Precise Deformation of Rheology MSD Model Calibrated by Randomized Algorithm

Hiroshi Noborio, Ryo Nogami and Ryo Enoki

Division of Information and Computer Science
Graduate School of Engineering
Osaka Electro-Communication University
Hatsu-Cho 18-8, Neyagawa, Osaka 572-8530, Japan

Abstract

In this paper, we propose and compare three kinds of mass-spring-damper (MSD) models of rheology object, and experimentally select the best model concerning to shape deformation and volume accuracy. The MSD model requires a few costs to calculate force propagation and shape deformation of rheology object. For this reason, the dynamic animation can be made by a personal computer within the video-frame rate (about twenty milli-seconds). Moreover, in order to maintain deformation precision, we calibrate all coefficients of dampers and springs under many experimental data by the randomized algorithm. Then in the set of simple pushing experimental operations, shape deformation and volume of virtual rheology object based on the best model is similar to these of real rheology object. This is a case study to generate dynamic animation efficiently and precisely by the MSD model.

1. Introduction

Real-time simulation of deformable object is a younger field. Dynamic animation is indispensable in robotics and virtual reality, which has been aggressively used in tele-operation, humanoid, assembly and task planning, computer animation, game and amusement and so on. The key trade-off occurs between calculation time and deformation/propagation accuracy. In general, shape deformation should be calculated in 33 milli seconds for the video-frame rate, and furthermore force propagation should be calculated in a few milli seconds for the haptic rendering. If this trade-off is broken, the animation becomes off-line, otherwise, it becomes on-line. The performance of modern computer and graphics hardware has made physical-based animation possible in real time. But even with today's best hardware and most sophisticated technique [1],[2],[3], only a few hundred elements with small deformations have been simulated in real-time. To represent models of elastic and visco-elastic objects, we have used one of four approaches, i.e., the mass-spring-damper (MSD) method [4],[5],[6],[7], the finite difference method (FDM) [8], the boundary element method (BEM) [9],[10], and the finite element method (FEM) [11],[12],[13],[14],[15]. The computation efficiency decreases in this order, and the deformation/propagation accuracy increases in this order. To solve such a trade-off between computation efficiency and shape accuracy, almost all the researchers focus on FEM to save computational complexity.

As the opposite major flow, we focus on MSD while maintaining deformation precision in this paper. First of all, we consider a rheology object and its deformation. Since a rheology object always leaves a residual displacement, its model should keep the displacement by many kinds of pushing operations. Although elastic and visco-elastic objects have been aggressively modeled, but a rheology object is seldom modeled [16],[17]. Also, calibrating many kinds of models from a lot of experimental data is not still established [6],[7].

On the observation, we propose an efficient MSD method for representing flexible deformations precisely. In order to watch shape deformation of rheology object within the video-frame rate (about 20 ms), we adopt the MSD. It is more efficient than the other models. They are typically done off-line, that is, computers spend a few seconds, minutes or hours to arrive at a single answer. The main defective point of MSD for the practical purpose is the lack of deformation accuracy. To overcome this problem, we calibrate three coefficients of each element from many experimental data, which

© The Eurographics Association 2003.

consists of one spring and two dampers. As the calibration approach, we adopt the randomized algorithm to investigate a set of good coefficients from deformations by many pushing operations [18],[19].

In this paper, section 2 describes a basic voxel/lattice structure which consists of three elements, i.e., two dampers and one spring. In addition, we explain how to calculate shape deformation and force propagation in the structure. Three elements are dependently used in the dynamic equation. It is represented as the quadratic differential equation. This can be approximately calculated by the fourth-ordered Runge-Kutta method. Moreover, we propose two extended structures composed of the same elements. Section 3 explains how to calibrate three coefficients of each element. In section 4, we compare three virtual rheology models with the real rheology object pushed by simple operations. First of all, we explain how to evaluate each difference between real and virtual rheology objects. Then, by an efficient randomized algorithm based on the difference, we calibrate three element coefficients so as to construct a virtual rheology object flexibly. Finally in section 5, we will give a few conclusions and future works.

2. Three Kinds of Mass-Spring-Damper Models

In this section, we propose three kinds of mass-spring-damper (MSD) models. First of all, we introduce our element with two dampers and one spring. Then, we explain three kinds of MSD models. The model 1 forms a basic voxel/lattice structure with three lengths of elements. In this model, we explain how to calculate propagation of internal forces in each rheology object. In order to maintain the volume constant condition in the model 2, we exchange four longest elements with eight half-length elements in each voxel in the model 1. In order to deform a few parts whose differences of virtual and real rheology objects are too large in the model 3, we add a set of extra internal forces initially in the model 1. Finally, we propose a digitalized approach to transmit a set of active external forces from a rigid body to a rheology object via their encountered surface. The rheology object is digitalized as a lot of mass points. Therefore, we develop a virtual (digital) force transmission which approximates to a real (analog) force transmission.

2.1. Our Element with Voigt Model and Damper

First of all, we introduce our element which consists of Voigt model and damper serially as shown in Fig.1. This is similar to elements proposed in two researches [16],[17]. In the paper [16], the element is tried, which consists of Voigt and Maxcell models serially. In the paper [17], the element is investigated, which consists of Voigt model and an adaptive damper serially. The adaptive damper flexibly controls coefficient of damper during pushing and releasing operations. In our element, the left Voigt model represents viscosity and elasticity, and the right damper expresses some residual displacement. The former Voigt model generates many kinds of elastic and visco-elastic materials. The latter damper makes many properties of rheology object.

