Real-Time Example-Based Materials in Laplace-Beltrami Shape Space

Jing Zhao[1,2] zhaojing@ysu.edu.cn
Fei Zhu[3,4] feizhu@pku.edu.cn
Yong Tang[1,2,*] tangyong@ysu.edu.cn
Liyou Xu[3,4] leopkucs@pku.edu.cn
Sheng Li[3,4] lisheng@pku.edu.cn
Guoping Wang[3,4,†] wgp@pku.edu.cn

[1]College of Information Science and Engineering of Yanshan University,Qinhuangdao,China,066004
[2]The Key Laboratory for Computer Virtual Technology and System Integration of Hebei Province,China,066004
[3]School of Electronics Engineering and Computer Science of Peking University,Beijing,China,100871
[4]Beijing Engineering Research Center for Virtual Simulation and Visualization,China,100871

Abstract

In this paper, we present a new real-time simulation framework for example-based elastic materials with different topology structures between the object and input examples. The geometry descriptions in the Euclidean space of all input shapes are projected into a common reduced shape space spanned by the Laplace-Beltrami eigenfunctions constructed on the volumetric mesh. The reduced shape interpolation space is scale and topology independent. The shape interpolation process is computed totally in the reduced subspace by solving a nonlinear energy optimization problem which is calculated in the model reduction subspace using cubature elements. To accelerate the energy solving process, we eliminate the transmission process between the reduced spaces and the Euclidean space by establishing a direct projection from Laplace-Beltrami shape space to the model reduction subspace. Experiments demonstrate that our method can achieve real-time efficiency while providing comparable simulation quality.

Keywords: deformable simulation, reduced space, example-based material, Laplace-Beltrami eigen-analysis

*Corresponding author.
†Corresponding author.

1 Introduction

The simulation of deformable objects is a common task in physically-based animation field because of its broadly used in computer graphics and virtual reality systems. The material properties tuning of the deformable behavior is a challenging and tedious work, while achieving art-directed deformation results becomes another intuitive choice.

Recently, Martin et al. [1] propose an approach of example-based elastic materials achieving artistic control. They construct an example space by interpolating example poses provided by users and generate an additional force which attracts the object towards the example space. It contains a time-consuming process to reconstruct a consistent geometric representation of interpolated example poses. Additionally, the input examples share the same topology with the simulation object. Zhu et al. [2] express shapes in a common shape space in which the basis are the coupled Laplace-Beltrami eigenfunctions. In the proposed shape space, different models can be animated using the same examples. Compared to Mar-

tin's method, the efficiency has been improved by using the reduced shape interpolation space. The whole simulation process is far from real-time efficiency since they obtain the interpolation shape by solving a constrained nonlinear-optimization problem. The energy computing is on the whole Euclidean space which is related to the vertices number of the object's surface mesh even though the process only contains several iteration steps. We achieve example-based elastic deformation at real-time rate, and at the same time we keep the different topology structures for both the examples and the simulation object using the Laplace-Beltrami shape space. For the details, in the simulation process, we take use of the nonlinear modal basis and the cubature elements to compute internal force for arbitrary materials presented by An et al.[3]. In the shape interpolation process, we obtain the target interpolation configuration to solve a constrained nonlinear optimization problem totally in the subspace. We solve the energy minimization issue in the subspace by proposing a direct projection from the Laplace-Beltrami shape space to the model reduction subspace.

2 Related Work

Example-Based Deformation Martin and his colleges [1] propose an example-based elastic material with the finite element method(FEM), which allows the deformation behavior to be implicitly controlled by specifying a set of example poses. They employ a nonlinear Green strain as the deformation measure and construct an elastic energy that pulls current configuration towards the preferred configuration defined by examples. It involves solving an expensive nonlinear optimization problem to reconstruct a consistent geometric representation of interpolated example poses. In their following researches [4], they utilize an incompatible representation for input and interpolate poses to interpolate between elements individually. The proposed method achieves significant performance improvements compared to previous work. Except for example-based elastic solid deformation materials, Fröhlich et al. [5] extend the deformation to discrete shells based on an extension of the discrete shell energy and Jones et al. [6] propose the example-based plastic deformation for rigid bodies based on linear blend skinning and an unmodified rigid body simulator. Koyama et al. [7] present a real-time example-based elastic deformation method using the shape matching framework and the linear interpolation of the example poses. Their method is based on a geometry framework and achieves real-time efficiency, while the simulation effect is not so good as Martin's method. The method proposed by Zhang et al. [8] also obtains a real-time simulation rate using a subspace integration. They formulate a new potential using example-based Green strain tensor which attracts the simulation model to the example-based deformation feature space. Wampler [9] introduces a novel approach for example-based inverse kinematic mesh manipulation which generates high quality deformations for a wide range of inputs. And the approach is fast enough to run in real time.

