Journal of Engineering Research & Education
Vol. 3, 2006 (146-163)

Simulation Model for Human Tissue

K. SUNDARAJ [1] AND C. LAUGIER [2]

[1]Kolej Universiti Kejuruteraan Utara Malaysia,
School of Mechatronics, Kompleks Pusat Pengajian KUKUM,
Jalan Kangar-Arau, 02600 Jejawi - Malaysia.
Email : kenneth@kukum.edu.my

[2] INRIA Rhone-Alpes, GRAVIR Laboratory,
655 Avenue de l'Europe, 38334 St Ismier - France.
Email : christian.laugier@inrialpes.f

Received : 14 January 2006 / Accepted : 22 November 2006

ABSTRACT

We present a novel online human tissue simulation model for virtual reality assisted medical robots. Online simulation of human tissue deformation during surgical training or surgical assistance is becoming increasingly important within the medical community. Unfortunately, even classical simulation models find human tissue to be computationally too costly for online simulation. In this paper, we simplify the complex bio-mechanical nature of human tissue within reasonable limits to develop a mathematical model which can be used for online simulation. This simplification is based on two principles; volume conservation and Pascal's Principle. Volume conservation is inherent to many organs in the human body due to the high concentration of blood (almost incompressible liquid) in them. Given an externally applied force, we use Pascal's Principle to obtain the global deformation vector at each time-step during simulation.

Keywords : Virtual Reality, Medical Simulators, Finite Elements.

INTRODUCTION

Computer assisted interventions are becoming increasingly important within the medical community. Within this context, online simulation of human tissue is one application which is still considered to be a bottleneck. Such a requirement can be found in surgical simulators or surgical navigators. These systems can be considered to belong to a bigger community, also known as medical robots. With this type of robots, medical professionals improve the quality of training, consultation or intervention without risking the patients life.

But human tissue is deformable. Hence, modeling and simulating deformation becomes essential and an integral part of these medical robots. In a surgical simulator for example, human tissue is subjected to deformation under different gestures such

as palpitate, twist, drag, cut, suture, stapling, etc. So, accurate models need to be designed to realize the physical and visual consistency between them.

To achieve this, virtual reality can be used as a tool for the online simulation of deformation. In this environment, human tissue is represented by a geometrical model and a physical model. Often, the combination of the two is referred to as a numerical model. In this numerical model, since human tissue is complex physically and geometrically, it is often divided into a set of smaller elements to facilitate analysis. Over the past fifteen years or so, many numerical models that are based on the concept of discrete elements have been proposed for human tissue; the more prominent ones being mass-spring networks (MSN) [5] [17] [12] [3] [7], finite element method (FEM) [1] [13] [2], elasticity theory method (ETM) [9] [8], tensor-mass model (TMM) [4] [10] and the hybrid elasticity model (HEM) [14] [6] [15] [11].

Unfortunately, most of the above mentioned models have not been effective in modeling human tissue for online simulation. Since the simulated object is itself very complicated and computation resources limited, it is natural to consider a trade-off between physical accuracy and computation efficiency. A compromise can be done by laying more emphasis on the areas of interest, for example linear elasticity, local deformation or volume conservation. Such an adaptive scheme has been studied for each of the above mentioned models. Although progress has been made, the need for alternative models can still be considered mandatory.

In this paper, we present a novel online human tissue simulation model. This new model is called the Volume Distribution Method (VDM) [17]. We are interested in modeling and simulating deformable objects which have an elastic skin as surface and are filled with an incompressible fluid. Human tissue can be considered as such an object. Let us take the liver for example. It is composed of three major parts; an elastic skin called Capsule of Glisson as the surface, the Parenchyma which is the interior and is full of liquid (≈95% blood) and a complex vascular network designated to irrigate the liver. These observations allow us to make some soft assumptions about the behavior of the liver. For example, due to the high content of blood in the liver, it is possible to deduce that the liver is incompressible i.e. the liver conforms to a change in shape but not to a change in volume. The elastic capsule indicates a behavior that obeys some form of Hook's Law.