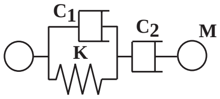

Figure 1: *Our basic element with two dampers and one spring.*

We briefly introduce what kinds of properties three coefficients generate. The larger the coefficient K is, the stronger the elasticity is. K controls displacement of deformation behavior. Moreover, the larger the coefficient C_1 is, the stronger the viscosity is. C_1 controls speed of the behavior. Finally, the larger the coefficient C_2 is, the larger the residual displacement is. If C_2 is small enough, the object appears elastic or visco-elastic property. On the other hand, if C_2 is large enough, the object appears plastic property. Furthermore, if K, C_1, C_2 are adequately selected within [100,3000], [500,10000], [500,20000], the virtual rheology object pushed freely is stably deformed as Fig.2(a). If some of K, C_1, C_2 are too large, each element becomes unstable and consequently the shape of rheology model is crushed as Fig.2(b). If K, C_1, C_2 are too small, each element looses elasticity, viscosity, and residual displacement, respectively.

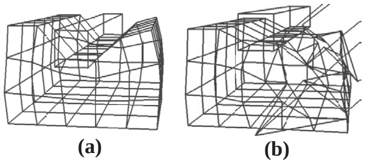

Figure 2: *(a) A stable shape of a rheology model. (b) An unstable shape of the model.*

2.2. A Basic Voxel/Lattice Model (Model 1)

A rheology object deforms in a 3-D environment. In order to describe several kinds of deformations flexibly, we adopt a symmetric voxel/lattice structure to describe a rheology object [16],[17]. In the structure, let us distribute mass points uniformly in a rheology object whose intervals are the same along X, Y, and Z axes (Fig.3(a)). Let N be the number of mass points and M_{object} be the total mass of rheology object. Therefore, each mass is given by $M = M_{object}/N$.

In our experiment, a real rheology object is made by mixing wheat flour and water. The rheology object (10cm × 6cm × 10cm divisions) is horizontally and vertically two times larger than its virtual rheology object (5cm × 3cm × 5cm divisions). Here, the unit length l is defined as 1cm in a virtual environment. Therefore, the former volume (600 $[cm^3]$ = water weight [g]) is eight times larger than the latter volume (75 $[cm^3]$). In the experiment, we use M = 6.0, M_{object} = 864 [g], N = 144 (= 6 × 4 × 6).

The elements are inserted between all the neighboring mass points as illustrated in Fig.3(b). The virtual rheology object is deformed by expanding and contracting all the elements. Let $P_{i,j,k}$ be position vector corresponding to mass point (i, j, k) ($1 \le i \le 6$, $1 \le j \le 4$ and $1 \le k \le 6$). Let us

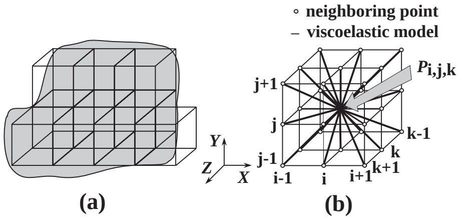

Figure 3: *(a) A voxel/lattice model of rheology object (b) Neighboring mass points connecting by many basic elements.*

derive quadratic differential equation of each mass at $P_{i,j,k}$. Each internal force acting on $P_{i,j,k}$ by the element between $P_{i,j,k}$ and its neighboring point $P_{i+\alpha,j+\beta,k+\gamma}$ is denoted by $F^{\alpha,\beta,\gamma}_{i,j,k}$. For each mass point, 6 elements whose distance is denoted as l (=1cm), 12 elements whose distance is denoted as $\sqrt{2}l$ (=$\sqrt{2}$cm), and 8 elements whose distance is denoted as $\sqrt{3}l$ (=$\sqrt{3}$cm) are located. Therefore, total internal force $F^{e}_{i,j,k}$ acting on $P_{i,j,k}$ is given by the sum of 26(= 6 + 12 + 8) internal forces $F^{\alpha,\beta,\gamma}_{i,j,k}$. Moreover, if the sum of active external forces at $P_{i,j,k}$ is denoted by $F^{\alpha}_{i,j,k}$, we can obtain the following quadratic differential equation. This summation is not so expensive on calculation time.

$$M\ddot{P}_{i,j,k} = F^{e}_{i,j,k} + F^{\alpha}_{i,j,k} \qquad (1)$$

In order to calculate next position $P_{i,j,k}$ ($1 \le i \le 6$, $1 \le j \le 4$ and $1 \le k \le 6$) at each mass, we should solve the above differential equation. This is done by the fourth Runge-Kutta method, but it is expensive. If a human operator gives an active external force at a mass point, its next position is calculated. By the expanding and contracting its neighbor elements, adequate internal forces are received at all connecting masses. Then, their next positions are calculated in parallel. This serial or parallel propagation starts from a set of pushed masses. Finally after determining all mass positions, we can describe a pushed virtual object in a 3-D graphics PC.

$$F^{e}_{i,j,k} = \sum_{\substack{\alpha,\beta,\gamma \in \{-1,0,1\} \\ (\alpha,\beta,\gamma) \neq (0,0,0)}} F^{\alpha,\beta,\gamma}_{i,j,k} \qquad (2)$$

We note that mass positions on an encountered surface between a rheology object and its pushing rigid body, and on the whole floor are fixed. This means the integrations by the Runge-Kutta method can be neglected at these points.

