Pacific Graphics (2015)
N. J. Mitra, J. Stam, and K. Xu (Editors)

Short Paper

Adaptive Hierarchical Shape Matching

Yuan Tian[1] Yin Yang[2] Xiaohu Guo[1] Balakrishnan Prabhakaran[1]

[1]University of Texas at Dallas, Richardson, Texas, U.S.A
[2]University of New Mexico, Albuquerque, New Mexico, U.S.A

Abstract

In this paper, we present an adaptive hierarchical method allowing users to interact with geometrically complex 3D deformable objects based on an extended shape matching approach. Our method extends the existing multi-resolution shape matching methods with improved energy convergence rate. This is achieved by using adaptive integration strategies to avoid insignificant shape matching iterations during the simulation. As demonstrated in our experimental results, the proposed method provides an efficient yet stable deformable simulation of complex models in real-time.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Animation—

1. Introduction

Simulating 3D volumetric deformation has long been an active topic in many research communities like computational mechanics, computer graphics, virtual reality and so on. Many computational methods, such as the finite element method (FEM) [CK05, BJ05] or the mass-spring system [THMG04, MZS*11] have been developed to model the dynamics of the soft objects. Among all types of methods, the shape matching [MHTG05] technique turns out to be a competitive candidate. One of the most attractive advantages associated with the shape matching approach is its unconditionally stable integration. This method is developed based on the *position based dynamics* (PBD) [BMOT13, MHHR07], is able to provide a fast, controllable, and unconditionally-stable dynamic simulation. Unlike mass-spring system, this method [MHTG05, MC11] is essentially a meshless method grouping the particle cloud into clusters. The computation associated with each cluster is independent making the shape matching based deformable model much more light-weight. River and James [RJ07] used overlapped clusters (lattice) to control the stiffness of the deformable object. Steinemann [SOG08] extended this work with dynamic adaptive selection of levels of details (LODs). Geometrically complex 3D models usually have a large number of clusters to capture detailed local deformation, which could lead to a slow deformation convergence. To improve the energy convergence rate, multigrid methods [BWD13, Mül08] have been adopted. In this paper, we propose a novel adaptive particle cluster hierarchy. The proposed method is able to effectively boost the convergency rate of the deformation energy during the simulation with three adaptive iteration strategies by tracking the variation of energy density and avoiding unnecessary computations accordingly. It inherits the unconditional stability and high efficiency of shape matching ap-

DOI: 10.2312/pg.20151278

proach, and it can be seamlessly integrated within the state-of-the-art shape matching based frameworks.

2. Adaptive Shape Matching using Cluster Hierarchy

In shape matching, each particle of the cluster is associated with a mass, an initial position, a current position as well as a goal position denoted with m_{ij}, $\mathbf{x}^0_{ij}$, $\mathbf{x}_{ij}$ and $\mathbf{g}_{ij}$, respectively for the jth particle in the ith cluster. The quadratic deformation energy/potential E_i is defined as the mass-weighted summation of square distance between current positions and the goal positions of all particles in cluster i:

$$\begin{aligned} E_i &= \sum_j m_{ij} \mid \mathbf{g}_{ij} - \mathbf{x}_{ij} \mid^2 \\ &= \sum_j m_{ij} \mid \mathbf{R}_i \mathbf{x}^0_{ij} + \mathbf{t}_i - \mathbf{x}_{ij} \mid^2 . \end{aligned} \tag{1}$$

where $\mathbf{R}_i \in \mathbb{SO}^3$ and $\mathbf{t}_i \in \mathbb{R}^3$ represent the best fitting rotation and the translation. We refer such computation for obtaining $\mathbf{R}$ and $\mathbf{t}$ of all clusters as the *shape matching* (SM). We call an individual loop of performing SM and neighbor averaging as an SM iteration. The procedure of SM iteration is similar to the so-called local/global optimization in recent contributions [BML*14], and it is guaranteed that each iteration will monotonically reduce the deformation potential of the entire voxel mesh. After SM iterations are completed, a forward Euler with time step size h is followed to update the velocity and displacement of all the particles.

In our method, a cluster hierarchy of multiple levels is constructed. The deformable simulation begins with the SM iteration at the top (coarsest) level. After sufficient energy reduction is observed, the algorithm proceeds to clusters at the next level while the results at upper level are inherited. Afterwards, the external forces will be incorporated with forward Euler at clusters of the bottom level, which triggers vibrational deformations due to the inertia terms and pulls the clusters away from their goal positions. At the next time step, the initial configuration of top-level clusters ($\mathbf{R}$ and $\mathbf{t}$) will be set as the the blended rotation (e.g. using Slerp [Sho85]) and translation of lower-level clusters. This procedure mimics the standard *V-cycle* in multigrid approach [BHM00] and has been adopted in many existing SM-based frameworks [SOG08, BWD13]. We refer this algorithm as hierarchical shape matching (HSM).