MATHEMATICAL FORMULATION

We begin by giving a formal description of this model. VDM is a surface based method that allows the computation of a global deformation vector produced by an external load vector. It only requires the surface to be discretized with the inside being transparent to the model. The interior of the object is assumed to be filled by some incompressible fluid. This fluid acts as the medium that transfers the change in energy experienced by the deformable object due to a change in state from equilibrium.

Notations

Consider the triangular surface mesh of a deformable object as shown in Figure 1 filled by some incompressible fluid. Let us now suppose that this surface is composed of vertices, which we will from now on refer to as nodes.

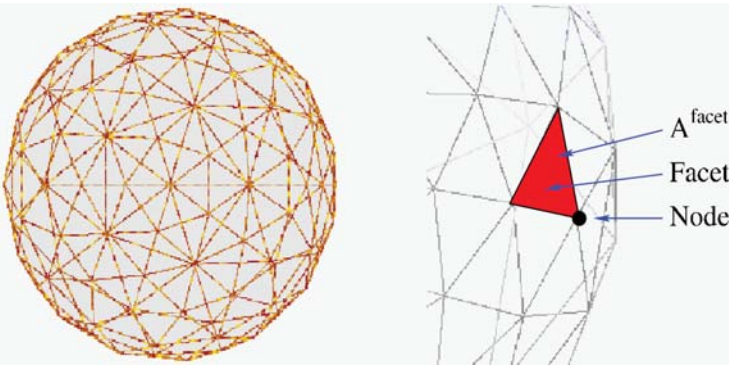

Figure 1. A deformable object with the surface discretized and a zoomed view of the surface. The interior of this object is filled with some incompressible fluid.

Let us now define the following notations for our deformable object. These notations (see Figure 2) are for a node i, on the surface of our deformable object:

- Volumic Pressure, P_{vp} - The pressure experienced by a node due to the displacement of the node.
- Volumic Tension, P_{vt} - The pressure experienced by a node due to the displacement of neighboring nodes.
- Contact Pressure, P_{cp} - The pressure experienced by a node due to contact.
- Fluid Pressure, P_{fluid} - The pressure exerted by the incompressible fluid.
- Environment Pressure, P_{ep} - The pressure exerted by the surroundings of the deformable object.
- Bulk Modulus, B_i - The ratio of pressure of node i, to the fractional volume compression of node i (equivalent to normal stress).
- Connectivity Bulk Modulus, B_{ij} - The ratio of pressure of node i, to the fractional volume compression between node i and node j (equivalent to shear stress).
- Area, A - The total surface area of the deformable object.
- Volume, V - The total volume of the deformable object.
- Distributed Area, A_i - The area assigned to a node i.
- Distributed Volume, V_i - The volume assigned to a node i.
- Facet Area, A^{facet} - The area of a facet.
- Displacement Vector, ΔL_i - The displacement vector of a node i.

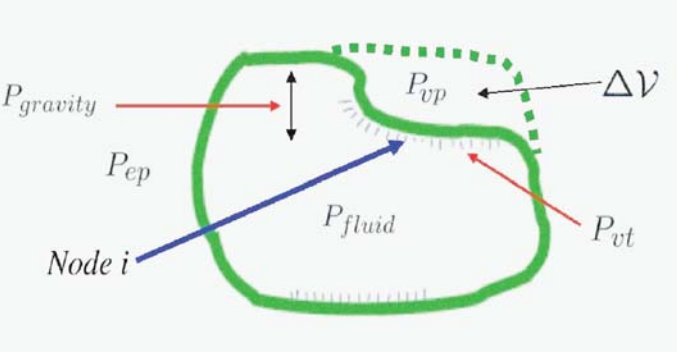

Figure 2. Volume Distribution Method (VDM) notations.

