

Dynamic Spherical Volumetric Simplex Splines with Applications in Biomedicine

Yunhao Tan*, Jing Hua†
Wayne State University

Hong Qin‡
Stony Brook University

Abstract

This paper presents a novel computational framework based on dynamic spherical volumetric simplex splines for simulation of genus-zero real-world objects. In this framework, we first develop an accurate and efficient algorithm to reconstruct the high-fidelity digital model of a real-world object with spherical volumetric simplex splines which can represent with accuracy geometric, material, and other properties of the object simultaneously. With the tight coupling of Lagrangian mechanics, the dynamic volumetric simplex splines representing the object can accurately simulate its physical behavior because it can unify the geometric and material properties in the simulation. The visualization can be directly computed from the object's geometric or physical representation based on the dynamic spherical volumetric simplex splines during simulation without interpolation or resampling. We have applied the framework for biomechanic simulation of brain deformations, such as brain shifting during the surgery and brain injury under blunt impact. We have compared our simulation results with the ground truth obtained through intra-operative magnetic resonance imaging and the real biomechanic experiments. The evaluations demonstrate the excellent performance of our new technique presented in this paper.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Physically based modeling

1 Introduction

Physically-based modeling and simulation of digitized real-world models are still extremely challenging tasks. Among many important aspects of simulation, the accuracy is of utmost importance since only physically realistic simulation can be used to represent the true reality and provide valuable information for the simulation-based assessment and analysis. In existing approaches, several different representations are typically required throughout the simulation of real-world models in computerized environments. That is to say, each stage within the entire physical simulation pipeline, including modeling (e.g., meshing, material modeling), simulation, analysis, visualization, typically takes as input a different representation of the modeled object, which requires costly and error-prone data conversions throughout the entire simulation process. It will certainly introduce error into the pipeline. For instance, in order to simulate the brain deformation, a linear solid mesh needs to be generated for finite element methods (FEMs) from the voxel-based representation of the brain representing the geometry of the brain (which has a highly convoluted cortical surface and many subtle sub-cortical structures). Then, manual material editing needs to be conducted to assign material properties to solid meshes. The FEM

properties are linearly interpolated during simulation and resampled once again to voxels' intensities for visualization. Certainly, conversions among volumetric datasets, solid meshes, finite elements, and voxels based on linear interpolation or resampling will introduce error. In addition, more errors will be brought into the pipeline as the constructed linear solid mesh may not well represent both geometry and material distribution simultaneously. The geometric, physical, and mechanical properties are not tightly integrated into the simulation. As a result, the current practice impedes the accurate modeling and simulation of digital models of real-world objects. With ever-improving computing power comes the strong demand for more accurate, robust, and powerful solid modeling and simulation paradigms that are efficacious for the modeling, simulation, analysis, and visualization of digital models of real-world objects.

To overcome the aforementioned deficiencies, we develop an integrated computational framework based on dynamic spherical volumetric simplex splines (DSVSS) that can greatly improve the accuracy and efficacy of modeling and simulation of heterogeneous objects since the framework can not only reconstruct with high accuracy geometric, material, and other quantities associated with heterogeneous real-world models, but also simulate the complicated dynamics precisely by tightly coupling these physical properties into simulation. The integration of geometric modeling, material modeling, and simulation is the key to the success of simulation of real-world objects. In contrast to existing techniques, our framework uses a single representation that requires no data conversion. The advantages of our framework result from many attractive properties of multivariate splines. In comparison with tensor-product NURBS, multivariate simplex splines are non-tensor-product in nature. They are essentially piecewise polynomials of the lowest possible degree and the highest possible continuity everywhere across their entire tetrahedral domain. For example, given an object of simplex splines with degree n , it can achieve C^{n-1} continuity. Furthermore, C^0 , other varying continuities, and even discontinuity can be accommodated through different knot and control point placements and/or different arrangements of domain tetrahedra in 3D. Furthermore, simplex splines are ideal to represent heterogeneous material distributions through the tight coupling of control points and their attributes. From dynamic simulation's point of view, they are finite elements which can be directly brought into finite element formulations and physics-based analysis without losing any information. Finite elements can be derived directly from the simplex spline representation, which can also be visualized via volumetric ray-casting without discretization [Hua et al. 2004]. Trivariate simplex splines are obtained through the projection of n -dimensional simplices onto 3D. Projecting them one step further onto 2D for visualization results in bivariate simplex splines of one degree higher than the original solid model, therefore, simplex splines facilitate the visualization task with an analytical, closed-form formulation. It is not necessary to perform any resampling and/or interpolation operations. Local adaptivity and local/global subdivision via knot insertion can be readily achieved.

On the application front, in recent years, tremendous efforts from biomedical research communities have been devoted into the brain simulation since accurate simulation of brain deformations can have many potential applications, e.g., computer-aided surgical planning/surgery, computer-assisted disease/injury positioning, accurate radiation therapy, and many other medical benefits [Maguire

*e-mail:tanyunhao@wayne.edu

†e-mail:jinghua@cs.wayne.edu

‡e-mail:qin@cs.sunysb.edu

et al. 1991]. Various methods are emerging for simulation of the brains in different physical environments. However, most brain volume simulation techniques still depend on linear geometric representation and FEMs as we have already described above. No advanced computational models are available for better simulation. As we all know, the brain is a highly convoluted organ rich of geometric, anatomical, and material variations. In order to obtain realistic deformation simulation of the brain, it is very important to construct a digital model which can simultaneously represent its geometry, imaging intensities, and material properties, and then integrate the properties into the biomechanic simulation. Consider that the human brain is topologically equivalent to a solid sphere, our proposed dynamic spherical volumetric simplex splines are perfect for modeling, simulation, and analysis of such an object. The spherical volumetric simplex splines are defined over a solid spherical tetrahedralization. In this paper, we apply and evaluate our simulation framework on various human brain deformations.

Our contributions in this paper can be summarized as follows:

- We develop a physical simulation framework which seamlessly integrates geometric properties, physical properties, and dynamic behaviors together. The consistent, uniform representation throughout each stage of modeling and simulation is a single degree n spherical volumetric DMS-spline. It is ideal for simulating complex, heterogeneous real-world objects.
- The heterogeneous model reconstructed from the digitalization of a real-world object is faithful and of high-fidelity in terms of its geometry and material distribution. The model reconstruction procedure is automatic, and the maximal fitting error to the original data can be controlled by user's specification interactively.
- During the simulation, the geometry and physical properties of the volumetric model can be computed using the analytic representation without any need for numerical approximations such as cubic interpolation or quadratic resampling. Hence, physical simulation, including all downstream processes, such as analysis and evaluation, can be achieved more accurately and robustly.
- We apply the dynamic spherical simplex splines scheme in the simulation and analysis of brain models. The unified scheme can achieve very accurate simulation compared with the ground-truth results because it can tightly integrate the geometric and material properties in the simulation. Our framework has great potential to provide simulation-based assessment for innovative computer-aided diagnosis of brain injury cases.

2 Previous Work

Our paper is related to the theory and application of multivariate simplex splines and physically based simulation. This section reviews the related, previous work in these fields.

2.1 Multivariate Simplex Splines

From projection's point of view, univariate B-splines can be intuitively formulated as volumetric shadows of higher dimensional simplices, i.e., we can obtain B-splines of arbitrary degree n by taking a simplex in the $(n + 1)$ -dimensional space and volumetrically projecting it onto R^1 . Motivated by this idea of Curry and Schoenberg, C. de Boor [de Boor 1976] presented a brief description of multivariate simplex splines. In essence, multivariate simplex splines are the volumetric projection of higher dimensional

simplices onto a lower dimensional space R^m . Simplex splines have many attractive properties such as piecewise polynomials over general tetrahedral domains, local support, higher-order smoothness, and positivity, making them potentially ideal in engineering design applications [Greiner and Seidel 1994]. From the point of view of blossoming, Dahmen *et al.* [Dahmen et al. 1992] proposed triangular B-splines. Later, Greiner and Seidel [Greiner and Seidel 1994] demonstrated their practical feasibility in graphics and shape design.

