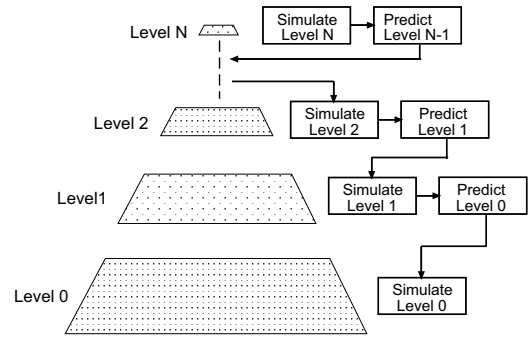


Figure 4: Simulation pyramid

```

Level = N
while Level ≠ -1 {
  if Level = N
    InitialStateVector[Level]=0
    StateVector[Level]=
      =Simulate(Level,InitialStateVector[Level])
  if Level ≠ 0
    InitialStateVector[Level-1]=Filter(StateVector[Level])
  Level - -
}

```

Figure 5: Hierarchical simulation pseudocode

Figure 5 illustrates the pseudocode of the hierarchical simulation procedure.

4. Collision detection module

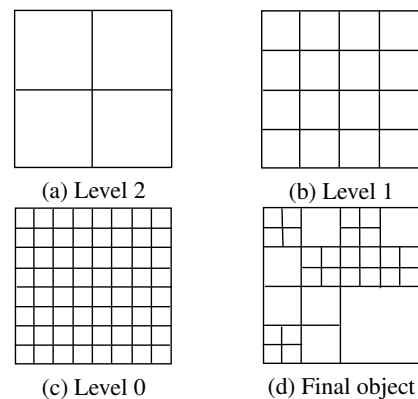


Figure 6: Collision hierarchy

Already existing collision detection schemes [BFA02], which proved to be very efficient, were extended and integrated in the hierarchical simulator in order to provide collision response to the animation procedure. The idea is that if the object is smooth in some regions, there is no need to use a dense mesh to describe it. Thus curvature criteria described

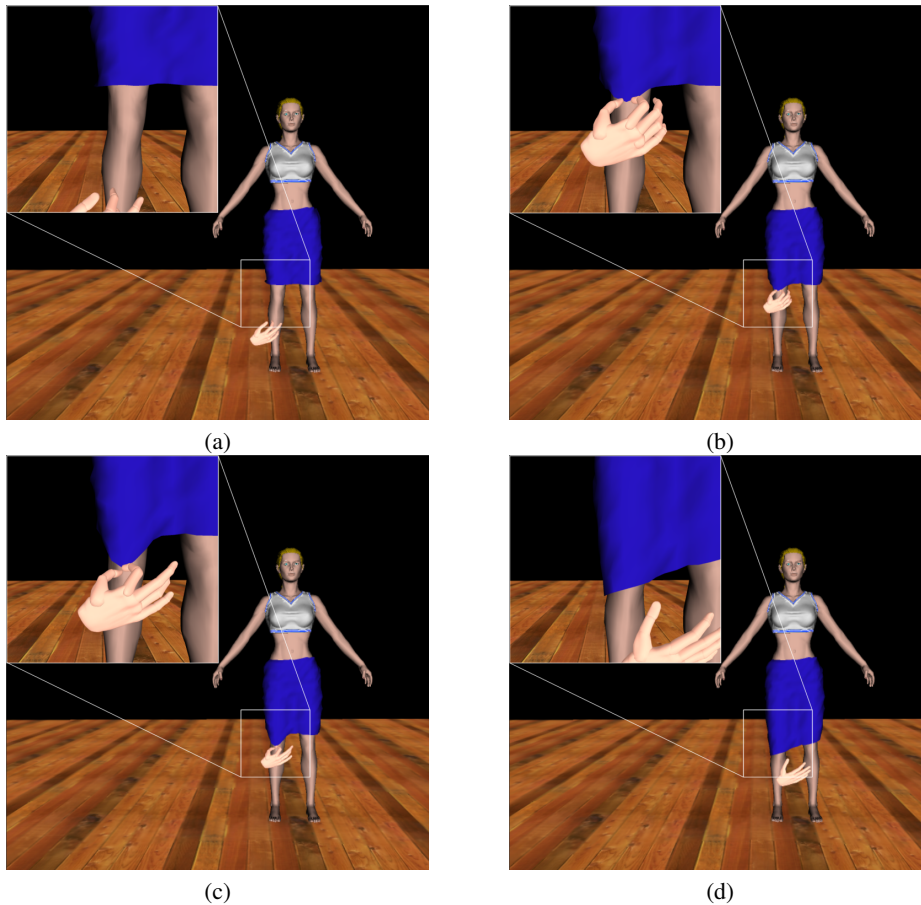


Figure 7: *Costume Editing: (a) Cloth mesh (b) Grabbing of a particle (c) Editing the mesh and (d) Modified cloth mesh*

