

A Smoothed Particle Hydrodynamics Algorithm for Haptic Rendering of Dental filling materials

Brian Tse¹, Alastair Barrow², Barry Quinn³ and William S. Harwin^{4*¶}

June 2015[¶]

Abstract

Using haptic interfaces to assist the training of skills within the curriculum of undergraduate dentists provides a unique opportunity to advance rendering algorithms and engineering of haptic devices. In this paper we use the dental context to explore a rendering technique called smoothed particle hydrodynamics (SPH) as a potential method to train students on appropriate techniques for insertion of filling material into a previously prepared (virtual) dental cavity. The paper also considers how problems of haptic rendering might be implemented on a Graphical Processing Unit (GPU) that operates in the haptics control loop. The filling simulation used 3000 particles to represent the cavity boundary (approx. 1400 particles), tool (approx. 42 particles) and filling material (approx. 1600 particles), running at an average of 447Hz. Novel smoothing function in SPH was developed and its flexibility is presented.

1 Introduction

Haptic interfaces have a good history of use as a skill training aid, in particular for medical, veterinary and

dental education. King's College London requires students to gain practical experience of cavity preparation and restoration in support of their theoretical education at the early stages of dental training and now use a haptic simulator as a viable method to support this teaching with a suite of workstations in use since 2009[11, 2].

Removing material is a relatively well explored process in haptic rendering, however tooth restoration also requires filling and curing of the cavity. In this paper we consider smoothed particle hydrodynamics (SPH) as a framework for applying soft body haptic rendering to the simulation of dental filling. A new adhesion model in SPH is used to replicate the soft body dynamics of dental materials. Finally the benefits of using GPU hardware for haptic rendering of SPH are evaluated.

Smoothed particle hydrodynamics facilitates the need to change the properties of the filling material during operation, thus enabling simulation of characteristics such as work hardening and curing of the filling material with ultraviolet light.

2 Background to smoothed particle hydrodynamics

Smoothed particle hydrodynamics (SPH) provide a useful framework for rendering fluids. According to Liu[7], Smoothed particle hydrodynamics was originally developed by Lucy [8], Gingold and Monaghan[4] as a mesh-free particle method for astrophysical analysis. The work was subsequently used for simulating fluids and is now widely used

^{*1}Brian Tse (briantse@design@gmail.com) received his PhD from The University of Reading, UK

^{†2}Alastair Barrow is a director of Generic Robotics Ltd, UK

^{‡3}Barry Quinn is with Kings College Dental School, London

^{§4}William Harwin (w.s.harwin@reading.ac.uk) is with The University of Reading, UK

[¶]Published in IEEE World Haptics Conference (WHC) (June 2015).

in computer animation. Cirio et al.[1] have implemented SPH in a realistic haptic simulation of fluids where they demonstrate stable rendering of fluid flow (pancake batter) using a 6DoF Virtuoso from Haption. Several authors give a general background to SPH[7, 3, 6, 9, 10].

Smoothed particle hydrodynamics provides a method for solving systems of differential equations such as the Navier Stokes equation. Kernels¹ are used to interpolate the value of a field variable and a well chosen Kernel allows the calculation to be confined locally to a particle. Differentiable kernels allow easy calculation of the gradient or Laplacian via the kernel rather than on the fields associated with each particle. Any particle j in a collection of N particles are attributed with a position \mathbf{r}_j , velocity \mathbf{v}_j , mass m_j , and density ρ_j . The value of a field f_j such as pressure can be also be attributed to each particle thus allowing interpolation of the field across all particles in the system ($\langle f(\mathbf{r}) \rangle$) to be computed using the following equation

$$\langle f(\mathbf{r}) \rangle = \sum_j^N m_j \frac{f_j}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h) \quad (1)$$

Where : $W(\mathbf{r} - \mathbf{r}_j, h)$ is a kernel described below (8, 9 and 10). Taking the gradient of the field can be done by noting that the only term dependent on position in equation 1 is the kernel $W(\mathbf{r} - \mathbf{r}_j, h)$. However some care is needed since numerical problems can lead to non symmetrical solutions or a non zero differential value in a constant field so the variation suggested by Müller[10] is usually considered where an average field is used for the interpolation.

$$\langle \nabla f(\mathbf{r}) \rangle = \sum_{j \neq i} \frac{f_i + f_j}{2} \frac{m_j}{\rho_j} \nabla W(\mathbf{r} - \mathbf{r}_j, h)$$

According to[7], there are several properties for constructing the smoothing function:

1. The integral of the smoothing function over the support domain must be equal to 1, satisfying the Dirac delta function.

