COMBINING DEFORMABLE AND RIGID BODY MECHANICS SIMULATION

Johan Jansson

Faculty of Information Technology
Uppsala University
Sweden

Joris S. M. Vergeest

Faculty of Design, Engineering and Production
Delft University of Technology
The Netherlands

Abstract

We present an interface between a deformable body mechanics model and a rigid body mechanics model. What is novel with our approach is that the physical representation in both the models is the same, which ensures behavioral correctness and allows great flexibility. We use a mass-spring representation extended with the concept of volume, and thus contact and collision. All physical interaction occurs between the mass elements only, and thus there is no need for explicit handling of rigid-deformable or rigid-rigid body interaction. This also means that bodies can be partially rigid and partially deformable. It is also possible to change whether part of a body should be rigid or not dynamically. We present a demonstration example, and also possible applications in conceptual design engineering, geometric modeling, as well as computer animation.

Key words:
mechanics model, deformation, collision, deformable bodies, geometric modeling, conceptual design, rigid body

1 Introduction

1.1 Background

One of the main reasons the electronic computer was invented was to perform very large computations which would have been impractical to perform manually. Previously, problems were solved by creating analytical models which were compact enough to solve problems in on paper. With electronic computation, we do not need to create new models for every problem we encounter, but can have small general analytical cores, and instead put the burden of the problem on the computation.

This philosophy is particularly well suited to (solid) mechanics. In most mechanics literature [8] [16], we can read that solid mechanics can be described by a small, very Newtonian model with particles interacting through $\vec{F} = m\vec{a}$. Nevertheless, for both historical and practical reasons, other models are used, resulting in incompatibilities.

There exists a multitude of different models in practical use, each supporting a particular domain, or having particular strengths or weaknesses, compared within the same domain. If several models could be used together simultaneously, it would be possible to select the most fitting model for each part of the mechanics problem, or simulation. To be able to solve this, a common interface has to be found, using force or energy for example, through which the models can communicate correctly and efficiently.

In this paper, we take a small step along this path, and define an interface between mass-spring models, which are typically used for dynamics problems where deformation is important, and rigid body models, which are used for dynamics problems where deformation is negligible.

1.2 Problem Statement

Generally, we want to find interfaces between mechanics models so that they can be used simultaneously, and interact with one another. We will examine some of the most widely used models, and determine if and how this can be possible. Specifically, we will look at interfacing the rigid body motion model and a volumetric mass-spring model.

1.2.1 Rigid Body Motion Theory

Rigid body motion theory is a fundamental and well-established part of physics. It is based on the approximation that for stiff materials, any force applied to a body produces a negligible deformation. Thus, the only change a force can produce is change in the center of mass motion and change in the rotational motion. This means that simulation of even complex bodies is relatively simple, and thus this method has become popular in the computer simulation field.

Given the forces acting on the body, the motion can be determined using $\vec{F} = m\vec{a}$ for translational motion, and a similar relation for rotational motion $\vec{\tau} = \tilde{I}\vec{\omega}'$.

The rigid body motion model has traditionally been applied in range analysis in CAD and for computer animation where deformation is not required.

If the deformation is not negligible, then the approximation does not hold, and we need to start over and come up with some other model. There exists many different models, but the two models which have emerged to become the most widely used in practice are: mass-spring models and statics models solved using the Finite Element Method (FEM).

1.2.2 Mass-Spring Theory

Mass-spring models represent bodies as discrete mass-elements, and the forces between them are transmitted using explicit spring connections ("spring" is a historical term, and is not limited to pure Hooke interactions). Given the forces acting on an element, we can determine its motion using $\vec{F} = m\vec{a}$. The motion of the entire body is then implicitly described by the motion of its elements.

Mass-spring models have traditionally been applied mostly for cloth simulation.

1.2.3 Statics Theory

Statics models are based on equilibrium relations, and thus make the approximation that the effect of dynamics are negligible. Relations between the strain and stress fields of a body are set up, and through specifying known values of these fields, through for example specifying forces acting on the body, the unknown parts can be determined. These relations form differential equations, and the known values are boundary values. The FEM is an effective method for solving boundary value problems, and has thus given its name to these types of problems.

