

# GEOMETRIC MODELING OF SHEET DEFORMATION WITHIN A VIRTUAL ENVIRONMENT

B.S. Mahal<sup>1</sup>, D.E.R. Clark<sup>2</sup> and J.E.L. Simmons<sup>1</sup>  
<sup>1</sup>Department of Mechanical & Chemical Engineering  
<sup>2</sup>Department of Mathematics  
Heriot-Watt University, Edinburgh Scotland, UK

## ABSTRACT

In this paper we present a real-time, computationally inexpensive environment for on-screen, accurate simulations of thermoplastic sheet materials on a conventional Personal Computer. A pseudo-immersive virtual environment is constructed to handle the implementation of a real-time aesthetically accurate deformation algorithm called MaSSE: Mass-Spring Simulation Engine. The motion of the sheet is controlled by gravity and by using a mouse-pointing device to manipulate the sheet within the environment. An application for the environment is centered on, although not restricted to, mechanical engineering-based real-time simulations of heat-sensitive sheet materials.

## 1. Introduction

The aim of the work is to build a computer-based geometric model utilizing Object Oriented technology which simulates the complex mechanical behaviour found within thermoplastic sheet materials over an appropriate temperature range. Attempts at modeling heat within Virtual Environments have remained within the realm of haptic interfaces [Dio97a], animation of turbulent hot gases[Fos97], [Dio97b], and simulations of fire in virtual buildings [Buk97]. These immersive simulations have tended to concentrate upon the representation and behaviour of fire within a confined space rather than the effects of heat and fire upon objects within the environment.

Object oriented development allows the system to be constructed in a modular structure with the environment being based on the classes Mass and Spring. The ultimate aim is that once the PC-based system has been developed it can be transferred to a full VR system.

The system utilizes the OpenGL environment, along with the GLUT toolkit to produce an open system that is portable across a diversity of platforms and which also takes advantage of the latest computer graphics based libraries and toolkits.

The objective is to produce an aesthetically accurate, real-time simulation of thermo-sensitive sheet behaviour in an environment that can be furnished with objects and set at various temperatures. Our approach differs from conventional finite-element-based systems [Eis96] that have the potential to produce highly accurate results but require high levels of computation that are awkward to implement in real time. Mass-spring-based systems have already been shown to work effectively within graphics-based animation and simulation [Ter96], portraying fast and aesthetically accurate behaviour of cloth [Hut96], [Lou95], [Vol96] and other sheet materials. The system comprises of a number of adaptable sheet models that are inserted into a virtual environment containing several environmental parameters. These include adjustable levels of gravity, three dimensional objects, and simulated heat. Sheet parameters include adaptable levels of elasticity and plasticity, combined with variable levels of moment stiffness.

Polyhedral objects are 'inserted' into the pseudo-immersive environment to demonstrate the effects of collisions with three-dimensional bodies. These objects also highlight the behaviour of the sheet as changes in sheet plasticity/elasticity take hold under changing temperature scenarios. A secondary form of interaction, using a mouse (initiated by the user) to manipulate the sheet in 3D space, is currently under development. This allows the simulation of wrapping, although some mouse functionality has already been incorporated to allow the user to "fly around" the setting and view the deformation process from any viewing perspective. The ability to fly around the sheet model allows the user to manipulate the sheet from any three-dimensional perspective, eliminating the need for a specialized environment [Fan95a].

The choice of a World on Window (WoW) non-immersive-based approach was made because of three perceived advantages. First, it allows the system to remain within the parameters of the project hypothesis (i.e. the production of an inexpensive, generic, real-time, accurate simulation) by avoiding dedicated, expensive hardware. Second, no system resources are diverted away from the application's numerical computation processing, to handle the immersion technology. Finally, at the start of the project it was deemed highly probable that immersion technology would leap forward over the life of the study. Therefore, by retaining a generic pseudo-immersive interface, the system could be adapted for the latest technology once its objectives have been achieved.

## 2. Mass-Spring Systems

Each *mass* instance (or sheet control point), when acting as an individual particle in free space obeys Newton's second law:

$$ma = f$$

where the mass ( $m$ ) of the particle multiplied by the particle's acceleration ( $a$ ) is equal to the sum of the forces ( $f$ ) acting upon it. Based upon this expression, a position and velocity matrix (or vector) is used to determine acceleration and force.

In conventional free-space systems that may contain  $n$  independent particles, the force calculation per control point is of order  $O(n)$ , which dramatically rises to the order of  $O(n^2)$  when considering interactions with other particles. Taking into account the effects of those particles close to a mass dramatically reduces this figure. To form a relationship between neighbouring sheet control points, *spring* objects are used to link masses, thus forming a coherent model structure.

Each *mass* instance (or sheet control point) has several properties that describe its current behaviour within free space at any instant. These parameters (which are mapped using three-dimensional vectors), include:

- The current location of the mass within the environment.
- The forces being applied to the mass.
- The current velocity of the mass, and
- The current acceleration of the mass.

A value for the actual mass of the mass instance/object is also maintained, together with positional data detailing the mass within the sheet structure. The mass instances are interconnected through a network of spring systems that also have properties that are used to describe the behaviour of the spring during the simulation.

Spring data includes information about:

- The position of the spring within the sheet model (using pointers to its interconnected masses)
- The axial stiffness of the spring, and
- The undisplaced length of the spring.