in [Pro97], with very strict curvature thresholds, are used to evaluate bending.

In particular, the mesh is divided into $N \times N$ regions as illustrated in Figure 6c. If very low curvature conditions are met for a region of the bottom level 0 (Figure 6c), it is sampled and merged with the adjacent three regions, if the curvature conditions are met for them too, a new region in level 1 (Figure 6b) is produced. This procedure is stopped if high curvature is detected for at least one of the four adjacent regions. At the end, the mesh of Figure 6d is used as input for the collision detection procedure.

This adaptive collision detection scheme reduces the computational effort about 8%. The computational overhead of this scheme for an object with many folds and wrinkles (curvature criteria is nowhere met) is about 3%.

The method described above is perfectly accurate but remains computationally intensive. A faster collision detection procedure has also been developed [MTS] and integrated to the proposed application, which executes the calculations about 20 times faster than the previously described method,

while retaining all necessary accuracy. A brief description of this method follows.

Initially, a bounding superquadric [MTS, SB90] is generated for each segment of the humanoid body, according to the H-Anim standard. The distance of the mesh from the superquadric is mapped on the superquadric surface. The collision detection procedure checks for collision between the avatar and each point of the cloth. It is applied in two steps: The first step consists of the evaluation of the superquadric inside-outside function [MTS]. If the point lies inside the superquadric, its distance from it and the minimum distance point on the superquadric surface are evaluated. If the retrieved distance is higher than the distance of the mesh and the superquadric at the minimum distance point, collision is detected. As previously mentioned, the superquadric based collision detection needs the avatar segments to be modelled with superquadrics.

The user of the proposed costume designer application can choose on of the two prescribed methods for collision detection. If a superquadric approximation of the avatar

is available, which is the case for most avatars stored in the database, the superquadric based collision detection is preferable.

5. Haptics Driven Cloth Editing

As previously stated, the system supports cloth editing using haptic devices, although it is possible to use standard pointing devices. The costume editing operation is depicted in Figure 7.

A specific portion of the cloth is depicted in its initial position in Figure 7a. The user grabs an area of the cloth using the haptic devices as it is depicted in Figure 7b. While the particle is selected, the user can perform any fitting or stretching to the cloth mesh using the haptic devices. An instance of this operation is depicted in Figure 7c. When the user decides to end this procedure, he/she releases the selected particle and the cloth stabilizes in a final position after physical modeling is applied once more to the cloth model. The cloth mesh after the editing procedure is depicted in 7d. This way, the user can freely manipulate the cloth mesh in any direction, if no collision of the selected particle with the avatar occurs. This ensures that the cloth will not penetrate the avatar mesh.

In the following the basic elements constituting the mesh editing procedure, which consist of the haptic interaction and the cloth mesh selection and update, are analyzed.

5.1. Haptic Interaction

Haptic perception incorporates both kinesthetic sensing (i.e. of the position and movement of joints and limbs) and tactile sensing (i.e. through the skin). The development of haptic, kinesthetic and tactile devices offers a new dimension of realism to virtual environments and these developments offer further potential applications for advanced multimedia environments. Force feedback haptic technologies are integrated to the costume designer application to enhance user interactivity with the virtual environment.

The costume designer application provides haptic interaction to the user using two off-the-shelf haptic devices, which are Phantom [Sen] and CyberGrasp [Imm] and are illustrated in Figure 8.

