3rd Workshop in Virtual Reality Interactions and Physical Simulation "VRIPHYS" (2006)
C. Mendoza, I. Navazo (Editors)

Deformable Hybrid Approach for Haptic Interaction

Maximo G. Mero[1,2] and A. Susin[3]

[1] Dept. Llenguatges i Sistemes Informàtics. Universitat Politècnica de Catalunya (UPC), Barcelona (Spain)
[2] Dept. Matemáticas e Informática. Universidad de Carabobo (Venezuela)
[3] Laboratori de Simulació Dinàmica (LABSID), Dept. Matemàtica Aplicada 1, UPC-Barcelona (Spain)

Abstract

A new hybrid approach for deformable models is presented here and carried out in a virtual reality environment, achieving real time performance with haptic interactions. Our implementation consists in using two approaches for the deformable model. The deformation is modelled using simultaneously a Finite Element Method and a Mesh Free Method.

With this Mesh Free method, particles are used to simulate large deformations in the volume region near the surface of the object. The remaining internal volume of the object can be modelled employing a coarse mesh using the Finite Element Method.

Categories and Subject Descriptors (according to ACM CCS): I.3.8 [Computer Graphics]: Applications

1. Introduction

The interactive applications developed in a real virtual environment are important in the field of medicine or engineering. In this way, there are tools, such as surgery simulators, that allow simulation and animation of volumetric deformable objects which can be manipulated in this virtual environment and with a real time haptic performance. One of the main characteristics of these simulations is the dynamic interaction between the deformable model and the possible external forces acting on it.

Our deformable model is based on physical properties. It is essentially built on techniques presented in computer graphics and mechanical engineering literature [RD89], [DJAK87], [DA88]. The dynamic behavior of the volumetric 3D deformable model is based on linear elastic mechanics.

For the simulation of deformable objects, one of the most usual numerical techniques is the Finite Element Method (FEM). O'Brien *et al.* [JJ99] and Debunne *et al.* [GMMA01] are good references about the continuous deformation problem in terms of FEM.Although [JJ99] is focused on the study of fractures of rigid materials, the elastic model can be used for simulating a major variety of deformable objects. It is known that [GMMA01] built a model for the human liver with the same finite element formulation. Also, a haptic model based on FEM has been implemented by Cotin *et al.* [SHN99] for surgical applications.

Topics like precision and speed need to be properly balanced when their performance in a virtual reality environment is the principal application goal. The adequate solution for combining both characteristics is to build a multiresolution model [GMMA01].The structure of such a model is organized in different layers from coarse to finer mesh. Computational accuracy depends on base functions and the number and size of the elements. When an external force is applied to the model in a delimited zone, the finer mesh is activated. The other resolutions are used to animate the model based on the distance from the force location and internal forces.

The different multiresolution models differ according to the relation of two consecutive mesh levels. This can be obtained from a refinement of the previous given mesh level or from a completely independent mesh (just meshing the same volume). This relation is critical in the transition zone where the two different meshes are activated. The main drawback of this approach is the complex data structure needed for efficient simulation.

Another recent deformation model is based on considering a discretization of the continuous material by using

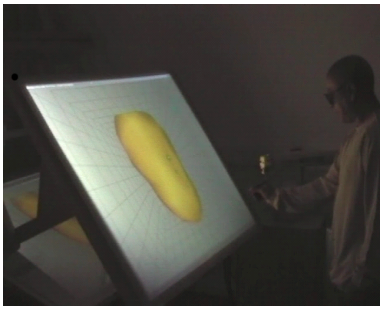

Figure 1: *The in-house workbench platform with the Fokker Haptic Master device.*

particles. In this approach, the material is a system of particles that includes deformation properties and their relations with neighboring ones. Smooth Particle Hydrodynamics (SPH) was first introduced by physicists for accurate simulation of fluid dynamics [J.M92]. Then Desbrun and Cany [MM99] presented these techniques to be used in the field of computer graphics. Recent papers [DPM*99], [MRA*04], [S.F01], present new applications of the MFM (see [TH03] for a classification of MFM methods).

In the works mentioned above, FEM or MFM is used exclusively to build 3D deformable objects. Besides, the multiresolution approaches use only FEM to model the different levels of detail (LODs). The idea of coupling two different models is developed in [CDA00] for FEM and mass-spring. Our proposal defines a hybrid model to take advantage of both FEM and MFM methods ([MRA*04], [WMJ94]). Thus, the inner region of the object is modelled using a FEM coarse mesh that will be a fast model and the domain between the surface and the internal mesh is built using an MFM. The particles zone will be the one reacting directly to user interaction and the deformation obtained will be more realistic than the one achieved only using coarse linear finite elements. FEM and MFM are simultaneously activated in order to maintain the material properties along the deformation. In addition, a transition zone between the two models has to be stated and a similar formalism for both methods is used in order to assure continuity in the simulation.

Our application has been implemented in an in-house designed workbench with a Fokker Haptic Master device for interaction (see figure 1). With this instrument the forces are computed according to the reaction of the model when the user interacts with the virtual object.

The paper is organized as follows: section 2 presents the elasticity theory background of deformable models.Section 3 shows the numerical approach for FEM and MFM models. Section 4 presents the hybrid model. Section 5 shows the dynamics. Section 6 shows a haptic model application. Results and conclusions are presented in the last sections.