Using this data, the system models the behaviour of the object (in this case a sheet) by determining whether the manipulation of any one particular mass will have an effect upon its surrounding masses. For a breakdown of calculations, consider two adjacent masses connected by a spring (see Figure 1), with  $\mathbf{f}$  denoting the force acting upon particle  $p$  from particle  $q$ ,  $s$  representing the spring's undisplaced or resting length. When stretched, the spring will be subject to Hooke's law (see Equation 1) and will generate a force that will act in the direction  $\mathbf{d} = \mathbf{p} - \mathbf{q}$ , along the line between the points. The symbol  $k_s$  represents the spring constant (or value for stiffness) and indicates that as the particles are moved further apart, the greater the force will be applied to attempt to bring the particles back together again.

$$\mathbf{f} = -k_s (|\mathbf{d}| - s) \frac{\mathbf{d}}{|\mathbf{d}|} \quad [1]$$

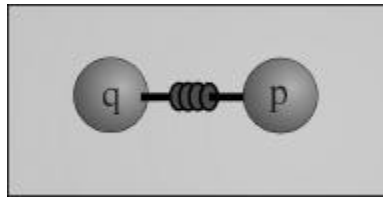


Figure 1: Two Masses interconnected via a Spring

If implemented in this simple form, the spring and mass system will start to oscillate, and, if left undisturbed, will continue forever. To cater for this eventuality, a level of damping must be retrofitted into Hooke's original equation (which has no inbuilt friction coefficient). This results in the equation:

$$\mathbf{f} = - \left( k_s (|\mathbf{d}| - s) + k_d \frac{\dot{\mathbf{d}} \cdot \mathbf{d}}{|\mathbf{d}|} \right) \frac{\mathbf{d}}{|\mathbf{d}|}$$

where  $k_d$  is the damping constant, and

$$\dot{\mathbf{d}} = \dot{\mathbf{p}} - \dot{\mathbf{q}} = \mathbf{v}_p - \mathbf{v}_q$$

where  $\mathbf{v}_p$  and  $\mathbf{v}_q$  represent the velocity of  $p$  and  $q$  respectively. This has the effect of generating a proportional force that acts in the same direction as the spring force.

Likewise, when dealing with elastic materials, if two masses are forced together so that the distance between them is less than the undisplaced length of the spring between them, a repulsive force is generated based upon:

$$f = -k_r \frac{d}{|d|^3}$$

### 3. Model Dynamics

The Mass-Spring Simulation Engine (MaSSE) was designed and created using a modular structure in order that the application may in future be extended to cater for a diversity of other shape simulations such as solid object deformation modeling. Figure 2 illustrates the flowchart.

Additional complexity required for modeling sheet behaviour such as bending moments is catered for in the structures employed to build the sheet models. The use of a third (depth) dimension allows a greater level of flexibility, since the inter-layer cross-springs can easily be modified to cater for different levels of modular stiffness. This is a break-away from the traditional mass-spring approach, which has tended to employ 2D structures with complex rotational stiffness calculations to simulate sheet bending [Bre97], [Gud97], [Ng95], [Pro97]. Instead, the present approach has, in essence, opted to use a simpler piece of code to calculate sheet behaviour. This is reused many times, rather than invoke a complex algorithm that is called but a few times per frame. This way, through careful optimization of the system, it is anticipated that significant improvements in sheet simulation may be achieved.

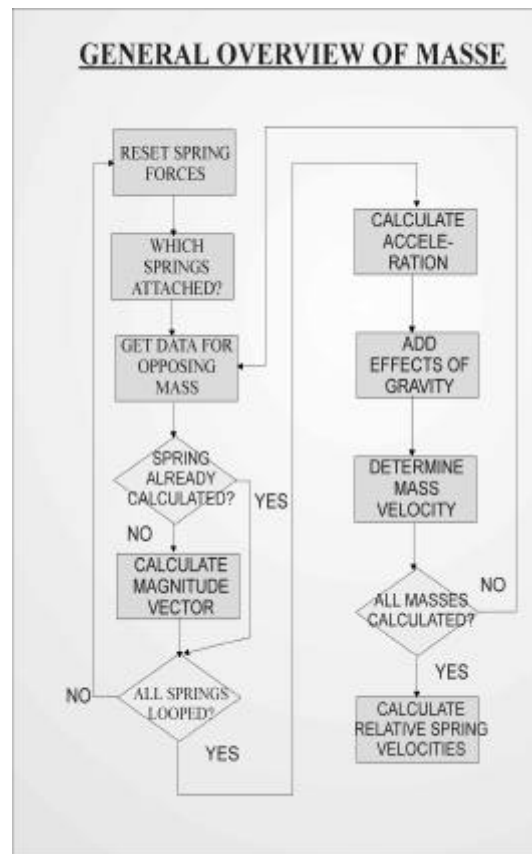


Figure 2: Dynamics Flowchart

## 4. Damping Model

At present, a simple generalised single degree of freedom model is used to represent frictional forces acting upon the sheet within the environment (see figure 3). However, to produce a more realistic simulation of viscoelastic sheet behaviour, a more complex model (see figure 14) would require incorporation into the system (See Section 10: Discussion).