Phantom is a grounded force feedback haptic device with one point of contact, six degrees of freedom and fixed workspace. Phantom provides access to very accurate tactile information such as friction, texture, etc. Cybergrasp is a force-feedback haptic interface for enabling CyberGlove users experience near-to-realistic force-feedback and to perceive with their hand the volume of computer generated objects. CyberGlove is a low-profile, lightweight glove with flexible sensors, which can measure the position and movement of the fingers and wrist. In order to obtain the position of the whole hand, the MotionStar WirelessTM tracker

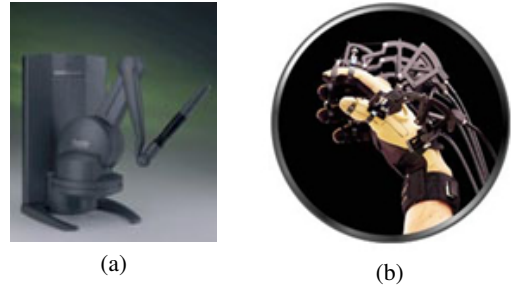


Figure 8: Haptic devices. a) Phantom, b) Cybergrasp

is used [Asc]. If the user interacts with the application using the Cybergrasp, a virtual hand is displayed in the virtual environment. If he/she uses the Phantom a cone appears inside the scene. The user of the Costume Designer Application is able to apply transformations to the cloth by selecting an area of it with the haptic devices. After selecting the area, as described in the sequel, the user is able to freely manipulate the cloth by stretching and moving it, while examining in real time the resulting transformations of the cloth mesh according to its physical parameters. The force feedback for the haptic devices is calculated from the physical model of the cloth as described in the sequel.

5.2. Cloth area selection and manipulation

The selection of an area of a cloth mesh, which will be manipulated seems to be a relatively easy and straightforward procedure. However, real-time interactive applications have to be very carefully designed in order to guarantee simulation stability and realism. In particular, if the selection procedure just selects a particle of the mesh, the resulting motion will be non-realistic. If an area of the mesh is selected to avoid non-realistic effects and the force of the manipulation is applied to this area only, the simulation stability will be jeopardized [VMT01] due to the high difference of the force field on the boundaries of the selection area.

To avoid such undesirable effects, the developed system initially selects all particles, which lie inside a predefined bounding sphere of the pointing device (haptic device, mouse) as illustrated in Figure 9. The user is capable of altering the size of the sphere so that he can choose the desired accuracy.

These particles are manipulated by the force \mathbf{F}_h exerted from the haptic device in the following way. Force \mathbf{F}_h is assumed to be exerted only in the center of the bounding sphere. To all vertices inside the sphere the force $\mathbf{F}(\mathbf{x})$ is exerted:

$$\mathbf{F}(\mathbf{x}) = \mathbf{F}_h \cdot \vec{g}(\mathbf{x}) \quad (1)$$

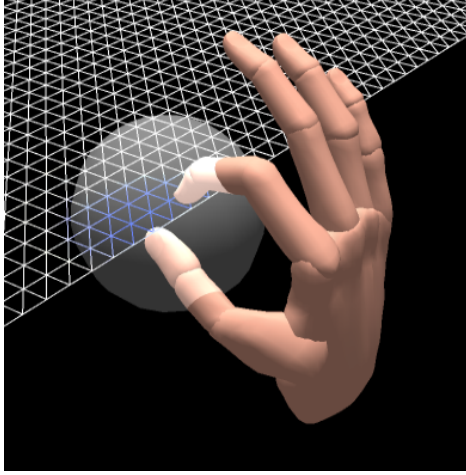


Figure 9: Cloth selection

$g(\mathbf{x})$ is the multivariate 3D gaussian distribution,

$$g(\mathbf{x}) = \frac{1}{\sqrt{2\pi C_x}} e^{\sqrt{(\mathbf{x}-\mathbf{m})^T C_x (\mathbf{x}-\mathbf{m})}} \quad (2)$$

where \mathbf{x} is the position of the vertex, \mathbf{m} the center of the bounding sphere and C_x the covariance matrix of \mathbf{x} . C_x is assumed to be diagonal and its diagonal elements are equal so as to produce a symmetrical normal distribution, because there should be no difference on the exerted force for different directions. Thus, $C_x = s \cdot \mathbf{I}$, where $s = r/2$ and r is the radius of the bounding sphere. The distribution $g(\mathbf{x})$ is normalized in the interval $[0, 1]$ so that the force in the center of the bounding sphere is \mathbf{F}_h . The resulting filter $\tilde{g}(\mathbf{x})$ is used in equation (4).

$$\tilde{g}(\mathbf{x}) = e^{\sqrt{(\mathbf{x}-\mathbf{m})^T C_x (\mathbf{x}-\mathbf{m})}} \quad (3)$$

The force value, which is fed onto the haptic devices in order to obtain force feedback, is calculated directly from the mass-spring model of the cloth. In particular, the summation of the forces applied from the mass-spring model to all particles inside the interaction sphere (Figure 9) is fed onto the haptic devices.

