Haptic Rendering Based on RBF Approximation from Dynamically Updated Data

Jan Fousek[1], **Tomáš Golembiovský**[1], **Jiří Filipovič**[1], **and Igor Peterlík**[2]

1 **Faculty of Informatics, Masaryk University, Czech Republic**
2 **Institute of Computer Science, Masaryk University, Czech Republic**

Abstract

In this paper, an extension of our previous research focused on haptic rendering based on interpolation from precomputed data is presented. The technique employs the radial-basis function (RBF) interpolation to achieve the accuracy of the force response approximation, however, it assumes that the data used by the interpolation method are generated on-the-fly during the haptic interaction. The issue caused by updating the RBF coefficients during the interaction is analyzed and a force-response smoothing strategy is proposed.

Keywords and phrases haptic rendering, radial-basis function approximation, precomputation, deformation modeling

Digital Object Identifier 10.4230/OASIcs.MEMICS.2010.26

1 Introduction and Related Work

The real-time haptic interaction with deformable objects is an interesting area of research with wide range of applications. However, the haptic rendering of response force that requires high refresh rate (at about 1 kHz), heavily restrict the cost of computations needed for updating the force. Nevertheless, the expensive computations must be involved when for example realistic deformation modeling is considered, motivated by construction of surgical simulators for training or operation planning. In this case, the behaviour of the scene is goverened by models emerging in the theory of elasticity, resulting in computationally expensive calculations that cannot be performed inside the haptic loop.

Besides various strategies such as simplification of the underlying mathematical models, an approximation from precalculated data has been successfully exploited. A pioneering work employing the force extrapolation to achieve the haptic refresh rate for interaction with linear anisotropic deformable model is given in [10]. In [6], the force response is computed in haptic loop by linear interpolation of precalculated deflection curves stored in mesh nodes. A data-driven haptic rendering based on the radial-basis function (RBF) interpolation of measured data is studied in [4]. The work is further extended in [5] by considering the interpolation for reproducing also the viscoelastic properties. A comprehensive system based on RBF-based neural networks is described in [1].

In this paper, an extension our earlier work based on precomputation and approximation of state spaces [8] is presented. Originally, the method employed fast polynomial interpolation using regularly distributed data generated during off-line phase. Two modifications of the approach has been proposed: first, in [9], it was shown that the accuracy of the method can be significantly improved by employing radial-basis function (RBF) interpolation. Second, in [3], the original method has been modified to avoid the time-consuming recalculation of state space by introducing on-line precomputation when the states needed for interpolation are generated directly during the haptic interaction. However, it was supposed that only

Sixth Doctoral Workshop on Math. and Eng. Methods in Computer Science (MEMICS'10)—Selected Papers.
Editors: L. Matyska, M. Kozubek, T. Vojnar, P. Zemčík, D. Antoš; pp. 26–31
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

polynomial interpolation can be used for that case. The main contribution of this paper is to show that two modifications described above can be combined resulting in technique taking advantage from both the on-line state generation and RBF approximation.

2 Preliminaries

2.1 Radial-Basis Function Interpolation and Haptic Rendering

The radial-basis function (RBF) interpolation is used for approximating a function $f(x)$ in arbitrary point x from a given vector of evaluations $\mathbf{f} = (f_1, \dots f_n)$ in arbitrarily distributed points $(x_1, \dots x_n)$. It is computed as

$$\tilde{f}(x) = p(x) + \sum_{i=1}^{n} \lambda_i \phi\left(|x - x_i|\right) \tag{1}$$

where ϕ is the radial-basis function, $p = c_0 + c_1 x \dots c_m x^m$ is a polynomial and λ_i are interpolation coefficients. The vectors $\boldsymbol{\lambda}$ and $\mathbf{c}$ of coefficients λ_i and c_i, respectively, are computed from the evaluation vector $\mathbf{f}$ as follows:

$$\begin{pmatrix} \mathbf{A} & \mathbf{P} \\ \mathbf{P}^\top & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ \mathbf{c} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ 0 \end{pmatrix} \tag{2}$$

where $A_{ij} = \phi\left(|x_i - x_j|\right)$ and $P_{ij} = p_j(x_i)$ for basis polynomial p_j. In our setting, linear polynomial p is considered together with linear ($\phi(r) = r$) and cubic ($\phi(r) = r^3$) radial-basis functions.

In [9], the RBF method is employed to approximate the components of the force response $f(x)$ for an arbitrary position x of the haptic interation point (HIP) during haptic loop. However, before the haptic interaction is executed, a large set of force responses f_i associated to various positions x_i of HIP (being referred as *state space*) is constructed numerically during an off-line phase.