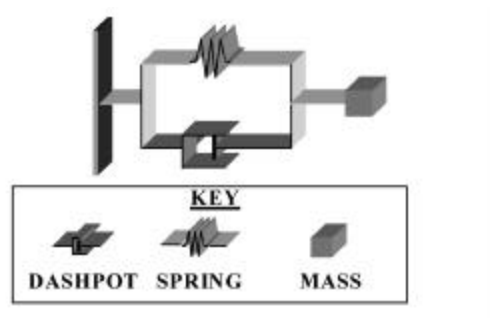


Figure 3: Simple Damping Model

## 5. Sheet Structures

Figure 4 represents a typical simple mass-spring implementation consisting of masses and springs in a two-dimensional layer [Gud97], [Pro97]. Like most elementary models its basic structure gave the mesh a fast, efficient form producing acceptable results (demonstrating spring mass-spring interaction and elasticity) within a short run-time. However, since its two-dimensional nature (i.e. its lack of depth perception) resulting in a failure to account for bending moment it was deemed an impractical choice for further investigation. In total, three sheet structures have been devised and tested, and these are examined below.

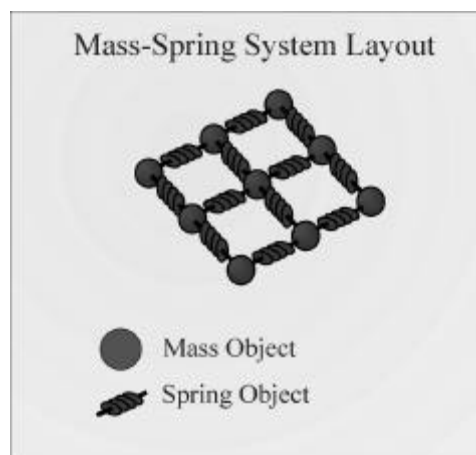
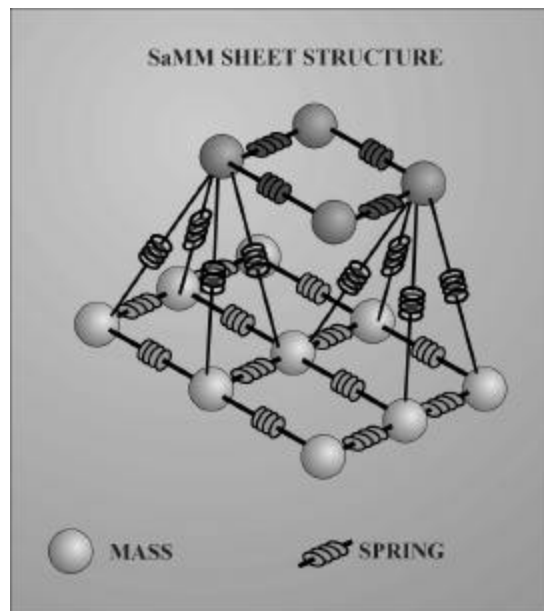


Figure 4: 2D Mass-Spring System

The square model structure, known as SaMM (see figure 5) is based upon the standard square-matrix sheet structure (see figure 4) with the addition of a third (depth) dimension, provided through the addition of a second (or multiple) layers of square matrices. Each mass placed in higher layers of the sheet is positioned above the centre of the square formed by the masses in the layer below. Each mass in the upper layer is then interconnected to the layer below through four springs, which link it together with the four masses that form the square in the lower level. The advantage of this approach is that it caters for bending moment in the sheet (through the use of the cross-layer springs) while maintaining a high degree of efficiency.



*Figure 5: Square (SaMM) Sheet Structure*

The square-lattice model structure known as SLaMM (see figure 6) is similar to the SaMM layout (see figure 5) but with the addition of a diagonal cross-member spring that interconnects the near-side left mass with its relative far-side right mass. The role of this cross-member (or spring brace) is to prevent the square subsystems of the layer from forming a flexible mechanism, instead creating a rigid structure. Visually the result of this additional spring is that it prevents the square subsystems bending diagonally to form a rhombus. Triangulation of the mass-spring system has proved to be an important advance in the design of the sheet structures resulting in experimentation with a more radical sheet.

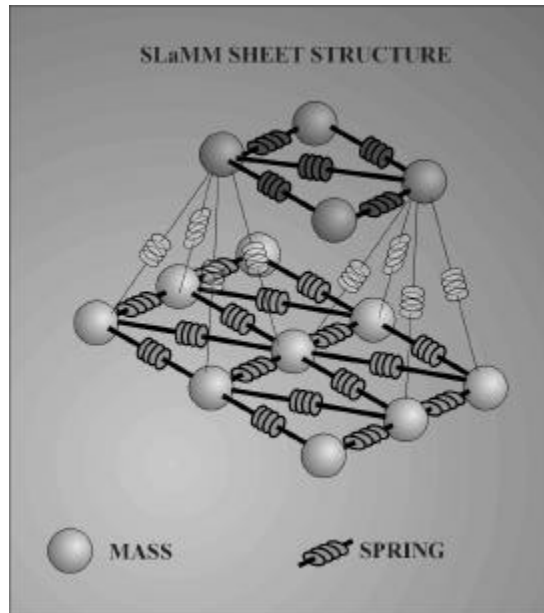


Figure 6: Square-Lattice (SLaMM) Structure

The triangular model, known as TRiMM (see figure 7), was brought about by experimentation with the SLaMM model and the effectiveness of triangular structures within the sheet mesh. TRiMM represents a radical change from earlier structures through its use of a triangular base for the sheet layer. Successive layers are then positioned in an over-lapping manner (similar to the SaMM and SLaMM structures). Again, each mass in the upper layer is connected (through a series of springs) to the lower layer via the three masses that form the triangle directly below which the mass is positioned.