2. Elasticity Theory Basics

The deformable model is based on continuum mechanics [Y.F65] in which the first assumption assumes that scale effects are significantly greater than the material composition scale. Therefore, the behavior of the material molecules or particles can be first modelled as continuous media. When an elastic body is deformed, internal restoring forces are produced. The strain at one point describes the change of distances between nearby points. The stress is associated with the forces that maintain the deformation. In a linear elastic body, the stress is linearly related to the strain.

Let $u = (u_1, u_2, u_3)^T \in \Omega \subset \mathbb{R}^3$ be a vector that denotes a location in the material coordinate frame. We can map the material location with the deformed one by $u \rightarrow u + \mathbf{x}(u)$, where the deformation of the material is defined by the function $\mathbf{x}(u) = (\mathbf{x}_1(u), \mathbf{x}_2(u), \mathbf{x}_3(u))$. The Jacobian of this map is

$$\mathbf{J} = \mathbf{I}_{3\times 3} + \nabla \mathbf{x}^T \tag{1}$$

where $\mathbf{I}_{3\times 3}$ is the identity and

$$\nabla \mathbf{x} = \begin{bmatrix} \frac{\partial \mathbf{x}_1}{\partial u_1} & \frac{\partial \mathbf{x}_1}{\partial u_2} & \frac{\partial \mathbf{x}_1}{\partial u_3} \\ \frac{\partial \mathbf{x}_2}{\partial u_1} & \frac{\partial \mathbf{x}_2}{\partial u_2} & \frac{\partial \mathbf{x}_2}{\partial u_3} \\ \frac{\partial \mathbf{x}_3}{\partial u_1} & \frac{\partial \mathbf{x}_3}{\partial u_2} & \frac{\partial \mathbf{x}_3}{\partial u_3} \end{bmatrix} \tag{2}$$

The strain tensor is associated with the change of length of the material. Consider a pair of points u and $u + du$ in the undeformed body, where $du = (du_1, du_2, du_3)$ is a small displacement. These are deformed to locations $u + \mathbf{x}(u)$ and $u + du + \mathbf{x}(u + du)$ by the function $\mathbf{x}(u)$.

The deformed configuration material vector in terms of the deformation gradient tensor and the reference configuration material vector is $d\mathbf{x} = \nabla \mathbf{x} du$.

The tensor

$$\varepsilon = \frac{1}{2}\left[(\nabla \mathbf{x})^T(\nabla \mathbf{x}) - \mathbf{I}_{3\times 3}\right] \tag{3}$$

is known as Green's strain tensor ε, and is used to measure large deformations of the material, as is pointed out in [GMMA01], [JJ99]. It is a 3×3 symmetric matrix and can be written by components as

$$\varepsilon_{ij} = \frac{1}{2}\left[\left(\frac{\partial \mathbf{x}}{\partial u_i} \cdot \frac{\partial \mathbf{x}}{\partial u_j}\right) - \delta_{ij}\right], \quad i, j = 1, 2, 3. \tag{4}$$

Here δ_{ij} is the Kronecker's delta function.

If we substitute (1) in the equation (3), we obtain the Green-Saint-Venant strain tensor in terms of displacement gradients

$$\varepsilon = \tfrac{1}{2}\left[(\nabla \mathbf{x})^T + (\nabla \mathbf{x}) + (\nabla \mathbf{x})^T(\nabla \mathbf{x})\right] \tag{5}$$

thus, by components

$$\varepsilon_{ij} = \frac{1}{2}\left[\frac{\partial \mathbf{x}_i}{\partial u_i} + \frac{\partial \mathbf{x}_j}{\partial u_j} + \sum_k \frac{\partial \mathbf{x}_k}{\partial u_i}\frac{\partial \mathbf{x}_k}{\partial u_j}\right]. \tag{6}$$

If we consider only small deformations, the second term in the equation (6) is ignored, resulting the Cauchy's infinitesimal strain tensor

$$\varepsilon_{ij} = \frac{1}{2}\left[\frac{\partial \mathbf{x}_i}{\partial u_i} + \frac{\partial \mathbf{x}_j}{\partial u_j}\right] \tag{7}$$

The Green's strain tensor ε defined in the equation (4) is the one employed in our work because it only measures deformations. It is invariant with respect to translation applied to $\mathbf{x}(u)$ and vanishes when the material is not deformed.

In linear elasticity theory a linear relation between the stress $\sigma^{(l)}$ and the strain ε is assumed

$$\sigma^{(l)} = \mathbf{C}\varepsilon \tag{8}$$

The linear operator $\mathbf{C}$ denotes a symmetric, positive definite elastic tensor that encodes the elastic relationship between $\sigma^{(l)}$ and ε. For isotropic materials it holds that

$$\sigma^{(l)} = 2\mu\varepsilon + \lambda \mathbf{tr}(\varepsilon)\mathbf{I}_{3\times 3} \tag{9}$$

where the positive coefficients μ and λ are called *Lamé* coefficients, μ represents the rigidity of the material while λ measures its ability to preserve volume. Here **tr** denotes the *trace* of a matrix, $\mathbf{tr}(\varepsilon) = \sum_i \varepsilon_{ii}$.