Finally in this model and the other models described in previous works [16],[17], we understand that shape of a calculated (virtual) rheology object unfortunately differs from shape of a practical (real) one. The shape differences mainly occur at two areas: (1) upper side around pushing surface; and (2) four sides of rheology object. This phenomenon occurs because of weak expansion forces. The reason is that volume of an experimental (real) object is always larger than volume which is eight times larger than volume of a calculated (virtual) object. To overcome this drawback, we construct two kinds of models in the following paragraphs.

2.3. Voxel/Lattice Model Including Volume Constant Condition in the Model 1 (Model 2)

In our previous work [20], we understand shape of a calculated (virtual) rheology object is not easily much the same as shape of an experimental (real) object. Especially, eight times larger volume of the former object is too small against volume of the latter object after the releasing. To overcome this drawback, we expand volume of each voxel by volume constant condition. The volume constant condition extends a voxel during deformation by eliminating four longest elements (whose distances are to be $\sqrt{3}$cm as shown in Fig.4(a)) and adding eight half-length elements from its center of gravity to eight vertices (whose distances are to be $\sqrt{3}/2$cm as illustrated in Fig.4(b)).

This technique has been already used in a mass-spring model [12]. This controls the isotropy or anisotropy of some elastic material. This idea can be straightforwardly extended to a mass-spring-damper model as this paper. Three coefficient of K, C_1 and C_2 in each of eight shorter elements are defined by multiplying α to K, C_1 and C_2 in each of four longest elements. Note that α is set as 0.5 because each of four elements is two times longer than each of eight elements.

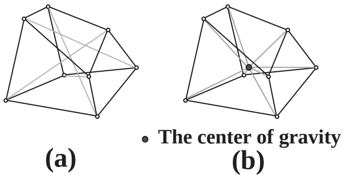

Figure 4: *(a) Four longest elements always connect two opposite vertices in a voxel. (b) Eight shorter elements connect from the gravity center of a voxel to its eight vertices.*

2.4. Voxel/Lattice Model Adding Extra Internal Forces in the Model 1 (Model 3)

With the support of volume constant condition, eight times larger volume of virtual (calculated) rheology object is similar to that of real (experimental) rheology object. However, this property is not always stable. It depends on the set of coefficients. The virtual object volume is sometimes too large because of stronger diagonal forces within each voxel. Moreover, shape of the former object still differs from that of the latter object. Especially, the swelling of four sides and that beside pushing surface are not enough. To overcome this tendency, we propose the model 3 to add extra internal forces into all mass points neighboring pushed mass points on the rigid body and the floor. The set of internal forces at masses neighboring masses pushed by a rigid body is the same against the set of active external forces received from the rigid body. Also, the set of internal forces neighboring masses on the floor is the same against the set of reactive external forces received from the whole floor.

In this paragraph, we explain how to calculate two kinds of extra internal forces in this model. If a rheology object is pushed by a rigid body, a volume between pushed area on the upper side and the bottom side on the floor is forced out. To express this phenomenon, we add two sets of special internal forces initially (Fig.5(a)). One is a set of internal forces at mass points neighboring upper mass points pushed by active forces F_{act} from a rigid body, and another is a set of internal forces at mass points neighboring bottom mass points pushed by reactive forces F_{rea} from a floor.

First of all, at each mass point neighboring mass point pushed by a rigid body along the vertical axis, e.g., Y-axis, we consider an extra internal force F whose direction is from G to the mass point. The point G is the intersection between line and plane (Fig.5(b)). The line includes the gravity center of encountered area between rheology and rigid objects, whose direction is coincident with a pushing direction. In this paper's experiments, the direction is restricted along the vertical axis, i.e., Y-axis. On the other hand, the plane is defined as the horizontal plane including the gravity center G_{object} of rheology object, i.e., XZ-plane. Then, we decompose each extra internal force F into three components F_x, F_y and F_z (Fig.5(c),(d)). As contrasted with this, at each mass point neighboring mass point pushed from the floor along the vertical axis, i.e., Y-axis, we consider another special internal force F whose direction is from G to the mass point. Then, we decompose each extra internal force F into F_x and F_z (Fig.5(c),(d)).

Secondly, at each mass point neighboring mass point pushed by a rigid body along one of the other axes, e.g., X-axis and Z-axis, we independently consider an extra internal force F whose direction is from G to the mass point. In addition, we decompose each extra internal force F into F_x, F_y and F_z (Fig.5(c),(e)). Finally, we reverse direction of the F_x along X-axis. As contrasted with this, at each mass point neighboring a mass point pushed from the floor along one of the other axes, e.g., X-axis and Z-axis, we independently consider another extra internal force F from G to the mass point. Moreover, we decompose each special internal force F into F_x and F_z (Fig.5(c),(e)). Finally, we reverse direction of the F_x along X-axis.

All masses neighboring masses on a rigid body are received by larger internal forces, whose distances are the unit length. Also, all messes neighboring masses on the rigid body are received by smaller internal forces, whose distances are $\sqrt{2}$ times larger than the unit length. In Fig.5(d),(e), the former masses are described as larger black circles, and the latter masses are illustrated as smaller black circles. The ratio between larger and smaller forces is 2 : 1. The sum of additional internal forces running at all the masses is the same against the sum of external active forces at masses around the rheology object pushed by the rigid object. Moreover, all masses neighboring masses on a rigid floor are received by larger internal forces, whose distances are the unit length. Also, all messes neighboring masses on the rigid floor are received by smaller internal forces, whose distances are $\sqrt{2}$ times larger than the unit length. In Fig.5(d),(e), the former masses are described as larger squares, and the latter masses are illustrated as smaller squares. The ratio between larger and smaller forces is 2 : 1. The sum of additional internal forces running at all the masses is the same against the sum of external reactive forces at masses around the rheology object pushed by the rigid floor.