The limitation of the previous example-based approaches is circumvented that all examples must have identical topology with the simulated object. Zhu et al. [2] propose a new method which allows all the examples are arbitrary in size, similar but not identity in shape with the object. They project all the geometry structures to a common shape space spanned by the Laplace-Beltrami eigenfunctions and interpolate the examples via a weighted-energy minimization to find the target configuration which guides the object to a desired deformation. The target configuration is solved by a nonlinear optimization problem and the nonlinear shape interpolation process is time-consuming since the nonlinear deformation energy between two shapes is defined on the surface mesh, which means the computation efficiency is relative to the resolution of the mesh.

Subspace Simulation A lot of researches have been proposed to improve the speed of simulating deformable models, especially for the subspace simulation approach which has gained popularity in computer graphics recently [10]. The modal analysis method was first proposed in graphics to build linear vibration modes as the subspace basis [11]. To handle rotational deformations, Choi and Ko develope the modal warping method by removing incorrect vertex deformation components caused by linear elasticity [12]. Barbič et al. [13] construct nonlinear de-

formation modes from modal derivatives for the St.Venant-Kirchhoff materials, since the internal force of the special material can be simply represented as a vector of cubic polynomials in the reduced coordinates. An et al. [3] calculate the reduced elastic force in the subspace by using cubature approximation elements. Their method is not limited to the StVK material and reduces the computational complexity of subspace simulation to $O(r^3)$, in which r is the number of modes in the subspace basis. A basis augmentation scheme is presented to capture local deformation caused by collision contact [14].

Shape Interpolation Shape interpolation is an important issue in geometry processing. Most of the shape interpolation schemes are based on the interpolation for the selected geometric quantities of a shape. The interpolation process is to compute the average quantities for all examples. The reconstruction process of the shape is typically done as a least-squares problem depending on the quantities of the vertices positions are linear or nonlinear interpolation. One effective research is depended on maintaining the rigid criteria called as-rigid-as-possible [15]. Their method computes preferred interpolations by local affine transformations of the local geometrical elements. Xu et al. [16] propose a nonlinear gradient field interpolation method. The geometric quantities conclude the vertex coordinates and surface orientation, and the reconstruction is a Poisson problem. Winkler et al. [17] utilize the edge lengths and dihedral angles of triangles of a surface mesh and Martin et al. [1] take use of the strain tensors of the tetrahedral mesh as the geometric interpolation quantities. Due to the nonlinear optimization problem in the reconstruction process, the methods run at real-time rates only for very coarse meshes. Von-Tycowicz et al. [18] restrict the shape optimization problem to a low-dimensional subspace that is specifically designed for the shape interpolation problem.

3 Background

In this session, we present an overview of the basic example-based elastic material simulation framework and the reduced subspace simulation method.

3.1 Example-Based Elastic Deformation

The key idea for the simulation of the example-based elastic materials is to add an additional elastic potential on the basic simulator of elastic solids, which attracts the deformable object towards the desirable deformation characterized by the examples. The equations of motion of an object discretize in space are given by

$$M\ddot{x} + D\dot{x} + f^{int} + \frac{\partial W_p}{\partial x} = f^{ext} \qquad (1)$$

Here M is the mass matrix, x and $\ddot{x}$ are the positions and accelerations of the object's n-odal degrees of freedom (DOFs), f^{int} represents the internal force, $W_p = W(X_{target}, x)$ represents the additional potential between the object's deformed configuration x and its closest target configuration X_{target} on the example manifold spanned by input examples, f^{ext} denotes the sum of external forces due to gravity, friction and contacts.