The viscous stress associated with the strain rate improve realism because it define the material's damping properties. The strain rate tensor measures the rate at which the strain ε is changing through the time

$$\frac{\partial \varepsilon}{\partial t} = \left(\frac{\partial \mathbf{x}}{\partial u_i}\cdot\frac{\partial \dot{\mathbf{x}}}{\partial u_j}\right) + \left(\frac{\partial \dot{\mathbf{x}}}{\partial u_i}\cdot\frac{\partial \mathbf{x}}{\partial u_j}\right) = \nu_{ij}, \quad i,j = 1,2,3 \tag{10}$$

where $\dot{\mathbf{x}} = \dfrac{\partial \mathbf{x}}{\partial t}$. Then the viscous stress is defined as

$$\sigma^{(\nu)} = 2\psi\nu + \phi \mathbf{tr}(\nu)\mathbf{I}_{3\times 3}. \tag{11}$$

The total stress tensor is obtained adding both the elastic and viscous stress $\sigma = \sigma^{(l)} + \sigma^{(\nu)}$,where ϕ and ψ are parameters that control how fast the material looses kinetic energy.

3. Numerical approach

This section introduces two numerical methods used to obtain approximation solutions to the partial differential equations that arise in scientific and engineering applications (FEM and MFM).

3.1. Finite Element Method

In this approach, the domain Ω is divided in small regions called finite element and the continuous function $\mathbf{x}(u)$ is expanded in basis functions N_j defined in the j element

$$\mathbf{x}(u) = \sum_j c_j N_j(u). \tag{12}$$

In this work, we have discretized the entire domain by using a tetrahedra mesh with linear shape functions. We introduced four nodes numbered $\mathbf{u}_1, \cdots, \mathbf{u}_4$ at the vertices of the element.

Imposing the usual interpolation conditions $N_j(\mathbf{u}_k) = \delta_{jk}, k = 1,2,3,4$, where $\mathbf{u}_k = (u_1^k, u_2^k, u_3^k)^T$ is a node in the undeformed configuration, one obtains

$$N_j(u) = \frac{D_{k,l,m}(u)}{C_{j,k,l,m}}, \quad (j,k,l,m) \text{ a permutation of } (1,2,3,4) \tag{13}$$

where

$$\begin{aligned} D_{k,l,m}(u) &= det\begin{bmatrix} [u] & [\mathbf{u}_k] & [\mathbf{u}_l] & [\mathbf{u}_m] \\ 1 & 1 & 1 & 1 \end{bmatrix} \\ C_{j,k,l,m} &= det\begin{bmatrix} [\mathbf{u}_j] & [\mathbf{u}_k] & [\mathbf{u}_l] & [\mathbf{u}_m] \\ 1 & 1 & 1 & 1 \end{bmatrix} \end{aligned} \tag{14}$$

and $det[\,]$ is the determinant of the 4×4 matrix.

A linear shape function is built and is uniquely determined on each face of the elements by placing nodes at the three vertices on the face. This guarantees continuity of bases constructed from the shape functions. The restriction of the solution $\mathbf{x}$ to a linear element e is

$$\mathbf{x}(u) = \sum_{j=1}^{4} c_j N_j(u). \tag{15}$$

Tetrahedral coordinates provide a natural way to define the linear shape functions within an element. The coordinates of a world point P can be expressed in tetrahedral coordinates $P = (\zeta_1, \zeta_2, \zeta_3, \zeta_4)$, where

$$\zeta_1 = \frac{V_{P234}}{V_{1234}}, \quad \zeta_2 = \frac{V_{P134}}{V_{1234}}, \quad ,\zeta_3 = \frac{V_{P124}}{V_{1234}}, \quad \zeta_4 = \frac{V_{P123}}{V_{1234}}. \tag{16}$$

Here V_{jklm} is the volume of the tetrahedron with vertices at j,k,l,m that can be computed as

$$V_{jklm} = \frac{1}{6}\left[(\mathbf{u}_k - \mathbf{u}_j)\times(\mathbf{u}_l - \mathbf{u}_j)\right]\cdot\left[\mathbf{u}_m - \mathbf{u}_j\right] \tag{17}$$

The transformation from world to tetrahedral coordinates (material coordinates) can be expressed as

$$\begin{bmatrix} u \\ 1 \end{bmatrix} = \begin{bmatrix} u_1^1 & u_1^2 & u_1^3 & u_1^4 \\ u_2^1 & u_2^2 & u_2^3 & u_2^4 \\ u_3^1 & u_3^2 & u_3^3 & u_3^4 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \zeta_1 \\ \zeta_2 \\ \zeta_3 \\ \zeta_4 \end{bmatrix} \equiv [\mathcal{B}(\mathbf{u}_i)]\,[\zeta] \tag{18}$$

where $\zeta = [\zeta_1 \quad \zeta_2 \quad \zeta_3 \quad \zeta_4]^T$ and $[\mathcal{B}(\mathbf{u}_i)]$ the matrix associated with the element nodes. A node $\mathbf{u}_i$ has a position in the material coordinates, a position in the world coordinates, $\mathbf{x}^i = (\mathbf{x}_1^i, \mathbf{x}_2^i, \mathbf{x}_3^i)^T$, and a velocity, $\dot{\mathbf{x}}^i = \mathbf{v}^i = (\mathbf{v}_1^i, \mathbf{v}_2^i, \mathbf{v}_3^i)^T$ in world coordinates. Thus, these tetrahedral coordinates may also be used to interpolate both the nodes and their velocities in the material coordinates

$$\begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix} = [\ x_1 \quad x_2 \quad x_3 \quad 1\ ]^T = \left[\mathcal{B}(\mathbf{x}^i)\right][\zeta] \tag{19}$$

$$\begin{bmatrix} \dot{\mathbf{x}} \\ 1 \end{bmatrix} = [\ v_1 \quad v_2 \quad v_3 \quad 1\ ]^T = \left[\mathcal{B}(\mathbf{v}^i)\right][\zeta] \tag{20}$$

