Computer Methods and Programs in Biomedicine (2005) **77**, 183—197

ELSEVIER

www.intl.elsevierhealth.com/journals/cmpb

Real-time deformable models for surgery simulation: a survey

U. Meier, O. López, C. Monserrat*, M.C. Juan, M. Alcañiz

Medical Image Computer Laboratory (MedICLab), Universidad Politécnica de Valencia, Valencia, Spain

Received 2 July 2003; received in revised form 5 November 2004; accepted 6 November 2004

KEYWORDS
Deformable model;
Elasticity;
Animation;
Surgery simulation;
Real time

Summary Simulating the behaviour of elastic objects in real time is one of the current objectives of computer graphics. One of its fields of application lies in virtual reality, mainly in surgery simulation systems. In computer graphics, the models used for the construction of objects with deformable behaviour are known as *deformable models*. These have two conflicting characteristics: interactivity and motion realism. The different deformable models developed to date have promoted only one of these (usually interactivity) to the detriment of the other (biomechanical realism). In this paper, we present a classification of the different deformable models that have been developed. We present the advantages and disadvantages of each one. Finally, we make a comparison of deformable models and perform an evaluation of the state of the art and the future of deformable models.

1. Introduction

Until less than two decades ago, the evolution of geometrical models in computer graphics had only allowed for the representation of rigid objects. In 1986, an initial *free-form deformation* method was presented that deformed arbitrary objects by distorting the space in which these were contained [1]. A year later, Terzopoulos [2] incorporated physical properties directly in a graphical object for the first time, and coined the term *deformable models*.

Deformable models can be defined in either one dimension (lines and curves), two dimensions (surfaces), or three dimensions (solid objects). Essentially, they are applied in three different areas of research: one of these areas is in the field of *object modeling* for pre-computed animations [3—5]. Another one is the area of *image segmentation*, for instance for an automatic 2D interpretation of the images provided by a camera supervising an automated production line, or for the 3D reconstruction of bones and organs from medical MRI or CT scans [6—12]. And the last is in the field of interactive mechanical simulations, i.e., to emulate the deformational behaviour of non-rigid objects due to external influences, both in deferred applications like surgery planning (e.g., simulation of the outcome of craniofacial surgery) [13—15] and in real-time applications. These real time applications include image guided surgery [16], minimally-

* Corresponding author. Tel.: +34 9633877007x83523; fax: +34 963877359.
E-mail address: cmonserr@dsic.upv.es (C. Monserrat).

0169-2607/$ — see front matter © 2004 Published by Elsevier Ireland Ltd.
doi:10.1016/j.cmpb.2004.11.002

invasive or tele-surgery [16—18], and surgery simulation [2,19—22].

From the number of applications, it is obvious that there is no single deformable model that is appropriate for all of the above mentioned problems. Instead, there are a variety of methods that are optimized in different ways to meet specific needs.

Even though virtual reality applications of deformable models are becoming more and more frequent, in many cases, the governing prerequisite for the simulation of mechanical deformations has been interactivity rather than precision. In addition to that, many of the modelled objects like garments or soft tissues do not possess easily describable properties. Consequently, exact methods, in general, cannot be applied as deformable models, and thus, other approaches must to be found [23].

Important progress in the field of deformable models has been made since the emergence of surgery simulation, with one of the first contributors being Cover et al. [24]. This is mostly due to the extreme prerequisites as far as computation time, complex properties of the simulated soft tissues, and intricate interactions with the virtual instruments are concerned. In fact, there have been many different points of departure in the research of adequate deformable models, focusing on anatomies and surgical techniques that are as different from each other as are eye surgery [25,26], knee arthroscopy [27,28], or hepatic laparoscopy [22,29]. The deformable models developed in this context can be divided into three basic groups: the ad hoc, heuristic methods; the more technical approaches based on a simplification of the continuum-mechanical model (see Fig. 1); and the hybrid methods which result from a combination of the two prior groups.

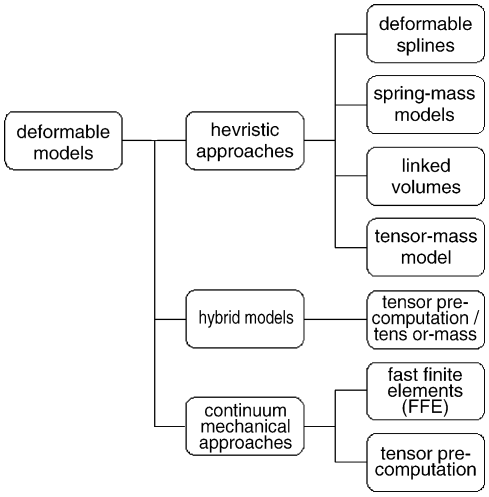

Fig. 1 Deformable models for surgery simulation.

The first group implicitly is based on the hypothesis that classical, relatively exact computation methods like Finite Element Methods (FEM) are far too complex to yield real-time performances. The models included in this group derive from alternative, rather straight forward modelling schemes for the geometry of deformable objects, allowing for the inclusion of elastic properties. Since both visual representation and basic haptic interaction are restricted to the surfaces of the respective organs, at first, they led to the appearance of *deformable splines* as an extension of the well-known graphical technique to represent smooth curves or surfaces [2]. Later, the iterative *spring-mass model* was developed, once again initially representing only surfaces [30,31].

The handling of solid objects as hollow shells can reduce the size of the problem to less than a hundredth part of its initial size [32]. However, this treatment intrinsically brings along limited realism, in particular with respect to the conservation of volumes. Consequently, there eventually appeared volumetric extensions of the spring-mass model like the voxel-based *linked volumes* algorithm [33] or the *tensor-mass model*, which are based on linear and non-linear relationships derived from continuum mechanics for tetrahedral elements [34,35].

The second and more recent group of deformable models is based on the hypothesis that the corresponding continuum-mechanics-based models are too complex, and hence, too slow on contemporary computers when scrupulously applied. That is, by sufficiently simplifying the assumptions on the mechanical behaviour of the modelled tissues, the continuum-mechanics-based models can be simulated today with certain realism. Parallel to the ever-increasing computational power of computer processors, eventually more and more complicated hypotheses on the respective properties of soft tissues will be incorporated. In fact, most deformable models that belong to this group are based on *finite element methods*, which require a discretization of the entire volume of the deformable objects [19,36,37]. A deformable model based on *boundary element methods* (BEM) has also been proposed, which offers the advantage of inherently taking the calculations of the interior of an object to its surface, and thus, only requiring a discretization of this surface.

Finally, there are also some hybrid methods that attempt to combine the advantages of both of these approaches. These methods are characterized by dividing a deformable object into different sections according to the expected kind of interaction with each of these sections, and to model each one of these with an appropriate model.

2. Heuristic approaches

2.1. Deformable splines

In general, *splines* serve to obtain smooth and rounded curves, surfaces, or volumes, which adjust themselves to a series of control points. By moving the control points, the form of the respective curve, surface, or volume changes accordingly. Among the different existent techniques are the well-known *Bézier curves* and *NURBS* (non-uniform rational B-splines). *Deformable splines*, also known as *active contours*, were the first deformable models (in the strict sense) to be developed [2,38—40], and were also the first models to be applied to the field of surgery simulation [24]. As a starting point, they employ classical splines to model a 3D object (or its surface). With the fundamental theorems of differential geometry regarding the equivalence of shapes, deformable splines then define a potential energy, which is proportional to the degree of elastic deformation. By using the Lagrange approach, this energy is finally minimized with respect to the displacements enforced in some control points, thus obtaining the corresponding deformation state.

The elevated number of parameters equips this deformable model with a substantial level of control over the shape and the physical properties of the mesh. However, these parameters can only be chosen arbitrarily and are difficult to determine empirically. In addition, the representation of an object as a smooth surface does not coincide with the rendering algorithms of modern graphic cards, which are oriented towards plane polygons. Thus, a further processing step is required. Even without this additional expense, deformable splines already have a very high computational cost. This is why solid objects usually have to be modelled as hollow shells, with the corresponding detriment to realism, particularly in the constancy of volume.

On the whole, deformable splines are more complex and computationally costlier than spring-mass type models, without actually offering better realism. This is why they are no longer employed.