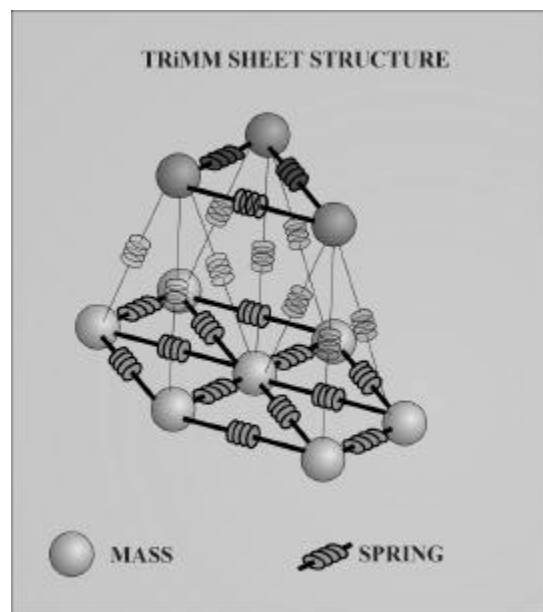


Figure 7: Triangular (TRiMM) Sheet Structure

## 6. Collision Detection

Due to the well-documented nature of collision detection algorithms, which result in an explosive growth of collision tests as the number of objects on the screen increases [Vol94], [Pro97], [Hub96], [Coh95], [Fan95b], two approaches were adopted in order to provide what is designed to be the fastest and most efficient solution for the given environment.

The first level of detection utilizes game rules to infer that specific sprites (objects on the screen that may be animating or moving) need not be collision tested against other sprites. These rules simply state that static sprites are not collision tested against other static sprites.

As with many collision detection routines, the focus of interest is positioned on the sprites within the screen space. The Sector method works by dividing the screen (or simulation view) into a grid of sectors. The algorithm then determines which sector each sprite is located within according to its screen position. A standard collision test [Fan95a] is then made against any sprites that are within the same sector. Inefficiencies occur when a sprite lies on the boundary of two or more sectors resulting in collision tests having to be performed between the sprite in question and every other sprite positioned within any of the bordering sectors.

## 7. Viewing Environment

The viewing environment is controlled by the use of a conventional 2D PC-based 2-button mouse. Rotation (or flight) around the environment is initiated by the user through depression of the left mouse button, combined with simultaneous movement of the mouse in the direction that the user would like the new camera perspective. For example, to move from the viewing perspective in figure 8 to that of figure 9, the user would press and hold down the right mouse button, then simultaneously move the mouse in an upward direction.

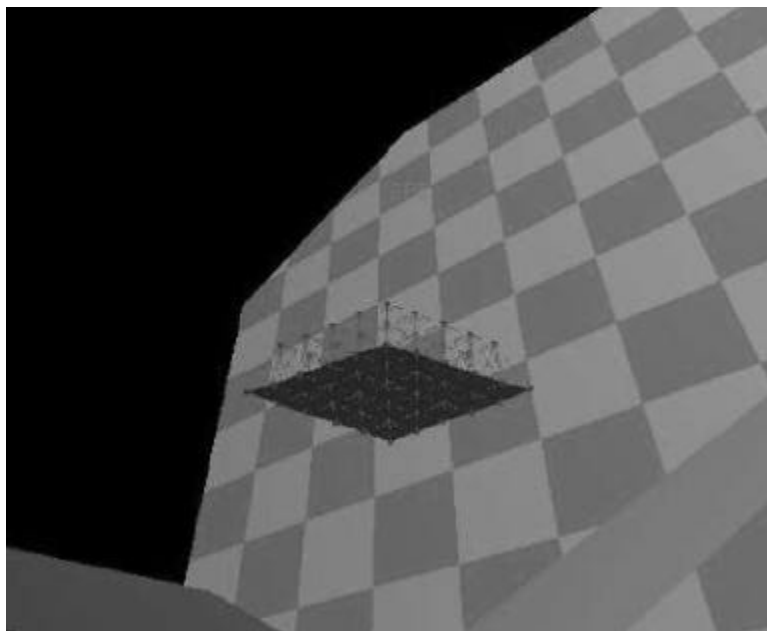
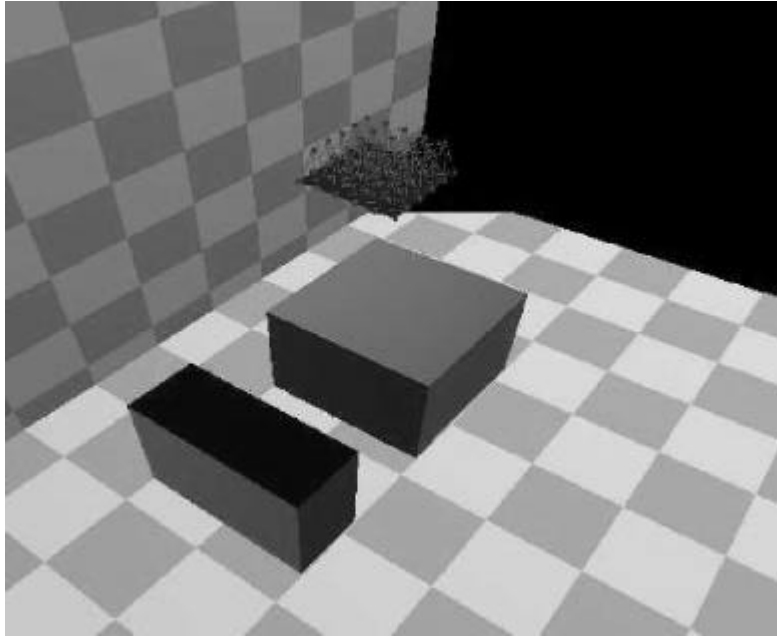