5.3. Model update

The model update module is responsible for the static handling of the cloth mesh, while it is edited from a user. The proposed application has two options for editing a mesh:

1. Using the haptic devices in order to fit a mesh to an avatar by performing stretching, shrinking etc. of the mesh, exactly as a human would do. When the user releases the mesh, it will return to its rest position as illustrated in Figures 10a and 10b.

2. Using the haptic devices in order to alter the structure of the mesh. I.e. to change the internal parameters of the mass-spring model of the cloth so that the edited cloth will not return to its initial position after the user releases it, but will stay approximately in the position where user left it as illustrated in Figures 10c and 10d.

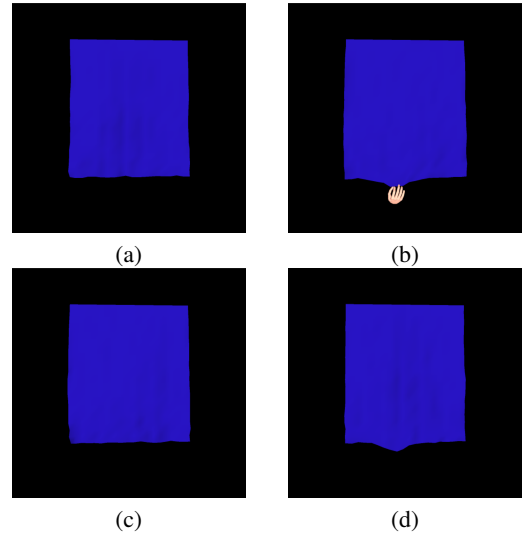


Figure 10: Model update. a) Cloth before editing, b) Cloth during editing, c) Released cloth without altering the structure (case 1), d) Released cloth with altered structure (case 2)

The first option requires only proper handling in order to add the exerted forces into the simulator. The second option is provided to the costume designer, because there are many cases where he wants to alter the structure of the mesh. The most common editing task is the stretching or compression of the mesh. If the designer stretches a part of the cloth, the exerted forces deform the mesh. However, in that case the stretching energy of the mesh is high and as a result the cloth will return approximately to its starting position when left from the user. In order to avoid this effect, the user is capable to eliminate the internal stretch energy [BW98] of the processed cloth area by simply pressing a button whenever necessary. Thus, the stretched area will not return to its initial position, since the internal structure parameters of the mesh (spring constants, spring natural lengths etc.) are altered in order to correspond to the new edited cloth mesh. More specifically, the force exerted to a mass point i yields from equation 4:

$$\mathbf{f}_i = \sum_j k_{ij} \left(\frac{\vec{x}_j \vec{x}_i}{\|\vec{x}_j \vec{x}_i\|} - l_{ij} \frac{\vec{x}_j \vec{x}_i}{\|\vec{x}_j \vec{x}_i\|} \right) \quad (4)$$

where \mathbf{x}_i is the position of particle i and k_{ij} and l_{ij} are the constant and rest length, respectively, of the spring connecting particles i and j . If a spring is stretched or compressed,

a force is exerted to its adjacent particles. This force can be eliminated if the rest length l_{ij} of the spring is set equal to the norm distance between particles i and j . This technique is used in the costume designer application in order to alter the structure of the mesh (case 2). I.e. the user has the ability to eliminate the forces exerted to the particles inside the bounding sphere (Figure 9).

6. Animation

One of the basic needs of a costume designer is the evaluation of the designed garment while a model wears it and performs some movements. This practise always reveals minor deficiencies and allows the designer to perform optimizations in a latter stage to the cloth pattern. Towards that cause the costume designer application supports the simulation of the cloth in terms of automatic fitting to the animating avatar.