2.2. Spring-mass model

As indicated by their name, *spring-mass models* are based on meshes composed of springs (or spring-damper elements) and discrete mass points. In general, they define a series of mass-free springs that are distributed over the surface of a modelled object and are welded in points or nodes to which a discrete mass is attributed. This way, the contour

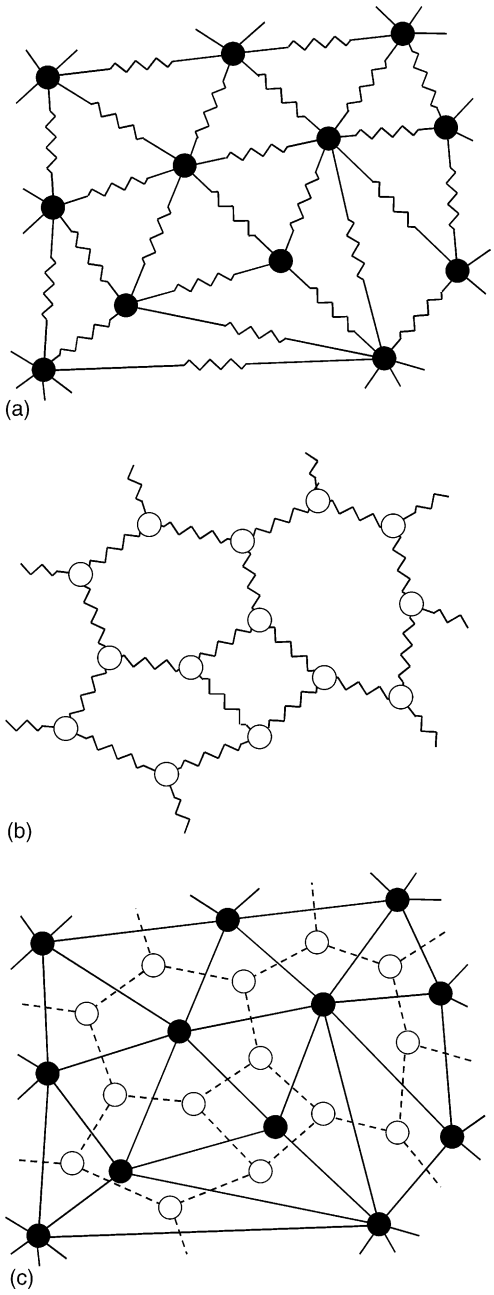

Fig. 2 Spring-mass-type meshes: (a) triangular mesh: springs forming triangular elements; (b) T_2-mesh: every node is connected to three of its neighbours; (c) duality between the two: nodes of the T_2-mesh white positioned in the centres of the triangular mesh elements (black nodes).

of the object can be discretized either into triangular patches [15,21,41—43], in which springs represent the edges of a polygonal surface mesh and welds represent the vertices (see Fig. 2a). Or, for simplified handling, the discretization can also be performed into T_i*-meshes*, which are characterized by each of its vertices being connected with a constant number of $i+1$ adjacent vertices [14,44,45].

In this case, T_2-meshes in which each node is connected with its three closest neighbours are usually chosen (see Fig. 2b). Because of their duality with triangular meshes, the T_2-meshes can best exploit the fact that graphic cards are typically optimized for triangular surface meshes by placing the nodes at the centres of the elements (see Fig. 2c).

Using the Newtonian law of motion, the force equilibrium for each node i of the mesh can be established:

$$m^i \frac{\mathrm{d}^2 \boldsymbol{r}^i}{\mathrm{d}t^2} + \gamma \frac{\mathrm{d}\boldsymbol{r}^i}{\mathrm{d}t} + \boldsymbol{f}^i_{\mathrm{int}} = \boldsymbol{f}^i_{\mathrm{ext}}, \tag{1}$$

where m^i is the mass of node i, $\boldsymbol{r}^i$ the current coordinates of node i, γ the viscous friction coefficient of the adjacent springs, $\boldsymbol{f}^i_{\mathrm{int}}$ the internal force acting on the node i and trying to maintain the initial position of the latter with respect to the nodes with which it is connected via springs, and $\boldsymbol{f}^i_{\mathrm{ext}}$ is the external force applied to the node i.

The internal force $\boldsymbol{f}^i_{\mathrm{int}}$, which is opposed to the deformations caused by the external forces, is determined by the degree of deformation of the springs. In general,

$$\boldsymbol{f}^i_{\mathrm{int}} = \sum_{j \in N_i} k^i_j \frac{|\boldsymbol{r}^j - \boldsymbol{r}^i| - |\boldsymbol{r}^j - \boldsymbol{r}^i|^0}{|\boldsymbol{r}^j - \boldsymbol{r}^i|} (\boldsymbol{r}^j - \boldsymbol{r}^i), \tag{2}$$

where N_i is the set of nodes with which the node i is connected via springs, k^i_j is the rigidity of the spring between the nodes i and j, and $|\boldsymbol{r}^j - \boldsymbol{r}^i|$ and $|\boldsymbol{r}^j - \boldsymbol{r}^i|^0$ are its current and initial length, respectively.

Finally, the set of differential Eq. (1) defined for each node is usually resolved by a discretization of time into intervals Δt, usually with finite Euler differences [15,31,42,46]. The stiffer the simulated object is intended to be, the smaller the Δt has to be in order to converge [47].

The application of a more sophisticated time discretization technique like Runge—Kutta has also been proposed. Generally speaking, it is much more stable than Euler's method, and hence, it permits the selection of a greater time step Δt. This is partially compensated for, by requiring a more frequent evaluation of the internal forces. Nevertheless, an acceleration factor of about two with respect to Euler's method can be achieved by employing a fourth order Runge—Kutta method [34,41].

The major advantage of this basic spring-mass model is the simplicity of its mesh. It is ideal for a direct rendering, and the resulting system of equations is also easy to obtain and program. In fact, the mesh has a markedly local structure since each node only depends on its direct neighbours.

This permits the representation of even great deformations with relatively high realism. In addition, any topological manipulations like incisions are easily represented, as the modifications in the system of equations are very confined. Nonetheless, when cutting into volumetric objects, their interior still has to be created [48].

However, there are disadvantages. The local structure of spring-mass meshes impedes the rapid global propagation of deformations, i.e., within each iteration step, the enforced displacement of one node is only propagated to the next surrounding ring of adjacent nodes. Together with the superficiality of the mesh, this causes a visibly limited volumetric behaviour. An object modelled in this way does not even approximately maintain a constant volume, and when acting on one of its sides, the opposing side only reacts after a noticeable delay. This effect is all the more visible the softer the object is and the larger its deformations are, thus limiting the otherwise elevated realism of large deformations (see Fig. 3).

Another important drawback with respect to the realism of spring-mass type deformable models is their predisposition to oscillate, which derives from their iterative structure. As mentioned before, a conversion of the solution largely depends on the selection of an appropriate time step and on the parameters of the mesh. Therefore, these do have to be determined experimentally for each case. Again, the simulation of incisions represents the worst-case scenario, as some springs have to be removed and others added. Thus, not only is it difficult to establish the required new parameters (specifically the initial lengths of the new springs) but the convergence of the over-

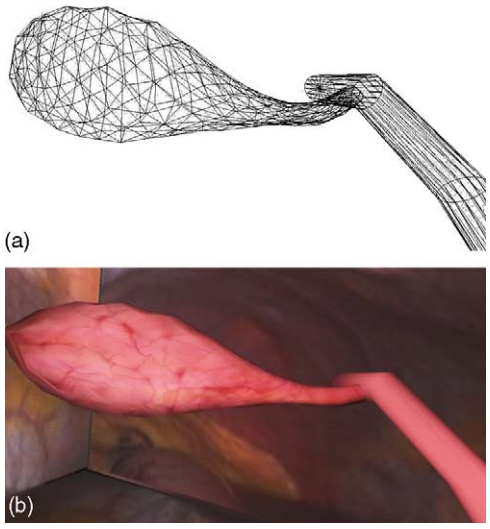

Fig. 3 Deformation of a gall bladder modelled with a spring-mass model: (a) wire frame view of undeformed gall bladder; (b) rendered view of deformed gall bladder.

all system of equations might also be significantly affected.

Despite these deficiencies, spring-mass type models are, at present, the most widely used deformable models and are not only applied to surgery simulation. As far as the number of already implemented interaction types are concerned, no other deformable model is as advanced as this one [41,49]. However, this success is only due to the simple mesh structure and the subsequently easy programmability of practically all possible interactions. For volumetric tissues like organs, the physical realism is very limited. And as the evolution of this deformable model has already reached its peak, it is not yet predictable whether it will be able to maintain its position in the future when other deformable models with higher physical realism will be able to simulate a similar range of interactions.