In contrast to theoretical advances, the application of simplex splines has been rather under-explored. Pfeifle and Seidel developed a faster evaluation technique for quadratic bivariate DMS-spline surfaces [Pfeifle and Seidel 1994] and applied it to the scattered data fitting of triangular B-spline [Pfeifle and Seidel 1996]. Recently, Rössl *et al.* [Pauly et al. 2002] presented a novel approach to reconstruct volume from structure-gridded samples using trivariate quadric super splines defined on a uniform tetrahedral partition. They used Bernstein-Bézier techniques to compute and evaluate the trivariate spline and its gradient. More recently, Hua *et al.* applied trivariate simplex splines to the representation of solid geometry, the modeling of heterogeneous material attributes, and the reconstruction of continuous volumetric splines from discretized volumetric inputs via data fitting [Hua et al. 2005]. Tan et al. applied the hierarchical simplex splines to volume reconstruction from planar images [Tan et al. 2007].

2.2 Physically Based Modeling and Simulation

Free-form deformable models were first introduced to the modeling community by Terzopoulos *et al.* [Terzopoulos and Fleischer 1988], and they have been improved by a number of researchers over the past 20 years. Celniker and Gossard developed an interesting prototype system [Celniker and Gossard 1991] for interactive free-form design based on the finite-element optimization of energy functionals proposed in [Terzopoulos and Fleischer 1988]. Bloor and Wilson developed related models using similar energies and numerical optimization [Bloor and Wilson 1990]. Welch and Witkin extended the approach to trimmed hierarchical B-splines for interactive modeling of free-form surfaces with constrained variational optimization [Welch and Witkin 1992]. Terzopoulos and Qin [Terzopoulos and Qin 1994; Qin and Terzopoulos 1995b] devised dynamic physical-based generalization of NURBS (D-NURBS). Later, they further developed dynamic triangular B-splines [Qin and Terzopoulos 1995a] paradigm for high topology surface modeling. The new paradigm on simplex spline finite elements is substantially more sophisticated and is expected to produce even more true-to-life simulation results.

As for simulation of digital models of real-world objects, researchers mainly focused on FEM meshing, which can represent the shape of the objects, and physical laws, which govern the model's behavior. Zhang *et al.* presented a method for 3D mesh generation from imaging data [Zhang et al. 2005]. They further designed an algorithm for automatic 3D mesh generation for a domain with multiple materials. In general, the main objective of FEM meshing is to construct a nicely-shaped elements which can represent both geometry and material of the real-world models for accurate and robust simulation. However, due to the linear representations, it cannot accurately represent the geometric and physical properties in simulation. For simulation-based assessment of real-world objects, e.g., the brain, the main goal is to obtain an objective analysis result from the realistic simulation of the objects. In brain deformation simulation, researchers have been using it for many clinical applications [Maguire et al. 1991].

3 Dynamic Spherical Volumetric Simplex Splines

In this section, we first briefly review the theoretical background of trivariate simplex splines. Then, we formalize them to the spherical volumetric simplex splines with details on spherical domain construction. We further generalize the splines with physical dynamics and develop dynamic spherical simplex splines which can be used for modeling and simulation of real-world models.

3.1 Trivariate Simplex Splines

A degree n trivariate simplex spline, $M(\mathbf{x}|\mathbf{x}_0, \dots, \mathbf{x}_{n+3})$, can be defined as a function of $\mathbf{x} \in \mathbb{R}^3$ over the half open convex hull of a point set $\mathbf{V} = [\mathbf{x}_0, \dots, \mathbf{x}_{n+3}]$, depending on the $n+4$ knots $\mathbf{x}_i \in \mathbb{R}^3, i = 0, \dots, n+3$. The trivariate simplex splines may be formulated recursively, which facilitates point evaluation and its derivative and gradient computation. When $n = 0$,

$$M(\mathbf{x}|\mathbf{x}_0, \dots, \mathbf{x}_3) = \begin{cases} \frac{1}{|\text{Vol}_{\mathbb{R}^3}(\mathbf{x}_0, \dots, \mathbf{x}_3)|}, & \mathbf{x} \in [\mathbf{x}_0, \dots, \mathbf{x}_3], \\ 0, & \text{otherwise,} \end{cases}$$

and when $n > 0$, select four points $\mathbf{W} = \{\mathbf{x}_{k_0}, \mathbf{x}_{k_1}, \mathbf{x}_{k_2}, \mathbf{x}_{k_3}\}$ from \mathbf{V} , such that \mathbf{W} is affinely independent, then

$$M(\mathbf{x}|\mathbf{x}_0, \dots, \mathbf{x}_{n+3}) = \sum_{j=0}^3 \lambda_j(\mathbf{x}|\mathbf{W})M(\mathbf{x}|\mathbf{V} \setminus \{\mathbf{x}_{k_j}\}), \quad (1)$$

where $\sum_{j=0}^3 \lambda_j(\mathbf{x}|\mathbf{W}) = 1$ and $\sum_{j=0}^3 \lambda_j(\mathbf{x}|\mathbf{W})\mathbf{x}_{k_j} = \mathbf{x}$.

The directional derivative of $M(\mathbf{x}|\mathbf{V})$ with respect to a vector \mathbf{d} is defined as follows:

$$D_{\mathbf{d}}M(\mathbf{x}|\mathbf{V}) = n \sum_{j=0}^3 \mu_j(\mathbf{d}|\mathbf{W})M(\mathbf{x}|\mathbf{V} \setminus \{\mathbf{x}_{k_j}\}), \quad (2)$$

where $\mathbf{v} = \sum_{j=0}^3 \mu_j(\mathbf{d}|\mathbf{W})\mathbf{x}_{k_j}$ and $\sum_{j=0}^3 \mu_j(\mathbf{d}|\mathbf{W}) = 0$.

3.2 Spherical Simplex Spline Volume

Generally, trivariate simplex spline can take as input any domain with arbitrary geometry and topology due to its non-tensor-product nature. Namely, spherical simplex spline volume is defined by trivariate simplex splines over a spherical volumetric domain.

3.2.1 Spherical volumetric DMS-splines

Now denote $\mathbb{S}^3 = \{\mathbf{x} \in \mathbb{R}^3, \|\mathbf{x}\| \leq c\}$ a solid sphere in \mathbb{R}^3 . Without loss of generality, let \mathbb{S}^3 be a unit solid sphere, i.e., $c = 1$. Let \mathbf{T} be an arbitrary ‘‘proper’’ tetrahedralization of \mathbb{S}^3 . Here, ‘‘proper’’ means that every pair of domain tetrahedra are disjoint, or share exactly one vertex, one edge, or one face. To each vertex \mathbf{t} of the tetrahedralization \mathbf{T} , we assign a knot cloud, which is a sequence of points $[\mathbf{t}_0, \mathbf{t}_1, \dots, \mathbf{t}_n]$, where $\mathbf{t}_0 \equiv \mathbf{t}$. We call \mathbf{t} primary-knot and $[\mathbf{t}_1, \dots, \mathbf{t}_n]$ sub-knots. For every tetrahedron $I = (\mathbf{p}, \mathbf{q}, \mathbf{r}, \mathbf{s}) \in \mathbf{T}$, we require

- all the tetrahedra $[\mathbf{p}_i, \mathbf{q}_j, \mathbf{r}_k, \mathbf{s}_l]$ with $i + j + k + l \leq n$ are non-degenerate.
- the set

$$\Omega = \text{interior}(\cap_{i+j+k+l \leq n} [\mathbf{p}_i, \mathbf{q}_j, \mathbf{r}_k, \mathbf{s}_l]) \quad (3)$$

is not empty.

- if I is a boundary tetrahedron, the sub-knots assigned to the boundary vertices must lie outside of \mathbb{S}^3 .

We then define, for each tetrahedron $I \in \mathbf{T}$ and $i + j + k + l = n$ (in the following, we use β to denote 4-tuple (i, j, k, l)), the knot sets

$$V_{\beta}^I = [\mathbf{p}_0, \dots, \mathbf{p}_i, \mathbf{q}_0, \dots, \mathbf{q}_j, \mathbf{r}_0, \dots, \mathbf{r}_k, \mathbf{s}_0, \dots, \mathbf{s}_l]. \quad (4)$$

The basis functions of normalized DMS-splines are then defined as

$$N_{\beta}^I(\mathbf{u}) = |\det(\mathbf{p}_i, \mathbf{q}_j, \mathbf{r}_k, \mathbf{s}_l)|M(\mathbf{u}|V_{\beta}^I). \quad (5)$$

These basis functions can be shown to be all non-negative and to form a partition of unity. The trivariate spherical DMS-spline volume is the combination of a set of basis functions with control points \mathbf{c}_{β}^I :

$$\mathbf{s}(\mathbf{u}) = \sum_{I \in \mathbf{T}} \sum_{|\beta|=n} \mathbf{c}_{\beta}^I N_{\beta}^I(\mathbf{u}). \quad (6)$$

The ‘‘generalized’’ control points \mathbf{c}_{β}^I are now $(k+3)$ -dimensional vectors, including control points (p^x, p^y, p^z) for the solid geometry, and control coefficients (g^1, \dots, g^k) for the attributes, where k denotes the number of attributes associated with the geometry. The spherical DMS-splines are ideal to model genus-zero, heterogeneous solid objects. The number of physical properties is application-oriented. For a concise expression of the formulation, without loss of generality, we will deal with only one physical attribute in the following formulas.