The animation of the avatar is compliant to the H-Anim [Han] standard. The user, has a variety of animations in his/her disposal, which are classified in two categories: a) basic and b) specialized animations. The first category includes basic movements like stand, walk, run, jump and kick. The second category involves area-specific animations and in the presented application is mainly focused in choreographes. It must be noted that in each instance of the animation, the cloth follows the avatar movement according to the physical simulation and collision procedures described in Sections 3 and 6. The animation of an avatar wearing a costume and performing a classical "walk" animation is depicted in Figure 11.

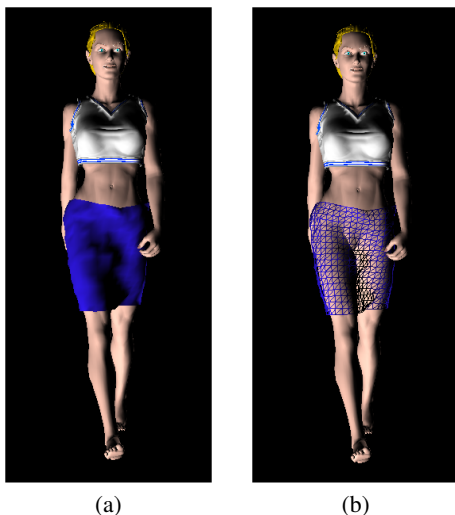


Figure 11: Walking avatar a) Snapshot, b) Wireframe

7. Case Study: Costume Designer application and the VR@Theater platform

7.1. The VR@Theater project

The Costume Designer application is part of a suite of applications for theater professionals called VR@Theater. Each application facilitates the artistic work of a specific theater professional; i.e. scenographer, director, choreographer, actor and costume designer. The application suite is complemented with a central database that holds all the content that is produced and authored by the individual applications. The applications are co-operative in means of content exchange through the central database. For example, the scene that a scenographer has produced can be imported by the other applications, the choreography that has been authored with the choreographer tool can be imported into the costume designer application (as a movement) and so on. The applications along with the central database form the VR@Theater platform and are the result of the project VR@Theater.

VR@THEATER focuses on the creation of a fully equipped design environment to offer to theater professionals various opportunities and significantly simplify their work. Specifically, scenographers have access to a large database of 3D stage items and are able to simulate the actual stage environment. Costume designers are able to select e-garments from a costume database, "fit" them on avatars, i.e. "virtual" representations of persons, and visualize results or even create animations. Directors are able to produce dynamic simulations by importing a "virtual" stage design, importing avatars (possibly wearing e-garments) and defining a "script", i.e. avatar animations, light and sound effects. Choreographers are able to enter the desired choreography moves with a user-friendly user interface, or even to record the movements of a specific performer using motion capture (mocap) techniques. Finally, actors can use an "actor-rehearsal" application, to rehearse their role and then watch a "virtual" preview of their act performed by his avatar reproducing his exact words, actions and facial expressions, within a virtual reconstruction of the actual stage environment.

7.2. The Costume Designer Application

The costume designer application is depicted in Figure 12. The user interface includes two major parts; the modeling and animation properties module and the 3D rendering module. The first module, includes the required controls for setting the parameters for the physical modeling and animation. More specifically, the user interface of the parameters module includes:

- Three scrolling lists of thumbnails of all the costumes, avatars and animations that are stored in the database.
- The parameter fields that describe the physical properties of the simulation.
- The controls to start, pause and stop the animation.