In conclusion, various methods have been proposed to compensate for the obvious superficiality of the basic spring-mass model. The simplest one is to introduce internal parent nodes into the mesh that connects the opposite sides of an object [31,41,50]. A higher connectivity of the different nodes (especially with distant nodes) largely deteriorates the simplicity of the system of equations, and its solution time is increased accordingly. Nevertheless, the selection of both positions and parameters of springs and nodes is still arbitrary and ad hoc. Also an optimization of the positioning process (for instance, with a simulated annealing process) is laborious and merely stochastic [51]. The major problems, arise when cutting into these modified spring-mass models. First, the interior springs that have to be cut are not easily determined, as the incision algorithms are typically directed towards the triangular surface mesh. Second, all optimization of the interior spring positions is annihilated.

Therefore, a complete discretization of the entire interior seems more appropriate in order to achieve greater volumetric behaviour. Thus, depending on the method used, the alternatives are either the linked volume technique or the tensor-mass model.

2.3. Linked volumes

2.3.1. Basic model

One possible volumetric extension of the basic spring-mass model is based on a discretization of the entire volume of a deformable object into evenly spaced, cubic elements (see Fig. 4a). The most direct approach is to lump the corresponding masses at the respective centers of these elements and to interconnect them with their neighbours in space by resorting to springs and dampers [27,52,53].

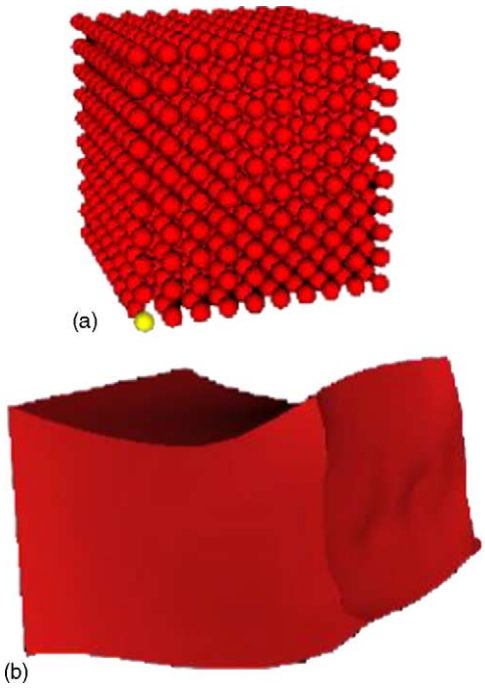

Fig. 4 Deformation of a cube discretized into linked volumes: (a) undeformed system; (b) deformed system, rendered with the help of an additional boundary representation of the surface.

Computationally, the only differences with respect to the basic spring-mass model are the considerably higher number of nodes and their increased connectivity. Thus, the overall performance is comparable, including the possibly critical stability, while the increased size of the problem significantly limits the resolution of the discretization. Current PCs can handle on the order of a thousand elements as has been demonstrated in tests carried out by the authors.

Yet, in order to achieve an acceptable visual realism with smooth surfaces, it is inevitable to resort to auxiliary boundary representations or surface maps (see Fig. 4b), another costly procedure. However, nowadays even complex interactions like cutting, carving, joining, or tearing can easily be represented by simply eliminating or adding the respective joints between elements. Thus, while the topological resolution is not crucial for the visual realism due to the surface maps employed, it is crucial for mechanical realism, as each volume element itself is rigid and indivisible, and practicable local refinement techniques have not yet been developed. At the same time, an increased number of elements slows down the propagation of defor-

mations. Consequently, the overall behaviour, and specifically the volumetric behaviour, of this approach is only improved to a certain extent with respect to the basic spring-mass model.

2.3.2. Chain mail algorithm

While the basic linked volume technique is a mere 3D extension of the spring-mass model, the *chain mail* algorithm takes a slightly different approach [27,33,52,55]. It uses to the same volumetric discretization, but instead of linking the different elements with springs, these are interconnected like links of a chain. That is to say, within a certain limit, each link can move freely without influencing its neighbours, while major displacements are propagated directly to the corresponding adjacent links (see Fig. 5).

To achieve an elastic behaviour, the chain mail algorithm resorts to a relaxation of the deformational energy accumulated in the system whenever there is some spare computational capacity. In similarity to the spring-mass type model, this internal energy is proportional to the displacement of the elements with respect to each other. And as with the basic linked volume technique, this energy is minimized iteratively until convergence. However, there are no forces associated to the links; for this reason, the force feedback is assumed to be proportional to the depth of penetration, i.e., the deepest displaced link. This depth is determined with the help of a distance map where the initial distances of all the elements with respect to the surface are tabulated. And like the basic linked volume technique, the chain mail algorithm further requires some additional surface mesh for rendering.

Just like the spring-mass model, the great advantage of the chain mail algorithm is its simplicity. Thus, the volumetric behaviour, including a rapid propagation of deformations and even a roughly constant volume, are automatically guaranteed by the chainlike structure of the model, which is also capable of emulating a quasi-viscose or plastic behaviour. Moreover, in conjunction with the linear or non-linear elastic relaxation, almost any mechanical behaviour can be simulated. And what is more, as with the basic linked volumes, even complex interactions can easily be represented by simply eliminating or adding some joints between elements.

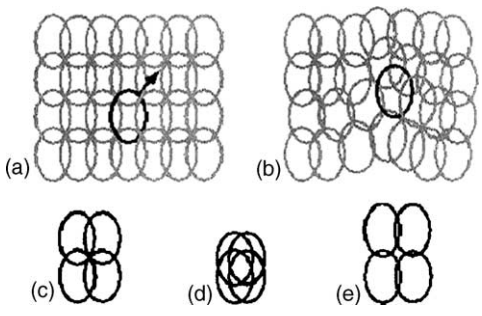

Fig. 5 Principle of chain mail algorithm [33]: deformation of a 2D mesh when moving a link in the direction of the arrow: (a) initial state; (b) deformed state: the neighbouring links are displaced to maintain maximum and minimum distances. Set of links: (c) initial state; (d) maximum compression; (e) maximum stretching.

Once again it is the resolution of the discretization that is the critical aspect of this model. While crucial for the mechanical realism, it is as limited as with the basic linked volumes technique. So far, chain mail approaches have only been used successfully in real time for 3D objects consisting of very few elements [52].

Like other heuristic models, the chain mail technique poses the problem of an arbitrary selection of parameters, which in turn affects the stability of the system. In this case, the margin of free mobility between the links also has an important damping effect. Thus, complex mechanical behaviour is still emulated rather randomly and cannot be directly influenced by the parameters [52].

Finally, the calculation of the propagations of imposed displacements for various simultaneous contact points can only be performed at high additional computational cost. The same is true when treating inhomogeneous tissues, for which the force feedback is not proportional to the depth of penetration. As there are no stresses associated to the different elements, tearing might well be simulated, but it occurs at random sites rather than where the maximum stress appears.

To sum up, the attractiveness of volume links stems from the straightforwardness of their mesh and the simplicity with which even complex interactions are modelled. However, while improving the volumetric character and allowing for more complex mechanical behaviour with respect to the basic spring-mass model, a very high price is paid to achieve this still-far-from-accurate volumetric behaviour, which represents an excessive amount of data to be processed. Ultimately, this is the reason why this deformable model is not yet very widespread.

2.4. Mass-tensor model

Another volumetric extension of the basic spring-mass model is based on a discretization of the entire interior of the deformable object into tetrahedrons. In a straightforward approach, point masses and springs are again positioned at the respective

tetrahedral vertices and edges [56]. The performance of such this model is comparable to that of the basic linked volumes technique (cf. Section 2.3.1). The different discretization technique does not require any additional surface mesh for its rendering. The tetrahedral mesh is much more difficult to create in the first place and to handle when simulating incisions. Even when minimizing the number of newly created tetrahedrons in such a situation, the corresponding increase of nodes is significant, thus further aggravating the scarcity of memory and resolution time [57]. Finally, since the performance of the model does not depend on the stochastic repartition of internal springs but rather on the resolution of the mesh, the problem of an arbitrary selection of parameters and the dependent stability of the system persists.

This is where the *mass-tensor* model comes in. In fact, it stems from the limitations encountered in the development of deformable models based on finite element methods (cf. Section 3), which also discretize 3D deformable objects into tetrahedrons [34]. By also using continuum mechanics, it establishes some simplified, linear relationships that describe the deformational behaviour of the different tetrahedrons. Summarized in tensors, the stiffness of each of these elements now replaces the springs with which the internal forces $\boldsymbol{f}^i_{\text{int}}$, acting at the different nodes and opposed to the deformations caused by external influences, are determined.