3.2.2 Initial Construction of Spherical Volumetric Domain

Theoretically, domain tetrahedralization, \mathbf{T} , can be an arbitrary tetrahedralization of a unit solid sphere, \mathbb{S}^3 , as aforementioned in Section 3.2.1. However, in practice, two important aspects of the domain tetrahedralization should be carefully considered:

- \mathbf{T} should be as uniform as possible, i.e., minimize $\frac{\max(\text{Vol}_{I \in \mathbf{T}})}{\min(\text{Vol}_{I' \in \mathbf{T}})}$. Uniform tetrahedralization at the same hierarchical level will decrease the recursion time while hierarchical structure is needed.
- \mathbf{T} should avoid bad-shaped tetrahedra in Delaunay tetrahedralization. Bad-shaped tetrahedra, for instance, slivers, will lead to instability of the program and increase numerical error during the evaluation.

Constrained Delaunay tetrahedralization [Edelsbrunner 2001] can observe the second requirement, but it will introduce very large and very small tetrahedra thus can not comply with the first requirement. Instead, we tetrahedralize a regular icosahedron and then make use of harmonic volumetric mapping to map the tetrahedralization to a solid sphere. As a result, the solid sphere tetrahedralization is uniform and its quality is better than what constrained Delaunay tetrahedralization can offer.

Figure 1 shows the flow of domain establishment and the knots distribution. Note that, in Figure 1(d), the sub-knots associated with boundary vertices are placed outside of the sphere. The uniform tetrahedralization may be subdivided and refined when necessary, e.g., modeling discontinuity as described later.

3.3 Model Reconstruction by Data Fitting

Besides constructing the initial unit sphere tetrahedralization as the parametric domain, another preliminary step prior to the reconstruction of the continuous volumetric model using spherical simplex splines is to find a volumetric parameterization between the physical model and domain space.

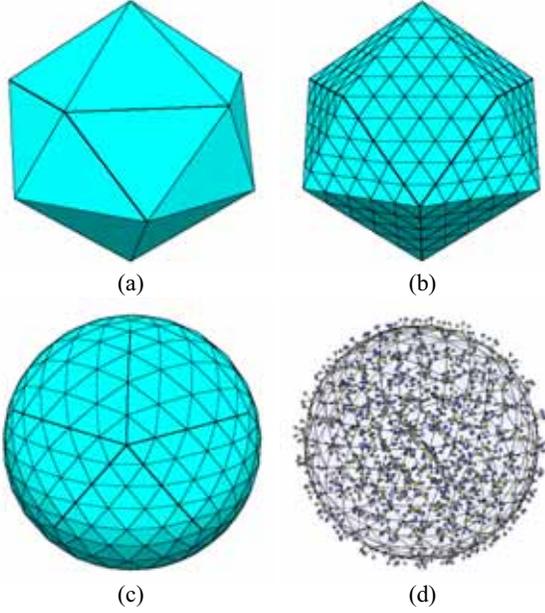


Figure 1: (a) A regular icosahedron, which is the best approximation of a solid sphere among all regular polyhedra; (b) Tetrahedralization of (a) is uniform and it is easy to implement; (c) Harmonic mapping from (b) to a unit solid sphere yields the domain tetrahedralization, consisting of uniform and well-shaped tetrahedra; (d) A domain with quadratic knot clouds assigned to (c).

3.3.1 Volumetric Parameterization

To find a volumetric parameterization of a genus-zero solid, harmonic volumetric mapping facilitates a viable solution. Harmonic volumetric mapping was first implemented for applications by Wang *et al.* [Wang et al. 2004b; Wang et al. 2004c]. They successfully exposed its merits by applying the approach to brain mapping which can be considered as a genus-zero volume. Recently Li *et al.* [Li et al. 2007] further extended the scheme to high-genus harmonic volumetric mapping and employed it in solid modeling applications.

Harmonic volumetric mapping can be formulated as follows:

Given two solid objects M_1 and M_2 , and their boundary surfaces ∂M_1 and ∂M_2 . Suppose that \vec{f}' is the conformal mapping [Gu et al. 2003; Wang et al. 2004a] between ∂M_1 and ∂M_2 , which is pre-computed. The harmonic volumetric mapping $\vec{f}: M_1 \mapsto M_2$ satisfies:

$$\begin{cases} \nabla^2 \vec{f}(v) = 0, & v \in M_1 \setminus \partial M_1, \\ \vec{f}(v) = \vec{f}'(v), & v \in \partial M_1, \end{cases}$$

where the ∇^2 is the Laplacian operator defined continuously in 3D as

$$\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2},$$

and $\nabla^2 \vec{f} = 0$ for $\vec{f} = (f_0, f_1, f_2)$ is equivalent to $\nabla^2 f_i = 0$ for all $i = 0, 1, 2$.

The algorithmic flow of harmonic volumetric mapping is concisely summarized as follows [Wang et al. 2004c]:

1. For each boundary vertex, $v, v \in \partial M_1$, let $\vec{f}(v) = \vec{f}'(v)$; for each interior vertex, $v, v \in M_1 \setminus \partial M_1$, let $\vec{f}(v) = \vec{0}$, compute the harmonic energy E_0 .

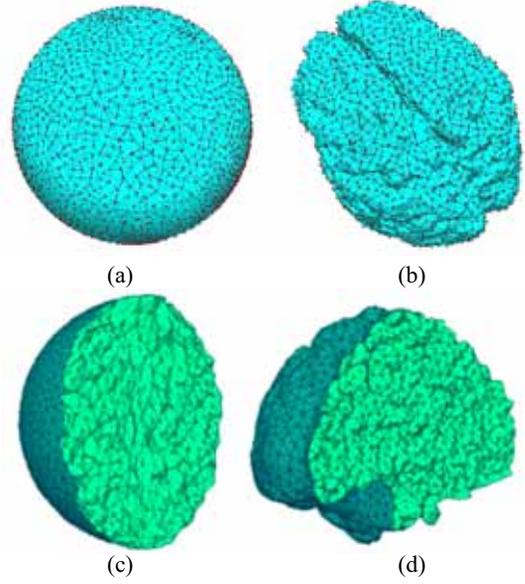


Figure 2: (a) The discretized point set in the spherical domain space; (b) The discretized data point set in the physical space, from the same angle of view as (a). (c-d) The shapes are cut into halves in order to show the interior mapping between the physical object and the parametric domain.

2. For each interior vertex, $v, v \in M_1 \setminus \partial M_1$, compute its derivative $D\vec{f}$, then update $\vec{f}(v)$ by $\delta \vec{f}(v) = -D\vec{f}(v)\delta t$, δt is the step length.
3. Compute the harmonic energy E ; if $E - E_0$ is less than user specified threshold δE , the algorithm stops; Otherwise assign E to E_0 and repeat 2 through 3.

Figure 2 shows the harmonic volumetric mapping from one brain to a solid unit sphere. After the mapping has been established, the point parameterization and correspondence between the domain and the object can now be stored as the input of our spherical simplex spline model reconstruction algorithm.

3.3.2 Fitting with Spherical Volumetric Simplex Splines

After harmonic volumetric mapping, a finite number of discretized sampling points of the physical object, $(x_i, y_i, z_i, \rho_i)_{i=1}^m$, and their parametric coordinates in the domain, $(u_i, v_i, w_i)_{i=1}^m$, can be retrieved. ρ_i denote a physical attribute. Note that, there could be multimodality physical attributes with more dimensions. In this case, all we need to do is to increase the dimensions and add the additional variables into the sampling. All the computation remains the same. Without loss of generality, we only consider one type of attribute here in order to simplify the mathematical notation. The sampling point pairs indicates the parameterization from the solid sphere domain to the to-be-modeled object. Volumetric simplex spline is an ideal tool for fitting the geometry as well as the physical properties of the volumetric object. In this section, we will describe how to fit spherical volumetric simplex splines to the real-world model.