3.2 Subspace Simulation

Model reduction is an efficient technique widely used to speed up the integration. The basic subspace simulation is to convert the displacement vector $u \in R^{3n}$ into a subspace spanned by a set of $r(r \ll 3n)$ representative deformation modes. These displacement vectors can be assembled into a $3n \times r$ matrix U, as the basis for subspace simulation. As the method described in [13], we use nonlinear modal basis to construct U, in which U is mass orthogonal: $U^T M U = I$, I is the $r \times r$ identity matrix. Then the full-space displacement u is converted to $u = Uq$, in which $q \in R^r$ represents the reduced coordinates in the subspace. By combining $u = Uq$ and $U^T M U = I$ with Equation 1, we obtain the governing equation in the subspace:

$$\ddot{q} + U^T D U \dot{q} + U^T f^{int}(Uq) + U^T \frac{\partial W_p}{\partial x} = U^T f^{ext} \qquad (2)$$

By using implicit integration, Equation 2 can be formulated into a dense $r \times r$ linear system, and the entire simulation can be orders of magnitude faster than the full-space simulation.

4 Example-Based Subspace Simulation

4.1 Real-time simulation framework

The previous example-based material simulation method has the same limitation that the input examples are constrained to share the same topology with the object. [2] proposed the Laplace-Beltrami eigen-analysis method to construct a scale and topology independent shape space. Our method proceeds also in two stages as: the pre-computation stage and the run-time simulation stage. Here we outline the procedures in each stage:

Pre-computation:

(1) Construct the non-inertial frame of the simulation model and align examples to the local frame;

(2) Compute m leading eigenfunctions and the corresponding eigenvalues for both the simulation model and input examples, and normalize the eigenfunctions;

(3) Align the examples' eigenfunctions with the object's top m eigenfunctions;

(4) Project the geometry of all examples onto their eigenfunctions and get corresponding coefficients as the shape descriptors in the non-inertial frame;

(5) Compute the nonlinear modal basis for the model;

(6) Compute the optimized cubature elements using the random forces and the input reduced modal basis.

Run-time:

(1) Obtain the non-inertial frame displacement of the current configuration and project the displacement to the Laplace-Beltrami shape space for the shape description;

(2) Find the target configuration according to the examples and object's current configuration in the Laplace-Beltrami shape space, and the shape interpolation process is computed in the non-inertial frame;

(3) Reconstruct the shape of the target interpolation configuration from Laplace-Beltrami space to the Euclidean space to calculate the example-driven force, which is computed by the linear force with the displacement of the current and target configuration by simplicity;

(4) Perform the rigid body simulation framework in inertial frame;

(5) Compute the deformed simulation with the example-driven force, the external forces and the forces caused by the rigid motion in non-inertial frame, step the deformed simulation;

(6) Project the non-inertial frame displacement to the inertial frame to render the deformation.

We demonstrate the runtime overview in Figure 1. Three spaces have been described here. In the Euclidean space, three input poses of the elastic cuboid are given as examples as geometry shapes, each with different number of vertices. All the geometry shapes are projected onto their first m eigenfunctions in the Laplace-Beltrami interpolation space(black points). In the interpolation process, we build a connection from Laplace-Beltrami shape space to the model reduction subspace to compute the elastic energy quickly. After the interpolation process, the target configuration (red point) in the Laplace-Beltrami space is reconstructed to the Euclidean space.

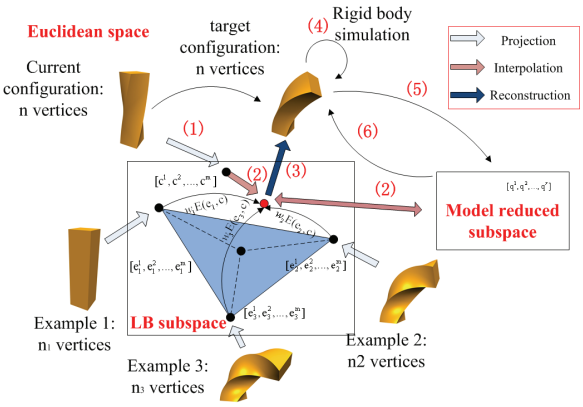

Figure 1: Overview of run-time operations.

4.2 Space description

For the whole simulation framework, we have three spaces and we will explain them in details. We have listed most of the variables of this paper in Table 1.

Laplace-Beltrami subspace Based on Laplace-Beltrami eigen-analysis method, we obtain the eigenfunctions on the volumetric mesh. There are several ways to approximate the Laplace-Beltrami operator and its eigenfunctions on dis-

Table 1: Symbols. This table summarizes some of the symbols.

