



Physically based modeling and simulation with dynamic spherical volumetric simplex splines

Yunhao Tan^a, Jing Hua^{a,*}, Hong Qin^b

^a Department of Computer Science, Wayne State University, Detroit, MI 48202, USA

^b Department of Computer Science, Stony Brook University, Stony Brook, NY 11794, USA

ARTICLE INFO

Article history:

Received 26 June 2008

Accepted 20 February 2009

Keywords:

Physically based modeling

Heterogeneous models

Biomechanic simulation

Biomedical imaging

ABSTRACT

In this paper, we present a novel computational modeling and simulation framework based on dynamic spherical volumetric simplex splines. The framework can handle the modeling and simulation of genus-zero objects with real physical properties. 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 the brain shifting during 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 real biomechanic experiments. The evaluations demonstrate the excellent performance of our new technique.

© 2009 Elsevier Ltd. All rights reserved.

1. Introduction

Modeling, simulation and assessment of digital representations of heterogeneous objects acquired from the real world are very challenging research tasks and have many potential applications. The fundamental objectives are to unambiguously model high-dimensional heterogeneous objects, accurately and effectively simulate their behaviors, and rigorously analyze their dynamic natures. Among many important aspects of physically based modeling and simulation, accuracy is of the utmost importance since only physically realistic simulations can be used to represent true reality and provide valuable information for 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 the 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, 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.

* Corresponding address: Department of Computer Science, Wayne State University, 5143 Cass Ave, 431 State Hall, Detroit, MI 48202, USA. Tel.: +1 313 577 9004; fax: +1 313 577 6868.

E-mail address: jinghua@cs.wayne.edu (J. Hua).

In order to bridge the gap and 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 cannot 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 the simulation. The integration of geometric modeling, material modeling, and simulation is the key to the success of the 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 the 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 a 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 [1]. 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 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 [2]. Various methods are emerging for the simulation of 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 with geometric, anatomical, and material variations. In order to obtain a 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.

As depicted in Fig. 1, the developed framework is fully automated without human intervention. The spherical domain is constructed from the subdivision of an icosahedron and harmonic

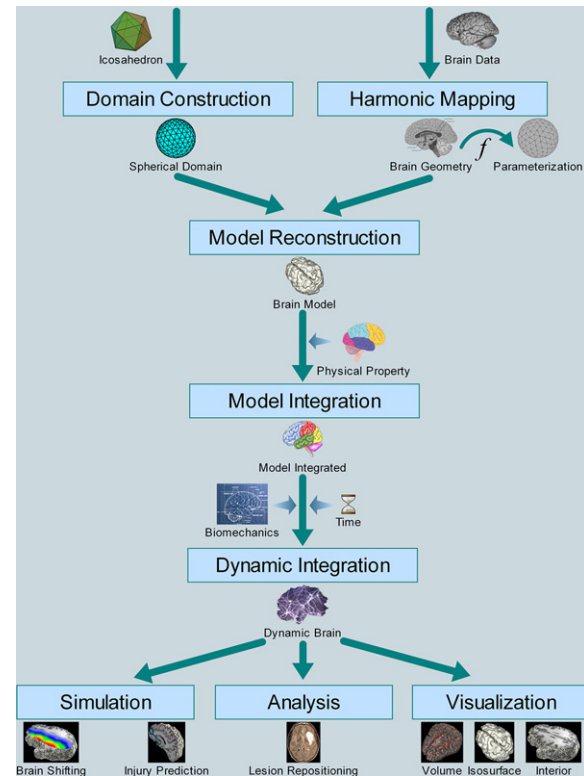


Fig. 1. Illustration of the pipeline of our DSVSS framework. The rectangles inside the pipeline indicate tasks, and the icons with text under them denote the inputs and outputs of tasks.

volumetric mapping. With spherical domain and harmonic volume parameterization, the continuous volumetric representation of the modeled object is obtained through fitting spherical volumetric simplex splines to the real-world volume data. Physical properties can then be integrated into the system to unify the geometric representation as well as the physical representation. With Lagrangian dynamics essentials integrated into the pipeline, the powerful framework yields the dynamic representation of the digital model. The dynamic representation of the digital model can facilitate multiple tasks such as model assessment, biomechanic simulation, and visualization.