Figure 8: Looking up through a filled SLaMM mass-spring sheet



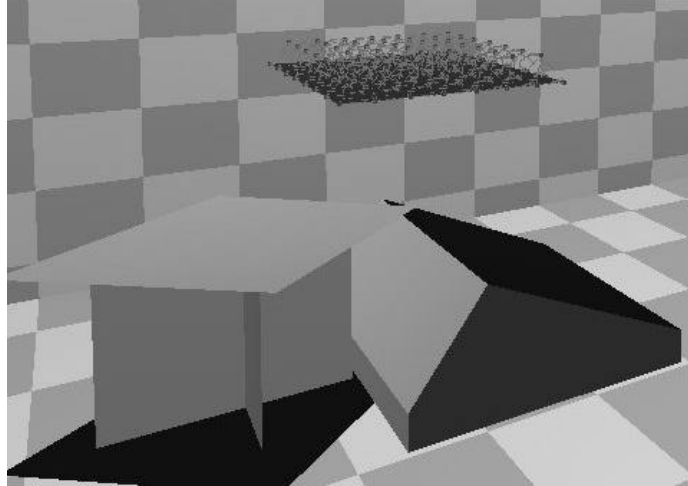


*Figure 9 – Viewing a SLaMM sheet fall from an elevated left-side position.*

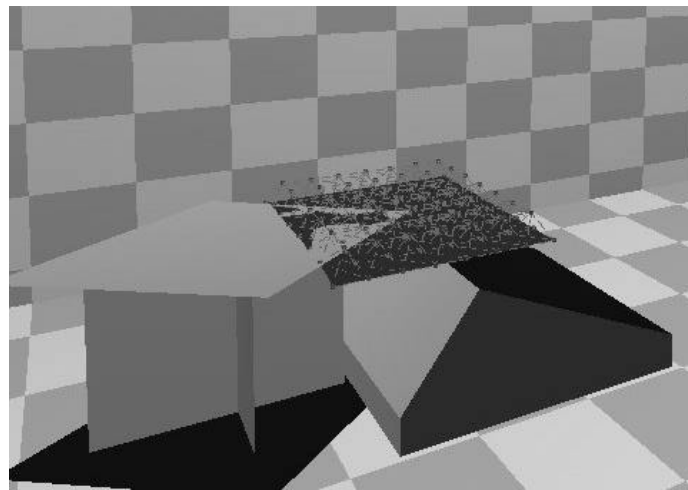
The diagrams illustrate on-going work, with a sheet manipulation algorithm currently being deployed to manipulate individual sheet elements when selected by the user. Using the viewpoint rotation and screen panning facility to isolate the model point to be controlled the user initiates the environmental manipulation tool. Due to the delicate nature of real-time simulation of complex systems such as this, it is not possible for the system simply to move the exact point of the sheet the user has selected at that particular timestep to the desired point of destination within the environment. Instead, at the time of selection a dynamic spring is created that inter-links the nearest mass to the point of selection to the actual selection location. This, it is believed, will prevent any numerical instability from being generated within the model during the execution of the simulation.

## **8. Simulation Demonstration**

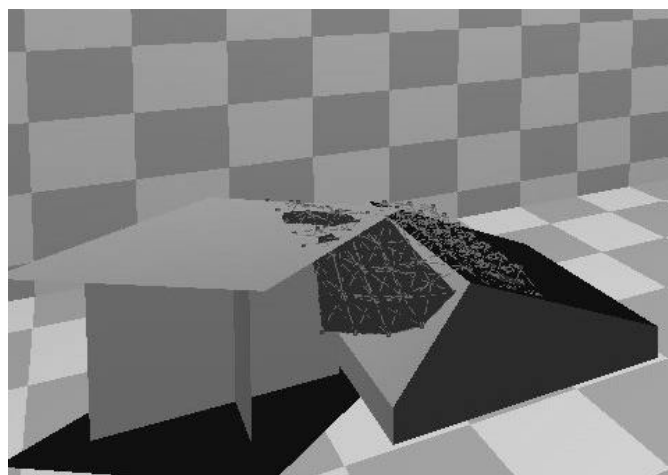
Figures 10 to 13 demonstrate the actions of a typical sheet simulation. When first initiated, the sheet is suspended above a set of polyhedra (see figure 10). Once gravity takes effect the sheet starts to fall over the polyhedra until a collision occurs. Careful analysis also allows the user to determine which masses were the subject of the collision as their colour is changed from red to green (see figure 11). As the sheet continues to fall, it experiences the first signs of deformation as the collided masses remain static whilst the remaining masses and springs continue to fall under gravity. At this point in the simulation the far side of the sheet makes contact with the second polyhedron (see figure 12). Figure 13 illustrates how the sheet continues to fall and further deform before coming to rest lying upon the polyhedra with masses suspended by their springs between the two polyhedra.