Instead of deriving a complex simultaneous description of global interrelationships as obtained with FEM, the mass-tensor model uses the same techniques as mass-spring type models when lumping masses at single points and iteratively solving the Newtonian law of motion (1) as a governing equation (cf. Section 4). Accordingly, even though they are computed with certain scientific rigour, the stiffness tensors still constitute an important influence on the stability of the system, i.e., on the maximum time step allowed for the iteration to converge. This time step becomes very small for materials that are not compressible (such as human tissues). This is why a recent improvement to the model heuristically introduces an additional internal force that later penalises any variation of the tetrahedral volumes during the deformation [35].

The result is a deformable model that no longer depends on the topology of the underlying mesh, but on the resolution of the mesh. It also exhibits significantly more volumetric behaviour than any spring-mass type models, as it essentially respects the incompressibility of tissues and propagates deformations much faster. Likewise, some material parameters can now be obtained empirically, and inhomogeneities can easily be considered by assigning different parameters to each tetrahedron. As the relatively costly computation of tensors is carried out before starting an actual simulation, the only way to simulate cutting processes in real time is to remove entire tetrahedrons from the mesh as when electrocauterizing.

However, in the mass-tensor model, the underlying linear elastic hypotheses, which are necessary to obtain simple and easily resolvable equations, also limit the physical realism to small deformations of an order of up to about 10%. For greater deformations of hyperelastic materials the rotational variance of the model becomes obvious, i.e., when revolving an entire object or only part of it, this is distorted in the rotational direction and its volume is increased. This important drawback is compensated for when using to a large deformation model with a strain tensor that is a quadratic function of the deformation gradient; for instance, the Green-St. Venant strain tensor. However, this improvement occurs at the expense of a largely increased computational cost (almost six times the computation time required for a mesh with some 6300 tetrahedrons) [35]. To solve this, first, the deformations are computed linearly. The computation is repeated non-linearly only for those parts of the organ that are deformed beyond a certain threshold. Thus, the total computation is reduced but it is still significantly more extensive, and hence, slower than the purely linear approach.

3. Continuum-mechanical approach

An entirely different way to approach the issue of deformable models is to directly base them on the laws of continuum mechanics, even though these have to be simplified significantly in order to obtain real-time performance. In fact, the simplest relations are obtained when assuming a linearly elastic material that has small, slow deformations and negligible internal forces like gravity. Under these circumstances, the resulting differential equation is the second order Navier equation:

$$(\lambda + \mu)\,\text{grad}\,(\text{div}\,\boldsymbol{u}) + \mu\,\Delta\boldsymbol{u} = 0, \tag{3}$$

where λ and μ are the Lamé constants of the material and $\boldsymbol{u}$ is the displacement vector of any point of the object with respect to its initial position.

There does not exist a general analytical solution for (3), and therefore a numerical solution scheme is required, with the two best known approaches being finite element methods and boundary element methods.

3.1. Finite element methods

3.1.1. Basic relationships

Originally, FEM were developed to approximately resolve differential equations defined for a certain domain and with some given corresponding boundary conditions. To do so, the entire domain, i.e., the deformable object, is discretized into a finite number of subdomains or *elements*. Then, the magnitude of interest, i.e., the displacement, is approximated with polynomial equations over each element and is represented as a function of the values at some corresponding control points or *nodes*. When forcing this quantity to be continuous over element boundaries, an approximated solution can be obtained by minimizing the inherently introduced error over the domain boundary.

In the simplest, linear case, the resulting system of equations is of the form:

$$\mathbf{K}\boldsymbol{U} = \boldsymbol{F}, \tag{4}$$

where $\boldsymbol{U}$ and $\boldsymbol{F}$ are the nodal displacement and force vectors, respectively, and where $\mathbf{K}$ is the symmetric and spare stiffness matrix. This, in turn, is banded with an adequate enumeration scheme for the finite elements.

During a simulation, most nodes are free, and therefore, have a zero force prescribed and an unknown displacement. The nodes that are virtually in contact with an instrument, in turn, have their displacement prescribed, while the resultant force to be fed back is unknown. Thus, $\boldsymbol{F}$ almost exclusively consists of zeros, with the exception of some unknown values for the contact nodes. $\boldsymbol{U}$, in turn is mostly comprised of unknowns with only a few known, but variable, prescribed values. As a consequence, (4) has to be reordered every time the set of constrained nodes changes in order to group all the known and unknown values together so that in what follows, $\boldsymbol{U}$ refers to the vector of unknown values and $\boldsymbol{F}$ refers to the vector of known values.

The properties of $\mathbf{K}$ constitute (4) a system that is relatively easy to solve. However, the solving time actually depends on the employed algorithm to a great extent. And even in the best case, a real-time application is only feasible for a system that is too small to allow for a general, direct application of the above equation [19,23]. This difficulty further demonstrates the importance of the substantial simplifications with respect to the underlying hypotheses about the mechanical behaviour of the modelled objects.

3.1.2. Fast finite elements (FFE)

Although numerically disadvantageous, by far, the fastest way to solve (4) repeatedly for the vector of unknowns $\boldsymbol{U}$ and a varying, known right-hand side vector $\boldsymbol{F}$, is the preliminary inversion of $\mathbf{K}$ before simulating any deformation, i.e., when time is not yet the governing factor [23]. Thus, for each input vector $\boldsymbol{F}$, it is only necessary to carry out the simple matrix–vector multiplication

$$\boldsymbol{U} = \mathbf{K}^{-1}\boldsymbol{F}. \tag{5}$$

However, $\mathbf{K}^{-1}$ is no longer sparse. Moreover, the required discretization of the entire volume also implies the creation of many nodes in the interior of the modelled object. Even though these contribute to the volumetric behaviour of the model, they are of no interest for a graphical representation nor for the mechanical interaction. Thus, (5) is of much larger than the one really required, and a compression of the system to the surface nodes can be carried out [19].

This way of solving the Navier Eq. (3) is also referred to as *fast finite elements* [19]. It is fast enough to allow for the real-time simulation of reasonable size, but requires a pre-computation of up to several hours. As far as the physical realism is concerned, the FFE very accurately comply with the preliminary hypotheses of the mechanical behaviour they are based on. However, these assumptions must be very restrictive so as to obtain a linear relationship. Consequently, neither great or rapid deformations, nor the partially viscous behaviour of tissues can be represented correctly.

The preliminary compression inevitably requires the constancy of the stiffness matrix $\tilde{\mathbf{K}}_{ss}$, as its modification would be too costly to be carried out in real time. This implies, however, that the finite element mesh has to be static and that the type of each node (prescription of either displacement or force) is fixed during the entire simulation. In other words, topological changes, such as cutting, cannot be considered nor can the contact area (i.e., the set of contacted nodes), be modified. This is why FFE are not really applicable to general surgery simulation.

3.1.3. Dynamic systems

The addition of a more dynamic behaviour with mass inertia and energy dissipation to this deformable model has also been proposed, albeit in an approximate way in order not to alter the simplicity of (3). For this purpose, the Newtonian law of motion (1) is used once again and solved with an explicit or semi-implicit Euler method. However, the

system becomes:

$$\mathbf{M}\ddot{\boldsymbol{U}} + \mathbf{D}\dot{\boldsymbol{U}} + \mathbf{K}\boldsymbol{U} = \boldsymbol{F}, \tag{6}$$

where the dots indicate temporal derivatives, and **M** and **D** are the diagonal mass and damping matrices. As far as the damping coefficients are concerned, these are supposed to be proportional to the corresponding nodal masses, i.e., $\mathbf{D} \equiv \alpha\mathbf{M}$.

In order to solve (6), suggested two different forms to apply Euler's method have been proposed to discretize time [23]. The explicit approach yields nodal displacements $\boldsymbol{U}_{t+\Delta t}$ at the instant $t + \Delta t$ which are directly obtained from the previous ones as

$$\boldsymbol{U}_{t+\Delta t} = \left(\frac{\alpha}{2} + \frac{1}{\Delta t}\right)^{-1} \left\{ \frac{2}{\Delta t} U_t + \left(\frac{\alpha}{2} - \frac{1}{\Delta t}\right) \boldsymbol{U}_{t-\Delta t} - \mathbf{M}^{-1}\,\Delta t[\mathbf{K}\boldsymbol{U}_t - \boldsymbol{F}_t] \right\}. \tag{7}$$

This method simply requires the evaluation of $\mathbf{K}\boldsymbol{U}$t at the moment prior to the current computational step. It is, hence, extremely simple and fast to carry out, even more so for a sparse stiffness matrix **K**, i.e., a non-compressed system. However, dynamic systems have has the same disadvantages as the spring-mass model except for the fact that the interior nodes improve the volumetric behaviour. Moreover, the global behaviour of the modelled objects no longer depends on the arbitrariness of the discretization.