In all trials, two kinds of special internal forces are initially added into the model 1. Therefore, we never add any external force after pushing a rheology object. For this reason, the sum of internal forces in the rheology object is always constant after pushing. As a result, **the conservation law of momentum** is exactly maintained during the deformation.

2.5. A Digital Operation Pushing a Rheology Object by a Rigid Body

The advantage of MSD model is to calculate force propagation and shape deformation efficiently. For this reason, we do not like to divide a virtual rheology object into a lot of voxels. As long as the number of voxels increases, calculation cost and memory storage in three models increase exponentially. To overcome this calculation explosion, we propose a digital operation pushing a rheology object by a rigid body.

In this research, we flexibly push a virtual rheology object by a rigid body, which are digitalized by the unit length l. First of all, surface around a rigid body is uniformly digitalized by the unit length l as illustrated in Fig.6(a). In general, since width and length of the surface cannot be divided exactly, rests m and n appear at ends of the surface ($m, n \leq l$). For this reason, there are major mass points, e.g., f_3^*, and are minor mass points, e.g., f_1^*, f_2^* and f_4^* around the surface. On the other hand, surface around a rheology object is also uniformly digitalized by the unit length l as illustrated in Fig.6(b). In this research, so as to keep a selected force precisely, we push a rheology object by a rigid body located on the tip of a robotic manipulator.

After transmitting many active forces from a digitalized rigid body to another digitalized rheology object, the rheology object starts to deform. In general, each active force f^* around the rigid body usually hits inside a cell around the rheology object. Therefore, a major force f^* should be distributed into four forces f_a, f_b, f_c and f_d at four vertices of a cell around the rheology object (Fig.6(c)). Moreover, magnitudes of four forces f_a, f_b, f_c and f_d are determined as follows: First of all, we define the total force received by each cell as the area $F = l^2$. Secondly, we determine $f_a = s \times t$ [area a] if $0.25 \times l \leq s$ and $0.25 \times l \leq t$, we regard $f_b = (l - s) \times t$ [area b] if $s \leq 0.75 \times l$ and $0.25 \times l \leq t$, we select $f_c = (l - s) \times (l - t)$ [area c] if $s \leq 0.75 \times l$ and $t \leq 0.75 \times l$, and finally we find $f_d = s \times (l - t)$ [area d] if $0.25 \times l \leq s$ and $t \leq 0.75 \times l$. Overall, magnitudes of four forces linearly correspond to their opposite areas. For example, magnitudes of f_a, f_b, f_c and f_d are linearly determined as areas a, b, c and d (Fig.6(c)). Moreover, we should consider a minor force f^*

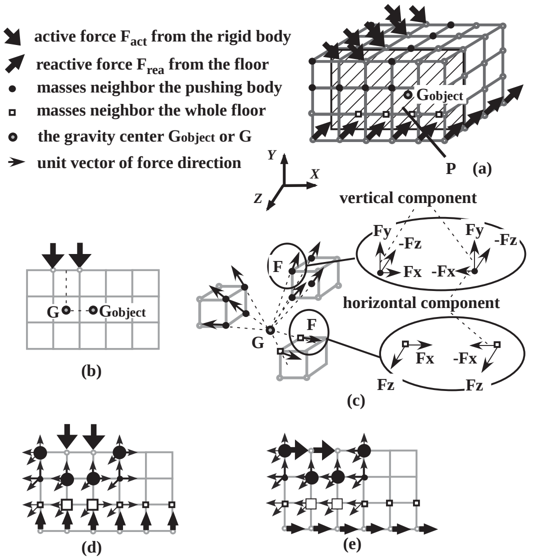
active force F_{act} from the rigid body
reactive force F_{rea} from the floor
masses neighbor the pushing body
masses neighbor the whole floor
the gravity center G_{object} or G
unit vector of force direction
G_{object}
P (a)
Y
X
Z
vertical component
Fy
-Fz
Fy
-Fz
Fx
-Fx
F
G
G_{object}
(b)
horizontal component
F
G
Fx
-Fx
Fz
Fz
(c)
(d)
(e)

Figure 5: *(a) There are two kinds of special internal forces affecting a virtual rheology object. (b) The point G is the intersection between line and plane. The line passes through the gravity center of encountered area between rheology and rigid objects, whose direction equals to the pushing direction. In this research, the line is limited in the vertical axis, i.e., Y-axis. On the other hand, the plane is defined as XZ horizontal plane including the gravity center G_{object} of rheology object. (c) An extra internal force F generated by an active external force consists of three components F_x, F_y and F_z. As contrasted with this, an extra internal force F generated by a reactive external force consists of two components F_x and F_z. (d) F_x, F_y and F_z at masses denoted as black circles are made for active external and vertical forces by a rigid body on an arbitrary sliced plane P. The magnitude of forces at larger black circles is twice bigger than that of forces at smaller black circles. On the other hand, F_x and F_z at masses denoted as squares are made for reactive external and vertical forces by a whole floor on an arbitrary sliced plane P. The magnitude of forces at larger squares is twice bigger than that of forces at smaller squares. (e) F_x, F_y and F_z at masses denoted as black circles are made for active external and horizontal forces by a rigid body on an arbitrary sliced plane P. F_x is to be the opposite force of X-component of the active force. The magnitude of forces at larger black circles is twice bigger than that of forces at smaller black circles. On the other hand, F_x and F_z at masses denoted as squares are made for reactive external and horizontal forces by a whole floor on an arbitrary sliced plane P. F_x is to be the opposite force of X-component of the active force. The magnitude of forces at larger squares is twice bigger than that of forces at smaller squares.*