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 simplex 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 a 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

This section reviews the previous work related to the theory and application of multivariate simplex splines and physically based modeling and simulation. In particular, we provide a brief background regarding brain simulation and its potential applications.

2.1. Multivariate simplex splines

From a 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 [3] 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 [4]. From the point of view of blossoming, Dahmen et al. [5] proposed triangular B-splines. Later, Greiner and Seidel [4] 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 [6] and applied it to the scattered data fitting of a triangular B-spline [7]. Recently, Rössl et al. [8] 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. Hua and Qin presented a volumetric sculpting framework that employs trivariate scalar nonuniform B-splines as an underlying representation [9,10]. More recently, they 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 [11]. Tan et al. applied the hierarchical simplex splines to volume reconstruction from planar images [12].

2.2. Physically based modeling and biomechanical simulation

Free-form deformable models were first introduced to the modeling community by Terzopoulos et al. [13], and they have been improved by a number of researchers over the past 20 years. Celniker and Gossard developed an interesting prototype system [14] for interactive free-form design based on the finite-element optimization of energy functionals proposed in [13]. Bloor and Wilson developed related models using similar energies and numerical optimization [15]. Welch and Witkin extended the approach to trimmed hierarchical B-splines for interactive modeling of free-form surfaces with constrained variational optimization [16]. Terzopoulos and Qin [17,18] devised dynamic physically based generalization of NURBS (D-NURBS). Later, they further developed a dynamic triangular B-splines [19] 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 have focused on FEM meshing, which can represent the shape of the objects, and physical laws and properties, which govern the model's behavior. Zhang et al. presented a method for 3D mesh generation from imaging data [20]. 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 nicely-shaped elements which can represent both the geometry and material of real-world models for accurate and robust simulation. However, due to its linear representations in general, it cannot accurately represent the geometric and physical properties of real-world objects. For simulation-based assessment of real-world objects, e.g., the brain, these FEM representations are not able to obtain an accurate and objective analysis result [21].

Biomechanical simulation of brain behaviors such as brain shifting and brain injury gains ever-increasing importance in recent years while these behaviors remain an unclear problem for public health professionals. Although impeded by that fact that brain material properties cannot be retrieved directly from the human brain in vivo, there is a certain amount of research which has been done either using animal brains or modifying brain biomechanical parameters to approach the real situation. Margulies et al. studied the relationship between non-preconditioned and preconditioned biomechanical response of brain tissue from pigs [22]. Later they further investigated the homogeneity of gray matter by measuring the stiffness of the cerebral cortex and comparing it to the thalamus of the porcine brain [23]. Many investigations have been conducted using mathematical finite element modeling [24,25,21]. In general, the cerebral tissues in their models were represented by homogeneous materials. Recent studies started to make distinctions between gray and white matter. In terms of applications, brain deformation simulation facilitates researchers and clinicians new prospects in clinical practice [2].

3. Dynamic spherical volumetric simplex splines

In this section, we first briefly review the theoretical background of volumetric 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. Volumetric simplex splines

A degree n volumetric 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 volumetric 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 $\sum_{j=0}^3 \lambda_j(\mathbf{x}|\mathbf{W})\mathbf{x}_{k_j} = \mathbf{x}$.