The problem of model reconstruction in our system can be stated as follows: given a set $P = \{\mathbf{p}_i\}_{i=1}^m$ of points, $\mathbf{p}_i = (x_i, y_i, z_i, \rho_i) \in \mathbb{R}^4$, find a trivariate DMS-splines volume $\mathbf{s}: \mathbb{R}^3 \rightarrow \mathbb{R}^4$ that approximates P .

Since we are interested in reconstructing the model with respect not

only to solid geometry but also to physical attributes, our spherical DMS-spline volumes are vector functions, i.e., the control points $\mathbf{c}_\beta^I \in \mathbb{R}^4$ are vectors. Unlike the existing fitting algorithms with simplex splines which usually find the parametric domain which is close to the original geometry of the to-be-fitted dataset [Hua et al. 2004; Hua et al. 2005], we use the position (u_i, v_i, w_i) within the solid sphere as the data point \mathbf{p}_i 's parametric value. Therefore, we need to minimize the following objective function:

$$\min E_{dist}(\mathbf{s}) = \sum_{i=1}^m (\mathbf{p}_i - \mathbf{s}(u_i, v_i, w_i))^2. \quad (7)$$

Equation (7) is a typical least squares problem. If the control points are treated as free variables, it falls into a very special category of nonlinear programming, i.e., unconstrained convex quadratic programming, which has the following form:

$$E_{dist} = \frac{1}{2} \mathbf{x}^T Q \mathbf{x} + \mathbf{c}^T \mathbf{x} + f,$$

where $\mathbf{x} = (\dots, c_\beta^I, \dots)^T$,

$$Q = \begin{pmatrix} \dots & \vdots & \dots \\ \dots & 2 \sum_{i=1}^m N_\beta^I(u_i, v_i, w_i) N_{\beta'}^I(u_i, v_i, w_i) & \dots \\ \dots & \vdots & \dots \end{pmatrix},$$

$$\mathbf{c} = (\dots, -2 \sum_{i=1}^m \mathbf{p}_i N_\beta^I(u_i, v_i, w_i), \dots)^T,$$

and $f = \sum_{i=1}^m \mathbf{p}_i^2$. Note that, Q is a positive definite, symmetric and sparse matrix. Interior-point method can solve this problem very efficiently.

After reconstruction procedure, we achieve an integrated representation incorporating the object's solid geometry, \mathbf{s} , and its material attribute, \mathbf{d} , at the same time. The scheme can be expressed as

$$\begin{bmatrix} \mathbf{s} \\ \mathbf{d} \end{bmatrix}(\mathbf{u}) = \sum_{I \in \mathbf{T}} \sum_{|\beta|=n} \begin{bmatrix} \mathbf{c} \\ \alpha \end{bmatrix} N(\mathbf{u} | V_\beta^I), \quad (8)$$

where \mathbf{c} and α are the control points and control coefficients for solid geometry and material attributes, respectively.

To model discontinuity in attribute field, we first detect where the discontinuity occurs, then decompose the original domain into two separated new domains with shared vertices and edges as the 2D illustration in Figure 3. This simple mechanism maintains the consistent structure of the domains. The evaluation, hierarchy structure, and data structure all remain the same. Therefore, we can perform the same evaluation on these two domains simultaneously as if the evaluation is performed on a single domain. With the association of different control coefficients, the functional evaluation can output a discontinuity in material field corresponding to the shared edges. This change will not affect the geometry of the DSVSS volume as long as the associated control points remain the same.

3.4 Dynamic Spherical Simplex Splines

In this section, we formulate our dynamic spherical volumetric simplex splines. We integrate mass, dissipation, and deformation energy into static simplex spline models, and employ Lagrangian dynamics to derive their equations of motion. Consequently, the static control points of the geometric model become generalized time-varying physical coordinates in the dynamic model.

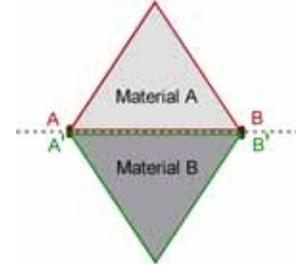


Figure 3: Modeling discontinuities with separated domain triangles. Even though A and A' are co-located, and B and B' are co-located, the domain triangles in red and green are belonged to two different domains.

3.4.1 Geometry and Kinematics of Simplex Spline Volumes

The dynamic simplex splines further extend the geometric simplex splines by incorporating time into the volume representation. Now the function of representation bears both parametric variable \mathbf{u} and time t as follows:

$$\mathbf{s}(\mathbf{u}, t) = \sum_{I \in \mathbf{T}} \sum_{|\beta|=n} \mathbf{c}_\beta^I(t) N_\beta^I(\mathbf{u}). \quad (9)$$

For simplicity of formulation expression, we define the vector of generalized coordinates of control points \mathbf{c}_β as:

$$\mathbf{c} = [\dots, \mathbf{c}_\beta^{I\top}, \dots]^\top, \quad (10)$$

where \top denotes transposition. We then express Equation (9) as $\mathbf{s}(\mathbf{u}, \mathbf{c})$ in order to emphasize its dependence on \mathbf{c} whose components are functions of time. Hence, the velocity of the dynamic simplex splines is:

$$\dot{\mathbf{s}}(\mathbf{u}, t) = \mathbf{J}\dot{\mathbf{c}}, \quad (11)$$

where the overstruck dot denotes a time derivative and Jacobian matrix $\mathbf{J}(\mathbf{u})$ is the concatenation of the vectors $\partial \mathbf{s} / \partial \mathbf{c}_\beta^I$. Assuming m tetrahedral in the parametric domain, β traverses $k = (n+1)(n+2)(n+3)/6$ possible tetrads whose components sum to n . Because \mathbf{s} is a 4-vector and \mathbf{c} is an $M = 4mk$ dimensional vector, \mathbf{J} is a $4 \times M$ matrix, which is expressed as

$$\mathbf{J} = \left[\dots, \begin{bmatrix} N_\beta^I & 0 & 0 & 0 \\ 0 & N_\beta^I & 0 & 0 \\ 0 & 0 & N_\beta^I & 0 \\ 0 & 0 & 0 & N_\beta^I \end{bmatrix}, \dots \right], \quad (12)$$

$$\text{where } N_\beta^I(\mathbf{u}) = \frac{\partial \mathbf{s}_x}{\partial \mathbf{c}_{\beta_x}^I} = \frac{\partial \mathbf{s}_y}{\partial \mathbf{c}_{\beta_y}^I} = \frac{\partial \mathbf{s}_z}{\partial \mathbf{c}_{\beta_z}^I} = \frac{\partial \mathbf{s}_d}{\partial \mathbf{c}_{\beta_d}^I}.$$

The subscripts x, y, z and d denote derivatives of the components of the 4-vector: Cartesian coordinates and physical property, respectively. Apparently, the solid volume can be presented as the production of the product of the Jacobian matrix and the generalized coordinate vector,

$$\mathbf{s}(\mathbf{u}, \mathbf{c}) = \mathbf{J}\mathbf{c}. \quad (13)$$

3.4.2 Lagrange Equations of Motion

Lagrange dynamics are widely used in physics-based shape design. In this section, we derive the equations of motion of dynamic simplex splines by applying Lagrangian dynamics [Gossick 1967]. We express the kinetic energy due to the prescribed mass distribution function $\mu(u, v, w)$, and a Raleigh dissipation energy due to a damping density function $\gamma(u, v, w)$. Both energy functions are defined over the parametric domain of the volume. The mass distribution function and damping density function are reconstructed with spherical volumetric simplex splines as well, as described in Section 3.3.2. 3D thin-plate-like energy under tension energy model [Celniker and Gossard 1991; Halstead et al. 1993; Welch and Witkin 1992; Terzopoulos 1986] is employed here in order to define an elastic potential energy,

$$U = \frac{1}{2} \iiint (\alpha_{1,1} \mathbf{s}_u^2 + \alpha_{2,2} \mathbf{s}_v^2 + \alpha_{3,3} \mathbf{s}_w^2 + \beta_{1,1} \mathbf{s}_{uu}^2 + \beta_{1,2} \mathbf{s}_{uv}^2 + \beta_{1,3} \mathbf{s}_{uw}^2 + \beta_{2,2} \mathbf{s}_{vv}^2 + \beta_{2,3} \mathbf{s}_{vw}^2 + \beta_{3,3} \mathbf{s}_{ww}^2) dudvdw. \quad (14)$$