Symbol	Dimension	Definition
m	scalar	num of aligned eigenfunctions
r	scalar	num of nonlinear modal basis
s	scalar	num of optimized cubature elements
n	scalar	num of vertices of the object
u	$3n$ vector	displacement in the fullspace
q	r vector	displacement in the model reduction subspace
l	$3m$ vector	shape description in the Laplace-Beltrami subspace
U	$3n \times r$ matrix	the nonlinear modal basis
L	$3n \times m$ matrix	top m eigenfunctions aligned between the examples and the object
T	$m \times r$ matrix	projection from Laplace-Beltrami shape space to model reduction subspace
$\tilde{x}(q)$	$3n$ vector	the vertex position in the non-inertia frame
v, ω	3 vector	linear and angular velocities in non-inertia frame of rigid simulation
$\dot{v}, \dot{\omega}$	3 vector	linear and angular acceleration velocities in non-inertia frame

crete representations of manifolds. We choose the linear FEM approximation for its highly accuracy [19, 20]. The normalized eigenfunctions construct the reduced interpolation shape space. The volumetric mesh in the Euclidean space can be projected to Laplace-Beltrami subspace, while the shape description of the reduced interpolation shape space can be reconstructed in the Euclidean space by using eigenfunctions and eigenvalues. The eigenfunctions are normalized and computed on the tetrahedral mesh, and the values of the eigenfunctions on each vertex is between 0 and 1. We render the eigenfunctions with different colors on each vertex using the piecewise linear function. We demonstrate the second eigenfunction for the interior of the armadillo model as Figure 2. The first constant eigenfunction is excluded from the computed eigenfunctions since it encodes only rigid motion of rotation and translation.

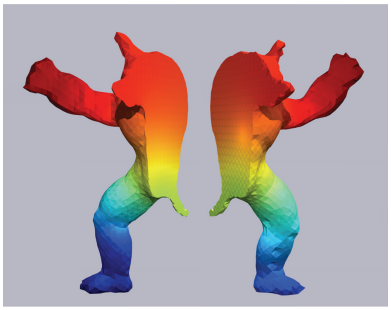

Figure 2: Rendering interior eigenfunction effect for the armadillo model.

The eigenfunction of different shapes need to be aligned before they can be used as the common basis. We align the eigenfunctions with the method in [21]. The registration process is convenient and can be computed in advance.

Full-space Full-space is the geometry description in Euclidean space and it is the transition space between the Laplace-Beltrami shape space and the model reduction subspace. Also we render the deformation displacement and excecute the rigid motion in this frame.

Reduced subspace constructed by modal basis The subspace is constructed by the nonlinear modal basis in the pre-computation process. In our current implementation, we use linear modal analysis and modal derivatives proposed in [13] to generate both linear and nonlinear deformation modes in the basis.

The spaces can be converted to each other as:

(1) Project to the Laplace-Beltrami space. The geometry description of the examples and the object in Euclidean space is projected onto the Laplace-Beltrami interpolation subspace by their own eigenfunctions. The projection is done through volume-inner product of the shape's geometry and its Laplace-Beltrami eigenfunctions, which can be described as:

$$(< p_x, \lambda_i L_i >, < p_y, \lambda_i L_i >, < p_z, \lambda_i L_i >) \tag{3}$$

where $1 \leq i \leq m$, $p(p_x, p_y, p_z)$ is the geometry vertices position in Euclidean space, λ_i,f_i is the i-th eigenvalue and its corresponding eigen-

function, and $<,>$ represents the volume-inner product. The description in Lapalace-Beltrami subspace is a $m \times 3$ vector.

(2) Reconstruct geometry shapes in the Euclidean space. By using the equation: $\sum_{i=1}^{m} l_i \frac{1}{\lambda_i} L_i$, we get a geometry description of Euclidean space.

(3) Project shape description from Laplace-Beltrami space to the model reduction subspace. As Zhu et al. [2] did in each time step, the shape descriptor is reconstructed in the Euclidean space and the nonlinear elastic energy is computed by a geometry deformation energy on the object's surface mesh, which is a time-consuming process. To accelerate the target shape interpolation process, we solve the elastic energy in the model reduction subspace described as equation (6) by using the optimized cubature elements. Since we have to utilize the Euclidean space as the transition space between the Laplace-Beltrami shape space and the model reduction subspace, which need to compute the large matrix-multiply operation twice with matrix L and matrix U, which are constant in the whole simulation process and can be computed beforehand. We propose a direct projection matrix from Laplace-Beltrami subspace to model reduction subspace by computing $T = L^T U$.