Our framework further improves the simulation efficiency. The key inspiration is that not all SM iterations play equally important roles in shaping the deformed geometry of the object. In fact, our experiment shows that a considerable amount of SM iterations (over 25%) could be avoided. This is achieved by checking three conditions during the multi-level SM iteration, namely the termination condition, the subdivision condition and the origination condition, which are to be discussed in detail. Typically, the hierarchy of three to five levels is used in our experiments. We may incorporate more overlapping particles as did in fast lattice shape matching (FLSM) [RJ07] to further tweak the "stiffness" of the object.

Termination Condition Following the intuition that the SM iteration goes to the finer level when current level iterations do not effectively reduce the energy potential, we evaluate the energy reduction rate r_l at lth level as:

$$r_l = \sum_i \Psi^k_{l,i} - \Psi^{k-1}_{l,i}, \tag{2}$$

where $\Psi_{l,i}$ defines the *energy density* of cluster i. Superscripts k and $k-1$ indicate the SM iteration index. $\Psi_{l,i}$ is computed as $\Psi_{l,i} = \frac{E_{l,i}}{n_i d_l^2}$, where n_i is the number of particles of ith cluster and d_l is the voxel size at level l. We use a threshold value T_r is used to examine the effectiveness of SM iteration. Iteration at current level l will be terminated and the simulator moves to the next level when the *termination condition*, $\mathcal{C}_t \equiv r_l < T_r$, is satisfied.

Subdivision Condition If clusters at coarser level already well capture the deformed mesh geometry, we should not perform the iteration at finer levels. In other words, SM iteration will only be applied to clusters whose energy density is larger than some threshold T_s i.e. $\mathcal{C}_s \equiv \Psi_{l,i} > T_s$, which is named as *subdivision condition*. All the child clusters of a cluster meats C_s are called *active clusters*. Figure 1 shows how the clusters are evolving and the SM information are passed as the Budda model is being bent.

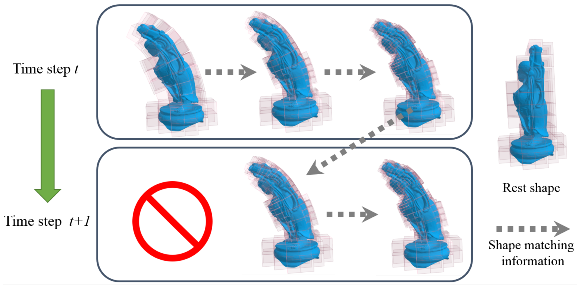

Figure 1: An example of the our SM iteration strategy. The dashed grey arrows indicate how the SM information are passed.

Origination Condition Due to the external force, the energy potential at the next step will increase. However it is possible that the external forces are subtle and regional, and only alter small local deformation patterns. For instance, a light breeze only sways the leaves of a willow tree while its major branches remain still (Figure). Such small deviations of particles' positions will induce larger perturbation of energy density for clusters at lower levels (because d_l is smaller) and blindly performing complete top-down SM iteration could induce redundant less-effective computations and slow the simulation performance. Based on this observation, we track the energy increase induced by forward Euler integration at all clusters undertaking external forces[†] as $\sum_i \Psi^*_{l,i} - \Psi_{l,i}$, where $\Psi^*_{l,i}$ denotes the energy density after applying the forward Euler. The *origination condition* will find a level whose energy reduction rate in previous time step is comparable with the energy increase: $\mathcal{C}_o \equiv r_l \propto \sum_i \Psi^*_{l,i} - \Psi_{l,i}$ and set it as the starting level.

Algorithm 1: Adaptive shape matching.

```
$l \leftarrow 0$;  /* $l$ is the current shape matching level */
while simulation is active do
    while $l < n$ do
        for all clusters at level $l$ do
            inherit R and t from parents;
        while $!\mathcal{C}_t$ do
            for all active cluster at level $l$ do
                shape matching iteration;
            update $r_l$;
        for all active cluster at level $l$ do
            if $\mathcal{C}_s$ then
                set child clusters as active clusters;
        $l \leftarrow l + 1$;
    /* now $l = n$ */
    for all clusters do
        forward Euler;
    update surface mesh;
    while $l > 0$ do
        if $\mathcal{C}_o$ then
            break;  /* starting level of next step found */
        else
            $l \leftarrow l - 1$;
```

Alg. 1 outlines our adaptive SM iteration strategy. Our method differs from existing methods [SOG08, BWD13] by not only addressing the questions of "where and when the iteration should end?" but also answering "where and when the iteration should start?" which further accelerates the energy convergency.