The subscripts on \mathbf{s} denote the parametric partial derivatives. The $\alpha_{i,j}(u, v, w)$ and $\beta_{i,j}(u, v, w)$ are elasticity functions which control tension and rigidity, respectively. Other energies, requiring greater computational cost, are also applicable, for instance, the non-quadratic, curvature-based energies in [Terzopoulos et al. 1987; Moreton and Sequin 1992]. Applying the Lagrangian formulation, we obtain the second-order equations of motion

$$\mathbf{M}\ddot{\mathbf{c}} + \mathbf{D}\dot{\mathbf{c}} + \mathbf{K}\mathbf{c} = \mathbf{f}_c, \quad (15)$$

where the mass matrix is

$$\mathbf{M} = \iiint \mu \mathbf{J}^T \mathbf{J} dudvdw, \quad (16)$$

the damping matrix is

$$\mathbf{D} = \iiint \gamma \mathbf{J}^T \mathbf{J} dudvdw, \quad (17)$$

and the stiffness matrix is

$$\mathbf{K} = \iiint (\alpha_{1,1} \mathbf{J}_u^T \mathbf{J}_u + \alpha_{2,2} \mathbf{J}_v^T \mathbf{J}_v + \alpha_{3,3} \mathbf{J}_w^T \mathbf{J}_w + \beta_{1,1} \mathbf{J}_{uu}^T \mathbf{J}_{uu} + \beta_{1,2} \mathbf{J}_{uv}^T \mathbf{J}_{uv} + \beta_{1,3} \mathbf{J}_{uw}^T \mathbf{J}_{uw} + \beta_{2,2} \mathbf{J}_{vv}^T \mathbf{J}_{vv} + \beta_{2,3} \mathbf{J}_{vw}^T \mathbf{J}_{vw} + \beta_{3,3} \mathbf{J}_{ww}^T \mathbf{J}_{ww}) dudvdw. \quad (18)$$

\mathbf{M} , \mathbf{D} and \mathbf{K} are all $M \times M$ matrices. The generalized force obtained through the principle of virtual work [Gossick 1967] done by the applied force distribution $\mathbf{f}(u, v, w, t)$ is

$$\mathbf{f}_c = \iiint \mathbf{J}^T \mathbf{f}(u, v, w, t) dudvdw. \quad (19)$$

4 Finite Element Framework

The evolution of the vector of generalized coordinates, $\mathbf{c}(t)$, is determined by the second-order nonlinear differential equation. Equation (15) with physical parameter dependent matrices, does not have an analytical solution. Instead, we obtain an efficient numerical implementation using finite-element techniques.

Standard finite element methods explicitly integrate the individual element matrices into the global matrices that appear in the discrete equations of motion [Kardestuncer 1987]. Although applicable in some environments, it is infeasible in our infrastructure because of its unacceptably high computational cost. Instead, we pursue an iterative matrix solver to avoid the cost of assembling the global matrices \mathbf{M} , \mathbf{D} , and \mathbf{K} , working instead with the individual dynamic simplex spline element matrices. We construct finite element data structures which facilitates the parallel computation of element matrices.

4.1 Data Structures for Dynamic Simplex Spline Finite Elements

We define an element data structure which contains the geometric specification of the tetrahedron patch element along with its physical properties. In each element, we allocate an elemental mass, damping, and stiffness matrix, and include the quantities such as the mass $\mu(u, v, w)$, damping $\gamma(u, v, w)$, and elasticity $\alpha_{i,j}(u, v, w)$ and $\beta_{i,j}(u, v, w)$ functions. A complete dynamic simplex spline consists of an ordered array of elements with additional information. The element structure includes pointers to appropriate components of the global vector \mathbf{c} . Neighboring tetrahedra will share some generalized coordinates.

4.2 Calculation of Element Matrices

We employ Gaussian quadrature [Press et al. 1986] to numerically evaluate the integral expressions for the mass, damping, and stiffness matrices associated with each element. In this section, we explain the expression of the element damping matrix in detail; the expressions of mass and stiffness matrix will follow suit. Assuming the parametric domain of the element is $I(v_0, v_1, v_2, v_3)$ where v_i denotes the vertex, the expression for entry d_{ij} of the damping matrix takes the integral form

$$d_{ij} = \int_{I \in T} \int_{I(v_0, v_1, v_2, v_3)} \gamma(u, v, w) f_{ij}(u, v, w) dudvdw, \quad (20)$$

where f_{ij} is evaluated using the recursive expression in Equation (1). Given integers N_g , we can find the corresponding Gauss weights a_g , and parametric abscissas u_g, v_g , and w_g such that d_{ij} can be approximated by

$$d_{ij} \approx \sum_{g=1}^{N_g} a_g \gamma(u_g, v_g, w_g) f_{ij}(u_g, v_g, w_g). \quad (21)$$

In our system, we choose N_g to be 10 for cubic dynamic simplex splines. Because of the irregularity of the knot distribution, many of the f_{ij} vanish over the sub-space of $I(v_0, v_1, v_2, v_3)$. We can further subdivide the $I(v_0, v_1, v_2, v_3)$ to minimize the numerical error.

4.3 Discrete Dynamics Equations

In this section, we will derive the discrete dynamics equations based on Equation (15). In order to integrate it in a simulation system, e.g., tissue simulation during surgery, it is important to provide users with visual feedback about the evolution state of the DSVSS model. Rather than using computation-intensive time integration methods which may traverse the largest possible time steps, it is

more crucial to provide a smoothly simulated display by maintaining the continuity of the dynamics from one step to the next. Therefore, it is much desirable to employ less costly yet stable time integration methods that take reasonable time steps.

The state of the dynamic simplex splines at time $t + \Delta t$ is integrated using prior states at t and $t - \Delta t$. To maintain the stability of the integration scheme, especially for high stiffness configurations with large elasticity functions, we use an implicit time integration method, which employs discrete derivatives of \mathbf{c} using backward differences. The velocity expression is

$$\dot{\mathbf{c}}^{t+\Delta t} \approx (\mathbf{c}^{(t+\Delta t)} - \mathbf{c}^{(t-\Delta t)})/2\Delta t \quad (22)$$

and the acceleration expression is

$$\ddot{\mathbf{c}}^{t+\Delta t} \approx (\mathbf{c}^{(t+\Delta t)} - 2\mathbf{c}^{(t)} + \mathbf{c}^{(t-\Delta t)})/\Delta t^2 \quad (23)$$

Then the time integration formula can be expressed as

$$(2\mathbf{M} + \Delta t\mathbf{D} + 2\Delta t^2\mathbf{K})\mathbf{c}^{(t+\Delta t)} = 2\Delta t^2\mathbf{f}_c + 4\mathbf{M}\mathbf{c}^{(t)} - (2\mathbf{M} - \Delta t\mathbf{D})\mathbf{c}^{(t-\Delta t)} \quad (24)$$

where the superscripts denote evaluation of the quantities at the indicated times. The matrices and forces are evaluated at time t . Our extensive experiments have shown that this discretization scheme produces satisfactory results. Instability due to large transient applied forces can be reduced by shortening the time integration step adaptively.

The equations of motion allow physically realistic simulation of real-world models with complex dynamics. However, it is possible to make simplifications to the equations of motion to further reduce the computational cost of solving Equation (24) when we simulate some more complicated volumes which bears more tetrahedra in its domain. In certain solid modeling and simulation applications where the inertial terms are not taken into count, the Equation (15) can be simplified by setting the mass density function to zero. Without computation of the acceleration terms or storage of mass matrices, the algorithm is more efficient. With zero mass density, Equation (15) simplifies to

$$\mathbf{D}\dot{\mathbf{c}} + \mathbf{K}\mathbf{c} = \mathbf{f}_c \quad (25)$$

Discretizing the corresponding derivatives of \mathbf{c} in Equation (25) with backward differences, the integration formula becomes

$$(\mathbf{D} + \Delta t\mathbf{K})\mathbf{c}^{(t+\Delta t)} = \Delta t\mathbf{f}_c + \mathbf{D}\mathbf{c}^{(t)} \quad (26)$$

5 Brain Simulation Using DSVSS Volume

With the reconstruction of brain model from both MRI data and material map using our spherical volumetric simplex splines, we can obtain an analytic representation simultaneously describing both geometric and physical properties of the brain. Thus, brain simulations, such as brain shifting, deformation, and brain injury prediction, can be achieved via the simulation-based analysis. In this section, we present the accurate brain reconstruction and simulation using our unified scheme, DSVSS volume.