Distributed Area and Volume

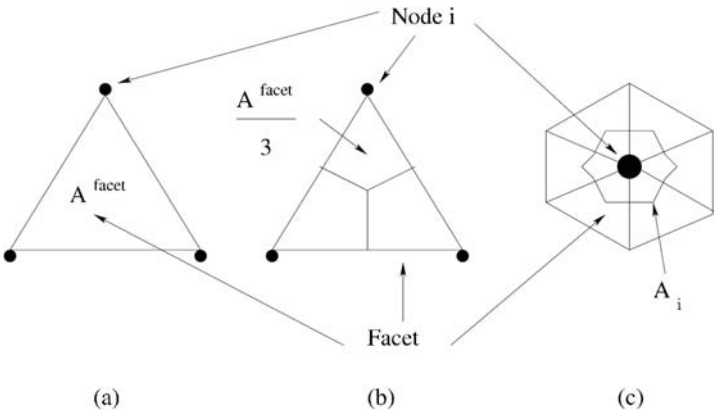

Figure 3. (a) The above facet has 3 nodes and area A^{facet}. (b) This area is distributed equally to each node, $A^{facet}/3$. (c) In the presence of neighboring facets, the sum is taken over all neighbors to obtain the distributed area of a node i, A_i.

Let us first derive A_i and V_i. Consider a deformable object with a discretized surface. Each node i is connected to j neighboring facets. These neighboring facets each have surface area A_j^{facet}. Within each facet, A^{facet} is distributed to each of the it's nodes equally. This is graphically shown in Figure 3. The distributed area for each node i is then obtained as follows:

$$A_i = \sum_j (A_{facetj} / 3) \quad (1)$$

$$\sum_I | A_i | = A \quad (2)$$

Once the area has been distributed, the volume can be distributed as well. Given the total volume of the geometrical mesh model as V and the total surface area A, we chose to distribute the volume as a function of the distributed area A_i. Then, the distributed volume for each node i is obtained as follows:

$$V_i = |A_i| V / A \quad (3)$$

$$\sum_i V_i = V \quad (4)$$

Bulk Modulus

The bulk modulus for a node B_i and the connectivity bulk modulus B_{ij} are the physical parameters of the object being modeled. These values are generally obtained from experiments or from the literature.

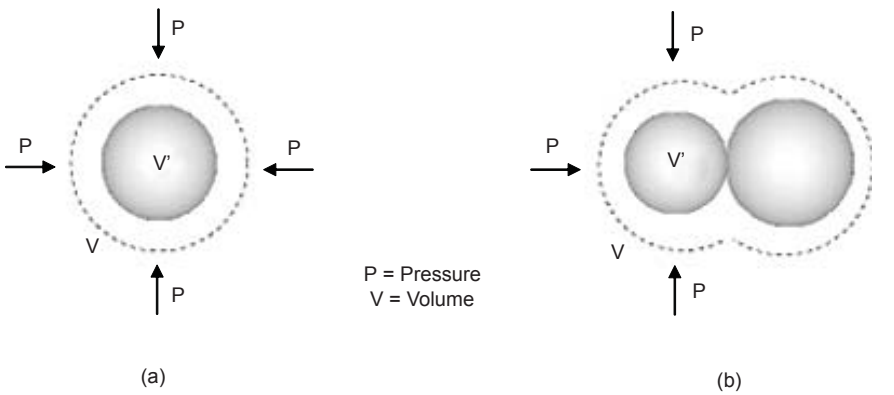

Figure 4. Physical definition of (a) bulk modulus and (b) connectivity bulk modulus.

The physical definition of bulk modulus can be understood by considering a piece of material of volume V such as the one shown in Figure 4(a). Given a change in pressure acting on this piece of material, a change of volume will be observed in this material. Let the final volume be denoted as V'. Then, the bulk modulus B can be defined as follows:

$$B = \Delta P/(\Delta V/V) \tag{5}$$

where $\Delta V=V-V'$. The physical meaning of connectivity bulk modulus can then be defined as the same influence to a change in volume, but this time in the presence of a neighboring piece of material (see Figure 4b).

Volumic Pressure

Consider a force per unit area applied to a node i on the surface of our deformable object. This force produces deformation. However, deformation of a node induces volumic change. Now, by introducing bulk modulus B_i for this node, we have:

$$P_{vp} = B_i / V_i \Delta V_i \tag{6}$$

where ΔV_i, the volumic change, is our measure of strain and V_i is the volume associated to a node. Now volumic pressure can alternatively be written in the following form:

$$P_{vp} = K_i \Delta V_i \tag{7}$$

where $K_i = B_i/V_i$. Note that K_i is dependent on V_i. We have derived the volumic pressure for a node.