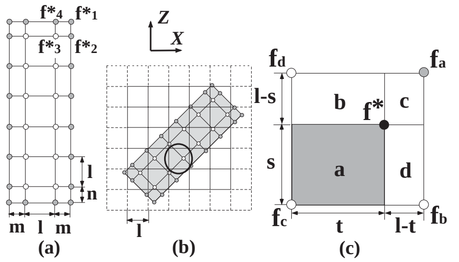

Figure 6: *(a) A set of cells around a rigid body. (b) A cell-based relationship between encountered surfaces of a rheology object and a rigid body. (c) An original force f^* is distributed into four forces f_a, f_b, f_c and f_d.*

located in the margin of pushing area. That is, we consider a horizontal force $f_h^* = m/l \times f^*$ if $0.25 \times l \leq m$, e.g., f_2^*, we consider a vertical force $f_v^* = n/l \times f^*$ if $0.25 \times l \leq n$, e.g., f_4^*, and we regard a diagonal force $f_d^* = (m^2+n^2)/l^2 \times f^*$ if $0.25 \times l \leq m$ and $0.25 \times l \leq n$, e.g., f_1^* (Fig.6(a)). This approach is an approximated transformation based on the balance of all analog forces and their distances via a digital encountered surface.

3. Modified Randomized Algorithm for Calibrating C1, C2 and K

The defective point of MSD is accuracy of force propagation and shape deformation. To overcome this, we calibrate K, C_1 and C_2 by minimizing the difference between shape deformations of real and virtual rheology objects in our modified randomized algorithm.

3.1. How to Calculate Shape Difference Between Real and Virtual Rheology Objects

In this research, a rheology object is precisely pushed by a rigid body located at the tip of a robotic manipulator (Fig.7(a)). The deformation, that is, the sequence of shapes is measured by two stereo vision camera systems *Digiclops* and its software development kit (SDK) *Triclops* (provided by Point Grey Research Inc, Canada). Each captures about three or more thousand points as shape of real rheology object in the real-time manner. After capturing the shape deformation, we finally measure how much total volume is changed before pushing and after releasing the rheology object. For this purpose, we use the following primitive method. First of all, we fill a ball with water, and then drop a deformed object into the ball (Fig.7(b)). Secondly, we gather overflowed water and measure its weight by a precise electric balance. As a result, we can understand volume of rheology object always decreases by about three percentages.

In order to evaluate a difference between real and virtual rheology objects, we summarize minimum distances from captured points to their nearest surfaces around a virtual rheology object. A virtual object consists of 5 × 3 × 5 hexahedrons which are individually deformed from initial cubes. Therefore in order to evaluate the difference, we calculate

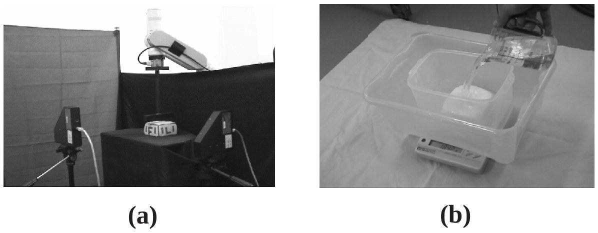

Figure 7: *(a)An experiment system: A real rheology object is pushed by a rectangular rigid body located at the tip of a robotic manipulator, and deformation of its sides is measured by two Digicrops cameras. (b)A photo when we measure volume of a rheology object before pushing and after releasing by overflowed water.*

the minimum of 5 × 3 × 5 shortest distances for each captured point to all hexahedrons by Lin-Canny closest point algorithm [18], and then we calculate the sum S of all the minimum distances for all captured points, which are smaller than the average error $0.05cm$ of $Digiclops$. In our calibration, we use the sum of four S at four times during pushing and after releasing. By minimizing the total sum in an efficient randomized algorithm, we can obtain a better set of three coefficients K, C_1 and C_2 of two dampers and one spring.

3.2. A Steepest Descendent Method

1. Two parameters T_{cal} (threshold of calculation time) and T_{ran} (driving distance of random walk) are given in advance.
2. Initialize coefficients K, C_1 and C_2 in a 3-D search space.
3. We calculate shape difference S between real and virtual rheology objects.
4. In order to find all the possible neighbors, we decrease and increase K, C_1 and C_2 by Δ. In this 3-D case, we obtain eight possibilities, that is, (K+Δ, C_1+Δ, C_2+Δ), (K+Δ, C_1+Δ, C_2-Δ), (K+Δ, C_1-Δ, C_2+Δ), (K+Δ, C_1-Δ, C_2-Δ), (K-Δ, C_1+Δ, C_2+Δ), (K-Δ, C_1+Δ, C_2-Δ), (K-Δ, C_1-Δ, C_2+Δ) and (K-Δ, C_1-Δ, C_2-Δ). Then, after calculating all sums at all the neighbors, we select their minimum.
5. If the minimum is smaller than S obtained in step 3, we move to the neighbor with the minimum by decreasing or increasing K, C_1 and C_2 by Δ, and return to step 3. Otherwise, the algorithm finishes.