$[\mathcal{B}(\mathbf{x}_i)]$ and $[\mathcal{B}(\mathbf{v}_i)]$ are used to determine position and velocity respectively. Let $[\Gamma] = [\mathcal{B}(\mathbf{u}_i)]^{-1}$, therefore we can write the position and velocity inside the element e in term of the material coordinates. Thus, the position and velocity of $\mathbf{x}(u)$ is:

$$\begin{bmatrix} \mathbf{x}(u) \\ 1 \end{bmatrix} = [\mathcal{B}(\mathbf{x}^i)][\Gamma] \begin{bmatrix} u \\ 1 \end{bmatrix}, \qquad \dot{\mathbf{x}}(u) = [\mathcal{B}(\mathbf{v}^i)][\Gamma] \begin{bmatrix} u \\ 1 \end{bmatrix}. \tag{21}$$

The strain and stress tensors (equations 4 , 9, and 11) are calculated using these expressions. The internal force is computed for each element using the total stress defined in the previous section

$$F^i = F^i_{(e)} = \frac{vol^{(e)}}{2} \sum_{j=1}^{4} \mathbf{x}^j \sum_{k=1}^{3} \sum_{l=1}^{3} \Gamma_{jl}\Gamma_{ik}\sigma_{kl}, \tag{22}$$

where $vol^{(e)}$ is the element volume.

3.2. Mesh Free Method

Another approach for modelling deformation is to discretize the continuous material as a set of particles, or more precisely point samples. These methods are called meshfree since they do not explicitly store neighborhood information (connectivity) and compute the basis functions in a different way than in FEM (where the neighbors are known a priori) ([AS00], [L.L77]). The most commonly used MFM is the Smooth Particle Hydrodynamic (SPH) [L.L77], [RJ82].

To obtain a discrete formulation of the deformation, the first step is to approximate the continuous function $\mathbf{x}(u), u = (u_1, u_2, u_3) \in \Omega$ as the contribution of the corresponding weighted values using a kernel function $W(u,h)$ with compact support, that is $W(u,h) = 0$, for $\| u \| \geq 2h$. Thus

$$\mathbf{x}(u) \approx \int_{\Omega} \mathbf{x}(y) W(u-y,h)dy. \tag{23}$$

The displacements, velocity and acceleration are interpolated using the kernel basis function W defined in (24). In the following, a short hand notation is used: $W^i = W(u-u^i,h)$, $W^{ij} = W(u^i-u^j,h)$ and $\mathbf{x}^j = \mathbf{x}(u^j)$. The superscript j refers to the neighboring particle of i, $u^i = (u_1^i, u_2^i, u_3^i)$.

A potential function frequently used for SPH simulations (see [BT57], [JBM77]) is the Lennard-Jones potential given by

$$W(r_{ij}) = \kappa \left[\left(\frac{1}{r_{ij}}\right)^3 - \left(\frac{1}{r_{ij}}\right) \right] \tag{24}$$

where r_{ij} is the distance between two particles i and j, and κ is the energy required to overcome the cohesion of particles (see figure 2)

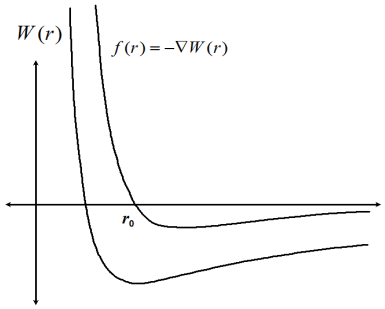

Figure 2: *Lennard-Jones's potential function $W(r)$ and the associated force function $f(r)$.*

In this work we use a linear approximation of the Lennard-Jones potential which fulfils kernel requirements. The volume integral in the equation (23) is approximated by a volume-weighted sum at particle i as

$$\begin{aligned} \mathbf{x}(u^i) &= \sum_{j \in I^h_{u^i}} \mathbf{x}(u^j) W(r^{ij},h) \triangle V^j \\ &= \sum_{j \in I^h_{u^i}} \mathbf{x}(u^j) N^h_j(u^i) \end{aligned} \tag{25}$$

where the neighborhood particles are defined by $I^h_u = \{j\ ;\ |u^j - u| \leq h\}$ and $\triangle V^j$ is the differential volume associated to particle j, and the sum is the total particles within the smoothing length h of the particle i. The volume of a particle is computed from its mass m and the density ρ at the particle as $\triangle V^j = \frac{m^j}{\rho^j}$. For a uniform initial grid, the mass

associated with a node m^j can be computed from the total number of particles within a known mass volume. The $N_j^h(u^i) = W(r^{ij},h)\triangle V^j$ functions are the base functions associated with the SPH interpolation.

The strain and stress tensors are calculated from the same expressions (4), (9) and (11) used for the FEM in the previous subsection. In an analogous way of the FEM approach, the differential form of the linear momentum balance can be expressed in a weak formulation

$$\int_\Omega \delta\mathbf{x}\rho\ddot{\mathbf{x}}dV - \int_\Omega \sigma\cdot\nabla\delta\mathbf{x}dV + \int_{\partial\Omega}\tau\delta\mathbf{x}dS = 0. \tag{26}$$

where $\tau = \sigma\cdot n$ is the traction on the surface with normal n. To obtain a weighted residual statement for the discrete system, the equation (25) is substituted in equation (26) and minimized at each node i obtaining a set of N discretized equations,

$$\begin{aligned}&\sum_{j\in I^h_{ui}} \rho^j W^{ij}\triangle V^i\triangle V^j\ddot{\mathbf{x}}\\ &= \sum_{j\in I^h_{ui}}\sigma^j\nabla W^{ij}\triangle V^i\triangle V^j - \sum_{j\in I^h_{ui}}\tau W^{ij}\triangle V^i\triangle S^j\end{aligned} \tag{27}$$