¹Since both the SPH algorithm and CUDA use the term Kernel the latter will be prefaced as CUDA kernels although it should be evident from context

2. Compact support (non zero in a limited space) and kernel symmetry.
3. Positive support domain of centered particle.
4. The effect of neighbour particles should decay smoothly as the distance increases.

3 Methods

We have adapted smoothed particle hydrodynamics algorithms to simulate dental filling. The demonstration hardware ran the SPH algorithm on a GPU with a second GPU for the visual rendering. Additional properties included adhesion/cohesion forces between the particles well as a modified kernel to allow efficient implementation.

3.1 Particle Interaction Forces

Four particle interaction forces were considered. These were gravity, pressure, viscosity and the forces due to cohesion and adhesion² (eqn 3). The force per unit volume due to pressure is $F_{press} = -\nabla P(\mathbf{r})$, where P is the pressure. The SPH formulation of F_{press} (eqn 5). In the case of viscosity $F_{vis} = \mu \nabla^2 \mathbf{V}(\mathbf{r})$, where \mathbf{V} is the velocity. The Laplacian of this vector field is also calculated on the Laplacian of the kernel. The viscous force is shown in eqn 4. Both the pressure and viscous force and their smoothing function used in this research were based on Müller[10] and Cirio[1] as it produces symmetrical forces and reduces numerical errors.

$$F_{total} = F_{LJ} + F_{vis} + F_{gravity} \quad (2)$$

where the F_{LJ} is the Lennard-Jones force consists of pressure and adhesion:

$$F_{LJ} = F_{adh} + F_{press} \quad (3)$$

$$F_{vis} = \sum \frac{\mu_i + \mu_j}{2} m_j \frac{V_j - V_i}{\rho_j} \nabla^2 W_{vis}(\vec{r}_{ij}, h) \quad (4)$$

²Cohesion and adhesion refer to the attraction between similar or dissimilar particles respectively. The term adhesion is used when describing situations that would also include cohesion

$$F_{press} = - \sum \frac{P_i + P_j}{2} \frac{m_j}{\rho_j} \nabla W_{spike}(\vec{r}_{ij}, h) \quad (5)$$

where:

μ_i, μ_j	: Viscosity associated with particle i and j
m_j	: Particle j mass
V_i, V_j	: Velocity of particle i and j
P_i, P_j	: Pressure of i and j
W_{spike}, W_{vis}	: Smoothing function used in pressure and viscosity force, shown in (9) and (10).

The particle mass density and pressure are shown in (6) and (7).

$$\rho_i = - \sum m_j W_{poly6}(\vec{r}_{ij}, h) \quad (6)$$

$$P_i = K_{P(i)}(\rho_i - \rho_{init}) \quad (7)$$

where:

W_{poly6}	: Smoothing function used in mass density calculation, shown in (8)
$K_{P(i)}$: Mass density recovery constant of particle i .

Different smoothing functions facilitate field calculations, and Müller based the pressure smoothing on Debrun's Spiky kernel and designed new smoothing kernels for mass density and viscosity. (See eqns 8,9 and 10)

$$W_{poly6}(\vec{r}_{ij}, h) = C_a \begin{cases} (h^2 - |\vec{r}_{ij}|^2)^3 & 0 \leq |\vec{r}_{ij}| \leq h \\ 0 & |\vec{r}_{ij}| > h \end{cases} \quad (8)$$

$$\nabla W_{spiky}(\vec{r}_{ij}, h) = C_b \frac{\vec{r}_{ij}}{|\vec{r}_{ij}|} \begin{cases} (h - |\vec{r}_{ij}|)^2 & 0 \leq |\vec{r}_{ij}| \leq h \\ 0 & |\vec{r}_{ij}| > h \end{cases} \quad (9)$$

$$\nabla^2 W_{vis} = C_c \begin{cases} (h - |\vec{r}_{ij}|) & 0 \leq |\vec{r}_{ij}| \leq h \\ 0 & |\vec{r}_{ij}| > h \end{cases} \quad (10)$$

Where C_a , C_b and C_c are normalization factor that depend on the dimension of the problem domain.