Volumic Tension

Consider now a group of neighboring nodes. These nodes are linked topologically by the surface of our deformable object. This surface is elastic and to represent this in 3D, a difference in volumic change can be used (similar to a difference in length for the 1D case). In this case, volumic tension P_{vt} can be written as:

$$P_{vt} = \sum_j B_{ij} / V_i (\Delta V_i - \Delta V_j) \tag{8}$$

for all neighboring node j of node i. B_{ij} is the connectivity bulk modulus constant between node i and j. Note that we have not made any assumptions on the nature of elasticity between nodes, they can in general be of linear elasticity as in this case or of nonlinear elasticity.

Stress Distribution

Stress σ, is a function of strain, and strain in the VDM model is a function of volumic change ∆V. Hence for a node i, stress can be measured by calculating the net volumic change of the node with respect to its neighbors j:

$$\sigma i = Bi/Vi\ (\Delta Vi - \sum_j \Delta Vj)$$

$$= Bi/Vi\ (Ai\,\Delta Li - \sum_j Aj\,\Delta Lj) \quad (9)$$

Equilibrium State

The equilibrium state within a node is obtained by considering the following:

$$Pext = Pint \quad (10)$$

where P_{ext} is the external pressure and P_{int} is the internal pressure. The external pressure associated to a node is affected by the surrounding environmental pressure P_{ep} and by the stress due to volumic change:

$$Pext = Pep + Pvp \quad (11)$$

while the internal pressure is due to pressure of the incompressible fluid P_{fluid} and the effects of gravity $P_{gravity}$:

$$Pint = Pfluid + Pgravity$$

$$= Pfluid + \rho g \delta \quad (12)$$

where ρ is the density of the incompressible fluid and δ is the measured hydrostatic distance of the node due to the contained fluid in our deformable object. By considering neighboring nodes, equilibrium is attained by including effects from the volumic tension. Hence equation (11) becomes:

$$Pext = Pep + Pvp + Pvt \quad (13)$$

Model Assemblage

To obtain the VDM model assemblage, we formulate and group the equations for the state of equilibrium of each node on the surface:

$$P_{ext} = P_{int} \tag{14}$$

where by substituting equation (12) and equation (13), we get:

$$P_{ep} + P_{vp} + P_{vt} = P_{fluid} + P_{gravity} \tag{15}$$

Applying this equation to a group of $\aleph$ nodes, the following can be written using index notations:

$$B_i / V_i \, \Delta V_i + \sum_j B_{ij} / V_i \, (\Delta V_i - \Delta V_j) - \Delta P_i = \rho_i g \delta_i \;\; \forall \, i = 1 \ldots \aleph \tag{16}$$

where:

$$\Delta P_i = P_{fluid_i} - P_{ep_i} \tag{17}$$

and j indicates all neighboring nodes for each node i. We will now use the following theorem as a boundary condition:

THEOREM 1 : Pascal's Principle states that a change in pressure ΔP, exerted on an enclosed static fluid, is transmitted undiminished throughout this medium and acts perpendicularly on the surface of the container.

By applying Pascal's Principle which gives constant change in pressure throughout the deformable object, the index i can be removed from ΔP_i. By doing this, the following set of equations is obtained:

$$B_i / V_i \, \Delta V_i + \sum_j B_{ij} / V_i \, (\Delta V_i - \Delta V_j) - \Delta P = \rho_i g \delta_i \tag{18}$$