4.3 Shape interpolation

The shape interpolation process is to find the optimized target configuration in the example manifold for the current configuration. In our system, we solve the shape interpolation problem totally in reduced space.

The deformable object is represented by a tetrahedral mesh with n vertices. The input examples are consist of k poses, represented as m-dimensional points $e_i (1 \leq i \leq k)$ in the shape space spanned by the eigenfunctions. We interpolate the examples to get the target configuration that guides the deformation of the object. Similar as [2] did, we also use nonlinear interpolation approach. The target configuration t is defined as the one that minimizes the sum of weighted deformation energies to all examples:

$$\min_{t} \sum_{i=1}^{k} \omega_i E(t, e_i) \tag{4}$$

where $E(\cdot,\cdot)$ is the non-linear deformation energy between two shapes, $\omega_i (1 \leq i \leq k)$ measures the guiding strength of the examples and $\sum_{i=1}^{k} \omega_i = 1$. The interpolation weight for each example weight according to its closeness with the object's configuration c in each time step:

$$\min_{\omega_i} \frac{1}{2} \left\| \sum_{i=1}^{k} \omega_i e_i - c \right\|_F^2 \tag{5}$$

In their method, the geometry energy is composed of three terms: stretching term, bending term and volume preservation term. Their geometry energy measurement is relative to the geometry description, and the resolution of the mesh will definitely decide the solving efficiency. To obtain a target interpolation configuration for the current deformation of the object, the nonlinear interpolation process is a nonlinear optimization which contains several iteration steps. It is time-consuming to compute the energy and its gradient in the Euclidean space at each iteration step. We accelerate the process by computing the energy with the cubature integration method provided by [3] in the model reduction subspace. As noted in 4.2, we have a direct projection from Laplace-Beltrami shape subspace to the model reduction subspace. With the projection, the elastic energy and its corresponding energy gradient can be computed easily for each target configuration. According to the method, the elastic force can be formulated in terms of a potential energy function, $E(q) : R^r \rightarrow R$, given by the domain integral,

$$\begin{aligned} E(t, e_i) &= \int_{\Omega} \Psi(X_{e_i}; q_{e_i}) d\Omega_X \\ &= \sum_{j=1}^{s} \omega_j \Psi_j(X_{e_i}; q_{e_i}) \end{aligned} \tag{6}$$

where the undeformed shape X_{e_i} is the ith geometry shape in the Euclidean space reconstructed from the Laplace-Beltrami space with the object's eigenfunction, $\Psi(X_{e_i}; q_{e_i})$ is the nonnegative strain energy density at material point X_{e_i} in the undeformed material domain Ω, s is the number of cubature elements and q_{e_i} describes the displacement between the target configuration and example e_i in the model reduction space. The optimizing cubature elements and

the corresponding weights obtained in the pre-computation process.

5 Results

We test our system on an Intel Core i5 2.8GHz processor. We use the OpenMp library to parallelize our simulation steps. Time step size is uniformly set to 0.001s for all simulations. Currently, we choose the StVK material model, while our proposed method is not limited to the using material. We use two input examples for all the experiments in the paper.

As shown in Figure 3, the first picture is the initial configuration, the second bar deformed under a twisted force on the bottom simulated in the full space, the third twisted bar is deformed with the same topology between the input examples and the object in the full space, the fourth one demonstrates the twisted bar deformed by different topology examples(cylinder examples) in the full space, and the last picture shows the twisted bar deformed on the same situation as the fourth one which is performed in the reduced space with our method. The last four pictures show similar deformation effects with different conditions, and our method with different input examples computes the interpolation shape in the model reduction space which achieves equally animation effect as in the full space.

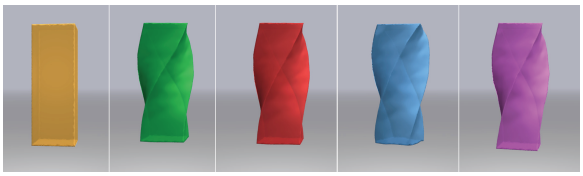

Figure 3: Bar deformed with different situations.

Without constrains As [22] pointed out, the simulation method in reduced subspace is not suitable for simulating the rigid motion of a deformable object, since incorporating rigid modes into the basis U will cause U to be time-dependent and we cannot afford updating U over time. We simulate the rigid motion separately from subspace deformation by rigid body dynamics, as did in [13, 23, 24]. The free motion simulation object contains the rigid motion and deformed motion. The shape interpolation process and the simulation process in the reduced space are computed in the non-inertial frame. Figure 4 and Figure 5 show the simulation effects for deformable objects without constrains.