2.2 State-space Construction

In [3], *on-line construction* of the state space has been introduced: instead of building the entire state-space during the off-line phase, the calculation of a force responses is performed directly during the interaction. The distribution of the points x_i of the state space for which the responses f_i are being calculated is determined on-the-fly by the actual position and motion of the HIP.

It has been shown that although the precomputation of a single state (i.e. the force response) takes from hundreds milliseconds to seconds, the interpolation runs stably at haptic refresh rate (1 kHz), provided the states are generated concurrently by dozens of processes running in distributed environment. Thus, the state space used by interpolation is gradually augmented: in our setting from about dozens to hundreds of new states were added to the actual space during one second. It was supposed that in each step of the haptic loop the HIP is located within the area enclosed by regularly-distributed known states, so the polynomial interpolation can be used to approximate the associated force response. However, due to this assumption, the HIP speed is restricted as quantitatively evaluated in [3].

3 RBF Approximation on Updated Data

3.1 From Interpolation towards Extrapolation

The issue addressed in this paper is represented by the limited speed of motion towards the horizon of the actually known state space. In contrast to the polynomial interpolation using the regularly distributed data, the approximation based on RBF is able to make use of data scattered in the space. Therefore, it is possible to leave the space fully populated by already known states and approximate the values of the force response in areas with sparsely distributed known states and even extrapolate outside the known state space. The downside of this approach is increase of the approximation error and possible loss of the accuracy.

Further, in contrary to the interpolation based on regular grid, the RBF method can make use of arbitrary number of known points. Therefore, two tasks arise to be performed by the process handling the interpolation :

1. select up to n nearest points, where n is constant and
2. compute the RBF coefficients for selected points by solving Eq. 2.

There is no need to explicitly switch between extrapolation and interpolation, the RBF method approximates the forces in the same way both outside and inside the area defined by the known states. When leaving the densely precomputed area and the approximation shifts to the extrapolation, numerous distributions of the nearest known points can occur. Following distributions describe the characteristic situations with respect to the HIP movement which can occur during the simulation (see figure 1 for illustration):

- HIP enters a **sparse area** where a limited number of states is available: RBF starts partially extrapolating from the remote points in the safe area,
- HIP gets into **gap**: RBF provides a bridge between two safe areas,
- HIP gets into **blank area** where no states are known: RBF works as pure extrapolation method.

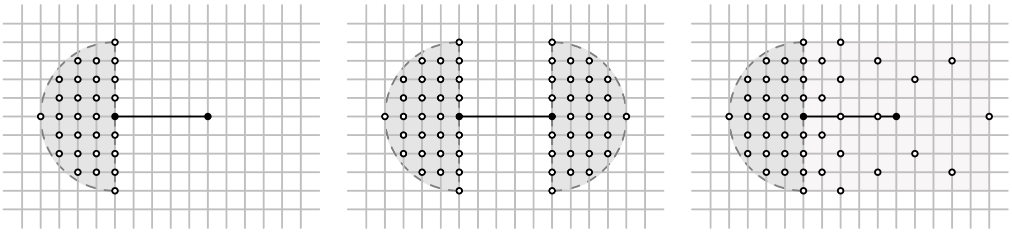

Figure 1 Characteristic situations when leaving the densely-populated area. The known states are shown on the regular grid together with the path of HIP. From left to right: sparse area, bridge and blank area.

3.2 Returning from Extrapolation towards Interpolation

The distribution of known states with respect to the position of HIP is updated for two reasons: either new states requested previously arrive from the solvers, or the position of the HIP changes rendering some new known states as nearer than some previously used states. In both cases, the blank area can change into sparse area or gap and further into safe area shifting the approximation back from extrapolation to interpolation.

To reflect the changes of the state space inside the approximation procedure, it is necessary to compute updated set of RBF coefficients. However, two issues arise here: first, the complexity of solving Eq. 2 is in $O(n^3)$ and therefore for a large number of states, recomputing the coefficients can be a computationally demanding task that cannot be done inside the haptic loop. Second, switching to a new set of RBF coefficients in one step can result in non-continuous change in the force response being perceived as a jump in the force feedback. Therefore the interpolation is implemented as two concurrent processes:

Updating process is event-driven: if a state space is updated or HIP travels a distance larger than predefined threshold, it selects n known states enclosing the HIP and computes the updated RBF coefficients $(\boldsymbol{\lambda}', \mathbf{c}')$ according to Eq. 2.