For the semi-implicit Euler method to solve (6), the product $\mathbf{K}\boldsymbol{U}$ is evaluated at the moment $t + \Delta t$ instead of at the moment t. By doing this, the following equation is obtained:

$$\hat{\boldsymbol{K}}\boldsymbol{U}_{t+\Delta t} = \hat{\boldsymbol{F}}_t, \tag{8}$$

where

$$\hat{\boldsymbol{K}} \equiv \boldsymbol{K} + \left(\frac{1}{\Delta t^2} + \frac{\alpha}{2\,\Delta t}\right)\boldsymbol{M} \tag{9}$$

and

$$\hat{\boldsymbol{F}}_t \equiv \boldsymbol{F}_t + \frac{2}{\Delta t^2}\mathbf{M}\boldsymbol{U}_t + \left(\frac{\alpha}{2\,\Delta t} - \frac{1}{\Delta t^2}\right)\mathbf{M}\boldsymbol{U}_{t-\Delta t}. \tag{10}$$

This is not an explicit system like (7); it is quite similar to the static system (4) which can actually be compressed and solved with a preliminary inversion of $\hat{\boldsymbol{K}}$, albeit with the same reservations as far as its practicability is concerned. However, as the dynamic system is not implicit, it may also oscillate. Thus, its stability largely depends on the selection of an appropriate time step Δt.

3.1.4. Tensor pre-computation

The most advanced FEM-based deformable model is more flexible than the ones referred to above, and at the same time offers most of the advantages of the other systems. It is based on the superposition principle [29]. When exposing a linear elastic body to a load, the body reaction (i.e., its deformation and the resultant force at the point of load application) is assumed to be proportional to the load. In addition to that, the reaction to the application of simultaneous loads is also assumed to be identical to the sum of reactions to the corresponding individual loads.

Then, assuming a unit displacement of every surface node subsequently and separately for the three basic directions, all the possible reactions of all the surface nodes from (4) can be progressively precomputed. The results are stored in the form of displacement tensors $\boldsymbol{T}^{\boldsymbol{U}}_{i,k} \in R^{3\times 3},\ i \neq k$ and force tensors $\boldsymbol{T}^{\boldsymbol{F}}_{k} \in \mathrm{R}^{3\times 3}$, where $\boldsymbol{T}^{\boldsymbol{U}}_{i,k}$ expresses the displacements of node i caused by unit displacements of the only constrained node, k, and $\boldsymbol{T}^{\boldsymbol{F}}_{k}$ expresses the corresponding reaction force at the node k itself.

Now, if at any moment of a simulation, a displacement $\bar{\boldsymbol{U}}_k \in \mathrm{R}^3$ is enforced at a single surface node k, the corresponding displacement and force tensors only have to be multiplied by $\bar{\boldsymbol{U}}_k$ to obtain the other nodal displacements and the reaction force $\boldsymbol{P}_k$ at the node k itself.

In the tensor technique, the restriction to surface nodes is trivial and does not require an additional compression. Moreover, a dynamic adaptation of the type of each node is finally possible when there is a variable set of contact nodes. This can occur when new instruments came into contact with a simulated organ and these slide along the surface, or when a contact is lost (variable prescription of either displacement or force). Also, multiple contacts can be easily controlled as long as there are not too many of them.

These advantages make this method the most practical FEM-based technique proposed so far [34]. However, it is still based on linearly elastic hypotheses. Accordingly, their realism is very limited for great or rapid deformations. What is more, topological changes, like the ones, which occur when cutting, can still not be represented in real time as all tensors would have to be recomputed, which is too costly an operation. This is why the application of the superposition principle might be quite suitable for certain applications where the geometry of the simulated tissues is not modified, e.g., with endoscopic simulators. However, it is not apt for more general applications.

3.1.5. Other approaches

Other FEM-based deformable models have also been proposed. However, some do not even focus on real-time computations as they have been designed for pre-operational planning (e.g., [15]). Others vary only slightly from the approaches presented above in the amount of pre-computations, which in general are less effective [25,36]. Another approach discretizes the surface of the modelled tissues only and treats them as thin-walled membrane-like structures filled with a liquid [32]. Thus, the size of the problem is largely reduced. However, since the non-linear mechanical hypotheses with which a relatively realistic behaviour is achieved are so complex, that the time saved by the reduced size is lost. No real-time simulations have been reported using this method.

While FEM-type deformable methods intrinsically have to be based on linear hypotheses to achieve real-time performance, some attempts have been made to include non-linear characteristics like anisotropy in an approximated way. Such attempts should be carried out once the linear computations as presented above have been performed, using characteristic curves [29]. Finally, some work has also been performed on the geometrical aspects of cutting into the 3D meshes used by FEM-type models [7,57,58]. These models concentrate on the complex problem of how to adjust a typically tetrahedral finite-element mesh to the arbitrarily propagating trace of shear or scalpel-type blades. However, despite progress in this area, none of these approaches is apt for real-time simulations due to the general inability of FEM-type models to allow for arbitrary topological changes during an actual simulation.

3.2. Boundary element methods

The last independent deformable model is based on the less well-known *boundary element methods* [22,59,60]. Like FEM, these were developed to approximately resolve differential equations. However, they use to so-called fundamental solutions that fulfil the corresponding differential equation within the interior of the respective domain. Thus, boundary value problems are reduced to precisely the boundary of these domains. As for the practical effects, this also means that only the surface of an object has to be discretized into patches or boundary elements. This way, the model automatically coincides with the most commonly used rendering algorithms, which use triangular surface meshes of the geometrical bodies to be represented

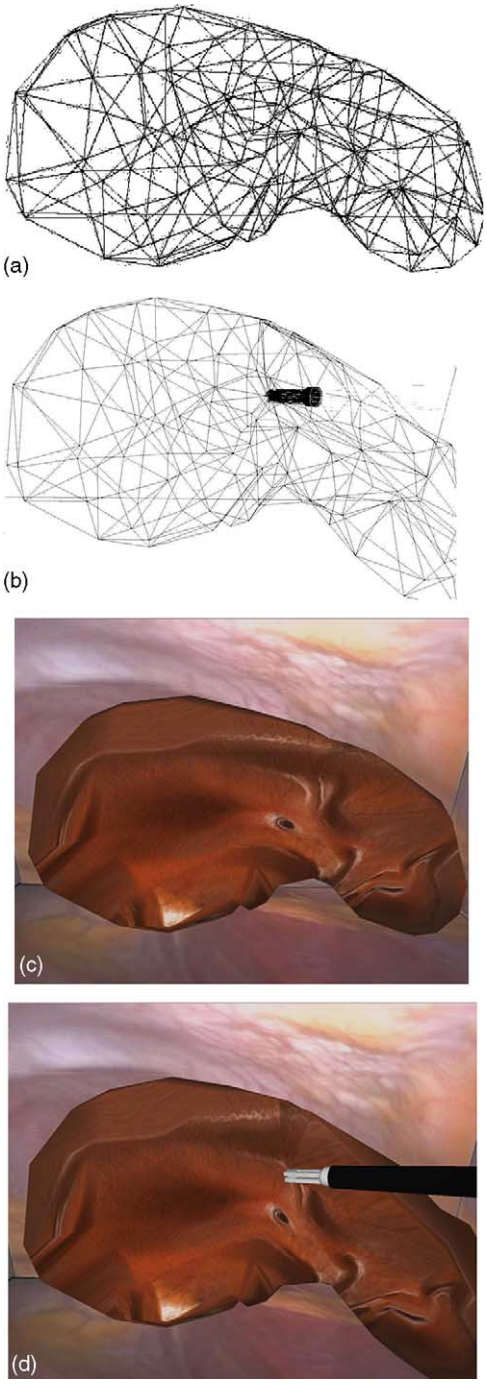

Fig. 6 Deformation of a liver modelled with a boundary element method (approximately 250 nodes): (a) wire frame view of undeformed liver; (b) wire frame view of deformed liver; (c) rendered view of undeformed liver; (d) rendered view of deformed liver.

(see Fig. 6). Hence, no costly additional boundary representation or other surface maps are required, nor does a complex discretization of the interior of a body become necessary. Furthermore, the dimension of the problem is reduced by one. However, the mathematical description is more complex, and ev-

ery node of this boundary mesh actually has a direct influence on all the other nodes.