The equation (27) can be assembled to obtain a mass matrix and force vector,

$$M\ddot{\mathbf{x}} = F^{ext} - F^{int}. \tag{28}$$

The individual components of the mass matrix are given by $M^{ij} = \rho^j W^{ij}\triangle V^i\triangle V^j$. The components of the internal force vector are given by

$$\begin{aligned}F^i_{int} &= \sum_{j\in I^h_{ui}}\sigma^j\nabla W^{ij}\triangle V^i\triangle V^j\\ &= \sum_{j\in I^h_{ui}} m^j m^i\left(\frac{\sigma^j}{\rho^j\rho^i}\right)\nabla W^{ij}\end{aligned} \tag{29}$$

4. Hybrid Model

Now a hybrid model can be defined by using the two models introduced in the previous sections. In the above section, the MFM equations were derived using an approach that parallels the derivation of the classical ones used for the FEM. In the MFM approach the particles are treated as elements with only one node and whose connectivity must be determined for each time step. A kernel sum approximation is used to compute the velocity gradient and stress divergence. The fact that the same constitutive relations are used for particles and finite elements allows to assume the same material properties in both methods.

The interpolation function $\mathbf{x}(u)$ in Ω, with $\Omega\in I\!R^3$ is built using both the FEM and MFM, the domain must include a set of $\{u^i\}_{i\in I_u^{N_i}}$ nodes with their associated base functions $N_i(u)$, which possess the information of the contributions of the finite element in $\mathbf{x}^f(u)$ as an approximation of the $\mathbf{x}(u)$ function. Therefore

$$\mathbf{x}^f(u) = \sum_{i\in I_u^{N_i}}\mathbf{x}(u^i)N_i(u). \tag{30}$$

A set of particles $\{u^j\}_{j\in I^h_{uj}}$ also exists with associated base functions $N_j^h(u)$ which possess the information of the contributions of the free mesh method in the approximation $\mathbf{x}^h$ of function $\mathbf{x}(u)$

$$\mathbf{x}^h(u) = \sum_{j\in I_u^h}\mathbf{x}(u^j)N_j^h(u). \tag{31}$$

In the most general case, the Ω domain is the union of two non disjoint sub-domains, $\Omega = \Omega_e\cup\Omega_p$. The Ω_e sub-domain is defined from the FEM base functions N_i as

$$\Omega_e = \{u\in\Omega \quad / \quad \exists \quad i\in I^{N_i};\quad N_i(u)\neq 0\},$$

Analogously, the Ω_p sub-domain is defined for the MFM base functions N_j^h as

$$\Omega_p = \{u\in\Omega \quad / \quad \exists \quad j\in I^h;\quad N_j^h(u)\neq 0\},$$

In the region Ω_e where only the finite element is present, the standard approximation of finite element (see figure 3) is chosen $\mathbf{x}(u)\approx\mathbf{x}^f(u)$. For the region where only the particles have influence, the standard approximation chosen is $\mathbf{x}(u)\approx\mathbf{x}^h(u)$.

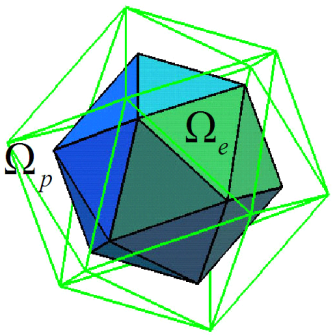

Figure 3: *Qualitative representation of the decomposition in two different regions. In the Ω_e region the finite elements methodology is employed. In the Ω_p region mesh free method is used.*

The coupling of the two models, as explained in the following sections, is achieved from the displacement of the

nodes and particles through the contact surface between Ω_e and Ω_p.

5. Dynamical Time Evolution

A deformable model is described by the position $\mathbf{x}(u)$, velocities $\dot{\mathbf{x}}(u)$ and accelerations $\ddot{\mathbf{x}}(u)$ associated with their mass elements m and forces F, as functions of material coordinates. We consider classical Newtonian dynamics for the time evolution

$$\dot{\mathbf{x}} = \mathbf{v}, \qquad \dot{\mathbf{v}} = \frac{F}{m}. \tag{32}$$

The total force $F = F_{int} + F_{ext}$ where F_{int} is the internal elastic force inside the object and F_{ext} is the external force applied on the object. A simple Euler's explicit scheme can be used to computed a numerical solution of (32) giving acceptable results in terms of speed and accuracy. Nevertheless, we use a Verlet scheme [PT01] in order to obtain a more stable behavior

$$\mathbf{x}_{t+\triangle t} = \mathbf{x}_t + \left(\mathbf{x}_t - \mathbf{x}_{t-\triangle t}\right) + \frac{F}{m}\triangle t^2 \tag{33}$$

where $\triangle t$ is the time step and $\mathbf{x}_t$ is the position of the particle i in the time t. This numerical scheme is used with the MFM model employing the internal elastic force (29).