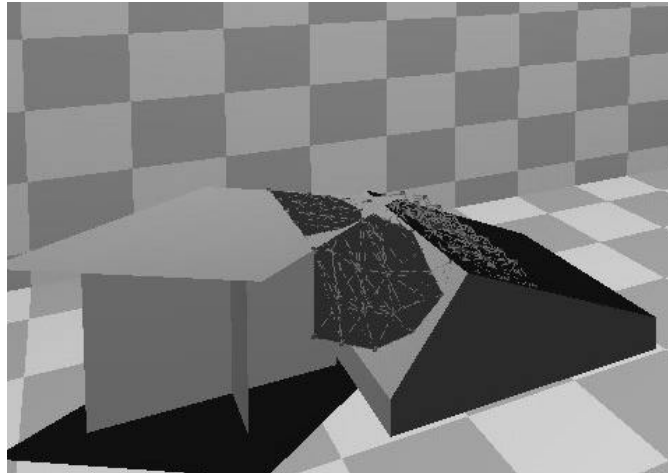
*Figure 10 – Filled sheet falls under effects of gravity*



*Figure 11 – Collision with the horizontal Polyhedron face*



*Figure 12 – Collision with the second Polyhedron*



*Figure 13 – Sheet settles over the first and second polyhedra*

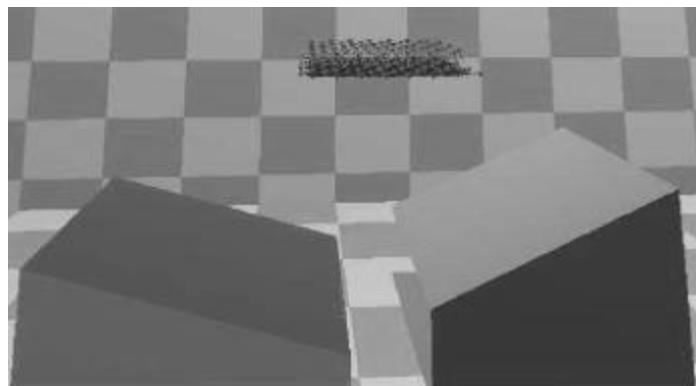
## 9. Experimental Results

The simulation presented in this paper was conducted on an Intel Pentium II 400 MHz portable computer, running Windows 98 (SR2) equipped with 64 MB RAM and a 2.5 MB VRAM Neomagic MagicMedic 256AV-display unit.

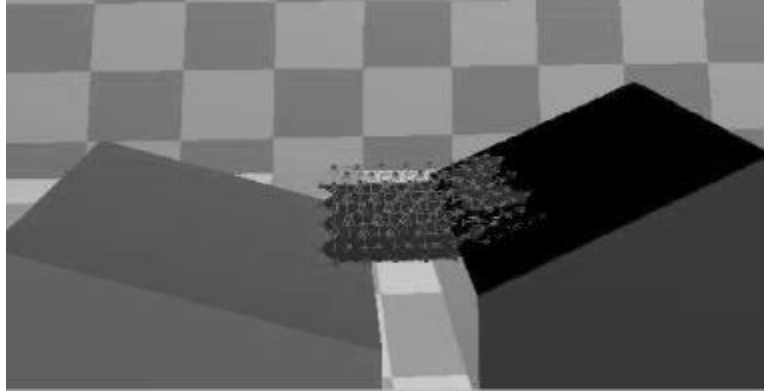
Experimental work has concentrated on modeling the performance of the system as a TriMM sheet falls under gravity from a predetermined distance above a single (or group) of polyhedral objects. The time is then measured from the point when the simulation is initiated to:

- When the sheet collides with the first Polyhedron. (See figure 14).
- When collision occurs with the second Polyhedron. (See figure 15).
- When the Sheet falls/stretches between the 2 polyhedra and contacts the floor. (See figure 16).
- When the simulation comes to rest as the damping coefficients take effect. (See figure 17).

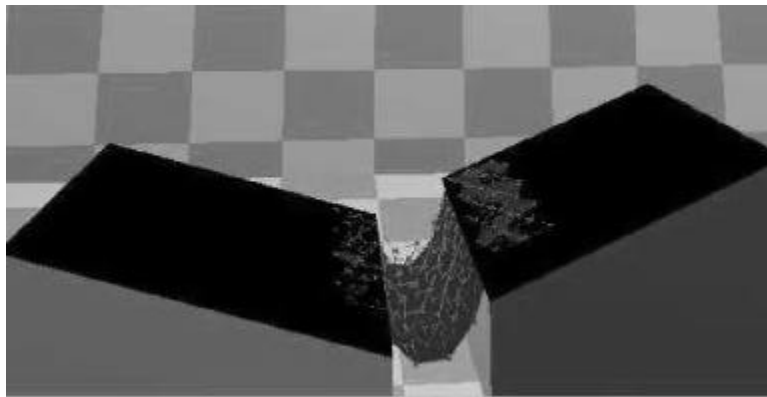
To aid in the identification of these stages, the colour of the polyhedron face in collision with a mass is changed to black highlighting that a collision has occurred.



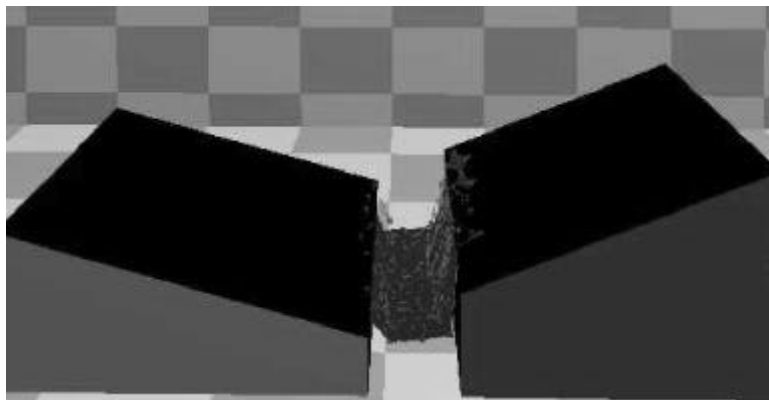
*Figure 14 – Filled 8x2x8 TRiMM sheet falls under effects of gravity*



*Figure 15 – Collision with first Polyhedron face*



*Figure 16 – Collision with the second Polyhedron*



*Figure 17 – Sheet begins to settle on floor as it slides down between the sides of the two polyhedra*

Parameters for the simulation are detailed below in Table 1, with the time results displayed in Figure 18. Table 2 details the dimensions of the sheets that were tested, with all sheets occupying the same physical area within the room environment. As the number of masses within the sheet are increased, the distance between the masses (together with the length of the interconnecting springs) are gradually reduced, thus resulting in a more detailed model.