In the simplest, linear case, one obtains,

$$\mathbf{H}\boldsymbol{U} = \mathbf{G}\boldsymbol{P} \tag{11}$$

as the resulting system of equations, where the influence matrices **H** and **G** are much smaller than the stiffness matrix **K** in the FEM case. However, in general, they are fully populated and do not possess any favourable properties such as symmetry or positive definiteness which could be taken advantage of for a simpler solution. $\boldsymbol{U}$ and $\boldsymbol{P}$, in turn, are the nodal displacement and traction (or surface force) vectors, respectively. Just as before, $\boldsymbol{U}$ mostly consists of unknown values except for the contact nodes, for which the displacement $\boldsymbol{U}^*$ is prescribed. $\boldsymbol{P}$, in turn, mostly consists of zeros, with the exception of unknown values for just the contact nodes. This means that (11) has to be reordered every time the set of constrained nodes with prescribed displacement $\boldsymbol{U}^*$ changes so as to group all known and unknown values together.

As was the case with the FEM, by far the fastest way to solve the above equation repeatedly for a slightly reordered vector of unknown values $\boldsymbol{U}'$ and a varying, known right hand side vector $\boldsymbol{U}^*$, is the preliminary inversion of the reordered $\mathbf{H}'$ before the actual simulation, therefore:

$$\boldsymbol{U}' = \mathbf{H}'^{-1}\boldsymbol{U}^*. \tag{12}$$

In general, this equation is comparable to the compressed system obtained with the fast finite element approach described above. Accordingly, the BEM approach is just as robust and quick as FEM. Also, both techniques stem from the same linearly elastic hypotheses so that their realism is equally limited for great or rapid deformations.

However, BEM tend to give even more exact results than FEM, in particular for force peaks like those created with the tips of surgical instruments. The most important difference with respect to the FEM approach is the fact that $\mathbf{H}'$ has been obtained from **H** and **G** by simple column exchanges and not by complex compression algorithms or tensor pre-computations. Thus, whenever the set of constrained nodes changes, only some columns of $\mathbf{H}'$ are modified, and the adaptation of $\mathbf{H}'^{-1}$ becomes possible in real time with the help of the Woodbury formula [61]. The same applies to topological changes, in which case, the matrix has to be amplified as well to consider new, additional nodes. Thus, local incisions can be represented with reasonable computational effort.

Finally, a quasi-viscosic behaviour is achieved by handling the computed static deformational state as a target state. The actual nodal positions $\boldsymbol{r}_t^i$ are then determined as an interpolation between the previous nodal positions $\boldsymbol{r}_{t-\Delta t}^i$ and the target positions $\tilde{\boldsymbol{r}}_t^i$ as

$$\boldsymbol{r}_t^i = \boldsymbol{r}_{t-\Delta t}^i + (\tilde{\boldsymbol{r}}_t^i - \boldsymbol{r}_{t-\Delta t}^i)\, \mathrm{e}^{-\alpha(d^i/\Delta t)}, \tag{13}$$

where d^i is the distance to the closest contact node, for which a displacement is prescribed, and Δt is the elapsed time since the last computation of the deformation state. The heuristic parameter α determines the speed with which the deformable object occupies the target state. Thus, contact nodes immediately reach their corresponding target positions, while other nodes require more time to do so the further away they are from any contact node. This heuristic extension of the static boundary element approach yields quite satisfying results as far as speed and even mechanical realism are concerned. However, there are satisfying results only as long as the modelled objects are not concave; then, the computation of the distances d^i, which has to be repeated every time the set of contact nodes changes, becomes quite laborious.

4. Hybrid model

The last deformable model developed to date is actually not a single independent method but rather a combination of two of the models presented above. It makes use of the fact that the mass-tensor model has been developed by starting from a finite element approach, i.e., from the same tetrahedral mesh types [34,62]. Thus, while FEM-based techniques are characterized by their speed and robustness, they also lack topological flexibility and deformational realism for great deformations. The mass-tensor model, in turn, is slower and less stable but allows for certain non-linear behaviour, and even cutting is possible, although only by removing entire elements.

Therefore, when limiting the area of an organ within which a cut can be expected during a simulation, this part can be modelled with a mass-tensor type model that tolerates such an intervention. The rest of the organ should be modelled with the tensor pre-computation approach in order to optimize computation time. As far as the interface between the two is concerned, this is designed to comply with the fact that the FEM-type method only supports prescribed displacements as boundary conditions. The displacements of the common nodes shared by both models are hence imposed by the mass-tensor model, which, in turn, receives the resulting forces as its input.

While the general properties of each technique are preserved, the overall computational performance of this approach is comparable to an accelerated mass-tensor model. To be efficient, however, the region attributed to the FEM-type model should comprise more than half of the organ. Thus, the pre-selection of sections where cuts might be performed and where they cannot be performed, has to be repeated for each scenario. What is more, nowadays, the pre-selection also has to be carried out manually because of the interior nodes. As a result, in practice this approach is inherently inefficient.

5. Comparison

None of the deformable models presented above simultaneously exhibits all of the sought-after characteristics required in surgery simulation such as speed, robustness, physiological realism, and topological flexibility. Therefore, a more thorough comparison will be carried out here. This is a difficult task, since only limited data has been published on the performance of each model.

In an attempt to make a more systematic comparison, first, a detailed list of the different features to be considered and their respective importance is elaborated. As far as the computations are concerned, both speed and robustness are the most important characteristics. In addition, the extent of possible pre-computations and the dimension of the system of equations, i.e., the amount of data to be handled must also be considered here.

The most important topological aspect of a deformable model is its capacity to cope with modifications of the mesh like when cutting. Moreover, its predisposition to be rendered and, to a lesser degree, the dimension of the mesh have a direct influence on the velocity of the model. And even though there is no fully automatic technique yet to create the mesh data for all kinds of anatomical structures from medical images, the creation of an appropriate mesh still involves quite different amounts of effort and shall therefore also be considered here.

Biomechanical realism, comprises a general plausibility of the deformations, i.e., consistency with the imposed movements of the instruments, immediacy, and globality. Therefore, incompress-

Table 1 Comparison between deformable models

	Weight	Deformable splines	Spring-mass model	Linked volumes	Tensor-mass model	Hybrid model	Tensor pre-computation	BEM
Computation								
Velocity	7	3	6	0	4	5	7	7
Robustness	5	3	2	4	4	4	5	5
Pre-computations	2	2	2	2	1	1	0	0
Dimension of system of equations	1	0	1	0	0.5	0.5	1	1
Total	15	8	11	6	9.5	105	13	13
Topology								
Flexibility (incisions)	6	3	5	6	3	2	0	4
Rendering	5	3	5	2	5	5	5	5
Mesh dimension	2	1	2	0	1	1	1	2
Mesh data acquisition	2	1	2	1.5	0.5	0.5	0.5	2
Total	15	8	14	9.5	9.5	8.5	6.5	13
Biomechanical realism								
Deformational behaviour	4	1.5	3	3	3	3	3	4
Volumetric constancy	4	3	4	4	4	3	0	0
Great deformations	3	0	0	3	4	3	2	2
Dynamics	2	1	2	2	2	1	0	1
Determination of parameters	2	0	0	0	1	1	2	1
Total	15	5.5	9	12	14	11	7	8
Total	45	21.5	34	27.5	33	30	26.5	34

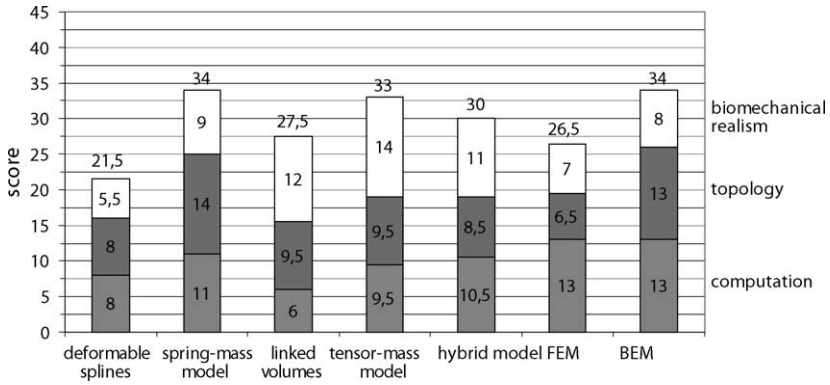

Fig. 7 Graphic comparison between deformable models.

ibility, ability to represent large deformations, and dynamical behaviour (viscoelasticity) have to be taken into account. Finally, the number of parameters controlling the deformational behaviour is important, as well as the way these are determined and whether they have an influence on the stability of the system of equations.

From the actual comparison in Table 1 and its visualization in Fig. 7, it becomes clear that there is no outstanding deformable model that is better than the rest. There are rather three different models that yield comparable overall results. Hence, depending on the feature to be emphasized, one model might be preferable over the other. For instance as a BEM-based model is only a non-iterative model, it is computationally quicker and more robust than the others. The spring-mass model, in turn, offers the highest topological flexibility and simplicity for rendering. Finally, the only approach that combines both good volumetric behaviour and realism of large deformations is the tensor-mass model.