Dental filling material includes adhesion and cohesion properties. The SPH method in[3] also considered these attraction properties and proposed the Spiky Smoothing Kernel. Müller and Circio also employed this kernel for liquid based simulation. This smoothing kernel allowed the particle interaction to produce a force and displacement characteristic similar to the Lennard-Jones model of particle potential. Their cohesion properties were described as a function of low fluid pressure, which was modified from a gas pressure function. The cohesion was achieved by adjusting the mass density recovery constant ($K_{P(i)}$) and the resting density (ρ_{init}) in (7). However, this method lacked the ability to control attraction and repulsion independently. Any changes to $K_{P(i)}$ and ρ_{init} would result in a change in both attraction and repulsion force. For material such as dental filling, this method required some modification.

In this research, a new adhesion/cohesion method model was developed within the SPH framework, based on Desbrun and Müller methods. The cohesion force here is described as a material physical connection rather than pressure gradient force. Instead of modelling the Lennard-Jones model in a single pressure function, the pressure and the adhesion force both have individual smoothing function to replicate the shape of particle repulsion and attraction. These two forces are modelled separately in order to achieve independent control. Care is needed to blend pressure and adhesion into a single kernel to ensure correct adjustable Lennard-Jones behaviour.

The pressure force equation in this work used the Spiky kernel to produce a pure repulsive force. This was achieved by setting the mass and resting density condition to $\rho_i \geq \rho_{init}$. ρ_{init} is a predefined initial density constant.

The adhesion force calculation only needs to be activated when two particles are in within a predefined distance. The adhesion force should then be blended with the pressure force to provide a continued motion for the particle. As the two particles move towards each other, the effect of pressure force dominates. If the adhesion force is activated, when two particle moved apart, the adhesion force would

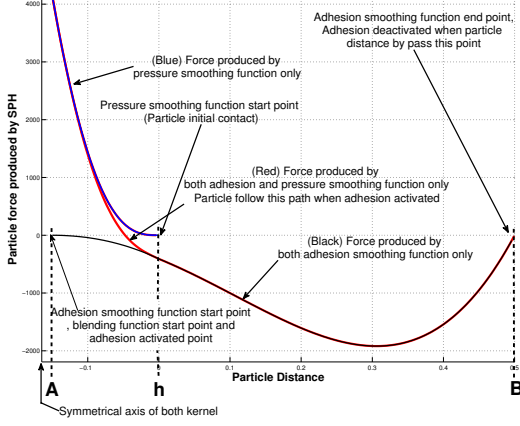


Figure 1: The combined pressure gradient (∇W_{spiky}) and adhesion kernel W_{adh} . Shows the smoothing function of the adhesion force, pressure force and the blending function of both forces. Note: the kernel axis symmetry as shown.

attract them towards each other. Fig. 1 demonstrates the pressure and adhesion smoothing function as well as the resulted blending force. The adhesion smoothing function is inverted to map the force direction for clearer presentation purposes. The loading and unloading of the particle interaction path are shown in Fig. 2 and the calculation of the adhesion force is given in eqn 11.

$$F_{Adh} = \sum \frac{K_{a(j)} + K_{a(i)}}{2} \frac{m_j}{\rho_j} W_{Adh}(\vec{r}_{ij}, h_a) \quad (11)$$

where:

$K_{a(i)}, K_{a(j)}$: Adhesion parameters for particle i and j
 W_{Adh} : Adhesion smoothing function

The calculation of adhesion smoothing function is shown in eqn 12 and design to be effect between support domain A and B from the particle, see Fig. 1.

$$W_{Adh}(\vec{r}_{ij}, h_a) = C_d \begin{cases} 0 & |\vec{r}_{ij}| < A \\ (\frac{\vec{r}_a}{Q})^\epsilon - (\frac{\vec{r}_a}{Q})^\zeta & A \leq |\vec{r}_{ij}| \leq B \\ 0 & |\vec{r}_{ij}| > B \end{cases} \quad (12)$$

where: $\epsilon > \zeta$, $Q = (B - A)$, $\vec{r}_a = (r_j - A)$ and,

A : Adhesion smoothing function starting point (activation distance).