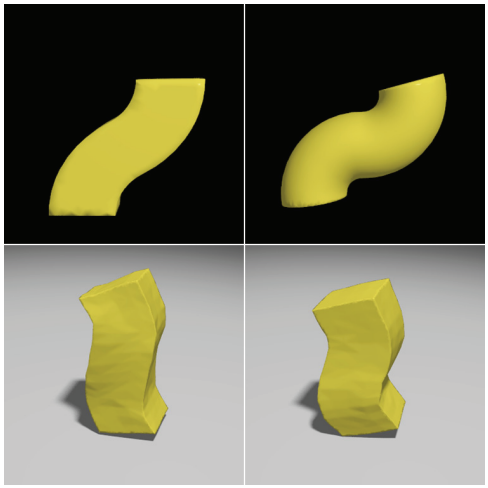

Figure 4: An elastic bar deforms under gravity using a S-cuboid shaped example and a S-cylinder shaped example.

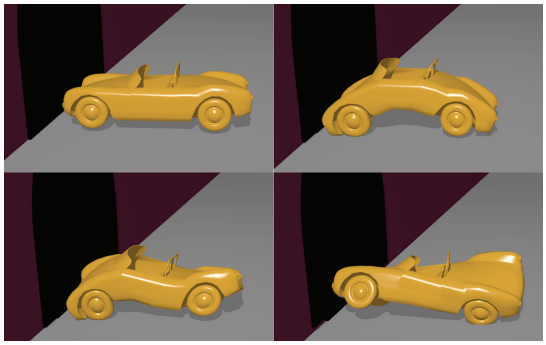

Figure 5: A car model hits the wall and reacts in diverse ways with no example(top left), an arc-shaped example(top right), a s-shape example(bottom left) and a twisted example(bottom right).

With constrains The object simulation with constrains is much more easier than the free motion one since the deformation excluded the rigid motion. As shown in Figure 6, the teddy model is fixed on the back. The first picture is the rest configuration, the second one shows the deformation of the left part of the model under the drag force enforced on the left leg, and the third one shows the global deformation between

the two legs and two arms.

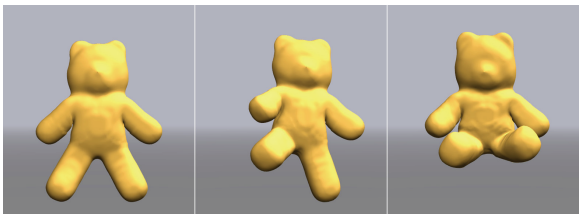

Figure 6: An elastic teddy deformed with local and global examples.

Local examples The method is easily extended to the local examples. We partition the entire simulation object into separate regions, each part is influenced by the corresponding example regions independently. We project each region on the entire object's Laplace-Beltrami eigenfunctions. The rest vertices displacement of the object excluded each region are set to zero. Each local region is handled independently, thus we achieve complex behaviors for different local examples. We demonstrate the local examples as Figure 7. The top three figures show different examples as the twist shape, the horizonal direction s-bend shape, and the vertical direction s-bend shape, respectively. The deformation result shows on the second line, the first picture is the initial configuration, the second picture shows the left region and the right region of the cross deformed with the first and second examples respectively, and the last one shows the middle region of the cross influenced by the third example. In Table 1, we list the detailed parameter settings and the performance data for all the examples presented in the paper. The results in this paper are produced with practical computation time. We can find out the time efficiency is obviously better than the method in [2]. We achieve a real-time simulation effect since we solve the energy minimization problem in the reduced subspace and we utilize the optimized cubature elements to find the target configuration. The optimized cubature elements can be computed in advance with the method proposed in [3, 25, 26].