Since the fluid is incompressible, we can add another boundary condition to our set of equations. The incompressibility of the fluid imposes the constraint that the volume of the deformable object is maintained at all times. By applying the principle of conservation of volume:

$$\sum_{i} \aleph \Delta V_i = 0 \tag{19}$$

Since we are interested to obtain the global displacement vector, equation (18) and equation (19) are then rewritten such that ΔL_i appears as the variable instead of ΔV_i. This is done by using the following equation:

$$\Delta V_i = A_i \Delta L_i + L_i \Delta A_i \tag{20}$$

If A_i is assumed to be known throughout the history of load application, then ΔA_i is null and equation (20) is reduced to:

$$\Delta V_i = A_i \Delta L_i \tag{21}$$

We now have $\aleph+1$ equations and $\aleph+1$ unknowns; ΔL for $i = 1 \ldots \aleph$ and ΔP. These $\aleph+1$ equations can be written in the following matrix form:

$$K \Delta L = R \tag{22}$$

The matrix K is the state matrix of the VDM assemblage, ΔL is the deformation vector matrix and the load vector assemblage R consists the hydrostatic pressure terms, $P_{gravity}$.

IMPOSING CONSTRAINTS

Once the state matrix K has been obtained, in the presence of constraints, the respective nodes in K has to be modified. We identify five types of constraints as shown in Figure 5, that can be subjected to our deformable object.

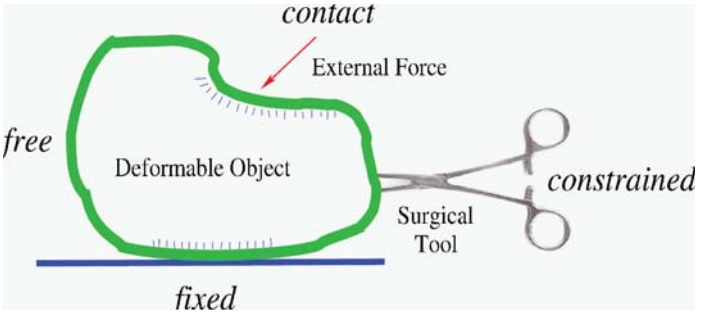

Figure 5. A resting deformable object that is touched by a finger and manipulated by a tool.

The following constraints manifest in this deformable object; FIXED: nodes that do not move, FREE: nodes that are free to move, CONTACT: nodes that are subjected to an external force, CONSTRAINED: nodes that are subjected to a displacement vector, ANISTROPIC: nodes that have preferred deformation direction.

FREE : Nodes that belong to this group are free to move. These nodes are not subjected to an external force and hence have P_{cp}= 0. Thus, they are governed by the following equation:

$$B_i/V_i\ (A_i \Delta L_i)+\sum_j B_{ij}/V_i\ (A_i \Delta L_i - A_j \Delta L_j) - \Delta P = \rho_i g \delta_i \qquad (23)$$

CONTACT : Nodes that belong to this group are subjected to an external load vector which can be modeled as a contact pressure P_{cp}. Hence, these nodes are governed by the following equation:

$$B_i/V_i\ (A_i \Delta L_i)+\sum_j B_{ij}/V_i\ (A_i \Delta L_i - A_j \Delta L_j) - \Delta P = \rho_i g \delta_i + P_{cpi} \qquad (24)$$

FIXED : Nodes in this group are fixed. They have null displacement vector and hence do not move during the simulation. To enforce this constraint during simulation, a penalty method is applied to the respective nodes. In this method, for a desired null displacement of node i, a penalizing term $\alpha >> 0$ is added to the diagonal term K_{ii} in the state matrix and R_i=0.

$$K_{ii} = \alpha\ ,\ R_i = 0 \qquad (25)$$

CONSTRAINED : The nodes belonging to this last group are subjected or constrained to a known displacement vector C. This happens when the nodes of our deformable object is fixed to a moving part. Again, to enforce this constraint, a penalizing term $\alpha >> 0$ is added to the diagonal term K_{ii} and $R_i = \alpha C$.

$$Kii = \alpha \ , \ Ri = \alpha C \tag{26}$$

ANISOTROPIC : We mentioned that the TMM model has been used to model anisotropic behavior. Anisotropy refers to the fact that deformation is experienced in a preferred direction. Soft tissue that are made up of nonhomogeneous material, for example fibers, exhibit this behavior. Some examples of soft tissue that fall into this category are muscles, ligaments and tendons.