Interpolating process is running at haptic rate: in each step, it computes the force response from the actual RBF coefficients $(\boldsymbol{\lambda}, \mathbf{c})$ according to Eq. 1. If updated set of coefficients $(\boldsymbol{\lambda}', \mathbf{c}')$ is available, it performs a *damped coefficient switch*, i. e. in following steps of the haptic loop it computes interpolated forces from both $(\boldsymbol{\lambda}, \mathbf{c})$ and $(\boldsymbol{\lambda}', \mathbf{c}')$ and calculates weighted average moving the weight in every tenth step (10 ms) between $(\boldsymbol{\lambda}, \mathbf{c})$ and $(\boldsymbol{\lambda}', \mathbf{c}')$ so that the force difference is lower than 7% which is a lower bound for the just noticeable difference [7, 11] for variety of haptical hand interactions.

By shifting the weight between two successive RBF coefficients on lower frequency than is the human perceptive resolution and by limiting the change of resulting force by the just noticeable difference we can assure the user will not experience twitches or vibrations caused by discontinuous changes in the force feedback.

4 Results

We have incorporated the modifications presented in previous section in our haptical simulator [2] and present here the evaluation of this implementation. Also to be able to study the effects of dynamical changes in the set of precomputed set of states and to simulate the on-line precomputation in deterministic way, we have modified the scheduler so that it was able to add the states to the known set with in advance given timing.

All the experiments have been performed on PC with 2x Dual Core AMD Opteron Processor 2218 2.6 GHz with 3 GB RAM using the model of the human liver with 1777 elements (see [9] for details). The total length of model was about 22 cm and we have experimented with force response associated with compression-like deformation between 0 and 6 cm. The state-space was given by a regular grid with cell size of 6 mm.

First, the experiments have confirmed that it is possible to compute the approximation using the RBF inside the haptic loop, whereas the RBF update spreads over dozens of iterations of the haptic loop. For example, for 100 states it takes in average 160 ms to update the RBF and 60 µs to approximate the forces, for 300 states it is 1.6 s and 200-250 µs, respectively.

Further, the magnitude of extrapolation error has been studied (Fig. 2). These results were obtained by displacing one node of the mesh by 6 cm in the direction of the x axis towards the center of the model considering three characteristic situations introduced in section 3.1. Note that presented values represent time independent equilibrium and is therefore not influenced by the damping effect. The absolute and relative errors are given w.r.t. the completely known state space. In all the cases state space of 100 nearest known states was used. In the first case the displacement path started at the border of the known space and continued towards space without any precomputed values. In the second case, it crossed the empty space towards another area continuously populated with known states.

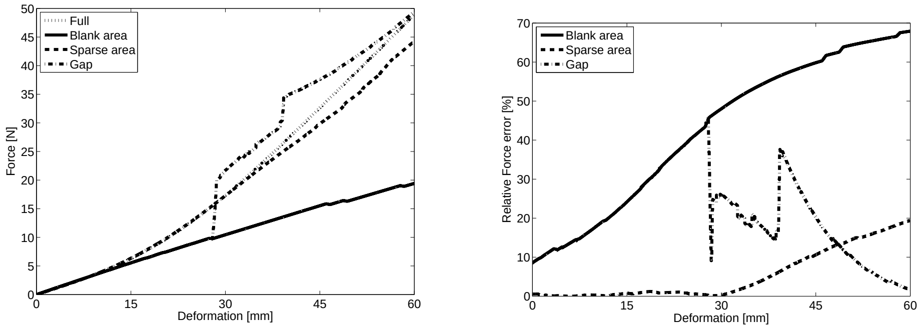

Figure 2 Resulting force (left) and relative error (right) of the extrapolation for the three characteristic situations from section 3.1.

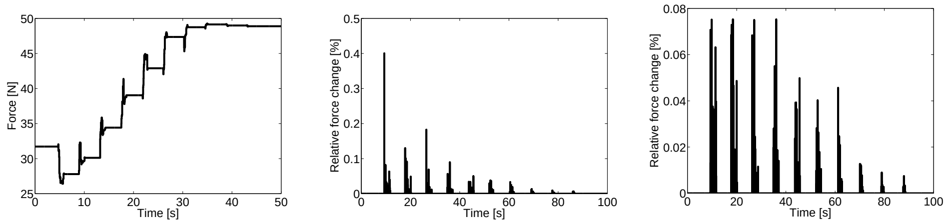

Figure 3 Damping the RBF coefficient switch: comparison of the force response change after comming sets of states. Absolute force (right) is presented together with ratio between resulting force for sucessive haptic frames an non-damped (middle) and damped (right) coefficient switching.