Statics models have traditionally been applied in stress and displacement analysis systems in CAD.

1.2.4 Interfaces

We can see similarities in these models. Forces are specified on a body to determine the sought after information. However, in the statics case, this information is stress and strain in other parts of the body, in the rigid body case it is only motion, and in the mass-spring case it is stress and strain (or displacement and force) over time, in addition to motion.

It is difficult to see how to interface the statics models with the other two, since it does not consider dynamics, while the others do. Between the rigid body models and the mass-spring models however, we can see a clear similarity: forces are specified to determine dynamical change of the bodies. The only difference between the models is that one assumes one of the possible changes, deformation, is negligible.

We now have a possible interface. What we now have to do is determine how our abstract discussion and terms function concretely, and determine how to solve any possible problems.

2 State of the Art

At least one attempt has been made to augment a rigid body model with a special module for deformable body collision [1]. It applies a two-phase model, where the first phase prevents inter-penetration, and the second phase calculates contact forces. Deformations are constrained to what can be represented by a global deformation function, which avoids the problem of calculating impulse propagation. It is not clear however, how general this approach is. The authors also state that allowing complex deformation functions will lead to a heavy computational burden.

Provot has described a mass-spring model used for cloth simulation [18], and Chen et. al. a model aimed at general objects [5]. It describes bodies as sets of point-masses, and the materialistic properties as a graph of springs over these sets (essentially a particle model as well). While the model is very useful for describing deformation of a single body, it does not cover collision at all, since there exists no concept of volume. Computationally, it can give rise to stiff differential equations, for very stiff springs for example, which requires finer time discretization, and thus more computation.

We have described a volumetric mass-spring model in [11] and [13], which extends the mass-spring model with collision. The mass-elements are given an interaction distance as a radius, and when two such radii intersect, a connection (spring) is created to produce the force interaction. We also informally describe how the mass-spring model relates to the atomic level of bodies, and how the model can be used to recursively increase or decrease resolution.

Terzopoulos, et. al. have developed a Lagrangian mechanics model aimed at animation [19]. They use continuous bodies, which are discretized in their solution method using the finite difference method. They treat elastically deformable bodies, and also collisions, which they handle by creating a force field around each body.

Kang and Kak [14] have developed a FEM system to create a geometric modeling system. The system presents the designer with an initial physical shape, represented by a FE shape. The user can then utilize a force-input interface, in their case a four-sensor plate, to manipulate nodes of the shape. The system could presumably be generalized to allow arbitrary input methods.

Baraff and Witkin propose “interleaved simulation” in [3] as a way to combine different mechanics models. The system is a black box system where the interface between the models is constraint forces. The system is shown to be general and efficient, but the solution method is asymmetric between two models, and thus is primarily suited for situations of large mass asymmetry between bodies belonging to different mechanics models.

O’Brien, Zordan and Hodgins present “coupled systems” in [17], again coupling different mechanics models, in the context of active vs. passive systems. Their interface between the models is force based. In their concrete example of the interaction between a basketball (a rigid body) and the net (a deformable mass-spring system), they compute the contact forces by imposing collision constraints. They do not however present a general model for interaction forces.

3 Proposed Approach

In the mass-spring model presented in [11], the only explicitly defined interactions are between individual elements. Higher-level behaviors such as contact and collision between bodies, self-collision, fracture etc. exist implicitly and do not need to be defined. We believe this property can be exploited when merging several models.

Our proposal for merging rigid body mechanics with this mechanics model is simply to use the mass-spring representation for describing a rigid body, but instead of computing the motion for each element individually, we use rigid body motion theory to compute the motion for an aggregate of elements.

This extension preserves all interaction possibilities from the mass-spring representation, and there is no need to explicitly determine rigid-deformable body interaction, or even rigid-rigid body interaction. Baraff and Witkin in [3] describe this explicit handling of interaction possibilities as a problem, and describe the resulting implementation difficulty as a “quadratic software explosion”. As mentioned however, this is completely avoided using our method.

We could go one step further, and also allow the statics model to be used. We could replace the mechanics simulation for an aggregate of elements with the statics model, and have the same benefits as in the rigid body case. However, as mentioned, there is still the problem of using the statics assumption of equilibrium in a dynamic system. This extension will have to be left for future research.