The VDM model can also be used to characterize anisotropic behavior. By varying the bulk modulus B_i and the connectivity bulk modulus B_{ij} associated to each node, deformation can be preferred in a particular direction as compared to another. This can be explained by noting that in the presence of an external load, a change in volume is experienced. This change in volume is distributed uniformly to the other parts of the surface by the pressure experienced by the incompressible fluid. Since, by definition, the bulk modulus relates pressure to the fractional change in volume, it can be used to control the amount of volume distributed, and hence deformation.

In practice, the bulk modulus B_i and the connectivity bulk modulus B_{ij} are tuned based on the characteristics of the simulated material, for example, fibers are only allowed to deform in their respective fibrous orientation.

SYSTEM RESOLUTION

In the previous section, the mathematical foundations of the VDM model was presented. The system of equations describing the equilibrium of the system was derived. In this section, we are interested in the methods of solving this set of equations for linear and nonlinear static analysis. A static analysis is sufficient because soft tissue is known to be well-damped and thus the viscoelastic effects can be neglected. Furthermore, a static resolution does not suffer from numerical instabilities related to the convergence of dynamic systems.

Linear Analysis

The linear analysis amounts to calculating small deformations and small strains due to external loads. Medical interventions that concurs small deformations are not uncommon. An example is the echographic exam of the human thigh. In these type of procedures, only small strains are imposed on the human body.

Let us recall the equation of equilibrium of the VDM model once again:

$$B_i/V_i\ (A_i\Delta L_i)+\sum_j B_{ij}/V_i\ (A_i\Delta L_i - A_j\Delta L_j) - \Delta P = \rho_i g \delta_i \quad (27)$$

where i=1... ℵ and $\Delta P = P_{fluid} - P_{ep}$. For small deformations, we assume that the area A^{facet} of our facets do not change. Hence, the boundary conditions can be reduced such that:

$$\sum_i \aleph A_i \Delta L_i = 0 \quad (28)$$

If $|\Delta L_i| <$ [epsilon] ∀, then A_i can be considered constant and the state matrix K is constructed only once in the beginning. This matrix is reused at each time step. The problem $K\Delta L = R$ is then solved optimally using standard numerical methods to obtain a solution.

A technique that is frequently used to precondition the state matrix K is the lower-upper (LU) decomposition. In this method, we suppose that K can be written as

$$LU = K \quad (29)$$

where L and U are the lower and upper triangular elements of K. By using the decomposition LU=K and after substitution into $K\Delta L = R$, we get:

$$K\Delta L = (LU)\Delta L = L(U\Delta L) = R \quad (30)$$

We first solve for the vector Y such that:

$$LY = R \quad (31)$$

and then solve:

$$U\Delta L = Y \quad (32)$$

Since K is constant, hence the matrices L and U are also constant. Thus, these matrices are also pre-computed. The pseudo-code of this resolution procedure is described in Figure 6.

```
Data: K, state matrix and R, the external load vector
Result: ΔL, the nodal displacement vector
begin
    A ← K
    B ← R
    if ( LU - A ) then
        while ( |B| ≠ 0 ) do
            LY = B
            UΔL = Y
end
```

Figure 6. Resolution of KΔL=R using a linear approach.

Quasi-Linear Analysis

The analysis presented earlier is only valid for small deformations and small strains (<10%) where we assume that A_i is constant. For well-damped soft tissue, this is perhaps a valid assumption for small deformations. However, for other kinds of large deformations, this assumption becomes invalid and volume conservation needs to be restated as follows:

$$\sum_i (A_i \Delta L_i + L_i \Delta A_i) = 0 \tag{33}$$

In this case, A_i is the nonlinear term and it has to be updated regularly. In the quasi-linear analysis, a selective method is used to update the nonlinear terms. Since the nonlinear A_i terms appear in almost all the nonzero terms of K, a choice has to be made on which terms that needs to be updated. To ensure that volume is conserved during our simulation, it is imperative that equation (33) is satisfied at all times. This is done by updating the A_i terms at each time-step. These terms only appear in the bottom row of matrix K, allowing us to use an optimal method to solve the system.