<b>Table 1 – Simulation Testing Variables</b>	
<b>VARIABLE</b>	<b>VALUE</b>
Spring stiffness (planar springs)	9.2 grams/mm
Spring stiffness (inter-layer springs)	6.2 grams/mm
Mass value for each Mass	1.0 grams
Sheet Size – X dimension (min-max)	-0.2 – 0.8
Sheet Size – Y dimension (min-max)	0.3 – 0.4
Sheet Size – Z dimension (min-max)	-0.2 – 0.8
Room Size – X dimension (min-max)	-6.5 – 6.5
Room Size – Y dimension (min-max)	-2.81 (floor) – 3.0
Room Size – Z dimension (min-max)	-2.99 (back wall) – 3.33
Air damping coefficient	0.97 (3%)
Sheet damping coefficient	6.0%
Timestep	0.03 seconds

<i>TABLE 2 – SIMULATION TESTING SHEET DIMENSIONS</i>		
<b>Sheet Dimensions (X x Y x Z)</b>	<b>No of Masses</b>	<b>No of Springs</b>
3 x 2 x 3	13	33
6 x 2 x 6	61	216
8 x 2 x 8	113	428
9 x 2 x 9	145	561
12 x 2 x 12	265	1068

The graphs in Figure 18 highlight the significant changes in processing time due to minor changes in the proportions of the sheet complexity. Table 2 does in fact justify this performance overhead when one considers the substantial increases in the number of Masses and springs in each of the models as their sizes are increased. Testing showed that sheets of size 6 by 2 by 6 masses (6x2x6) or 8 by 2 by 8 masses (8x2x8) appear to give the most realistic behaviour whilst retaining a realistic sense of real-time processing.

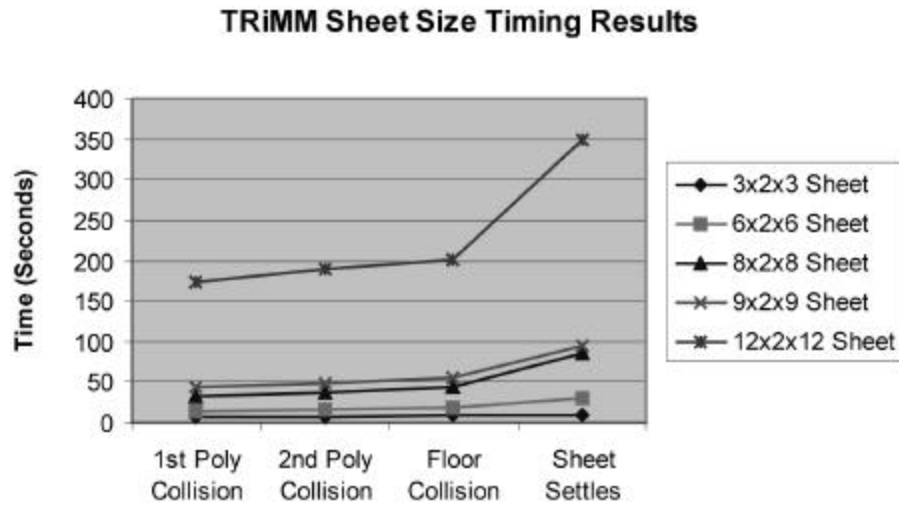


Figure 18 – Simulation Timings for 5 Varying Sheet Sizes

## 10. Discussion

Preliminary results of the prototype described in this paper have so far proven to be positive with all indications showing that the underlying hypothesis is achievable.

After further optimisation and testing of the sheet dynamics, the next stage is the installation of a virtual heat source. This source will then be used to simulate the effects of modeling plastic/elastic deformation properties of a thermally sensitive sheet. To achieve this aim a more advanced damping model is required. This new model (see figure 19) is an extension of the Voigt-Kelvin element spring and dashpot model which has been combined with a Maxwell element system. It is anticipated that this new damping structure will prove beneficial once the heat simulation components of the system are included to allow the sheet to reproduce viscoelastic behaviour.

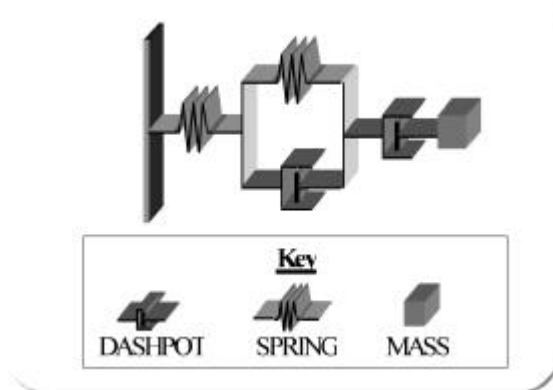


Figure 19: Advanced Damping Model

## 11. Acknowledgement

The Authors acknowledge the generous assistance of Dr Grant Fraser and Mr Ross Brown in the advancement of this work.