B : Adhesion smoothing function end point.

ϵ, ζ : User adjustable adhesion behaviour. In this application $\epsilon = 3$ and $\zeta = 2$

If the adhesion is activated, the blending force occurs between the adhesion smoothing function starting point and the pressure smoothing function support domain (h). The blending of the pressure and adhesion force are shown in (13).

$$F_{LJ} = \begin{cases} F_{press} = 0 & |\vec{r}_{ij}| > h \\ F_{press} + F_{adh} & A \leq |\vec{r}_{ij}| \leq h \\ F_{adh} = 0 & |\vec{r}_{ij}| < A \end{cases} \quad (13)$$

3.2 Algorithmic Implementation

The SPH algorithmic structure used to render the dental the filling simulation was based on the Nvidia particle simulation structure[5]. This research extended the CUDA particle algorithm to include SPH and achieve loop times compatible with haptic rendering. We also implemented a less efficient algorithm where every inter particle distance is considered and used in the force calculations.

The structure of the GPU requires a different approach to programming. The concept of broad and narrow phase collision detection is well suited to computation on GPU hardware, where particles likely to collide are first identified in broad phase. During narrow phase processing properties relevant to SPH are calculated including mass density and pressure. The optimised SPH implementation used 6 steps. These were:

1. Broad phase collision detection. Based on the Nvidia grid method.
2. Reorganize particles and cell.

3. First pass narrow phase collision detection. Calculation of mass density.
4. Calculation of particle pressure field.
5. Second pass narrow phase collision detection. Calculation of adhesion, viscous and pressure forces.
6. Use sum of forces from previous simulation loop to compute particle acceleration and integration for new particle velocity and position.

The concept of broad and narrow phase collision detection reduces the read/write operation time and global memory space on GPU, maximized the computation efficiency in terms of particles number. The mass density computation must be done first as the values are required in the second pass calculations.

Different particles are needed to represent the tool, filling material and cavity. We employed the ghost particle methods[7] to implement cavity walls and the haptic tool. The cavity walls remained static during the simulation whereas the tool particles were updated based on the position of the haptic device.

Step 1. Broad phase collision detection, and

Step 2. Rearrange and organize particle and cell information

The broad phase collision detection was based on the particle simulation by Nvidia[5] using the Parallel Radix Sort algorithm (provided in the CUDA Thrust Library). This provided fast data access for the narrow phase collision detection.

Step 3. First pass narrow phase collision detection and calculation of mass density and

Step 4. Calculate particle pressure

Each particle can only interact with other particles in the 27 neighbouring cells (including the Home Cell). The narrow phase collision detection uses the sorted

Home Cell/Particle ID list to identify other particles within these neighbouring cells. It first calculated the mass density and then used it to calculate the pressure value.

3.3 Step 5. Second pass narrow phase collision detection and calculation of adhesion, viscous and pressure force

The second pass narrow phase collision detection is then performed to once again retrieve neighbouring particle information for the calculation of the particle forces. The second pass was required so as to compute adhesion. This includes activation and deactivation of adhesion status, based on the relative position between two particles.

Once a neighbouring particle has been identified, the distance to that particle is computed and if less than the smoothing length, the value for the viscous and pressure smoothing function can be calculated. If the distance between the two particles is closer than an adhesion threshold (A), the adhesion computation is activated.

Step 6. Integrate force to output new position

The force on the particles can now be calculated using (3). The forces acting on those particles representing the tool were summed up and sent to the haptic devices. The resultant forces of the filling material particles were then used in the Euler integration to update all their velocity and position. An updated position received from the haptic interface is used to update the position of all the tool particles via their fixed homogeneous transform. To ensure cavity boundary and fill material stiffness is modulated correctly the particles associated with the tool and boundaries should be evenly spaced.