The Sherman-Morrison (SM) method can be used for this purpose. In this method, we suppose that the inverse matrix of the initial state matrix K^{-1} has been found. This can be done using any numerical method. Then the solution to the system is given as follows:

$$\Delta L = K^{-1} R \tag{34}$$

Since K is updated at each time-step with the new A_i terms, K^{-1} has to be recomputed to obtain a new solution. Now, given a change to our original matrix K of the form:

$$K\,(K+u\otimes v) \tag{35}$$

where in our case, vector u is a unit vector and v is the components of the error vector that must be added, the following can be obtained:

$$\begin{aligned} z &\equiv K\text{-}1\cdot u \\ w &\equiv (K\text{-}1)^T\cdot v \\ \lambda &= v\cdot z \end{aligned} \tag{36}$$

Then the desired change in the inverse is given by:

$$K\text{-}1 \longrightarrow K\text{-}1 + \{K\text{-}1\cdot u\}\otimes\{(K\text{-}1)^T\cdot v\}/1+(K\text{-}1\cdot u) \tag{37}$$

Hence, adding the matrix $u\otimes v$ to the original state matrix is equivalent to updating the bottom row of K with the new distributed areas. The pseudo-code of this resolution procedure is described in Figure 7.

```
Data : K, The statc matrix and R, the external load vector
Result : △L, the nodal displacement vector
begin
    Given K ⟼ (K + u ⊗ v)
    A ⟵ K
    B ⟵ R
    while (|B| ≠ 0) do
        Compute Error Vector V
        Compute K-1
        Solve △L = K -1 R
end
```

Figure 7. Resolution of K∆L=R using a quasi-linear approach.

Nonlinear Analysis

In the nonlinear analysis, all nonlinear terms are updated at each time-step. This amounts to simulating large deformations and large strains. In this case, for large systems, a simple inversion or preconditioning of the state matrix at each time-step may be computationally expensive for interactive-time applications. However, the rapid increase in computational power has popularized iterative methods as a resolution scheme.

The basic idea behind this iterative method is the minimization of the residual r^i at each iteration i, defined as:

$$r_i = R - K\Delta L_i \tag{38}$$

It can be shown that this guarantees exact convergence for linear systems of equation in at most n iterations, where n is the size of the linear system. However, due to floating point errors, exact convergence is improbable in practice and the solution is obtained when the error drops below a defined tolerance.

We chose the Bi-Conjugate Gradient (BCG) iterative method as the optimal resolution scheme. This method is attractive for large sparse systems because only the nonzero terms of the state matrix is stored; hence minimal memory. For real-time solution of very large systems, even a solution in n iterations may be too expensive. However, in interactive-time applications the solution ΔL only changes minimally from one time-step to another. Then, by using the previous result of the displacement vector as the starting guess for ΔL^0, we can achieve dramatic gains in speed after finding the first solution. The number of iterations needed to minimize the error below a certain tolerance is very much smaller than the value of n. The pseudo-code of this iterative resolution procedure is described in Figure 8.

```
Data: K, the state matrix and R, the external load vector
Result: ΔL, the nodal displacement vector
begin
    A ←— K
    B ←— R
    while ( |B| = 0 ) do
        while ( |ΔL^(i+1)| - |ΔL^i| > Є) do
            Improve r^i = B - AΔL^i
            Improve Estimated Solution ΔL^(i+1)
end
```

Figure 8. Resolution of KΔL=R using an iterative approach.

CONCLUSION

In this paper, we have presented VDM, a new physical model suitable for soft tissue simulation. This model is surface based, hence it's complexity is in general one order of magnitude lower than a classic volumic model like the Finite Element Method (FEM). VDM uses bulk variables like pressure, volume and bulk modulus as model parameters. Pascal's principle and volume conservation are used as boundary conditions. The intrinsic parameters in VDM for soft tissue depends on the organ being simulated. If for example, a liver is being modeled, it being 95% irrigated by blood, requires only the density ρ and bulk modulus B of blood. Such parameters can be obtained from the literature. Furthermore, experiments to determine these values can be conducted for verification purposes. These advantages make VDM an interesting alternative for soft tissue simulation.