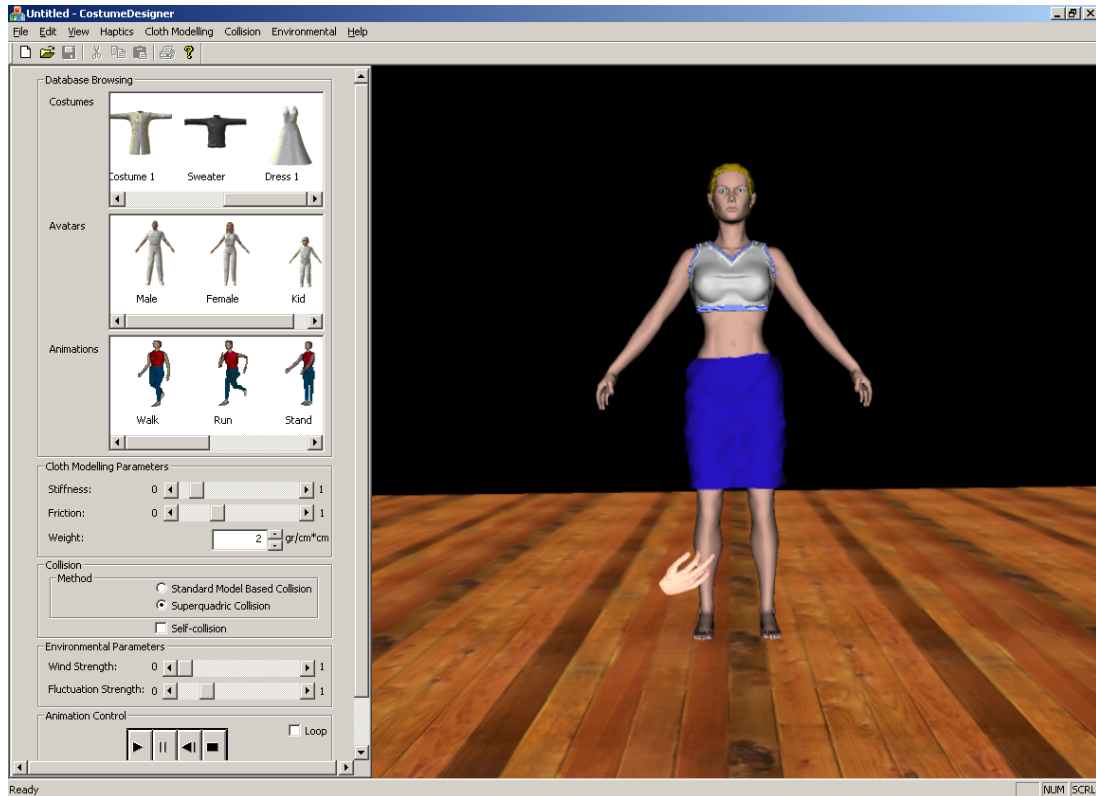


Figure 12: The Costume Designer Application Environment.

- The appropriate menus to connect with the haptic devices.

The 3D rendering window performs the required rendering operations and animations in real time. It is based on the open source 3D graphics engine OGRE (Object Oriented Graphics Rendering Engine) [Ogr].

The costume designer application has been evaluated at the Center of Higher Education in Theater Studies in Greece. The users characterized the application as very friendly and easy to use. Most of them claimed that the most impressive and useful feature of the costume designer application was the cloth editing with haptics. They mentioned that the interactive altering of the structure of a mesh using haptic feedback was very realistic and useful when designing costumes, because it is possible to generate different clothes by altering the structure of existing meshes.

8. Conclusions

In this paper an efficient framework for interactive cloth simulation was presented. A hierarchical simulator was utilized in order to effectively simulate clothing. The user of the application can interact with the virtual environment using haptic devices. The interaction can be performed either in order

to help the fitting of the cloth to the avatar, exactly as a human does when wearing a cloth, or in order to edit the cloth meshing and alter its structure. Experimental evaluation has shown that the application is user friendly and appropriate for cloth editing, if the cloth mesh is not excessively altered.

9. Acknowledgements

This work was supported by the Greek Secretariat for Research and Technology funded project VR@THEATER.

References

- [Asc] Ascension technologies corp., motionstar wirelessTM installation and operation guide.
- [BFA02] BRIDSON R., FEDKIW R., ANDERSON J.: Robust treatment of collisions, contact and friction for cloth animation. *ACM Trans. Graph. (SIGGRAPH Proc.)* 21 (2002), 594–603.
- [BW98] BARAFF D., WITKIN A.: Large steps in cloth simulation. In *Comput. Graph (SIGGRAPH Proc.)* (1998), vol. 32, pp. 106–117.
- [CC03] CHITTARO L., CORVAGLIA D.: 3d virtual