4 Theory

For the deformable mechanics model, we adopt the formalisms from [11]. To simplify the discussion, we remove non-crucial features from the model, specifically damping and friction. These can be transparently added back at the end of the discussion, since they have no impact on this particular extension.

The deformable body model is formally a superset of a mass-spring model. It adds a radius and fracture distance to the point mass entity. We call this new entity an "element", to reflect that it is no longer dimensionless. The spring entity is similarly extended with a fracture distance. We call this entity a "connection", to differentiate between pure Hookean interaction.

The reasoning behind these additions is that once two elements are within a certain distance from each other, the interaction is significant enough to explicitly be taken into account. Thus a connection element is created between the elements with properties derived from the elements. Conversely, once the elements move a certain distance apart, the interaction is no longer significant, and the connection is destroyed. These distances need not be the same, and thus we have two distance parameters.

4.1 Formalized Definitions for Deformable Bodies

Element:

An element e is a set of parameters $\{\ \vec{p},\ \vec{v},\ m,\ r,\ k,\ t,\ C\ \}$.

$\vec{p}$ position
$\vec{v}$ velocity
m mass
r radius
t fracture distance
C set of connections connected to the element

Connection:

A connection c is a set of parameters $\{\ e_1,\ e_2,\ k,\ l,\ t\ \}$.

$e_1,\ e_2$ elements comprising the connection
k Hooke spring constant
l nominal distance
t fracture distance

Deformable Body:

A deformable body B_d is a set of parameters $\{\ E,\ C\ \}$.

C set of connections comprising the body
E set of elements comprising the body

A deformable body is thus simply an aggregation of elements, and their connections. This aggregation does not affect the behavior of the elements in any way. If we describe a set of elements as one deformable body, or as n deformable bodies (a partition of the set), the quantitative behavior is exactly the same.

4.2 Formalized Definitions for Rigid Bodies

Rigid Body:

A rigid body B_r is a set of parameters $\{\ O,\ \vec{p},\ \vec{v},\ \vec{\omega},\ \tilde{I},\ m,\ E,\ C\ \}$.

O orientation of the body
$\vec{p}$ position of center of mass
$\vec{v}$ velocity of center of mass
$\vec{\omega}$ angular velocity
$\tilde{I}$ (tensor of) inertia
m mass
C set of connections comprising the body
E set of elements comprising the body

The representation for a rigid body is the same as for a deformable body, only that the distances between its elements are constrained to be fixed. Unlike a deformable body, a rigid body is treated as a single entity when determining its motion. Therefore we need extra parameters which are associated with it. All of these parameters exist implicitly in the elements E of the body, but are explicitly defined to ease treatment.

For cases where a body can be either a deformable or rigid body, we make a definition of “body”, which is identical to the definition of a deformable body:

4.3 Formalized Definitions for Bodies

Body:

A body B is a set of parameters $\{\ E,\ C\ \}$.

C set of connections comprising the body
E set of elements comprising the body

4.4 Mechanics of a Deformable Body

We have that the mechanics of a deformable body is determined by the mechanics of each of its elements $e_i \in E$. For an individual element e, we have, according to Newton’s second law:

$$\vec{F} = m\vec{p}'' \tag{1}$$

The force F is the sum of all forces acting on the element. We adapt the force model from [11] into a simplified form as follows:

$$\vec{F} = \vec{F}_C + \vec{F}_G \tag{2}$$

Where F_C is the force from the connections of the element and F_G is a simplified gravitational force.

To simplify notation, we define the subscript e to be the considered element, p to be the opposite element in a connection, and c to be a property of the current connection in the iterated sum. We iterate over the connection set C of the element:

$$\vec{F}_c = \sum_{i=0}^{|C_e|} -k_c(||\vec{p}_e - \vec{p}_p|| - l_c)\frac{\vec{p}_e - \vec{p}_p}{||\vec{p}_e - \vec{p}_p||} \tag{3}$$

Gravitation is a one-dimensional field.

$$\vec{F}_G = m\vec{g} \tag{4}$$

4.5 Mechanics of a Rigid Body

For a rigid body, we can only describe the mechanics of the center of mass and orientation. Thus, the treatment becomes similar to that of a single element. Any force $\vec{F}$ applied on a point $\vec{p}_a$ of a rigid body B_r can be decomposed into two components, parallel and perpendicular to the line between $\vec{p}_a$ and the center of mass $\vec{p}_{com}$ of the body:

$$\vec{F} = \vec{F}_\perp + \vec{F}_{||} \tag{5}$$

$\vec{F}$ itself is the same force as in the previous section on deformable bodies. Thus, the application point $\vec{p}_a$ of a connection force between an element e of B_r, and an element of some other body B is actually the position $\vec{p}$ of e.