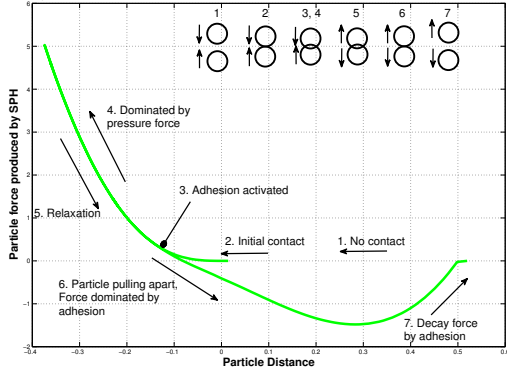


Figure 2: Interaction between two particles, from non-adhesive state to adhesion state separation

3.4 Explanation of SPH interaction force

Fig. 1 shows the forces produced by the adhesion and pressure smoothing function as well as the blending forces between when combined ³. Fig. 2 demonstrates 7 key stages of particle interaction. When the particles are not in contact (1), no adhesion or pressure force arises. During initial contact, only pressure force is produced (2). When the particle passes the predefined adhesion threshold, adhesion activates (3). As the particles get closer together, pressure force dominates (4). The tool particle is then relaxed (5) and blending force occurs. As two particles separates, adhesion force occurs (6) and got stronger before decay (7).

4 Implementation

Two GPUs were used in this research, the first GPU (a Nvidia GTX560) was responsible for graphical rendering output, the second GPU (a Nvidia GTX580) was solely responsible for filling material interaction, which included all the computation of the haptic rendering forces and all the inter particle force calcula-

tions. This computational architecture reduced the work load on the CPU to being primarily a method to manage the data flow between GPUs, the haptic hardware and long term memory storage.

The computer CPU was then managed in three threads. A haptic interface thread was responsible for communicating forces and positions with the devices (Novint Falcon ⁴ or Force Dimension Omega ⁵). A Haptic GPU thread managed the stability of the SPH calculation and the CUDA kernel calls. Although the SPH calculation was done entirely on the GPU (GTX580), the calling of the GPU function had to be done by CPU. This thread was also responsible for copying memory data between the GPU and CPU. The particle positions were then passed on to a graphic rendering thread for display via the second GPU (GTX560).

5 Result

In order to approximate the algorithm performance accurately the particle were setup so that each particle has to naturally collide with as many particles as possible. These results are shown in Fig. 3. The results show that on the test hardware, loop times of 400 to 600 hertz are possible.

5.1 Dental Filling Simulation

A desktop based dental filling simulation was developed, shown in Fig. 4. The tooth triangle mesh model was generated via high resolution CT scan. A total of 3000 particles were used, about 1400 representing the cavity boundary, 42 the tool, and 1600 the fill material. New material dropped into the cavity when user requested. The user compacted material layer by layer until the cavity was filled. The filling process is illustrated in Fig. 5. A video also shows the SPH algorithm in operation and a haptic device used to simulate the cavity filling process.

³The smoothing function are symmetrical, only half of the smoothing function are shown for presentation purposes.

⁴<http://www.novint.com/>

⁵<http://www.forcedimension.com/>

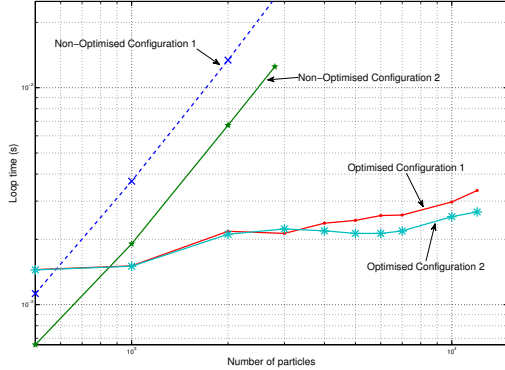


Figure 3: Algorithm Performance showing an efficient and inefficient implementation. Configuration 1 uses a GTX 560 for the haptic SPH, configuration 2 uses a GT580 for the haptic SPH.



Figure 4: Haptic dental filling simulation setup. Simulation ran on both the Novint Falcon and the Force Dimension Omega.