6 Conclusions

We present a real-time framework for example-based elastic materials with different topology

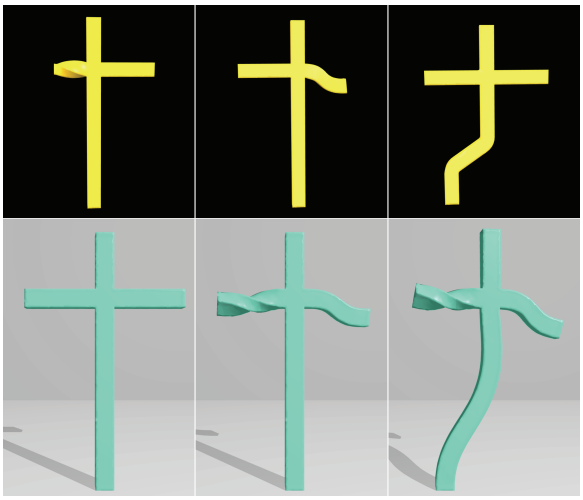

Figure 7: Eigenfunctions of the three local regions on the cross shape are computed, respectively.

structures between the simulation object and input examples with FEM. We compute the eigenfunctions on the tetrahedral mesh with the linear FEM approximation for the Laplace-Beltrami Operator eigen-analysis method to construct the shape interpolation subspace. We propose the direct projection from the Laplace-Beltrami subspace to the model reduction subspace, which is convenient to compute the elastic energy with the optimized cubature elements. Finally, we separate the rigid motion from the deformable motion with the inertia frame and the non-inertial frame. Our experiments demonstrate the flexibility and the highly simulation efficiency for the example-based materials to some extend.

In the subspace simulation method for deformable objects, an important question is whether collisions can be handled in the subspace as well. We do not take much time on the collision problem. For simplicity, we choose the penalty force for collision force in the full-space which provided some collision-artifact. This problem can be extended in the subspace to improve the efficiency as [27, 28] did in the future. Finally, although our system separates rigid motions from non-rigid motions, it is not strictly correct and the result can be different from the result of full-space simulation. Sometimes the system will unstable on the rotation caused by the collision force and example force.

Model	DOFs	Tets	Ex.	m	r	s	Zhu et.al[2]		our method	
							t_{interp}	t_{total}	t_{interp}	t_{total}
bar-cuboid	981	1053	327	10	30	60	51.2	55.5	35.1	41.5
bar-cylindar	981	1053	2478	10	30	60	266.8	271.6	32.7	39.4
cross	4683	4912	3172	10	42	21	564.1	592.4	11.9	18.3
car	2181	2369	1019	6	20	23	207.7	221.6	16.7	21.2
teddy-global	1773	1755	11193	6	30	30	70.3	83.5	28.7	33.4
teddy-local	1773	1755	11193	6	30	30	302.8	313.7	33.2	38.4

Table 2: Summary of all results presented in the paper. The columns indicate the number of DOFs of the object, the number of tetrahedral elements of the object, the vertices number of the last example, the number of the selected eigenfunctions, the number of modal basis, the number of cubature elements, average time(in milliseconds) for example interpolation and total time for one simulation step of [2] and our method, respectively.

Acknowledgements

We would like to thank Steven S. An for sharing his cubature source code and the anonymous reviewers for their comments and suggestions. This work was partially supported by Grant No. 2015 BAK01B06 from The National Key Technology Research and Development Program of China, and Grant No. 61232014, 61421062, 61170205, 61472010 from National Natural Science Foundation of China.

References

[1] Sebastian Martin, Bernhard Thomaszewski, Eitan Grinspun, and Markus Gross. Example-based elastic materials. In *ACM Transactions on Graphics (TOG)*, volume 30, page 72. ACM, 2011.

[2] Fei Zhu, Sheng Li, and Guoping Wang. Example-based materials in laplace–beltrami shape space. In *Computer Graphics Forum*, volume 34, pages 36–46. Wiley Online Library, 2015.

[3] Steven S An, Theodore Kim, and Doug L James. Optimizing cubature for efficient integration of subspace deformations. In *ACM Transactions on Graphics (TOG)*, volume 27, page 165. ACM, 2008.

[4] Christian Schumacher, Bernhard Thomaszewski, Stelian Coros, Sebastian Martin, Robert Sumner, and Markus Gross. Efficient simulation of example-based materials. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 1–8. Eurographics Association, 2012.

[5] Stefan Fröhlich and Mario Botsch. Example-driven deformations based on discrete shells. In *Computer graphics forum*, volume 30, pages 2246–2257. Wiley Online Library, 2011.

[6] Ben Jones, Nils Thuerey, Tamar Shinar, and Adam W Bargteil. Example-based plastic deformation of rigid bodies. *ACM Transactions on Graphics (TOG)*, 35(4):34, 2016.