3.3. Our Randomized Algorithm

1. We select arbitrary K, C_1 and C_2 within a given 3-D search space, whose ranges are $K^{min} \leq K \leq K^{max}$, $C_1^{min} \leq C_1 \leq C_1^{max}$ and $C_2^{min} \leq C_2 \leq C_2^{max}$.
2. We calculate the S for the K, C_1 and C_2. By the steepest descendent method described above, we get one of the local minima whose value is the smallest S and set $S_{ran} = S$. Then, if calculation time equals to or is larger than T_{cal}, the algorithm ends, otherwise, move to step 3.
3. We randomly increase and decrease three coefficients K, C_1 and C_2 T_{ran} times by Δ. Then, we calculate S for K, C_1 and C_2. Then, if $S \leq S_{ran}$ is satisfied, we return to step 2, otherwise, continued to step 3.

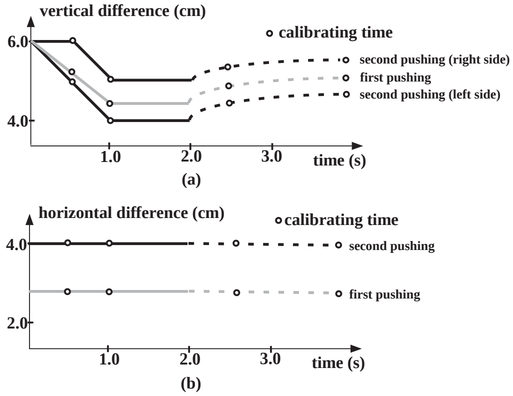

Figure 8: *Two pushing operations are described as the gray and black lines. The vertical/horizontal differences of pushing and releasing are described as the whole and dot lines. (a) Y-axis direction. (b) X-axis direction.*

4. Comparative Results

In this section, we compare three MSD models with each other concerning to computation time, memory storage and shape accuracy. The deformation of virtual rheology object is calculated and visualized by a 3-D graphics software OpenGL in a personal computer (CPU: Pentium4 2.26GHz, Main memory: 1024MB) with a 3-D graphics acceleration board (NVDPIA Quadro 2EX, 32MB).

4.1. Computation Complexity

As mentioned previously, a real rheology object pushed and released by a rigid body deforms during 4 seconds in all the experiments. On the other hand, 2000 deformations of a virtual rheology object under the models 1, 2, and 3 are calculated during about 40 seconds in the CG environment. As shown in Table 1, we check calculation time less than 20 milli-seconds per one deformation. The speed is enough to make dynamic animation because it is smaller than the video-frame rate, i.e., 33 milli-seconds. Furthermore, we show memory storage of the models 1, 2 and 3 is too small and also relatively constant (Table 1).

The numbers of masses and elements in the models 1 and 3 are the same, but numbers of masses and elements in the models 1 and 2 differ from each other. As mentioned previously, force calculation at each mass is not expensive, but position calculation at the mass is time consuming because of solving the differential equation. The force calculation is always necessary in all the elements, but the position calculation is not always necessary in all the masses. For example, since position of each mass on the floor and the pushing body is fixed and can be calculated without the integration. Moreover in the model 2, since the mass position within each voxel is calculated as the gravity center of the voxel, the integration is not necessary.

For this reason, calculation costs of three models totally depend on the number of mass points $N_{cal} = (N_x \times N_y \times N_z) - f_{out} - (N_x \times N_z)$ (N_x, N_y, N_z: the numbers of masses along X, Y, Z axes, respectively, f_{out}: the number of masses pushed by a rigid body). As shown in Table 1, since $f_{out} = 12$ is in the first pushing under the $6 \times 4 \times 6$ model, the number of masses to need the integration is denoted as $N_{cal} = (6 \times 4 \times 6) - 12 - (6 \times 6) = 96$. Similarly, the number of masses to need the integration is denoted as 182 and 304 for $7 \times 5 \times 7$ and $8 \times 6 \times 8$ models, respectively. As described in Table 1, calculation time of each model is proportionally evaluated by the equation $N_{cal} = (N_x \times N_y \times N_z) - f_{out} - (N_x \times N_z)$.

Since numbers of mass points in three kinds of models are almost the same, memory storage is also the same. The memory storage m is approximately evaluated by summing the mass storage m_N (N: the number of masses) and basic software storage m_S including C++ compiler (VC++ 6.0). The former is proportional to the number of masses, but the latter is invariable. Using the results in Table 1, we calculate $m_S = 19000[KB]$ and $m_N = 16 \times N[KB]$.

Table 1: *Calculation time* $[msec]$ *per one deformation and memory storage* $[KB]$ *in PC (CPU: Pentium4 2.26GHz, Main memory: 1024MB)*

Calculation time for the first pushing $[msec]$			
Number of total masses	$[6 \times 4 \times 6]$	$[7 \times 5 \times 7]$	$[8 \times 6 \times 8]$
(model1)	18.508	35.125	60.316
(model2)	19.553	37.141	62.805
(model3)	18.790	35.860	60.328
Memory storage $[KB]$			
(model1)	21416	23044	25260
(model2)	21452	23048	25360
(model3)	21492	23044	25315

4.2. Deformation Accuracy

In this paragraph, we describe a global aspect of a rheology object by changing three coefficients. In the same randomized algorithm, we use the same parameters $\Delta = 10$, $T_{cal} = 30\ [hour]$, and $T_{ran} = 100[number]$. The search space consists of three intervals $[K^{min}, K^{max}]$, $[C_1^{min}, C_1^{max}]$ and $[C_2^{min}, C_2^{max}]$ which are defined by [100,3000], [500,10000] and [500,20000]. If each interval is divided by $\Delta = 10$, the search space includes candidate points whose number is 537225000. The resolution $\Delta = 10$ is experimentally selected in order to find the optimal or a near-optimal solution. If $\Delta < 10$, the number of candidate points is too large to find the solution. If $\Delta > 10$, the magnitude of voxel resolution is too large to find it.