5.1 Fitting Spherical Volumetric Simplex Splines to Brain Data

Taking a set of high-resolution brain SPGR MR scans, we first strip away the skull and only retain the brain volume as shown in Figure 4(a). With the initial tetrahedralization of the brain model and harmonic volumetric mapping, we can obtain the parameterization of the data points of the brain tetrahedralization as described in Section 3.3, i.e., the parameterization describes the correspondence between the brain data points and parametric coordinates in the sphere domain. Fitting spherical volumetric simplex splines to the geometric representation, we can reconstruct the geometry of the brain nicely as shown in Figure 2. To model the intensities (for visualization purpose) and material distribution (for simulation purpose), we can start with the same spherical tetrahedral domain, and then subdivide and refine the domain [Tan et al. 2007], when necessary, to model more sophisticated material variations or discontinuities as described in Section 3.3.2. Note that, the intensities and material of brain structures are related since the imaging procedure can be considered as a function mapping of the material maps to scanned images. So the required domain for intensities and material distributions are very similar. Figure 4 shows the reconstruction result with different rendering techniques and Equation(27) shows the reconstructed representation,

$$\begin{bmatrix} \mathbf{s} \\ \mathbf{d} \\ \mathbf{I} \end{bmatrix}(\mathbf{u}) = \sum_{I \in \mathbf{T}} \sum_{|\beta|=n} \begin{bmatrix} \mathbf{c} \\ \alpha_d \\ \alpha_I \end{bmatrix} N(\mathbf{u}|V_\beta^I), \quad (27)$$

where \mathbf{s} denotes the solid geometry of the brain, \mathbf{d} denotes the reconstructed physical attributes of the brain, and \mathbf{I} denotes the reconstructed image intensities from the high-resolution SPGR MRI sequence. \mathbf{c} , α_d and α_I are the control points and control coefficients. The accuracy of the data fitting is documented in the experimental result section. After obtaining high-quality DSVSS volume representation of the brain model, we can use it to simulate brain deformation during surgery for computer-assisted surgical planning/surgery, or even for an innovative simulation-based diagnosis for brain injury under blunt impact.

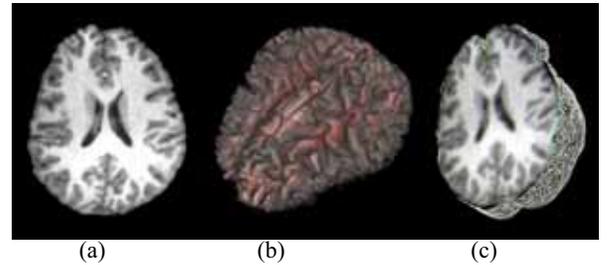


Figure 4: (a) An axial view of a slice high-resolution brain SPGR MRI dataset; (b) Volume visualization of the reconstructed DSVSS volume; (c) The volume is split to show its reconstructed interior intensities.

5.2 Brain Shifting during Surgery

As known by brain surgery professionals, after a patient's skull is open, the brain will behave increasing deformation, known as brain shifting, during ongoing surgical procedures, predominantly due to the gravity and the drainage of cerebrospinal fluid. This will inevitably lead to the repositioning of the surgical targets embedded in brain. As a compensation to increase the spatial accuracy of modern neuronavigation systems, intraoperative magnetic resonance imaging (IMRI) is widely used for quantitative analysis and

visualization of this phenomenon [Nimsky et al. 2000]. Nevertheless, despite its virtually real-time aspects, IMRI only provides very low-resolution intraoperative MR image which can never substitute the high-resolution pre-operative SPGR MR image used to determine with high accuracy key dimensions of the brain and the locations of the surgical targets embedded in the brain. We employ our dynamic spherical volumetric simplex splines model into the brain simulation to compute the brain shifting.

In our framework, brain shifting can be simulated by applying constant gravity force \vec{G} to the brain. The material properties that we used in our experiments were obtained from the *biomechanics group* at Wayne State University (WSU). After setting up the physical parameters of an individual brain, we also need to take the nature boundary of the brain, the skull, into consideration. The fact is that no matter what manner the brain behaves deformation, it lies inside the skull, i.e., its nature boundary will not exceed the skull. Therefore, spatial geometric constraints need to be enforced. We add the soft constraints with forces. When there is shifting outside the boundary, we insert corresponding forces along the opposite direction of the movement to the simulation procedure.

Figure 5 illustrates the brain shifting simulation using our framework when taking out the resected skull over the right temporal lobe. The green contour shows the deformation clearly. Our shifting simulation results highly agree with the fact captured by IMRI. Based on 20 simulation experiments, quantitative comparison between the IMRI volumes and our simulated brain volumes by co-registration shows that our system can achieve an excellent accuracy of 92.2%. The result shows that it is effective to use our model to recover motion and deformation from image data. Figure 6 depicts another brain shifting simulation. The skull is imaginarily resected over the left temporal lobe. The color map is blended into the figure to better visualize the deformation scale. Note that, when surgical tools are operating in the brain, there will be larger shifting and deformation.

As demonstrated from the available comparison and evaluation, our framework can accurately simulate the deformation of the brain (e.g. $s(t)$) and simultaneously present high-quality and high-resolution visualization using the transformed SPGR image intensities, I , modeled in the reconstructed simplex spline volume (see Equation 27). It is very promising to use the framework in both surgical planning (e.g., predicting the shifting of the targets) and computer-assisted surgery (e.g., repositioning the targets with high-resolution display, I , automatically computed based on the realistic deformation of the reconstructed brain, $s(t)$).

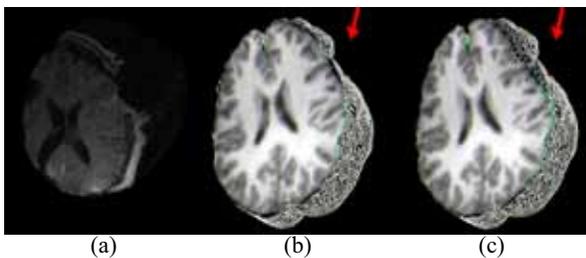


Figure 5: (a) One slice view of IMRI image; (b) The reconstructed DSVSS volume, where the cross-sectional view displays the DSVSS-captured image intensities reconstructed from the pre-operative high-resolution SPGR images; (c) The brain deformation simulated using our system, where the cross-sectional view is captured, from the same view angle as (b), to show the displacement from (b), and the green contour indicates the extent of displacement at the boundary. In (b) and (c) the red arrow denotes the orientation of gravity, and its position denotes the resected skull.

5.3 Brain Injury Prediction

Here, we refer the brain injury prediction as a procedure of finding out the extent and location of the injury in the brain during a blunt impact. The injury frequently occurs to automobiles drivers during the collision and sports players during the acute sports activities such as football. Current brain surgeons and professionals rely indispensably on those modern neuroimaging and neuronavigation systems to pinpoint the injury. Clinically, the identification of the site and extend of injury within the brain without subjecting the patient to an imaging scanning, has its advantages. For instance, head injured patients are difficult to control and may not remain still long enough for the completion of the scanning. In some severe cases, time is so limited that patients even can not afford such a pre-operative scanning. Thus the demand of simulation-based Computer Aided Diagnosis (CAD) solution goes up to high gear. Oftentimes, the solution is referred as “brain injury modeling”.

One critical issue about BIM technique is to derive a patient-specific brain model based on a template model, thus skipping neuroimaging and neuronavigation, and saving computational time as well as pre-operation time. One widely employed way is to modify the exterior surface of each substructure from a general brain model followed by re-generation of the mesh. Ferrant et al. [Ferrant et al. 2000] and Miga et al. [Miga et al. 2003] developed their approaches respectively using this approach by meshing the entire brain without considering anatomical structures and material difference. Obviously, this approach is not accurate since the brain geometry, structures, and heterogeneous material variations are not considered. We employ our dynamic spherical simplex splines-based simulation framework to handle the situation. As for developing a patient-specific model, our method can quickly modify the control points/coefficients according to the data fitting of the available data or information of the patient.