However, this comparison does not reflect the complexity of the different models. Thus, the triangular surface meshes required by a BEM-based model or a spring-mass approach are still relatively easy to create and to modify, while the tensor-mass model involves a complex tetrahedral mesh. As far as the system of equations is concerned, the spring-mass type system is the easiest one to create, while the BEM-based model entails the most complex precomputations.

6. Conclusions

At present, there does not yet exist an optimum deformable model that complies with all the different requirements of surgery simulation. The existing approaches emphasize one aspect to the detriment of others. Thus, basically it is the simplicity and small size of the spring-mass type mesh and the corresponding system of equations that make this deformable model the most widely used one for applications in surgery simulation. This approach has been successful in modelling even complex interactions with relative ease making.

However, the preceding evaluation is based on the performances with currently available hardware configurations. In the future, the computational power of processors along with their memory is expected to increase rapidly. This additional capacity will not only be used to increase the complexity of surgical simulation scenarios but also offers the potential to increase the realism of deformable models. However, the models that are currently considered to be the best may not be have greatest potential for improvement. The spring-mass model has already almost reached its maximum capacity. The linked volume technique (with its comparable simplification) has a high potential of becoming the most widely used approach as it offers a largely improved volumetric behaviour.

Likewise, more complex techniques, such as the tensor-mass model or the continuum-mechanical approaches, will have to offer a better performance than the spring mass or linked volume models to make the required extra effort worthwhile. Thus, the highest potential might be with the BEM-based model, as it requires a mesh that is comparably as simple as the spring-mass model. However, its future largely depends on its capacity to overcome the current restriction to linearity, which affects mostly the realism of large deformations. In the long run, however, we expect continuum-mechanical methods to be able to accurately simulate the deformation of human tissues in surgery simulation.

References

[1] T.W. Sederberg, S.R. Parry, Free-form deformation of solid geometric models, Comput. Graph. 20 (4) (1986) 151—160.

[2] D. Terzopoulos, et al., Elastically deformable models, Comput. Graph. 21 (4) (1987) 205—214.

[3] S. Coquillart, Extended free-form deformation: a sculpturing tool for 3D geometric modeling, Comput. Graph. 24 (4) (1990) 187—196.

[4] S. Coquillart, P. Jancène, Animated free-form deformation: an interactive animation technique, Comput. Graph. 25 (4) (1991) 23—26.

[5] W.H. Hsu, Direct manipulation of free-form deformations, Comput. Graph. 26 (2) (1992) 177—184.

[6] M.-E. Algorri, F. Schmitt, Deformable models for reconstructing 3D data, in: N. Ayache (Ed.), Proceedings of the Computer Vision and Robotics in Medicine (CVRMed: 1), Lecture Notes in Computer Science, vol. 905, Springer, Berlin, 1995.

[7] D. Serby, M. Harders, G. Székely, A new approach to cutting into finite element models, in: Proceedings of the Medical image computing and computer-assisted intervention (MICCAI: 4), Lecture Notes in Computer Science, vol. 2208, Springer, Berlin, 2001, pp. 425—433.

[8] J.M. Gauch, Hybrid boundary-based and region-based deformable models for biomedical image segmentation, SPIE 2299 (1994) 72—83.

[9] V. Grau, Interpretación Automática de Imágenes Médicas mediante Técnicas Multiescala, Ph.D. Thesis, Universidad Politécnica de Valencia, 2001.

[10] F. Leitner, P. Cinquin, Complex topology 3D-objects' segmentation, SPIE 1609 (1991) 16—26.

[11] J.V. Miller, et al., Geometrically deformed models: a method for extracting closed geometric models from volume data, Comput. Graph. 25 (4) (1991).

[12] M. Neveu, D. Faudot, B. Derdouri, Recovery of 3D deformable models from echocardiographic images, SPIE 2299 (1994) 367—376.

[13] M. Bro-Nielsen, Modelling elasticity in solids using active cubes—application to simulated operations, in: N. Ayache (Ed.), Proceedings of the Computer Vision, Virtual Reality and Robotics in Medicine (CVRMed: 1), Lecture Notes in Computer Science, vol. 905, Springer, Berlin, 1995, pp. 535—541.

[14] H. Delingette, G. Subsol, J. Pignon, A Cranofacial Surgery Simulation Testbed, INRIA, Research Report RR2199, 1994 http://www.inria.fr/rrrt/rr-2199.html.

[15] E. Keeve, S. Girod, B. Girod, Cranofacial surgery simulation, in: K.H. Böhme, R. Kikinis (Eds.), Proceedings of the Visualization in Biomedical Computing (VBC: 4), Lecture Notes in Computer Science, vol. 1131, Springer, Berlin, 1996, pp. 541—546.

[16] G. Székely, J. Duncan, Soft Tissue Deformation, Medical Image Analysis, 5, editorial, 2001, p. 229.

[17] H. Fischer, Sensor-Aktorsysteme für den Einsatz in der laparoskopischen Chirurgie, Scientific Report FZKA 5898, Forschungszentrum Karlsruhe, Karlsruhe, 1997.

[18] N. Suzuki, et al., Simulator for virtual surgery using deformable organ models and force feedback system, in: J.D. Westwood, et al. (Eds.), Proceedings of the Medicine Meets Virtual Reality (MMVR: 6), Studies in Health Technology and Informatics, vol. 50, IOS Press, Amsterdam, 1998, pp. 227—233.

[19] M. Bro-Nielsen, Surgery simulation using fast finite elements, in: K.H. Böhme, R. Kikinis (Eds.), Proceedings of the Visualization in Biomedical Computing (VBC: 4), Lecture Notes in Computer Science, vol. 1131, Springer, Berlin, 1996, pp. 529—534.

[20] J.P. Gourret, N. Magnenat-Thalmann, D. Thalmann, Modeling of contact deformations between a synthetic human and its environment, Comput. Aided Des. 23 (7) (1991) 514—520.

[21] Y. Lee, D. Terzopoulos, K. Waters, Realistic modeling for facial animation, Comput. Graph. 29 (1) (1995) 55—62.

[22] C. Monserrat, U. Meier, F. Chinesta, M. Alcañiz, V. Grau, A fast real time tissue deformation algorithm for surgery simulation, in: H.U. Lemke, et al. (Eds.), Proceedings of the Computer Aided Radiology and Surgery (CARS: 11), Elsevier, Amsterdam, 1997.

[23] M. Bro-Nielsen, Finite Element Modelling in Surgical Simulation, Proc. IEEE 86 (3) (1998) 490—503.

[24] S.A. Cover, N.F. Ezquerra, J.F. O'Brian, R. Rowe, T. Gadacz, E. Palm, Interactively deformable models for surgery simulation, IEEE Comput. Graph. Appl. November 93 (1993) 68—75.

[25] Y. Cai, C.-K. Chui, Y. Wang, Z. Wang, J.H. Anderson, Parametric Eyeball Model for Interactive Simulation of Pphthalmologic Surgery, in: W.J. Niessen, M. Viergever (Eds.), Proceedings of the Medical Image Computing and Computer-Assisted Intervention (MICCAI: 4), Lecture Notes in Computer Science, vol. 2208, Springer, Berlin, 2001, pp. 465—472.

[26] M.A. Sagar, D. Bullivant, et al., A virtual environment and model of the eye for surgical simulation, Proc. SIGGRAPH (1996) 205—212.

[27] S.F.F. Gibson, et al., Simulating arthroscopic knee surgery using volumetric object representations, real-time volume rendering and haptic feedback, in: J. Troccaz, E. Grimson, R. Mösges (Eds.), Proceedings of the CVRMed-MRCAS'97, Lecture Notes in Computer Science, vol. 1205, Springer, Berlin, 1997, pp. 369—378.

[28] A.D. McCarthy, R.J. Hollands, A commercially viable virtual reality knee arthroscopy training system, in: J.D. Westwood, et al. (Eds.), Proceedings of the Medicine Meets Virtual Reality (MMVR: 6), Studies in Health Technology and Informatics, vol. 50, IOS Press, Amsterdam, 1998, pp. 302—308.

[29] S. Cotin, H. Delingette, N. Ayache, Real-time elastic deformations of soft tissues for surgery simulation, IEEE Trans. Visualization Comput. Graph. 5 (1) (1999) 62—73.

[30] U. Kühnapfel, C. Kuhn, M. Hübner, H.G. Krumm, B. Neisius, CAD-based Simulation and Modelling for Endoscopic Surgery, Proc. Med. Tech., SMIT'94 Berlin 1994.