There are several possible resolution methods for the analysis of soft tissue simulation using the VDM model. A static analysis is sufficient in many cases as soft tissue is known to be well damped and the viscoelastic effects can be neglected. This linear analysis is suitable for soft tissue undergoing small deformation. There are many examples of medical simulation that restrict deformation within the linear limits. A quasi-linear analysis allows a quick resolution technique for any deformation with volume conservation as the main aspect. In this method there is a trade-off between the computational time and realism. On the other hand, the nonlinear method allows us to solve for large deformations and large strains without much loss of accuracy. Since it is an iterative method, the accuracy of the results depend on the size of the system solved and the constraint of interactive-time in the application concerned.

This model also has its limitations. Unlike FEM, VDM is not very accurate. This can be expected as VDM being surfaced based, is used to simulate volumic objects. The law of conservation of volume (a geometrical constraint) does not guarantee that the stress state (a physical constraint) of the object is at the best minimum level. But if the object is simulated at small time-steps, VDM is found to produce similar results like FEM. For fast soft tissue simulation, VDM can be considered as an alternative choice.

REFERENCES

1. Aulignac, D. (2001). *Modelisation de l'interaction avec des objets deformables en tempreel pour des simulateurs medicaux*. Institute National Polytechnique de Grenoble France.

2. C. Mendoza, C. L. (2003). *Simulating soft tissue cutting using finite elements models.* Paper presented at the IEEE International Conference on Robotics and Automation.

3. Casson, F. B.-d. (2000). *Simulation Dynamique de Corps Bioogies et Changements de Topologie Interactifs.* University de Savoie France.

4. Cotin, S. (1999). *Modeles Anatomiques deformables en temps reel.* University de Nice, France.

5. D. Terzopoulos, K. W. (1990). Physically-based facial modelling analysis and animation. *Journal of Visualization and Computer Animation.*

6. D.C. Tseng, J. Y. L. (2000). *A hybrid physical deformation modeling for laparoscopic surgery simulations.* Paper presented at the IEEE International Conference of the Enigineering on Medic Biology Society.

7. F. Boux-de Casson, C. L. (1999). *Modelling the dynamics of a human liver for a minimally invasive simulator.* Paper presented at the Medical Image Computing and Computers Assisted Intervention

8. G. Debunne, M. D., M.P. Cani (1999). *Interactive multiresolution animation of deformable models.* Paper presented at the EAGG Conference on Eurographics.

9. G. Debunne, M. D., M.P. Cani and A. Barr. (2001). *Dynamics real-time deformations using space and time adaptive sampling computer Graphics.*

10. G. Picinbono, H. D., and N. Ayache. (2002). *Nonlinear and anistropic elastic soft tissue models for medical simulation.* Paper presented at the IEEE International Conference on Robotics Automation.

11. H. Delingette, S. C., and N. Ayache. (1999). *A hybrid elastic model allowing real-time cutting, and force-feedback for surgery training and simulation.* Paper presented at the CGS Conference on Computer Animation

12. J. Brown, K. M. (2001). *A microsurgery simulation system.* Paper presented at the Medical Image Computing and Computer Assisted Intervention.

13. M. Bro. Neilsen, S. C. (1996). *Real-time volumetric deformable models for surgery simulation using finite elements and condensation.* Paper presented at the EAGG Conference on Eurographics.

14. S. Cotin, H. D. (1999). Real-time elastic deformations of soft tissue for surgery simulation. *IEEE Transactions on Visualization and Computer Graphics.*

15. S. Cotin, H. D., and N. Ayache (2000). A hybrid elastic model allowing real time cutting, deformation and force-feedback for surgery training and simulation. *Journal of Visual Computer.*

16. Sundaraj, K. (2004). *Real-time dynamic simulation and 3D interaction of biological tissue:Applicayion to medical simulators.* Institute Nationale Polytchnique de Grenoble, France.

17. Waters, K. (1987). *A muscle model for animating three dimensional facial expression.* Paper presented at the ACM SIGGRAPH.