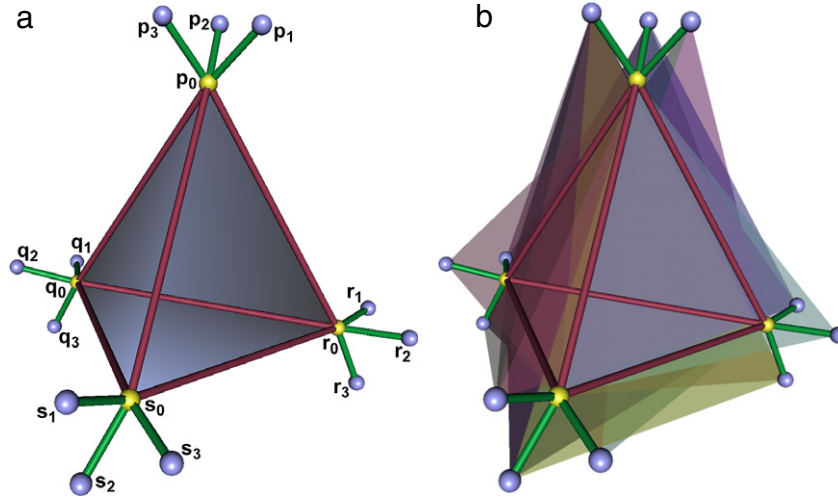


Fig. 2. (a) A domain tetrahedron demonstrated with its knot clouds assigned and labeled. The underlying tetrahedron, $(\mathbf{p}, \mathbf{q}, \mathbf{r}, \mathbf{s})$, is rendered as a shaded tetrahedron; (b) The virtual tetrahedra of the domain tetrahedron are visualized in different colors with the opacity value of 0.4. In (a) and (b), the degree of the domain is cubic hence each vertex of the tetrahedron has three sub-knots. The primary-knots are presented with yellow dots while sub-knots are depicted with blue dots. The red lines connecting the primary-knots indicate the underlying tetrahedron. Each green line here denotes the association between the primary-knot and the sub-knot. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

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

3.2. Spherical simplex spline volume

Generally, a volumetric simplex spline can take as input any domain with arbitrary geometry and topology due to its non-tensor-product nature. The spherical simplex spline volume is defined by volumetric simplex splines over a spherical volumetric domain. Here, we choose the sphere domain since mapping most organic objects in the biomedical research field to a sphere results in less distortion and more uniform distribution of sampling points, which reduces the difficulty in the fitting procedure. Note that, our volumetric simplex spline volumes represent not only boundary geometry, but also interior geometry. They can represent physical or material attributes over the entire solid as well.

3.2.1. Spherical volumetric simplex splines

Now let $\mathbb{S}^3 = \{\mathbf{x} \in \mathbb{R}^3, \|\mathbf{x}\| \leq c\}$ denote 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. Fig. 2(a) shows 4 vertices with cubic knot clouds associated, which are labeled as \mathbf{p} , \mathbf{q} , \mathbf{r} , or \mathbf{s} group, respectively. The primary-knots are rendered with yellow dots and sub-knots with blue dots. We will use these two colors to differentiate the primary-knots and sub-knots in the rest of the illustrations.

For every tetrahedron $I \in \mathbf{T}$, assume $I = (\mathbf{p}, \mathbf{q}, \mathbf{r}, \mathbf{s}) = (\mathbf{p}_0, \mathbf{q}_0, \mathbf{r}_0, \mathbf{s}_0)$. We call $(\mathbf{p}, \mathbf{q}, \mathbf{r}, \mathbf{s})$ the underlying tetrahedron. All the other tetrahedra $[\mathbf{p}_i, \mathbf{q}_j, \mathbf{r}_k, \mathbf{s}_l]$ with $0 < i + j + k + l \leq n$ are called the virtual tetrahedra. Fig. 2(a) shows the underlying tetrahedron with shading. Fig. 2(b) demonstrates the virtual tetrahedra rendered with different shading colors.

Then for every tetrahedron I , 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, i.e., the underlying tetrahedron and virtual tetrahedra should be valid.
- the set

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

is not empty.

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

The condition that Ω is nonempty states that the sub-knots associated with different vertices of I are all separated from each other. The underlying tetrahedron I and its virtual tetrahedra have the same orientation. As shown in Fig. 2(a), in our framework, the orientation of the tetrahedron is defined as, observed from \mathbf{p}_i , the triangle formed by $(\mathbf{q}_j, \mathbf{r}_k, \mathbf{s}_l)$ is clockwise-oriented where $i + j + k + l \leq n$.