[31] C. Kuhn, U. Kühnapfel, H.-G. Krumm, B. Neisius, A 'Virtual Reality' Based Training System for Minimally Invasive Surgery, in: H.U. Lemke, et al. (Eds.), Proceedings of the Computer Assisted Radiology (CAR: 10), Elsevier Science, Amsterdam, 1996, pp. 764—769.

[32] S. De, M.A. Srinivasan, Thin Walled Models for Haptic and Graphical Rendering of Soft Tissues in Surgical Simulations, in: J.D. Westwood, et al. (Eds.), Proceedings of the Medicine Meets Virtual Reality (MMVR: 7), Studies in Health Technology and Informatics, vol. 62, IOS Press, Amsterdam, 1999, pp. 94—99.

[33] S.F.F. Gibson, 3D ChainMail: a fast algorithm for deforming volumetric objects, in: Proceedings of the Symposium on Interactive 3D Graphics, ACM SIGGRAPH, 1997, pp. 149—154.

[34] S. Cotin, H. Delingette, N. Ayache, Efficient Linear Elastic Models of Soft Tissues for Real-Time Surgery Simulation, INRIA Technical Report RR-3510, 1998, http://www.inria.fr/RRRT/RR-3510.html.

[35] G. Picinbono, H. Delingette, N. Ayache, Real-time large displacement elasticity for surgery simulation: non-linear tensor-mass model, in: S.L. Delp, A.M. DiGivia, B. Jaramaz

(Eds.), Proceedings of the Medical Image Computing and Computer-Assisted Intervention (MICCAI: 3), Lecture Notes in Computer Science, vol. 1935, Springer, Berlin, 2000, pp. 643—652.
[36] J. Berkley, S. Weghorst, H. Gladstone, G. Raugi, D. Berg, M. Ganter, Fast Finite Element Modelling for Surgical Simulation, in: J.D. Westwood, et al. (Eds.), Proceedings of the Medicine Meets Virtual Reality (MMVR: 7), Studies in Health Technology and Informatics, vol. 62, IOS Press, Amsterdam, 1999, pp. 55—61.
[37] S. Cotin, H. Delingette, N. Ayache, Real time volumetric deformable models for surgery simulation, in: K.H. Böhme, R. Kikinis (Eds.), Proceedings of the Visualization in Biomedical Computing (VBC: 4), Lecture Notes in Computer Science, vol. 1131, Springer, Berlin, 1996, pp. 535—540.
[38] D. Terzopoulos, K. Fleischer, Deformable Models, The Visual Computer, Springer, Berlin, 1988, pp. 306—331.
[39] D. Terzopoulos, A. Witkin, Physically based models with rigid and deformable components, IEEE Comp. Graph. Appl. 8 (6) (1988) 41—51.
[40] D. Terzopoulos, A. Witkin, M. Kass, Constraints on deformable models: recovering 3D shape and nonrigid motion, Artif. Intell. 36 (1988) 91—123.
[41] J. Brown, S. Sorkin, J.-C. Latombe, K. Montgomery, M. Stephanides, Algorithmic tools for real-time microsurgery simulation, Med. Image Anal. 6 (2000) 289—300.
[42] M. Downes, et al., Virtual environments for training critical skills in laparoscopic surgery, in: J.D. Westwood, et al. (Eds.), Proceedings of the Medicine Meets Virtual Reality (MMVR: 6), Studies in Health Technology and Informatics, vol. 50, IOS Press, Amsterdam, 1998, pp. 316—322.
[43] K. Mori, Y. Seki, J.-I. Hasegawa, J.-I. Toriwaki, H. Anno, K. Katada, A method for shape deformation of organ and its application to virtualized endoscope systems, in: H.U. Lemke, et al. (Eds.), Proceedings of the Computer Aided Radiology and Surgery (CAR: 11), Excerpta Medica, International Congress Series, vol. 1134, Elsevier Science, Amsterdam, 1997, pp. 189—194.
[44] H. Delingette, Simplex Meshes: A General Representation for 3D Shape Reconstruction, INRIA, Research Report RR2214, 1994, http://www.inria.fr/rrrt/rr-2214.html.
[45] C. Monserrat, et al., Aproximación mediante Modelos Deformables a la Reconstrucción 3D de la Anatomía Dental, in: Proceedings of the Congreso Español de Ingeniería Gráfica (CEIG: 6), Universidad Politécnica de Valencia, Valencia, 1996.
[46] C. Monserrat, Modelos Deformables de Tejidos Elásticos en Tiempo Real, Ph.D. Thesis, Universidad Politécnica de Valencia, Valencia, 1999.
[47] H. Delingette, Toward realistic soft-tissue modelling in medical simulation, Proc. IEEE 86 (3) (1998) 512—523.
[48] W. Lin, R.A. Robb, Dynamic volume texture mapping and model deformation for visually realistic surgical simulation, in: J.D. Westwood, et al. (Eds.), Proceedings of the Medicine Meets Virtual Reality (MMVR: 7), Studies in Health Technology and Informatics, vol. 62, IOS Press, Amsterdam, 1999, pp. 198—204.
[49] U. Kühnapfel, H.K. Çakmak, H. Maaß, Endoscopic surgery training using virtual reality and deformable tissue simulation, Comput. Graph. 24 (2000) 671—682.
[50] U. Kühnapfel, B. Neisius, Real-time graphical computer simulation for endoscopic surgery, in: Proceedings of the Medicine Meets Virtual Reality (MMVR: 2), San Diego, 1994.
[51] O. Deussen, L. Kobbelt, P. Tücke, Using simulated annealing to obtain good approximations of deformable bodies, in: D. Terzopoulos, D. Thalmann (Eds.), Proceedings of the EuroGraphics Workshop Computer Animation and Simulation, Springer, Berlin, 1995.
[52] S.F.F. Gibson, Using linked volumes to model object collisions, deformation, cutting, carving, and joining, IEEE Trans. Visualization Comput. Graph. 5 (4) (1999) 333—348.
[53] J. Jansson, J.S.M. Vergeest, A Discrete Mechanics Model for Deformable Bodies Computer-Aided Design, vol. 34, Elsevier, 2000, pp. 913—928.
[55] S.F.F. Gibson, Volumetric object modelling for surgical simulation, Med. Image Anal. 2 (2) (1998) 121—132.
[56] F. Boux de Casson, C. Laugier, Modelling the dynamics of a human liver for a minimally invasive surgery simulator, in: C. Taylor, LA. Colchester (Eds.), Proceedings of the Medical Image Computing and Computer-Assisted Intervention (MICCAI: 2), Lecture Notes in Computer Science, vol. 1679, Springer, Berlin, 1999, pp. 1166—1174.
[57] A.B. Mor, T. Kanade, Modifying soft tissue models: progressive cutting with minimal new element creation, in: S.L. Delp, A.M. DiGivia, B. Jaramaz (Eds.), Proceedings of the Medical Image Computing and Computer-Assisted Intervention (MICCAI: 3), Lecture Notes in Computer Science, vol. 1935, Springer, Berlin, 2000, pp. 598—607.
[58] H.-W. Nienhuys, A.F. van der Stappen, A surgery simulation supporting cuts and finite element deformation, in: Proceedings of the Medical Image Computing and Computer-Assisted Intervention (MICCAI: 4), Lecture Notes in Computer Science, vol. 2208, Springer, Berlin, 2001, pp. 145—152.
[59] U. Meier, C. Monserrat, N.-C. Parr, F.J. García, J.A. Gil, Real-Time Simulation of Minimally-Invasive Surgery with Cutting Based on Boundary Element Methods, in: W.J. Niessen, M. Viergever (Eds.), Proceedings of the Medical Image Computing and Computer-Assisted Intervention (MICCAI: 4), Lecture Notes in Computer Science, vol. 2208, Springer, Berlin, 2001, pp. 1263—1264.
[60] C. Monserrat, U. Meier, M. Alcañiz, F. Chinesta, M.C. Juan, A new approach for the real-time simulation of tissue deformations in surgery simulation, Comp. Meth. Progr. Biomed. 64 (2001) 77—85.
[61] W.H. Press, et al., Numerical Recipes in C, second ed., Cambridge University Press, Cambridge, 1992.
[62] S. Cotin, H. Delingette, N. Ayache, et al., Efficient linear elastic models of soft tissues for real-time surgery simulation, in: J.D. Westwood (Ed.), Proceedings of the Medicine Meets Virtual Reality (MMVR: 7), Studies in Health Technology and Informatics, vol. 62, IOS Press, Amsterdam, 1999, pp. 100—101.