3. Experimental Results

The proposed framework was implemented using Microsoft Vistual C++ 2010 on a 64-bit Windows 7 PC equipped with an Intel Xeon 2.8 GHz CPU and 6.0 GB RAM. Only single tread was used in our experiment and reported data. Our experiments use 3 models including willow tree (5k faces, 4k vertices), elephant (85k faces, 42k vertices), and dinosaur (123k faces, 58k vertices). Table 1 shows the detailed information of cluster hierarchy setup as well as the comparative time performance. As highlighted in the Table 1, the proposed adaptive iteration rule has a notable advantage of efficiency over the classic HSM method.

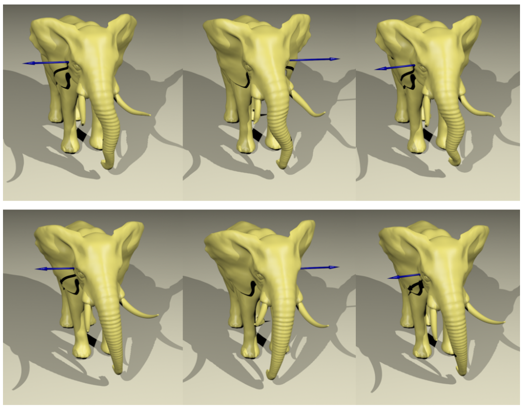

Figure 2: Snapshots of the simulation results using NSM method (top) and our method (bottom). Scripted forces are highlighted as blue arrows in the figure.

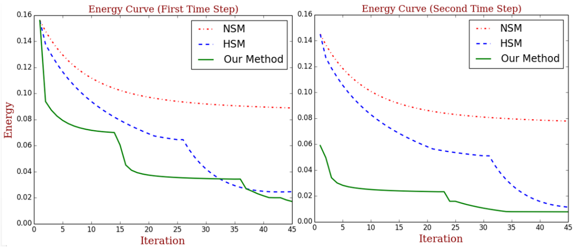

Figure 3: Energy changes over iterations of the first two time steps of elephant model showed in Figure. 2.

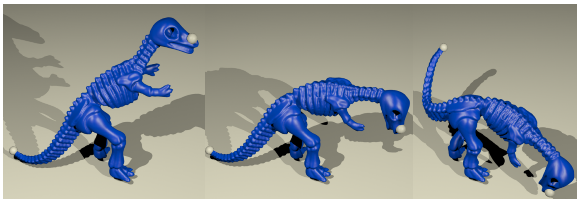

Figure 4: Two haptic devices interact with the deformable dinosaur model. Realistic results are produced even under extreme deformation.

Figure 2 shows a comparison of using our method and the naive shape matching (NSM) approach. The forces, indicated as blue arrows in the figure, are applied trying to shake the nose of the elephant. At each time step, the surface mesh is updated after each cluster completes just one shape matching. Using the NSM (the top row), we can clearly see an unnatural wave-like deformation propagation at the nose. With our method, a more natural result is produced (the bottom row). If we want to achieve the same energy reduction as the one after 15 iterations with our adaptive method, NSM will need over 2,200 iterations. In terms of computation efficiency, classic HSM is about 190 times faster than NSM while our method is up to 310 times faster. Table 1

[†] Since forces are applied to particles at the bottom clusters, all of their parent clusters are also considered.

Model	# Cluster	Comp. intensity (# SM iteration)			Time benchmark		
		NSM	HSM	Our method	NSM	HSM	Our method
Willow tree	$703/3k/10k$	$10.9m$	$35k$	$29k$ ($\sim 372\times$)	$3.81s$	$16ms$	$11ms$ ($\sim 354\times$)
Elephant	$202/954/5k$	$6.9m$	$27k$	$22k$ ($\sim 314\times$)	$1.86s$	$10ms$	$6ms$ ($\sim 310\times$)
Dinosaur	$258/841/3k/17k/98k$	$48m$	$66k$	$59k$ ($\sim 822\times$)	$12.60s$	$22ms$	$16ms$ ($\sim 788\times$)

Table 1: Time and computation performance. *Comp. intensity*: number of SM iteration required on average at each step in order to achieve energy convergency; *Time benchmark*: the average computation time simulating the 3D model for a single time step.