The formation of Ω can be intuitively described as: starting from the underlying tetrahedron, we chop it using the triangle faces from each virtual tetrahedron, and only keep the part inside of the triangle faces. As the starting shape of the underlying tetrahedron is convex, the chopping operation will not affect its convexity. Hence the final shape of Ω is a convex, solid polyhedron if nonempty.

Fig. 3(a) and (b) depicts the Ω of a cubic domain tetrahedron, with and without virtual tetrahedra rendered, respectively. Note that if $\Omega \neq \emptyset$, Ω must be a convex solid polyhedron formed by the interior of the underlying tetrahedron and virtual tetrahedra. Fig. 3(a) and (b) illustrate the Ω as a blue, convex, and solid polyhedron.

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 are

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

For an example in Figs. 2 and 3, as the degree of the domain is cubic, V_{β}^I has 16 elements: 4 primary-knots and 12 sub-knots.

The basis functions of normalized simplex splines are then defined as

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

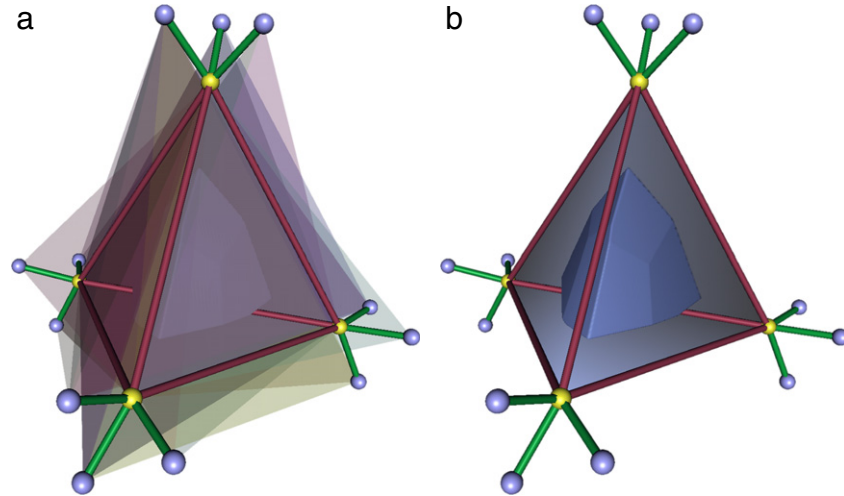


Fig. 3. (a) The Ω of a domain tetrahedron, formed by the interior of the underlying tetrahedron and virtual tetrahedra, is rendered with a blue polyhedron. The virtual tetrahedra are visualized in different colors with the opacity value 0.1; (b) The virtual tetrahedra are removed to better visualize the Ω . (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

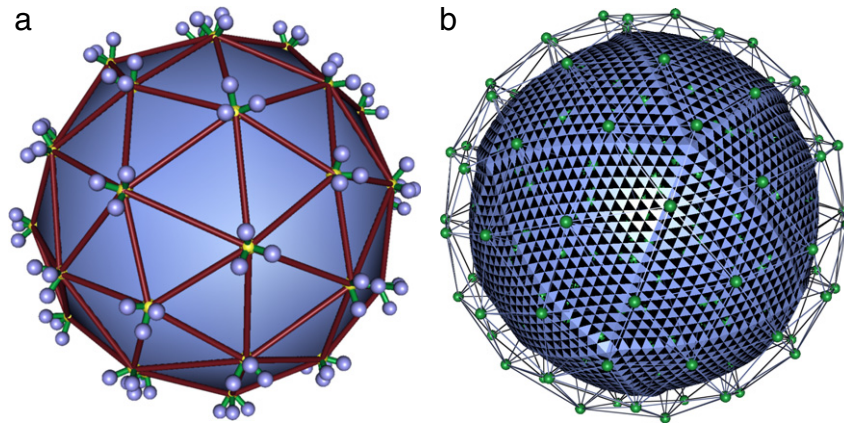


Fig. 4. (a) The spherical domain with assigned knot clouds for defining spherical volumetric simplex splines. The yellow and blue dots denote primary-knots and sub-knots, respectively; (b) The spherical simplex spline volume defined upon the domain in (a). The green dots denote the control points. The evaluated spherical volume simplex volume is scaled to show its nonempty interior property. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

These basis functions can be shown to be all non-negative and to form a partition of unity. The volumetric spherical simplex volume is the combination of a set of basis functions with control points \mathbf{c}_β^l :

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

The “generalized” control points \mathbf{c}_β^l 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 simplex 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.