In the same way, to increase stability, we use a *semi-implicit* scheme for the FEM model following the work of [GMMA01]. The *semi-implicit* scheme can be stated as

$$\begin{aligned}
\tilde{\mathbf{v}}_{t+\triangle t} &= \mathbf{v}_t + \frac{F}{m}\triangle t, \\
\mathbf{v}_{t+\triangle t} &= \left(\mathbf{I}_{3\times 3} - \frac{(\triangle t)^2}{m}[\mathbf{J}_F(x)]\right)^{-1}\tilde{\mathbf{v}}_{t+\triangle t}, \\
\mathbf{x}_{t+\triangle t} &= \mathbf{x}_t + \mathbf{v}_{t+\triangle t}\triangle t.
\end{aligned} \tag{34}$$

where t stands for the present time simulation.

A first approximation of the force applied at a node when it moves is given by Jacobian matrix force $[\mathbf{J}_F(x)]$ associated with each node. This matrix can be precomputed when the object is in rest position assuming that its neighbors are fixed, and considering that it remains constant during the simulation. The internal elastic force with this model is defined in (22)

6. Haptic Hybrid Model Application

In this application, it is assumed that the deformable object is a volumetric body with a topological structure equivalent to the region between two concentric ellipsoids. All in all, our goal is to obtain a haptic representation of a heart data set to simulate deformation and force feed-back during user interaction. These data, from our point of view, consist only of two surface meshes: internal and external. Thus, we have built a hybrid model defining two inner volume regions between both surfaces: region Ω_e, the most inner zone, tetrahedralized and simulated with FEM, and region Ω_p, the nearest to the outer surface, filled with particles and simulated with MFM.

In order to achieve a fast simulation when the user touches the model (assuming only local deformation), only the closest part to the contact point in region Ω_p is activated. Therefore, as a preprocess, for each external surface point p_s it is necessary to determine a set of particles belonging to region Ω_p, associated with this point (see figure 4). For this reason, for each p_s a neighbor set of points $nv(p_s) = \{p_1^s, \cdots, p_l^s\}$ is defined which includes the vertices of the surface triangle containing p_s and, eventually, their first-connected vertices (depending on the mesh size).

To build this application we considered the boundary of $nv(p_s)$ defined by triangles and the pair of p_i, p_j adjacent points. The segment $\overline{p_i - p_j}$ will be projected in a radial direction towards the object's mass center. Therefore, the projected segments will determine a polygonal curve $C(p_s)$ in the outer boundary of region Ω_e. Then, a volumetric region $RV(p_s)$ included in Ω_p (filled with particles) is defined by the regions inside $C(p_s)$ and $nv(p_s)$ during the projection process.

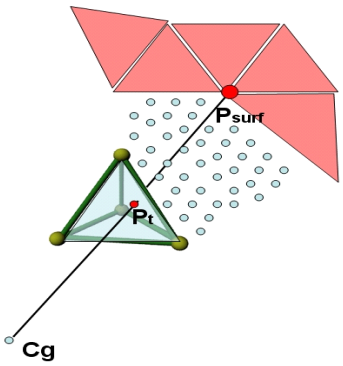

Figure 4: *Projection of external mesh points. Particles are included between the surface of the object and the inner region defined by finite elements.*

6.1. Dynamics of the hybrid model

To simulate (using MFM) only in the volumetric region $RV(p_s)$, some restrictions of particle motion have to be imposed.The boundaries of $RV(p_s)$ are considered as elastic regions that can not be crossed therefore, a similar approach to [SET02] is utilized. Thus, the active particles using MFM are identified as inner or boundary particles. To continue

this construction, the $RV(p_s)$ volume is embedded in a voxelized bounding box $B(p_s)$. The voxels corresponding to the $RV(p_s)$ and inner boundaries are identified. The active particles start in these voxels.

Dynamics of the inner particles are defined from the MFM model but in order to enforce a non-rigid preservation of the boundary region and prevent the particles from moving away, it is necessary to define a gradient vector field, figure (5). Therefore, at each voxel a gradient vector is defined and could be seen pointing to the nearest voxel in the boundary. This gradient vector is used to define a force which is applied to the boundary particles and also to the ones which traverse a boundary voxel.

The coupling between both models in each time step is achieved from conservation of Linear Momentum. The displacement of MFM particles that cross a boundary voxel through a tetrahedron face is averaged by using their baricentric coordinates and applied to the FEM nodes in this face. Analogously, the displacement of the faces of the tetrahedra moves all the activated MFM particles that are in the boundary voxels using the normal face direction.

To obtain a more global motion of the object surface when no particles are activated, we consider a direct connection between the external surface and the internal FEM model. As a preprocessor, each external vertex is linked to an internal tetrahedron face using its projection in the mass center direction. This defines a dumped spring-like connection which is used to move the external surface far from the activated zone, according to the FEM motion.

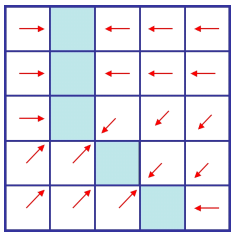

Figure 5: *Gradient vector field defined to preserve $RV(p_s)$ boundary (in blue).*

7. Results

As a first approach for testing the finite element method, we built an ellipsoidal model (Ellipsoids Model) shown in figure (6) consisting of two concentric ellipsoids. The interior of the inner ellipsoid is empty and the volume between these two ellipsoids is divided in 188 tetrahedra elements with 276 nodes. This model has been used as the interior of the virtual object described in the previous section, see figure (8). After some initial tuning, the elasticity parameters, time simulation in frame per seconds (f/s), number of elements (e) and particles (p) are shown in table I .