For the mechanics of the center of mass, we have, as before:

$$\vec{F}_{||} = m\vec{p}'' \tag{6}$$

For the mechanics of the orientation, we have:

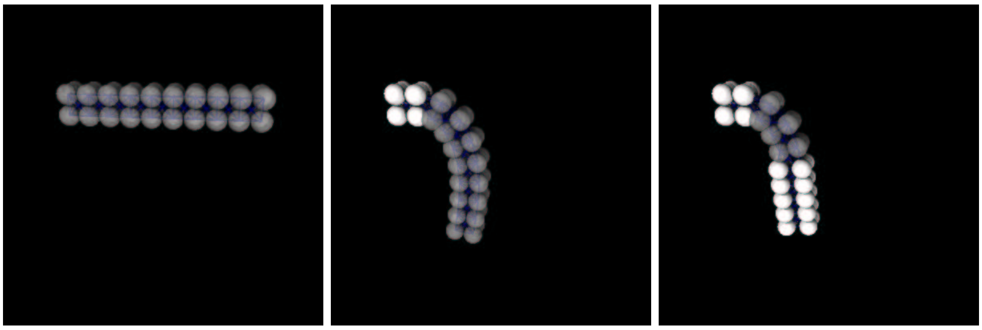

Fig. 1. A beam is fixed at one end and then bent using two methods; by making the whole beam deformable and by making the beam part deformable and part rigid.

$$\vec{\tau} = \tilde{I}\vec{\omega}' \tag{7}$$

Where:

$$\vec{\tau} = \vec{r} \times \vec{F}_{\perp} \tag{8}$$

And:

$$\vec{r} = \vec{p}_{com} - \vec{p}_a \tag{9}$$

4.6 Example

To help visualize the representations and the mechanics we show a simple example involving a beam. In figure 1 we can see a beam represented by elements and connections.

Assume we want to create a bend in the beam in a specific region. Using only the deformable model we fix the left end of the beam and pull down on the right end. This creates the expected curved shape along the entire free part of the beam. If we combine the deformable and rigid models we can again fix the left end, but now also make the right part of the beam rigid. Pulling down on the right end again bends the beam in the expected way, but only the deformable part is curved.

4.7 Implications

From this discussion we can see the model makes mathematical sense, but now we have to consider what the semantics of the model really are.

To begin with, we seem to have an apparent contradiction. For rigid bodies, we make the approximation that no significant deformation takes place. In this model however, a rigid body is deformable in the interface between another body (also between two rigid bodies), due to the deformable body representation of all bodies. These two statements seem to cause a contradiction.

If we look closer, we see there is no contradiction. The “rigid” property of bodies in the model is not something which can exist in physics, it is an idealization to simplify computation. Thus, the material properties are those of a deformable material (any material is deformable), but instead of explicitly computing the behavior of the body, we assume that for certain types of materials, the behavior will be close enough to the motion produced by the rigid body approximation. Thus, the force interaction will be that of a deformable body, while the motion will be that of a rigid body.

5 Implementation

This new “merged” model can be implemented using two separate self-contained implementations, one for each of the sub-models, and then an implementation of an interface layer.

5.1 Implementation of the Deformable Body Model

We use the implementation described in [11] and [13], which implements the volumetric mass-spring model previously described. It uses the forward Euler method for integrating the differential equation system, and a hierarchical partitioning algorithm for determining intersection between the elements. It allows us to produce a volumetric mass-spring representation from a b-rep shape, through discretization.