6 Discussion

Tooth restoration training using haptics has primarily focused on dental drilling and cavity preparation. There are a number of possible reasons for this, in-

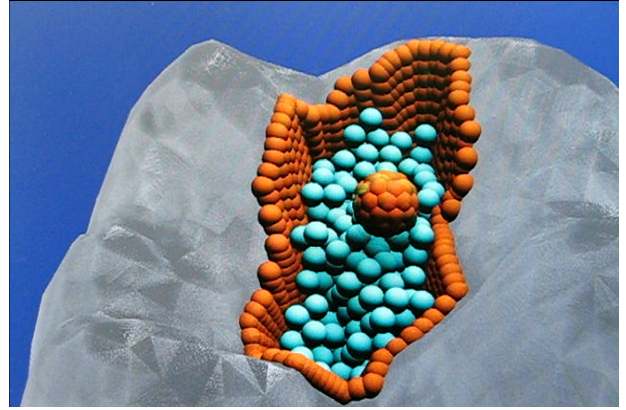


Figure 5: Haptic dental filling simulation - compacting material layer by layer. The tool is represented by the orange ball of particles and is controlled by the haptic device. The filling material is shown as blue particles, and the static cavity wall particles are shown in orange.

cluding: the algorithms for rigid body deformation are less computationally expensive than those, for example, for filling; arguably there is greater manual skill and procedural knowledge required in the preparation of the cavity thus potentially a greater benefit to training; and also, drilling is particularly associated with dentistry for non-dental professionals and simulation developers may simply unaware of the role played by different parts of the tooth restoration procedure. However the quality of the application and compaction of the material in the cavity remains a vital part of tooth restoration. As with any simulated procedure, dental students using a haptic filling trainer can repeat, practice and perfect their technique.

6.1 Algorithm Performance

The algorithm was implemented using GPU parallel toolkit. The Parallel Radix Sort requires the number of GPU threads launched to be the same as the number of grid cells. The numbers of threads launched for other operations in section (implementation), are proportional to the number of particles in the sys-

tem. This makes the algorithm to be adaptable for different GPU platform. The performance remained stabled at around 450Hz between 3000 to 7000 particles. The regression of the lines gives the exponent of the performance result to be about 0.25. Graphically rendering of the scene graph was running at 30fps, which is above the acceptable level.

6.2 Haptic Rendering and Material Representation

The haptic device interfacing thread and the GPU SPH calculation thread ran at 1.5kHz and 447Hz respectively. The SPH thread sent forces via communication class object to the haptic interface as soon these were available. The simulation testing shows that haptic interaction between the tool, boundary and the material were stable and the decay properties of the smoothing function aided this contact stability.

The material in this SPH simulation was weakly compressible. When two particles moved toward each other, the pressure force increased as predicted. However, if the particles continue and overlap completely, no further pressure increase will occur. As the motions carry on, the particle positions swap and will experience a sudden surge of pressure which can result in instability. This was overcome by the fast SPH calculation speed and high pressure gain value. The new algorithm presented here gives the ability to control adhesion and pressure force independently which allows greater flexibility in the design of the interaction. The blending function of the two forces allows particle motions to be continuous. The particle interaction path from contact, activation of adhesion to separation includes a hysteresis effect based on displacement to emphasise the adhesion properties of the material.

It is important to understand that SPH particles are interacting with each other via a force field, not the physical particle contact. In this algorithm, h was set to the particle radius, making the force field the same as particle volume. The h can be extended depending on the application. Consideration in graphical rendering is needed to ensure the surface of the particle volume is represented correctly. Therefore, the users do not interact the surface particle force

field without surface contact.

6.2.1 Representation of Filling Material, Cavity and Tool Using SPH

The associated video shows that SPH is suitable for representing dental filling material. The forces due to pressure, adhesion and viscous force can be simulated as a visco-plastic fluid that emulates the behaviour of the filling material and allows simulation of the haptic interaction with user.

Particles were used to represent the cavity and tool boundary vertices. The location of these vertices required careful consideration, even distribution was preferred. Boundary particles were intentionally overlapped slightly to prevent the filling particles leaking through. The particles of the cavity, tool and filling each had different properties in operation. In comparison to filling material particles, the pressure parameters associated with the cavity and tool particles were higher, but the adhesion was not present. Therefore, no adhesion occurs when the tool was in contact with the cavity. However, adhesion would still occur between tool/material or cavity/material interaction.