In our framework, we compute the stress field of the human brain under blunt impact using our DSVSS volume. Because the human brain has highly heterogenous physical properties in different areas of the brain, such as the white matters, the gray matters, the cerebellum, the brainstem, the lateral ventricles, the third ventricles, the bridge veins, and so on. From this perspective, brain structures under direct impact are not necessary the parts where brain injuries occur. With our unified solid representation through dynamic spherical volumetric simplex splines, blunt-impact injury can be simulated using our framework by applying an instantaneous impact to the brain model under given approximate impact conditions. The model incorporated in our framework can not only assist the physician in identifying the location and extend of damaged area without pre-operative scanning but also enable the designer of automobiles and helmets to improve the human-centered design of head-protective facilities.

Figure 7 demonstrates a brain injury prediction with an imaginary blunt-impact on the frontal lobe. Time interval here is 3ms. Note that, we assume that the brain always lies inside the skull during the simulation. As in brain shifting simulation, we add corresponding contacting forces into the simulation when the brain is shifting outside the boundary. The corresponding contacting forces is along the opposite direction and linear to the extent of the brain movement. Figure 7(b-j) shows the stress fields of the brain in each time step. Redder area indicates higher stress, which is a sign for a higher possibility of injury and bleeding. In the figure, the thalamus is under bigger stress as well beside the place under direct blunt impact. The result complies with the ground truth captured from the real biomechanic experiments on a human corpus model. Quantitative evaluation of our simulation result is obtained through the comparison with the ground truth. Figure 8 shows two stress evolution curves of one landmark inside right thalamus under the specified blunt im-

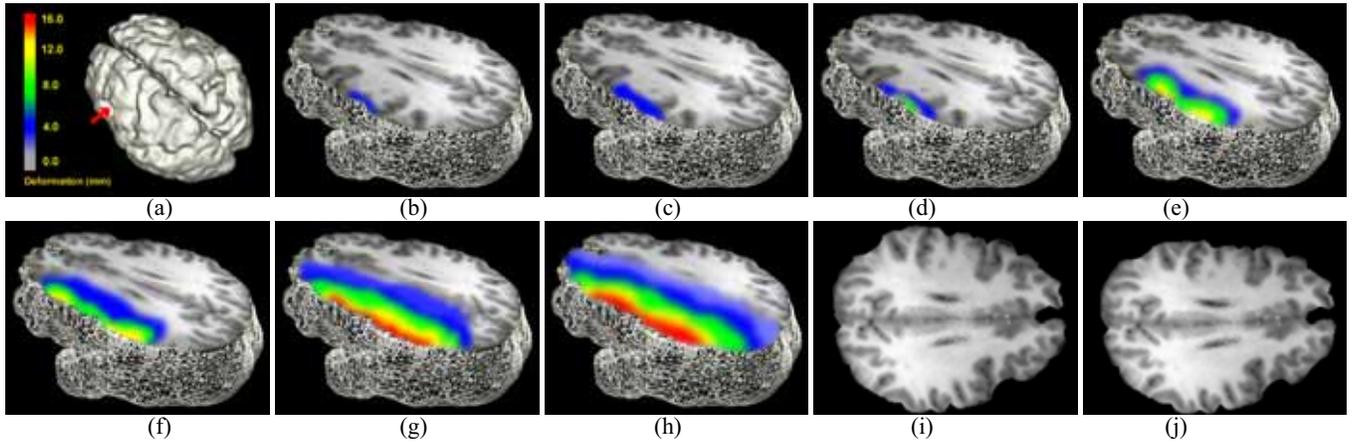


Figure 6: (a) The color map used to describe the deformation scale. The red arrow on the ISO-surface indicates the position where skull is resected. (b)-(h) Brain shifting simulation with a time interval of 5ms. (i)-(j) To better visualize the deformation, cross-section views of the first key frame (b) and last one (h) are retrieved. Deformed junction between the two hemispheres indicates the global brain shifting.

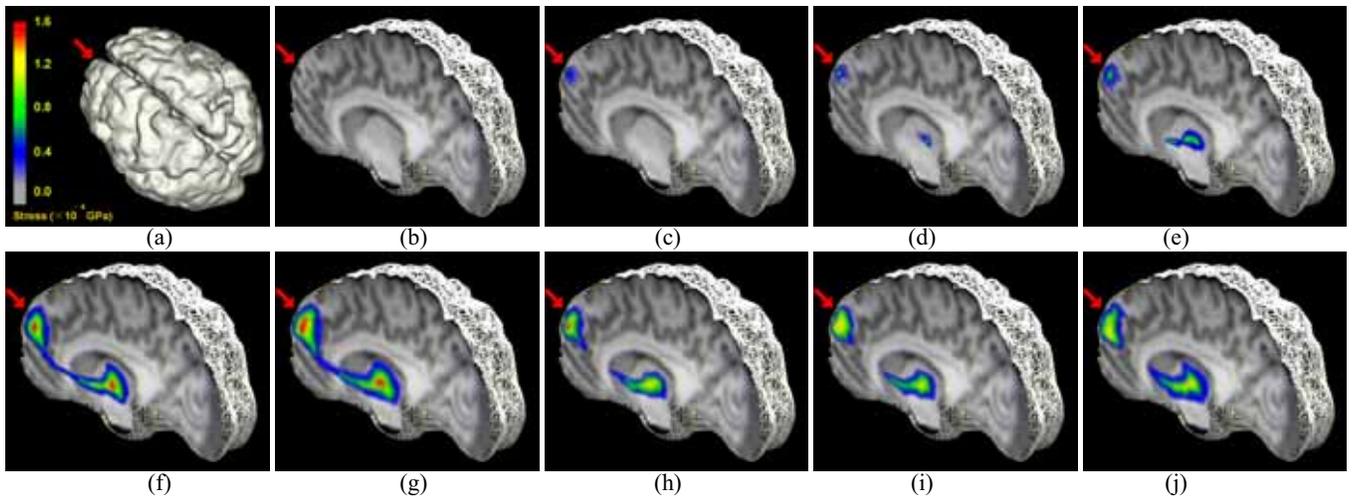


Figure 7: (a) The color map used to describe the stress field. The red arrow on the ISO-surface indicates the position where a blunt impact occurs. (b)-(j) Brain injury simulation with a time interval of 3ms. Imaginary blunt impact occurs at front lobe. Simulation results indicate that in addition to the spot directly under the impact, there are some other positions where bleeding may happen.

fact in Figure 7. The green one is the ground truth obtained from the real biomechanic experiments and the red one is the result simulated using our framework. The result curves demonstrated that our simulation can obtain an accurate and satisfactory result, which has great potential for computer-aided diagnosis of brain injury under blunt impact.

6 Experimental Results

We have implemented a prototype system on a 3.4GHz Pentium IV PC with 2GB RAM. The system is written in VC++ and VTK 4.2. We perform experiments on several brain datasets. In order to compare the reconstruction qualities for patient-specific cases, we uniformly scale the brain geometric and physical fields into a unit cube.

Table 1 shows the configuration of DSVSS volumes reconstructed from different datasets. The performance statistics of our fitting algorithm is also included. From the table, one can observe that, compared with discrete mesh representation, our spherical volumetric

simplex spline based representations have low storage requirements and can achieve high accuracy, e.g., fitting r.m.s. error $\leq \times 10^{-4}$.

The simulation results on brain shifting and brain injury prediction have been already illustrated in Section 5. Our brain simulation results exhibit a reliable approximation of how brain behaves shifting and how brain could be injured in the real world. More experiments on brain injury modeling will be conducted when more ground truth experimental datasets become available.

7 Conclusion

In this paper, we have developed a novel simulation framework based on dynamic spherical volumetric simplex splines. We have introduced an automatic and accurate algorithm to fit the digital models of real-world objects with a single spherical volumetric DMS-spline which can represent with accuracy geometric and material properties of objects simultaneously. With the integration of the Lagrangian mechanics, the dynamic volumetric simplex spline representing the real-world object can accurately simulate its phys-

Subject	Degree	Data Points	Tetrahedra	Control Points	Knots	Fitting r.m.s. Error
A	2	60298	2500	3871	1683	3.0375×10^{-4}
B	3	72357	2500	12431	2244	2.1483×10^{-4}
C	2	79593	4320	6525	2769	1.9743×10^{-4}
D	3	86226	4320	21117	3682	1.5290×10^{-4}

Table 1: Statistics of 3D reconstruction.