To calculate the sum of differences between real and virtual rheology objects at each candidate point, we need about 40 seconds. In this research, we use $T_{cal} = 30\ [hour]$ to investigate a better set of three coefficients, and therefore we can check candidate points whose number is about 2700. The search density is too sparse and therefore the randomized algorithm selects a near-optimal solution by using $T_{ran} = 100[number]$ and eliminating candidate points whose sum S is larger than the present minimum sum S_{ran}.

In order to ascertain goodness of pushing a rheology object by a rigid body, we prepare two kinds of pushing. In both operations, external forces at masses around a rheology object act along the Y-axis. In the first operation, all masses around a rheology object are simultaneously pushed by a rigid body (Fig.9). In this case, we compare which model is the best. Shape and volume differences between real and virtual rheology objects are given in Table 2, and also shapes of both objects after releasing are described in Fig.10. As shown in Table 2 and Fig.10, the model 3 is the best concerning to shape accuracy and volume consistency.

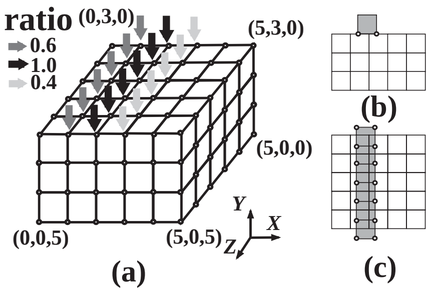

Figure 9: *(a) 3-D view for the first pushing. (b) Front view. (c) Upper view.*

Table 2: *Calibration results for the first pushing in three models. (S: The sum of error distances)*

The number of captured points is $N = 15235$						
Calibration result	S $[cm]$	S/N $[cm]$	K $[gf/cm^3]$	$C1$ $[gfs/cm^3]$	$C2$	$Volume$ $[cm^3]$
(model 1)	2097.40	0.138	1990	510	3470	68.49
(model 2)	2308.62	0.152	1720	1300	780	83.88
(model 3)	1785.78	0.117	3000	1380	1550	69.95
The number of points whose error is more than 0.25cm						
	first	second	third	fourth	total	
(model 1)	513	381	335	267	1496	
(model 2)	498	451	459	388	2796	
(model 3)	266	224	242	177	895	

In the second operation, right and left endpoints of a rigid body contact a rheology object at different times (Fig.11). In this case, shape and volume differences between real and virtual rheology objects are given in Table 3, and also shapes of both objects after releasing are described in Fig.12. As shown in Table 3 and Fig.12, the model 3 is the best concerning to shape accuracy and volume consistency.

Table 3: *Calibration results for the second pushing in three models. (S: The sum of error distances)*

The number of captured points is $N = 14551$						
Calibration result	S $[cm]$	S/N $[cm]$	K $[gf/cm^3]$	$C1$ $[gfs/cm^3]$	$C2$	$Volume$ $[cm^3]$
(model 1)	2483.15	0.171	1970	510	3470	67.95
(model 2)	2746.35	0.189	1780	980	720	88.95
(model 3)	2051.96	0.141	1950	7520	8930	70.28
The number of points whose error is more than 0.25cm						
	first	second	third	fourth	total	
(model 1)	1260	983	724	751	3718	
(model 2)	1002	1629	755	832	4218	
(model 3)	837	604	496	485	2422	

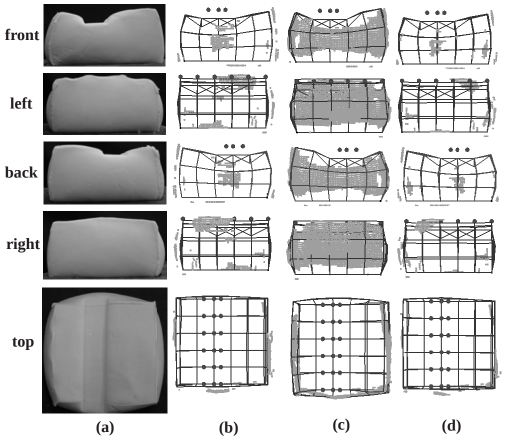

Figure 10: *The distance error between real and virtual rheology objects for the first pushing. (a) Real rheology object. (b),(c),(d) Virtual rheology objects which are colored by gray, whose errors are larger in the proposed models 1, 2 and 3.*

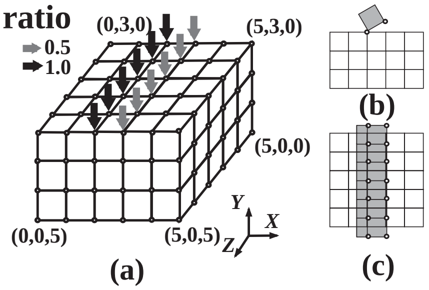

Figure 11: *(a) 3-D view for the second pushing. (b) Front view. (c) Upper view.*

5. Conclusions and Future Works

In this paper, we represent a rheology object by three kinds of MSD models, and calibrate three coefficients of two dampers and one spring by an efficient randomized algorithm based on many experimental results. Since force propagation of rheology object during each deformation is efficient in the MSD model, we can watch many deformations in the video-frame rate by a personal computer with a popular 3-D graphics acceleration board. Moreover, by the careful calibration as pre-processing, shape and volume of virtual (calculated) rheology object are quite similar to those of real (experimental) one during and after simple pushing operations. As a result, the model 3 is the best concerning to shape and volume accuracy if and only if force directions acting at masses are along the vertical axis, i.e., Y-axis, and also force magnitudes are not so large.