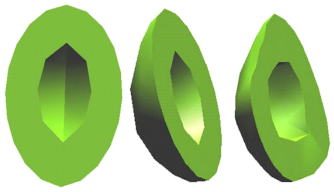

Figure 6: *Ellipsoids Model. Transversal section of the two concentric ellipsoids at rest (left and center) and the deformable model after an external force is applied on it (rigth).*

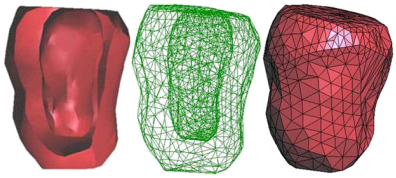

Figure 7: *Full FEM Model. An actual left ventricle heart model filled with tetrahedra. The right picture show the internal and external surfaces of the deformable object. The center picture show its internal tetrahedral structure. The left picture show a external surface of the deformable object.*

For our second example (Full FEM Model) shown in figure (7), we have used real data from a human patient. We have filled with tetrahedra the whole region between the two surfaces corresponding to the inner and outer surface of the left ventricle of the patient. In this example, the total number of tetrahedra is 1352 with 353 nodes and the obtained frame rate per second is about 35f/s. Although this result is adequate for realistic visual response effects, it is not enough for haptic interaction.

The hybrid left ventricle model of the heart (Hybrid Model) presented in this paper is shown in figure (8). The region defined between the tetrahedra mesh and the exterior surface of the object does not have a uniform thickness. For this reason, in this case, the amount of particles to be activated for a haptic action ranges between 50 and 600, being 200 the average amount of activated particles for the interaction. The total amount of particles initialized in the preprocess step in the region Ω_p is 27988. Using a personal computer with a Pentium IV 2.8GHz processor, a suitable haptic frame rate is obtained, even in the worse case, with an interaction response of 330 frames per second.

In figure (9) is shown the sequence of four frame images from an animation of our hybrid model when an external force is applied to the virtual left ventricle representation.

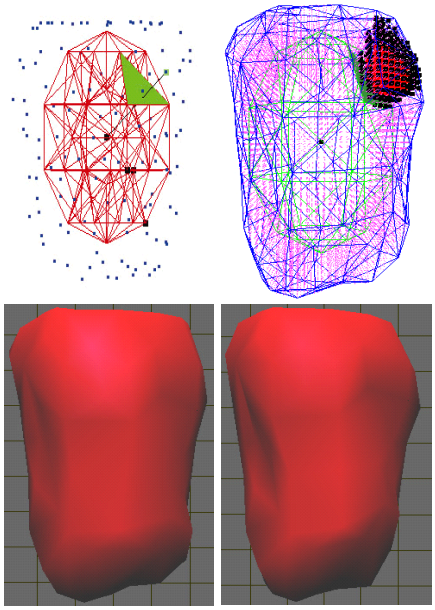

Figure 8: *Hybrid model. Internal deformable model of tetrahedra and activated particle region (left). Surface visualization at rest and after interaction (right).*

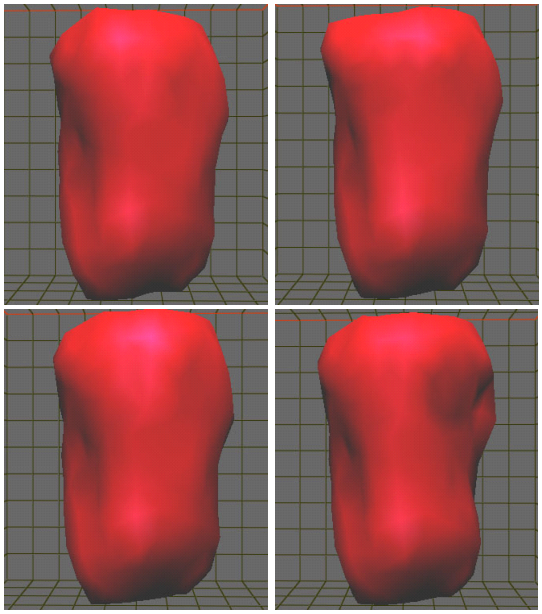

Figure 9: *Frames obtained from an animation when an external force is applied to the Hybrid model.*

Table 1: *Parameter values used in simulations*

	Ellipsoids Model	Full FEM Model	Hybrid Model
λ	5.5	0.04	5.0
μ	2.5	1.6e-3	2.0
Φ	9.0e-5	1.0e-4	9.0e-5
Ψ	9.0e-5	1.0e-4	9.0e-5
Running time	980 f/s	35 f/s	(330-600)f/s
Elements-Particles	188 e	1352 e	188 e; 200 p

8. Conclusions and Future Work

We have presented a new hybrid deformable model suitable for haptic interaction. Our approach combines the known methods of FEM and MFM in contrast with other multiresolution strategies based on FEM. We have also shown that high frame rates can be achieved with this hybrid methodology better than the ones obtained with only a finer tetrahedra FEM mesh.

This method has been implemented in an virtual reality environment, see figure (10), in which the user can interact with the model obtaining force-feedback response in real time. We have applied this hybrid methodology to construct a volumetric object which fits an actual left ventricle of a human heart. This application is embedded in a more general framework where a complete medical application, including actual human left ventricle SPECT images, surface extraction and volume representation, is being designed.

We are working now in improving both efficiency and deformation sensibility using different size partitions on the particles zone. Another possible future work could be to exploit that both linear FEM and particle methods are well suited for implemetation on the graphics processing unit (GPU). Faster computation times would simplify the subdivision strategy used for the MFM region.

Acknowledgments

This research is partially supported by the TIN2004-08065-C02-01, and FISs 03/0102 and the Univ. of Carabobo.