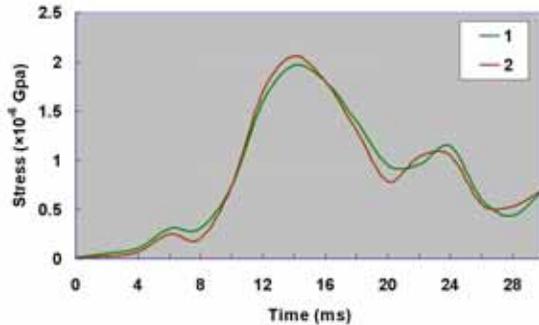


Figure 8: Comparison of stress evolutions of the right thalamus. The green one is the ground truth obtained from the real biomechanic experiments and the red one is the result simulated using our framework.

ical behavior. We have applied the framework in the biomechanics simulation of the brain, such as brain shifting during the surgery and brain injury under sudden impact. We have compared the simulated results with the ground truth obtained through interactive magnetic resonance imaging and the ground truth from real biomechanic experiments. The experimental results have demonstrated the excellent performance of our technique, which can be effectively used in deformation-based brain simulation and simulation-based diagnosis/assessment. The robustness and accuracy result from the tight integration of the geometric and material properties into the simulation. In the near future, we will investigate more powerful simulation schemes based on our novel digital representations. Hierarchical simulation will also be explored to speed up the simulation for real-time applications. On the application side, we will develop a DSVSS model of an entire head, which allow us to simulate more behaviors of the brain.

Acknowledgements

The authors would like to thank Drs. E. Mark Haacke and Liying Zhang at the Biomedical Engineering Department of WSU for providing us all the datasets and real biomechanic experimental results for evaluation. This work is supported in part by the grants received from the National Science Foundation, the National Institute of Health, the Michigan Technology Tri-Corridor, and the 21st Century Jobs Funds.

References

- BLOOR, M., AND WILSON, M. 1990. Representing PDE surfaces in terms of B-splines. *Computer-Aided Design* 22, 6, 324–331.
- CELNIKER, G., AND GOSSARD, D. 1991. Deformable curve and surface finite elements for free-form shape design. *Computer Graphics* 25, 4, 257–266.
- DAHMEN, W., MICCHELLI, C. A., AND SEIDEL, H.-P. 1992. Blossoming begets B-spline bases built better by B-patches. *Mathematics of Computation* 59, 199, 97–115.
- DE BOOR, C. 1976. Splines as linear combinations of B-splines. In *Approximation Theory II*, Academic Press, New York, 1–47.
- EDELSBRUNNER, H. 2001. *Geometry and Topology for Mesh Generation*. Cambridge University Press. Edited by P.G.Ciarlet and A.Iserles and R.V.Kohn and M.H.Wright.
- FERRANT, M., WARFIELD, S., NABAVI, A., JOLESZ, F., AND KIKINIS, R. 2000. Registration of 3D intraoperative MR images of the brain using a finite element biomechanical model. In *MICCAI*, 19–28.
- GOSSICK, B. 1967. *Hamilton's principle and physical systems*. Academic Press. New York and London.
- GREINER, G., AND SEIDEL, H. P. 1994. Modeling with triangular B-splines. *IEEE Computer Graphics and Applications* 14, 2, 56–60.
- GU, X., WANG, Y., CHAN, T., THOMPSON, P., AND YAU, S. 2003. Genus zero surface conformal mapping and its application to brain surface mapping. In *Information Processing in Medical Imaging*, 172–184.
- HALSTEAD, M., KASS, M., AND DEROSE, T. 1993. Efficient, fair interpolation using catmull-clark surfaces. In *Computer-Graphics Proceedings, Annual Conference Series, Proc. ACM-Siggraph93*, 35–44.
- HUA, J., HE, Y., AND QIN, H. 2004. Multiresolution heterogeneous solid modeling and visualization using trivariate simplex splines. In *Proceedings of the Ninth ACM Symposium on Solid Modeling and Applications*, 47–58.
- HUA, J., HE, Y., AND QIN, H. 2005. Trivariate simplex splines for inhomogeneous solid modeling in engineering design. *ASME Transactions: Journal of Computing and Information Science in Engineering* 5, 2, 149–157.
- KARDESTUNCER, H. 1987. *Finite Element Handbook*. McGraw-Hill. New York.
- LI, X., GUO, X., WANG, H., HE, Y., GU, X., AND QIN, H. 2007. Harmonic volumetric mapping for solid modeling applications. In *Proceedings of the 2007 ACM Symposium on Solid and Physical Modeling (SPM'07)*, 109–120.
- MAGUIRE, G., NOZ, M., RUSINEK, H., JAEGER, J., KRAMER, E., SANGER, J., AND SMITH, G. 1991. Graphics applied to medical image registration. In *IEEE Computer Graphics Application*, vol. 11, 20–28.
- MIGA, M., SINHA, T. K., CASH, D. M., GALLOWAY, R., AND WEIL, R. 2003. Cortical surface registration for image-guided neurosurgery using laser-range scanning. *IEEE Transactions on Medical Imaging* 22, 8, 973–985.

- MORETON, H., AND SEQUIN, C. 1992. Functional optimization for fair surface design. *Computer Graphics* 26, 2, 167–176.
- NIMSKY, C., GANSLANDT, O., CERNY, S., HASTREITER, P., GREINER, G., AND FAHLBUSCH, R. 2000. Quantification of, visualization of, and compensation for brain shift using intraoperative magnetic resonance imaging. *Neurosurgery* 47, 5, 1070–1080.
- PAULY, M., GROSS, M., AND KOBBELT, L. 2002. Efficient simplification of point-sampled surfaces. *IEEE Visualization 02 Proceedings*, 163–170.
- PFEIFLE, R., AND SEIDEL, H.-P. 1994. Fast evaluation of quadratic bivariate DMS spline surfaces. In *Proceedings of Graphics Interface '94*, 182–189.
- PFEIFLE, R., AND SEIDEL, H.-P. 1996. Scattered data approximation with triangular B-splines. *Advance Course on Fairshape*, 253–263.
- PRESS, W., FLANNEY, B., TEUKOLSKY, S., AND VETTERING, W. 1986. *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press. Cambridge.
- QIN, H., AND TERZOPOULOS, D. 1995. Dynamic manipulation of triangular B-splines. In *In Proceedings of Third Symposium on Solid Modeling and Applications (Solid Modeling '95)*, 351–360.
- QIN, H., AND TERZOPOULOS, D. 1995. Dynamic NURBS with geometric constraints for physics-based shape design. *Computer Aided Design* 27, 2, 111–127.
- TAN, Y., HUA, J., AND DONG, M. 2007. 3D reconstruction from 2D images with hierarchical continuous simplicies. *The Visual Computer* 23, 9-11, 905–914.
- TERZOPOULOS, D., AND FLEISCHER, K. 1988. Deformable models. *The Visual Computer* 4, 6, 306–331.
- TERZOPOULOS, D., AND QIN, H. 1994. Dynamic NURBS with geometric constraints for interactive sculpting. In *ACM Transactions on Graphics*, vol. 13, 103–136.
- TERZOPOULOS, D., PLATT, J., BARR, A., AND FLEISCHER, K. 1987. Elastically deformable models. *Computer Graphics* 21, 4, 205–214.
- TERZOPOULOS, D. 1986. Regularization of inverse visual problems involving discontinuities. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, 413–424.
- WANG, Y., GU, X., CHAN, T., THOMPSON, P., AND YAU, S. 2004. Intrinsic brain surface conformal mapping using a variational method. In *SPIE International Symposium on Medical Imaging*.
- WANG, Y., GU, X., CHAN, T., THOMPSON, P., AND YAU, S. 2004. Volumetric harmonic brain mapping. In *In ISBI'04: IEEE International Symposium on Biomedical Imaging: Macro to Nano*, 1275–1278.
- WANG, Y., GU, X., AND YAU, S. 2004. Volumetric harmonic map. *Communications in Information and Systems* 3, 3, 191–202.
- WELCH, W., AND WITKIN, A. 1992. Variational surface modeling. *Computer Graphics* 26, 2, 157–166.
- ZHANG, Y., BAJAJ, C., AND SOHN, B. 2005. 3D finite element meshing from imaging data. In *Computer Methods in Applied*