Fig. 4(a) illustrates a spherical volumetric simplex spline and its domain with its cubic knot clouds associated. As observed in the figure, the sub-knots assigned to the boundary vertices of the sphere domain are positioned outside of the sphere. Fig. 4(b) shows the control space and the evaluated spherical volumetric simplex volume.

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 $\frac{\max(\text{Vol}_{l \in \mathbf{T}})}{\min(\text{Vol}_{l \in \mathbf{T}})}$. Uniform tetrahedralization at the same hierarchical level will decrease the recursion time while a hierarchical structure is needed.
- \mathbf{T} should avoid bad-shaped tetrahedra in Delaunay tetrahedralization. Bad-shaped tetrahedra, for instance, slivers, will increase numerical error during the evaluation.

Constrained Delaunay tetrahedralization [26] can observe the second requirement, but it will introduce very large and very small tetrahedra thus cannot 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.

Fig. 5 shows the flow of domain establishment and the knots distribution. Note that, in Fig. 5(d), the sub-knots associated with

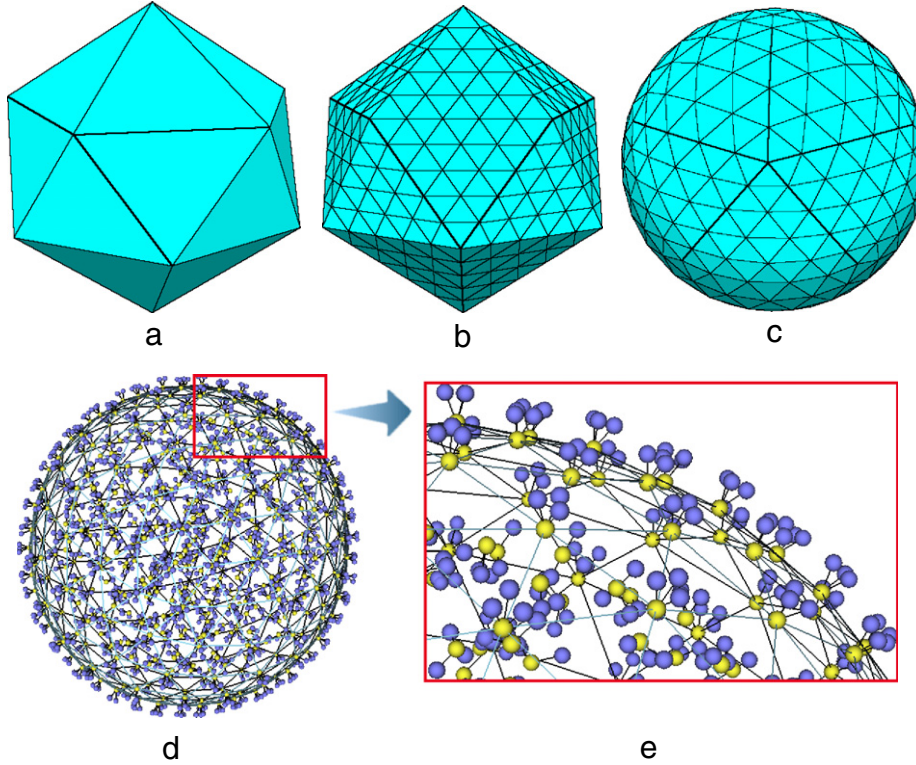


Fig. 5. (a) A regular icosahedron, which is the best approximation of a solid sphere among all regular polyhedra; (b) Tetrahedralization of (a) is uniform and it is easy to implement; (c) Harmonic mapping from (b) to a unit solid sphere yields the domain tetrahedralization, consisting of uniform and well-shaped tetrahedra; (d) A domain with cubic knot clouds assigned to (c); (e) A close view of the domain picked from (d).