In the case of sparse area, the density of known states descended continuously along the deformation path.

The error of the extrapolation in the case of blank area grew rapidly with the length of deformation while few sparsely distributed known states along the path were enough for the RBF to approximate the resulting force with relatively low error. Also note two rapid changes in the approximated force in the case of the gap crossing. These come from the change in the set of nearest points used for the approximation: the first jump corresponds to the point where first points from the other side of the gap are added, the second jump then similarly corresponds to the removing of the last points from the starting side of the gap.

Finally, the damped switch of the RBF coefficients was verified experimentally. The HIP was displaced similarly to the blank area setup in the area without any precomputed states extrapolating from distant continuously precomputed area. Then in intervals of 10 s groups of states were added to the known state space starting with most distant states to the HIP filling the empty space completely in the end. The Fig. 3 depicts jumps in the absolute value of feedback force together with relative change of the force between two successive iterations of the haptic loop and it can be clearly seen, that without damping the change exceeds the just noticeable difference by far, whereas the proposed damping was able to spread the jump in the force in several successive frames of the haptic feedback.

5 Conclusions and Future Work

In this paper, an extension to the haptic rendering method based on precomputation and approximation has been proposed. It was shown that the RBF interpolation, which improves the accuracy of the approximation, can be employed also in the case when the states are generated directly during the haptic interaction. Possible issues were identified and resolved: the RBF coefficients were recomputed by additional thread running on lower frequency and the switch to the updated coefficients were damped in order to avoid artificial discontinuities in the force response. The technique proposed in the paper was verified by experiments presented in result section.

Our future work will focus on finding new scheduling strategies considering e. g. motion direction heuristics, sparse area precomputation or prioritizing and preempting some states. We also plan to improve the speed of RBF coefficient update by using some advanced technique for matrix inversions.

Acknowledgements The financial support provided by Ministry of Education, Youth and Sport of the Czech Republic under the research program MSM0021622419 is acknowledged. The access to the METACentrum computing facilities provided under the research intent MSM6383917201 is acknowledged.

References

1 D. Deo and S. De. Phyness: A physics-driven neural networks-based surgery simulation system with force feedback. In *Proc. of 3rd Joint Eurohaptics Conference and Haptic Symposium*, pages 30–34, 2009.

2 J. Filipovič, I. Peterlík, and M. Madzin. Parallel real-time deformation simulator. `http://parides.sourceforge.net/`, 2010.

3 J. Filipovič, I. Peterlík, and L. Matyska. On-line precomputation algorithm for real-time haptic interaction with non-linear deformable bodies. In *Proc. of The 3rd Joint EuroHaptics Conf. and Haptic Symposium*, pages 24–29, 2009.

4 R. Hover, M. Harders, and G. Szekely. Data-driven haptic rendering of visco-elastic effects. In *Proc. of Symp. on Haptic Interf. for Virt. Env. and Teleop. Systems*, pages 201–208, 2008.

5 R. Höver, G. Kósa, G. Székely, and M. Harders. Data-driven haptic rendering–from viscous fluids to visco-elastic solids. *IEEE Trans. Haptics*, 2(1):15–27, 2009.

6 M. Mahvash and V. Hayward. Haptic simulation of a tool in contact with a nonlinear deformable body. In *Proc. of Surg. Sim. and Soft Tissue Def.*, pages 311–320, 2003.

7 X.D. Pang, H.Z. Tan, and N.I. Durlach. Manual discrimination of force using active finger motion. *Perception & Psychophysics*, 49(6):531–540, 1991.

8 I. Peterlik and L. Matyska. An algorithm of state-space precomputation allowing non-linear haptic deformation modelling using finite element method. In *Proc. of the 2nd Joint EuroHaptics Conference and Haptic Symposium*, pages 231–236, 2007.

9 I. Peterlík, M. Sedef, C. Basdogan, and L. Matyska. Real-time visio-haptic interaction with static soft tissue models having geometric and material nonlinearity. *J. of Comp.& Graph.*, 34(1):43 – 54, 2010.

10 G. Picinbono, J-C. Lombardo, H. Delingette, and N. Ayache. Improving realism of a surgery simulator: linear anisotropic elasticity, complex interactions and force extrapolation. *Jour. of Vis. and Comp. Anim.*, 13(3):147–167, july 2002.

11 H.Z. Tan, X.D. Pang, and N.I. Durlach. Manual resolution of length, force, and compliance. *Advances in Robotics*, 42:13–18, 1992.