- clothing: from garment design to web3d visualization and simulation. In *Proceedings of the eighth international conference on 3D Web technology* (2003), pp. 73–84.
- [DHQ04] DUAN Y., HUA J., QIN H.: Hapticflow: Pde-based mesh editing with haptics. *Computer Animation and Virtual Worlds 15* (2004), 193–200.
- [DSB99] DESBRUN M., SCHRODER P., BARR A.: Interactive animation of structured deformable objects. In *Proceedings of the 1999 Conference on Graphics Interface* (1999), pp. 1–8.
- [Han] *ISO/IEC FCD 19774, Humanoid Animation (H-Anim)*.
- [HB00] HOUSE D., BREEN D.: *Cloth Modeling and Animation*. A.K. Peters Ltd., 2000.
- [Imm] Immersion technologies inc., virtual hand suite 2000 user & programmers guide, <http://www.immersion.com/3d/support/documentation.php>.
- [KSFS03] KECKEISEN M., STOEVS S. L., FEURER M., STRASSER W.: Interactive cloth simulation in virtual environments. In *Proceedings of the IEEE Virtual Reality 2003* (2003), IEEE Computer Society, p. 71.
- [KTS01] KOMPATSIARIS I., TZOVARAS D., STRINTZIS M.: Hierarchical representation and coding of surfaces using 3-d polygon meshes. *IEEE Trans. on Image Processing* 10, 8 (August 2001), 1133–1151.
- [MDDB00] MEYER M., DEBUNNE G., DESBRUN M., BARR A.: Interactive animation of cloth-like objects in virtual reality. *The Journal of Visualization and Computer Animation* 12, 1 (2000), 1–12.
- [MDM*02] MULLER M., DORSEY J., MCMILLAN L., JAGNOW R., CUTLER B.: Stable real-time deformations. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2002), pp. 49–54.
- [MTS] MOUSTAKAS K., TZOVARAS D., STRINTZIS M.: Efficient hierarchical simulation of cloth and deformable objects using a novel collision detection algorithm. *IEEE Trans. on Visualization and Computer Graphics*, submitted.
- [MTS04] MOUSTAKAS K., TZOVARAS D., STRINTZIS M.: Fast Hierarchical Simulation of Cloth and Deformable Objects using an Optimal Pyramidal Representation. In *International Conference on Computer Animation and Social Agents, CASA2004* (Geneva, Jul. 2004), pp. 163–170.
- [NFTS04] NIKOLAKIS G., FERGADIS G., TZOVARAS D., STRINTZIS M. G.: A mixed reality learning environment for geometry education. In *Lecture Notes in Artificial Intelligence* (June 2004), Springer Verlag.
- [Ogr] Ogre (object oriented graphics rendering engine), www.ogre3d.org.
- [Pro95] PROVOT X.: Deformation constraints in a mass-spring model to describe rigid cloth behavior. In *Proc. of Graphics Interface '95* (1995), pp. 147–154.
- [Pro97] PROVOT X.: Collision and self-collision handling in cloth model dedicated to design garment. In *Proc. of Graphics Interface '97* (1997), pp. 177–189.
- [SB90] SOLINA F., BAJCSY R.: Recovery of parametric models from range images: The case for superquadrics with global deformations. *IEEE Trans. Pattern Anal. Mach. Intell.* 12, 2 (1990), 131–147.
- [Sen] Sensable technologies inc., <http://www.sensable.com/>.
- [TNF*04] TZOVARAS D., NIKOLAKIS G., FERGADIS G., MALASIOTIS S., STAVRAKIS M.: Design and implementation of haptic virtual environments for the training of visually impaired. *IEEE Trans. on Neural Systems and Rehabilitation Engineering* 12, 2 (June 2004), 266–278.
- [TS00] TZOVARAS D., STRINTZIS M.: Optimal construction of reduced pyramids for lossless and progressive image coding. *IEEE Trans. on Circuits and Systems II* 47, 4 (April 2000), 158–166.
- [TW88] TERZOPOULOS D., WITKIN A.: Physically based model with rigid and deformable components. *IEEE Computer Graphics and Applications* (Dec. 1988), 41–51.
- [VB02] VILLARD J., BOROUCHE H.: Adaptive meshing for cloth animation. In *Proceedings of the 11th International Meshing Roundtable (IMR2002)* (2002), pp. 243–252.
- [VMT00] VOLINO P., MAGNENAT-THALMANN N.: Implementing fast cloth simulation with collision response. In *Computer Graphics' International* (Jul. 2000), pp. 257–266.
- [VMT01] VOLINO P., MAGNENAT-THALMANN N.: Comparing efficiency of integration methods for cloth simulation. In *Computer Graphics International 2001* (2001), pp. 265–274.