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.

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. [27,28]. 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. [29] 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 , and suppose that \vec{f}' is the conformal mapping [30,31] 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 harmonic volumetric mapping f here minimizes a harmonic energy $E(f)$ [28], which is defined as

$$E(f) = \sum_{(u,v)} k(u,v)(f(u) - f(v))^2, \quad (7)$$

where $k(u,v)$ is the string constant defined in the edge between u and v . Here, f can be solved using a steepest descent algorithm.

The algorithmic procedure of harmonic volumetric mapping is concisely summarized as follows:

- (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 using Eq. (7).
- (2) For each interior vertex, $v, v \in M_1 \setminus \partial M_1$, compute its derivative $D\vec{f}$ using a steepest descent algorithm, then update $\vec{f}(v)$ by $\delta \vec{f}(v) = -D\vec{f}(t)\delta t$, 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 step (2) through step (3).

Fig. 6 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

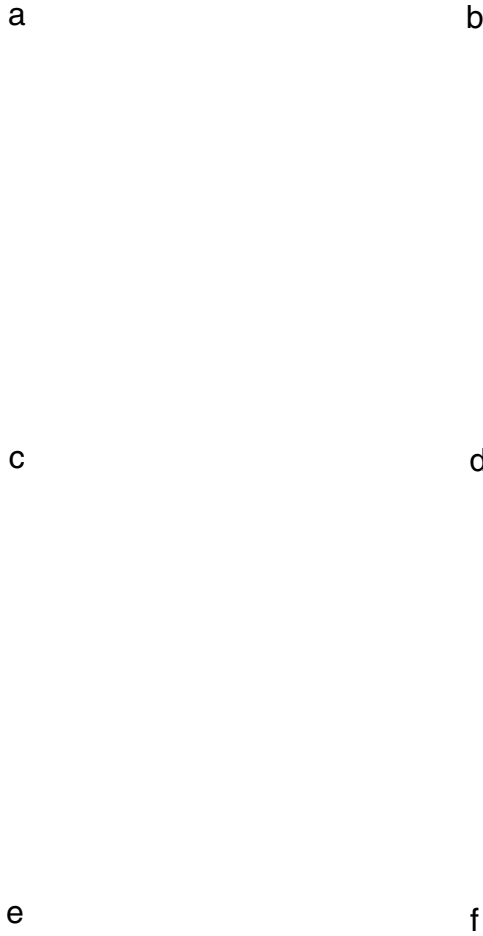


Fig. 6. (a) The discretized point set in the spherical domain space; (b) The discretized data point set in the physical space, from the same angle of view as (a); (c–f) The shapes are cut into halves sagittally (c–d) and axially (e–f) in order to show the interior mapping between the parametric domain and the physical object.

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, $\mathbf{p}_i = (x_i, y_i, z_i, \rho_i) \in \mathbb{R}^4$, and $G = \{\mathbf{g}_i\}_{i=1}^m$, $\mathbf{g}_i = (x_i, y_i, z_i) \in \mathbb{R}^3$ denoting the pure geometry extracted from the sampling points, find a volumetric simplex splines volume $\mathbf{s} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ that approximates G .

Since we are interested in reconstructing the model with respect to its solid geometry, our spherical simplex spline volumes

are vector functions, i.e., the control points $\mathbf{c}_\beta^j \in \mathbb{R}^3$ 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 [1,11], we use the position (u_i, v_i, w_i) within the solid sphere as the data point \mathbf{g}_i 's parametric value. Therefore, we need to minimize the following objective function:

$$\min E_{\text{dist}}(\mathbf{s}) = \sum_{i=1}^m (\mathbf{g}_i - \mathbf{s}(u_i, v_i, w_i))^2. \tag{8}$$

Eq. (8) 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:

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

where $\mathbf{x} = (\dots, \mathbf{c}_\beta^j, \dots)^T$,