References

[AS00] A.HUERTA, S.FERNÁNDEZ: Enrichment and coupling of the finite element and meshless methods. *Int. Jour. for Numerical Methods in Engineering 48* (2000), 1615 – 1636. 11

[BT57] B.ALDER, T.WAINWRIGHT: Phase transition for a hard sphere system. *J. Chem. Phys 27*, 5 (1957), 1208–1209. 11

[CDA00] COTIN S., DELINGETTE H., AYACHE N.: A hybrid elastic model allowing real-time cutting, deformations and force-feedback for surgery training and simulation. *The Visual Computer 16*, 8 (2000), 437–452. 9

[DA88] D.TERZOPOULOS, A.WITKIN: Physically based models with rigid and deformable components. *IEEE*

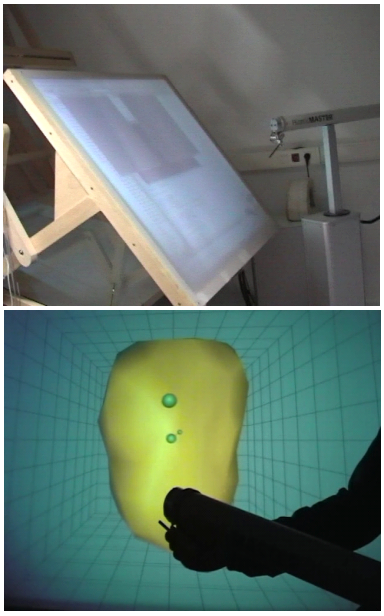

Figure 10: *Hybrid Model. Two images of the simulator running in the workbench with the Fokker Haptic Master device.*

Computer Graphics and Aplications 8, 6 (1988), 41–51. 8

[DJAK87] D.TERZOPOULOS, J.PLATT, A.BARR, K.FLEISCHER: Elastically deformable models. *ACM, Proceedings of SIGGRAPH 87 4* (1987), 205 – 214. 8

[DPM*99] D.STORA, P.AGLIATI, M.CANI, F.NEYRET, J.GASCUEL: Animating lava flows. In *Proc. Graphics Interface'99* (1999), pp. 203–210. 9

[GMMA01] G.DEBUNNE, M.DESBRUN, M.CANI, A.BARR: Dynamic real-time deformations using space & time adaptive sampling. In *SIGGRAPH '01 Conf. Proc.* (New York, NY, USA, 2001), ACM Press, pp. 31–36. 8, 9, 13

[JBM77] J.MCCAMMON, B.GELIN, M.KARPLUS: Dynamics of folded proteins. *Nature 267*, 1 (1977), 585–590. 11

[JJ99] J.O'BRIEN, J.HODGINS: Graphical models and animation of brittle fracture. In *SIGGRAPH'99 Conf. Proc.* (1999), 137 – 146. 8, 9

[J.M92] J.MONAGHAN: Smoothed particle hydrodynamics. *Annu. Rev. Astron. and Astrophysics 30* (1992), 543 – 574. 9

[L.L77] L.LUCY: A numerical approach to the testing of the fission hypothesis. *The Astronomical J. 82* (1977), 1013 – 1024. 11

[MM99] M.DESBRUN, M.CANY: *Space-Time Adaptive Simulation of Highly Deformable Substances*. Tech. Rep. 3829, INRIA, 1999. 9

[MRA*04] M.MÜLLER, R.KEISER, A.NEALEN, M.PAULY, M.GROSS, M.ALEXA: Point based animation of elastic, plastic and melting objects. In *Proc SCA '04 symposium on Computer animation* (2004), ACM Press, pp. 141–151. 9

[PT01] P.BATCHO, T.SCHLICK: Special stability advantages of position Verlet over velocity Verlet in multiple time step integration. *J.Chem. Phys. 15* (2001), 4019 – 4029. 13

[RD89] R.COOL, D.MALKUS: Concepts and applications of finite element analysis. In *Concepts and applications of finite element analysis*, Wiley J., Sons., (Eds.). 1989. 8

[RJ82] R.GINGOLD, J.MONAGHAN: Kernel estimates as a basis for general particle methods in hydrodynamics. *J. Comput. Physics. 46* (1982), 429 – 453. 11

[SET02] STAHL D., EZQUERRA N., TURK G.: Bag-of-particles as a deformable model. In *Proc. VISSYM '02 symposium on Data Visualisation* (2002), pp. 141–150. 13

[S.F01] S.FERNÁNDEZ: *Mesh-Free Methods and Finite Elements, friend or foe?* PhD. Universidad Politecnica de Catalunya, Spain, 2001. 9

[SHN99] S.COTIN, H.DELINGETTE, N.AYACHE: Real-time elastic deformations of soft tissues for surgery simulation. *IEEE Transactions on Visualization and Computer Graphics 5*, 1 (1999), 62 – 73. 8

[TH03] T.FRIES, H.MATTHIES: *Classification and Overview of Meshfree methods*. Tech. Rep. 2003-3, Institute of Scientific Computing Technical University Braunschweing. 2003. 9

[WMJ94] W.ATTAWAY, M.EINSTEIN, J.SWEGLE: Coupling of smooth particle hydrodynamics with the finite element method. *Nuclear Engineering and Design 150* (1994), 199 – 205. 9

[Y.F65] Y.FUNG: Linear elasticity. In *Foundations of Solid Mechanics*, Inc. P. H., (Ed.). Englewood Cliffs, N.J, 1965. 9