[7] Yuki Koyama, Kenshi Takayama, Nobuyuki Umetani, and Takeo Igarashi. Real-time example-based elastic deformation. In *Proceedings of the 11th ACM SIGGRAPH/Eurographics conference on Computer Animation*, pages 19–24. Eurographics Association, 2012.

[8] Wenjing Zhang, Jianmin Zheng, and Nadia Magnenat Thalmann. Real-time subspace integration for example-based elastic material. In *Computer Graphics Forum*, volume 34, pages 395–404. Wiley Online Library, 2015.

[9] Kevin Wampler. Fast and reliable example-based mesh ik for stylized deformations. *Acm Transactions on Graphics*, 35(6):235, 2016.

[10] Eftychios Sifakis and Jernej Barbic. Fem simulation of 3d deformable solids: a practitioner's guide to theory, discretization and model reduction. In *ACM SIGGRAPH 2012 Courses*, page 20. ACM, 2012.

[11] A. Pentland and J. Williams. Good vibrations: modal dynamics for graphics and animation. *Acm Siggraph Computer Graphics*, 23(23):207–214, 1989.

[12] Gyu Choi Min and Hyeong Seok Ko. Modal warping: real-time simulation of large rotational deformation and manipulation. *IEEE Transactions on Visualization & Computer Graphics*, 11(1):91–101, 2005.

[13] Jernej Barbič and Doug L James. Real-time subspace integration for st. venant-kirchhoff deformable models. In *ACM transactions on graphics (TOG)*, volume 24, pages 982–990. ACM, 2005.

[14] David Harmon and Denis Zorin. Subspace integration with local deformations. *Acm Transactions on Graphics*, 32(4):96, 2013.

[15] Olga Sorkine and Marc Alexa. As-rigid-as-possible surface modeling. *Eurographics Symposium on Geometry Processing*, pages 109–116, 2007.

[16] Dong Xu, Hongxin Zhang, Qing Wang, and Hujun Bao. Poisson shape interpolation. *Graphical Models*, 68(3):268–281, 2006.

[17] T. Winkler, J. Drieseberg, M. Alexa, and K. Hormann. Multi-scale geometry interpolation. In *Computer Graphics Forum*, pages 309–318, 2010.

[18] Christoph Von-Tycowicz, Christian Schulz, Hans-Peter Seidel, and Klaus Hildebrandt. Real-time nonlinear shape interpolation. *ACM Transactions on Graphics (TOG)*, 34(3):34, 2015.

[19] Martin Reuter, Franz Erich Wolter, and Niklas Peinecke. Laplace-beltrami spectra as 'shape-dna' of surfaces and solids. *Computer-Aided Design*, 38(4):342–366, 2006.

[20] Martin Reuter. Hierarchical shape segmentation and registration via topological features of laplace-beltrami eigenfunctions. *International Journal of Computer Vision*, 89(2-3):287–308, 2010.

[21] A. Kovnatsky, M. M. Bronstein, A. M. Bronstein, K. Glashoff, and R. Kimmel. Coupled quasi-harmonic bases. In *Computer Graphics Forum*, pages 439–448, 2013.

[22] Jernej Barbič and Yili Zhao. Real-time large-deformation substructuring. *Acm Transactions on Graphics*, 30(4):1–8, 2011.

[23] Danny M. Kaufman, Shinjiro Sueda, Doug L. James, and Dinesh K. Pai. Staggered projections for frictional contact in multibody systems. *Acm Transactions on Graphics*, 27(5):32–39, 2008.

[24] Xiaofeng Wu, Rajaditya Mukherjee, and Huamin Wang. A unified approach for subspace simulation of deformable bodies in multiple domains. *ACM Transactions on Graphics (TOG)*, 34(6):241, 2015.

[25] Christoph Von Tycowicz, Christian Schulz, Hans Peter Seidel, and Klaus Hildebrandt. An efficient construction of reduced deformable objects. *Acm Transactions on Graphics*, 32(6):1–10, 2013.

[26] Yin Yang, Dingzeyu Li, Weiwei Xu, Yuan Tian, and Changxi Zheng. Expediting precomputation for reduced deformable simulation. *Acm Transactions on Graphics*, 34(6):1–13, 2015.

[27] Jernej Barbič and Doug L. James. Subspace self-collision culling. *Acm Transactions on Graphics*, 29(4):1–9, 2010.

[28] Yun Teng, Miguel A. Otaduy, and Theodore Kim. Simulating articulated subspace self-contact. *Acm Transactions on Graphics*, 33(4):70–79, 2014.