Finally as several future works, another structure (e.g., nested or non-nested tetrahedral meshes [2]) and another element (e.g. mass-spring element, Voigt and Maxwell element [16],[17]) should be tested by our experimental calibration. In addition, as the calibration algorithm, we should try to use another optimal algorithm such as GA (generic algorithm).

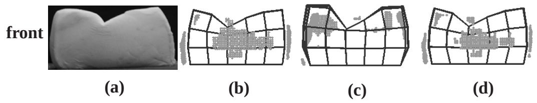

Figure 12: *The distance error between real and virtual rheology objects for the second pushing. (a) Real rheology object. (b),(c),(d) Virtual rheology objects which are colored by gray, whose errors are larger in the proposed models 1, 2 and 3.*

Acknowledgements

The authors thank Professor S.Hirai and H.Tanaka (Ritsumeikan University) for fruitful discussions. This research is supported in part by 2002 Grants-in-aid for Scientific Research from the Ministry of Education, Science and Culture, Japan (No.14550247).

References

1. Y.Zhuang, "Real-time simulation of physically-realistic global deformations," *Doctoral Thesis, UC Berkeley*, 2000.
2. G.Debunne, M.Desbrun, M.Paule Cani and A.Barr, "Dynamic real-time deformations using space and time adaptive sampling," *Computer Graphics (Proc. SIGGRAPH)*, pp.31-36, 2001.
3. X.Wu, M.S.Downes, T.Goktekin and F.Tendick, "Adaptive nonlinear finite elements for deformable body simulation using dynamic progressive meshes", *Computer Graphics Forum (Proc. Eurographics)*, pp.349-358, 2001.
4. K.Waters, "A muscle model for animating three-dimensional facial expression," *Computer Graphics (Proc. SHIGGRAPH)*, pp.17-24, 1987.
5. E.Promayon, P.Baconnier and C.Puech, "Physically-based deformations constrained in displacements and volume," *Computer Graphics Forum (Proc. Eurographics)*, pp.155-164, 1996.
6. J.Louchet, X.Provot and D.Crochemore, "Evolutionary identification of cloth animation models", *Computer Animation and Simulation (Eurographics)*, pp 44-54, 1995.
7. J.Louchet, M.Boccara, D.Crochemore and X.Provot, "Building new tools for synthetic image animation using evolutionary techniques", *Artificial Evolution '95 (Springer Verlag: Eds. J.M.Alliot, E.Lutton, E.Ronald, M.Schoenauer and D.Snyers)*, pp 273-286, 1996.
8. D.Terzopoulos, J.Platt, A.Barr and K.Fleischer, "Elastically deformable models," *Computer Graphics (Proc. SIGGRAPH)*, pp.205-214, 1987.
9. M.B.Nielsen and S.Cotin, "Real-time volumetric deformable models for surgery simulation using finite elements and condensation," *Computer Graphics Forum (Proc. Eurographics)*, pp.57-66, 1996.
10. D.James and D.K.Pai, "ArtDefo, accurate real time deformable objects," *Computer Graphics (Proc. SIGGRAPH)*, pp.65-72, 1999.
11. D.Terzopoulos and K.Fleisher, "Modeling inelastic deformation: viscoelasticity, plasticity, fracture," *Computer Graphics (Proc. SIGGRAPH)*, pp.269-278, 1988.
12. G.Debunne, M.Desbrun, M.-P.Cani, A.Barr, "Adaptive simulation of soft bodies in real-time," *Computer Animation 2000*, pp.133-144, 2000.
13. M.Muller, J.Dorsey, L.McMillan, "Real-time simulation of deformation and fracture of stiff materials", *Proceedings of Eurographics, Computer Animation and Simulation (CAS)*, pp.113-124, 2001.
14. M.Muller, J.Dorsey, L.McMillan, R.Jagnow and B.Cutler, "Stable real-time deformations", *Proceedings of ACM SIGGRAPH Symposium on Computer Animation (SCA)*, pp 49-54, 2002.
15. J.J.Berkley, "Haptic rendering of deformable bodies using real-time finite element analysis: an application to surgical simulation," *Doctoral Thesis, University of Washington*, 2002.
16. S.Tokumoto, Y.Fujita and S.Hirai, "Deformation modeling of viscoelastic objects for their shape control," *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pp.1050-1057, 1999.
17. S.Tokumoto, S.Hirai and H.Tanaka, "Constructing virtual rheological objects," *Proc. World Multiconference on Systemics, Cybernetics and Infomatics*, pp.106-111, July, Auland, 2001.
18. M.C.Lin and J.F.Canny, "A fast algorithm for incremental distance calculation," *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pp.1008-1014, 1991.
19. J.-C.Latombe, "Robot motion planning", *Kluwer Academic Publishers.*, 1991.
20. H.Noborio, R.Enoki, S.Nishimoto and T.Tanemura, "On the calibration of deformation model of rheology object by a modified randomized algorithm," *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2003 (to appear).