## REFERENCES

- [Bre97] Breen, D., et al. Can We Get There from Here? : Current Challenges in Cloth Design, Modeling and Animation. in Computer Graphics Annual Conference. 1997.
- [Buk97] Bukowski, R. and Sequin, C. Interactive Simulation of Fire in Virtual Building Environments. in SIGGRAPH 97. 1997.
- [Coh95] Cohen, J., et al. I-COLLIDE: An Interactive and Exact Collision Detection System for Large-Scale Environments. in ACM Interactive 3D Graphics Conference. 1995.
- [Dio97a] Dionisio, J., et al., The Virtual Touch: Haptic Interfaces in Virtual Environments. Computers and Graphics, 1997. 21(4): p. 459-468.
- [Dio97b] Dionisio, J., Virtual Hell: A Trip Through the Flames. IEEE Computer Graphics and Applications, 1997(May-June): p. 11-14.
- [Eis96] Eischen, J., Deng, S. and Clapp, T., Finite-Element Modelling and Control of Flexible Fabric Parts. IEEE Computer Graphics and Applications, 1996(September): p. 71-80.
- [Fan95a] Fang, J.J., D.E.R. Clark, and J.E.L. Simmons, A Pseudo-Immersive Virtual Environment - A Shell for Research. IJVR, 1995.
- [Fan95b] Fang, J.J., Clark, D.E.R. and Simmons, J.E.L., Collision Detection Methodologies for Rigid Body Assembly in a Virtual Environment. Virtual Reality, 1995. 1(1): p. 41-48.
- [Fos97] Foster, N. and Metaxas, D., Modelling the Motion of a Hot, Turbulent Gas. in Computer Graphics Annual Conference. 1997. Los Angeles, California, USA.
- [Gud97] Gudukbay, U., Ozguc, B. and Tokad, Y., A Spring Force Formulation for Elastically Deformable Models. Computers & Graphics, 1997. 21(3): p. 335-346.
- [Hub96] Hubbard, P.M., Approximating Polyhedra with Spheres for Time-Critical Collision Detection. ACM Transactions on Graphics, 1996. 15(3): p. 179-210.
- [Hut96] Hutchinson, D., Preston, M. and Hewitt, T., Adaptive Refinement for Mass/Spring Simulations. in Seventh Eurographics Workshop on Animation and Simulation. 1996.
- [Lou95] Louchet, J., Provot, X. and Crouchemore, D., Evolutionary identification of cloth animation models. Computer Animation and Simulation, Eurographics, 1995: p. 44-54.

- [Ng95] Ng, H.N., Grimsdale, R.L., and Allen, W.G., A System for Modelling and Visualization of Cloth Material. *Computers and Graphics*, 1995. 19(3): p.423-430.
- [Pro97] Provot, X., Collision and Self-Collision Handling in Cloth Model Dedicated to Design Garments. in *Graphics Interface '97*. 1997.
- [Ter96] Terasawa, M., Physically-Based Computer Animation Controlled by Motion Features. *International Journal of Shape Modelling*, 1996. 2(4): p. 247-256.
- [Vol94] Volino, P. and Thalmann, N., Efficient Self-Collision Detection on Smoothly Discretized Surface Animations using Geometrical Shape. *Eurographics*, 1994. 13(3): p. 155-166.
- [Vol96] Volino, P., et al., An Evolving System for Simulating Clothes on Virtual Actors. *IEEE Computer Graphics and Applications*, 1996(September): p. 42-51.

## BIOGRAPHIES



**Bhopinder S. Mahal** graduated from Staffordshire University with a HND in Computer Science in 1992 and a B.Sc in Information Systems in 1994. He received a Masters degree in Human Computer Interaction from Heriot-Watt University in 1995 and has worked in IT systems at Gardiner-Caldwell Communications, the Forshungszentrum Informatik (Computer Science Research Centre) at the University of Karlsruhe, Germany, as well as the University of Edinburgh. Currently a Doctorate Researcher at Heriot-Watt University in Edinburgh he is completing his Ph.D. entitled Bending and Deformation of Heat Sensitive Material within a Virtual Environment.

*Contact information:*

Department of Mechanical & Chemical  
Engineering  
Heriot-Watt University  
Edinburgh EH14 4AS  
Scotland, UK  
Phone: +131 449 5111 x4353  
Fax: +131 451 3129  
Email link: <mailto:bsm@cee.hw.ac.uk>





**Douglas E.R. Clark** Received B.Sc. in Mathematics from London University in 1967 and a Ph.D. from University of Durham in 1970. A 1970 recipient of a Royal Society European Fellowship to continue research in the Geometry of Manifolds at the Mathematics Institute in the University of Amsterdam, Netherlands. Lecturer in Mathematics at Heriot-Watt University since 1971. Research interests include Differential Geometry, Low Dimensional Topology and Computational Geometry.

*Contact information:*

Department of Mathematics  
Heriot-Watt University  
Edinburgh EH14 4AS  
Scotland, UK



**John E.L. Simmons** is Vice-Principal and Professor of Mechanical Engineering of Heriot-Watt University, Edinburgh. His research interests are in the areas of Manufacturing Technology and Management Systems. Prior to joining the academic sector Professor Simmons held Production and Design Management positions in UK manufacturing companies.

*Contact information:*

Professor John Simmons, Vice Principal  
Heriot-Watt University  
Edinburgh EH14 4AS  
Scotland, UK  
Tel: 0131-451 3001  
Fax: +131 451 3129  
Email link: [mailto: j.simmons@hw.ac.uk](mailto:j.simmons@hw.ac.uk)