reports the detailed simulation performance. On average, our method is orders-of-magnitude faster than NSM and consistently outperforms HSM by $30-50\%$. In some extreme cases (e.g. very large/subtle deformations occur), our method could bring better performance improvement due to the adaptive iteration strategies used. Figure 3 shows how the deformation potential is reduced along the simulation at the first two times steps using NSM, HSM and our method. The curves correspond to the elephant animation shown in Figure 2. Our method is particularly good at simulating the geometrically complex model with enriched local details. The teaser figure shows the snapshots of a willow tree swaying in the wind. We refer readers to the accompanying video for details, where we show three different scenarios with light, medium and large winds. The proposed adaptive simulation strategy is able to well accommodate various wind fields of different intensity and natural result is produced while the simulation is still efficient. In Figure 4, two haptic devices participate in the interaction with the dinosaur model simultaneously. It can be seen that our system is rather robust even under extreme deformations imposed by the user.

4. Conclusion and Future Work

In this paper, we present a novel system to simulate deformation of geometrically complex objects based on adaptive hierarchical shape matching. We construct a multi-resolution hierarchy of particle clouds with three adaptive strategies to boost the energy convergence speed while the locally detailed deformation is still well captured. Our approach works well with existing methods such as FLSM. There are many future works that can improve the current system. First of all, we still lack of a good representation to accurately incorporate elastic materials with different parameters such as Young's modules or Poisson's ratio. Another promising direction is to further extend our system to the networked environment in a tele-immersive virtual environment. In this case, special cares need to be taken in order to handle the connection stability of the network and provide a high-quality user experience.

5. Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No. 1012975. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[BHM00] Briggs W. L., Henson V. E., McCormick S. F.: *A Multigrid Tutorial*, 2 edition ed. SIAM: Society for Industrial and Applied Mathematics, Philadelphia, PA, June 2000. 2

[BJ05] Barbič J., James D. L.: Real-time subspace integration for St. Venant-Kirchhoff deformable models. *SIGGRAPH '05 24*, 3 (Aug. 2005), 982–990. 1

[BML*14] Bouaziz S., Martin S., Liu T., Kavan L., Pauly M.: Projective dynamics: Fusing constraint projections for fast simulation. *ACM Trans. Graph. 33*, 4 (July 2014), 154:1–154:11. 2

[BMOT13] Bender J., Müller M., Otaduy M. A., Teschner M.: Position-based methods for the simulation of solid objects in computer graphics. In *EUROGRAPHICS 2013 State of the Art Reports* (2013), Eurographics Association. 1

[BWD13] Bender J., Weber D., Diziol R.: Fast and stable cloth simulation based on multi-resolution shape matching. *Computers & Graphics 37*, 8 (2013), 945 – 954. 1, 2, 3

[CK05] Choi M. G., Ko H.-S.: Modal warping: Real-time simulation of large rotational deformation and manipulation. *IEEE Trans. Visual. Comput. Graph. 11*, 1 (2005), 91–101. 1

[MC11] Müller M., Chentanez N.: Solid simulation with oriented particles. *ACM Trans. Graph. 30*, 4 (July 2011), 92:1–92:10. 1

[MHHR07] Müller M., Heidelberger B., Hennix M., Ratcliff J.: Position based dynamics. *J. Vis. Comun. Image Represent. 18*, 2 (Apr. 2007), 109–118. 1

[MHTG05] Müller M., Heidelberger B., Teschner M., Gross M.: Meshless deformations based on shape matching. In *ACM Transactions on Graphics (TOG)* (2005), vol. 24, ACM, pp. 471–478. 1

[Mül08] Müller M.: Hierarchical position based dynamics. *VRIPHYS 8* (2008), 1–10. 1

[MZS*11] McAdams A., Zhu Y., Selle A., Empey M., Tamstorf R., Teran J., Sifakis E.: Efficient elasticity for character skinning with contact and collisions. In *ACM Transactions on Graphics (TOG)* (2011), vol. 30, ACM, p. 37. 1

[RJ07] Rivers A. R., James D. L.: Fastlsm: Fast lattice shape matching for robust real-time deformation. In *ACM SIGGRAPH 2007 Papers* (2007), SIGGRAPH '07. 1, 2

[Sho85] Shoemake K.: Animating rotation with quaternion curves. *SIGGRAPH Comput. Graph. 19*, 3 (July 1985), 245–254. 2

[SOG08] Steinemann D., Otaduy M. A., Gross M.: Fast adaptive shape matching deformations. In *Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2008), SCA '08, pp. 87–94. 1, 2, 3

[THMG04] Teschner M., Heidelberger B., Muller M., Gross M.: A versatile and robust model for geometrically complex deformable solids. In *Computer Graphics International, 2004. Proceedings* (2004), pp. 312–319. 1