Thus far there has been no formal calibration of the parameters in the simulation to any practical dental material other than via the collective experience of dental tutors and engineers within the Haptel consortium. A further process is needed to determine a set parameters appropriate to teaching the task of cavity restoration with additional validation needed to ensure that learning via simulation transfers to practice. Likewise there has been no effort to present the haptic simulation in a visually appropriate way and methods such as those described by[12] should be used to present a better visual form.

7 Conclusion

This paper discusses haptic rendering in dental filling. This is an aspect of the tooth restoration training that had not been explored in the field of haptic. Smoothed particle hydrodynamics was selected as the modelling method and this work demonstrated that

smoothed particle hydrodynamics is an appropriate method for representing filling materials in haptic interaction. Physical properties such as pressure, viscosity and gravity forces were included in the model. A novel adhesion/cohesion model was developed to extend the traditional smoothed particle hydrodynamics framework to simulate the mechanical connection between material particles. The results show that the algorithm can simulate the properties of filling material and has the flexibility to model other soft matter. Parallel programming using multiple GPUs was employed as the computational platform in order to achieve real-time stable haptic rendering. In this case 3000 particles were rendered at about 447 Hz for haptic display.

Although focused on dental filling the techniques discussed in this paper should allow rendering of other material interactions. It may be possible to extend the method to include the hardening process of the material to be achieved simply through a parameter change. The use of GPU processing in haptic rendering is not wide spread, but this hardware is highly suited to real-time computations over large numbers of particles and polygons. This along with the tools, availability and cost make it an attractive solution, however there is an additional burden to adapt the algorithms to run efficiently on the GPU architecture.

Acknowledgements

The Authors are grateful to the ESRC and the ES-PRC for the funding for the HapTEL project. We are also pleased to acknowledge the support of Margaret Cox, Mark Woolford, Jonathan San Diego, and all the HapTEL researchers.

References

- [1] G. Cirio, M. Marchal, S. Hillaire, and A. Lecuyer. Six degrees-of-freedom haptic interaction with fluids. *Visualization and Computer Graphics, IEEE Transactions on*, 17(11):1714 – 1727, nov. 2011.
- [2] Mathieu Desbrun and Marie-paule Gascuel. Smoothed Particles: A new paradigm for animating highly deformable bodies. In *In Computer Animation and Simulation 96 (Proceedings of EG Workshop on Animation and Simulation)*, pages 61–76. Springer-Verlag, 1996.
- [3] R. A Gingold and J. J. Monaghan. Smoothed particle hydrodynamics - Theory and application to non-spherical stars. *Monthly Notices of the Royal Astronomical Society*, 181:375–389, 1977.
- [4] Simon Green. Particle simulations using cuda. Technical report, Nvidia.com, July 2012. http://docs.nvidia.com/cuda/samples/5_Simulations/particles/doc/particles.pdf.
- [5] Micky Kelager. Lagrangian fluid dynamics using smoothed particle hydrodynamics, 2006.
- [6] G. R. Liu and Liu M. B. *Smoothed Particle Hydrodynamics - a meshfree particle method*. World Scientific Publishing Co. Pte. Ltd., 2003.
- [7] L. B Lucy. A numerical approach to the testing of the fission hypothesis. *Astronomical Journal*, 82:1013–1024, 1977.
- [8] J.J. Monaghan. Notes [on] smoothed particle hydrodynamics IPAM. Technical report, IPAM Institute for Pure & Applied Mathematics, UCLA, March 2005.
- [9] Matthias Müller, David Charypar, and Markus Gross. Particle-based fluid simulation for interactive applications. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, SCA '03, pages 154–159, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.
- [10] Brian Tse, William Harwin, Alastair Barrow, Barry Quinn, Jonathan San Diego, and Margaret Cox. Design and development of a haptic dental training system: haptel. In *Proceedings of the 2010 international conference on Haptics - generating and perceiving tangible sensations*:

Part II, EuroHaptics'10, pages 101–108, Berlin, Heidelberg, 2010. Springer-Verlag.