5.2 Implementation of the Rigid Body Model

We use the “Dynamo” implementation described in [4]. It is a typical rigid body motion simulation system. We specify the position, orientation, velocity, angular velocity, mass and inertia tensor $\tilde{I}$ to produce a rigid body representation and its initial state. We can apply forces at points of the rigid body. Due to this, and the initial state, motion is produced, which the system determines according to a specified integrator. We can then access the state at each discrete time interval in the solution process.

The convention of the system is to specify $\tilde{I}$ in the basis where the base vectors lie along the principal axes of the body (principal basis). In this basis, $\tilde{I}$ is diagonal, and treatment is eased.

Incidentally, the base vectors of the eigenspace of $\tilde{I}$ form the principal basis, and the eigenvalues of $\tilde{I}$ form the diagonal of $\tilde{I}$ expressed in the principal basis. Thus, all we have to do is find the eigenspace and the eigenvalues of $\tilde{I}$, which is a symmetric matrix. We have used Lapack++ [6], which provides these functions directly.

5.3 Implementation of the Interface

The interface is the novel part of the implementation. It is not very advanced however. It consists of a number of conversions between the deformable body representation and the rigid body representation:

(1) Initially, we require a conversion between a set of elements to a rigid body representation. This is done only once for each body, in the initialization stage of the simulation state.
(2) The element interactions produce forces on the elements, which need to be converted into the rigid body force application context.
(3) Finally, when the state of a rigid body has changed due to the applied forces, a conversion back to the element representation is required.

For each body, we will have two bases: one global basis G, which the original elements are expressed in, and one local basis P, which is the principal basis of the body. We will denote a vector $\vec{u}$ in the global basis G as $[\vec{u}]_G$, and in the local principal basis P as $[\vec{u}]_P$.

The rigid body calculations for each body will be done in P, local to each body, while the deformable body calculations will be done in G. This means we will have to convert the relevant attributes between the bases.

5.4 Initialization Conversion

We are given a set of elements E. For this procedure, they can be treated as pure point masses. It is trivial to find the rigid body properties from a set of point masses [16]. All we have to do is convert the position $[\vec{p}]_G$ of each element to $[\vec{p}]_P$, and then find the rigid body properties. Thus, we now have a rigid body B_r.

We do not perform conversion of element velocities in the initialization, since there may not exist a consistent conversion to a rigid body velocity/angular velocity. Instead we require a separate specification of the initial velocity/angular velocity of the body.

5.5 *Force Conversion*

Each element in E of B_r has a force $[\vec{f}]_G$ applied on its position $[\vec{p}]_P$. We convert the force and position, and apply the force on B_r. Thus, we apply $[\vec{f}]_P$ on point $[\vec{p}]_P$ of B_r,

5.6 *Rigid Body State Conversion*

During the motion of the rigid body, the orientation O and the position of the center of mass $\vec{p}$ of the rigid body change. Thus, what we need to do is convert these changes back into the deformable body representation.

5.6.1 *Position*

The rigid body model implementation operates purely in the basis P. $[O]_P$ is a transformation from the original orientation of the elements in B_r, and $[\vec{p}]_P$ represents a translation in the same way. Thus, if O is a transformation matrix, and $\vec{p}_e$ the position of an element e in B_r, then the position of e in the deformable model is $[O]_G[\vec{p}_e]_G + [\vec{p}]_G$.

5.6.2 *Velocity*

If we also need to know the velocity of the converted elements, the velocity needs to be computed. The velocity and angular velocity of the rigid body is $[\vec{v}]_P$ and $[\vec{\omega}]_P$ respectively. Thus, the velocity $[\vec{v}_e]_G$ of an element e in the deformable model is: $[\Big[\vec{\omega}]_P \times [\vec{p}_e]_P + [\vec{v}]_P\Big]_G$

6 Applications

6.1 *Simulation Optimization*

The main advantage of this model is complexity reduction in the simulation, caused by the rigid body approximation. If simulation time is not an important issue, we should not use the rigid body approximation, and instead simulate the more correct behavior using the deformable body model.

Additionally, we can dynamically select which mechanics model should be used for a given aggregate of elements. We can select periods of time in a simulation when deformation is negligible, and thus optimize these periods by using the rigid body model for the selected aggregates.

6.2 Example

We have created a situation where a beam is resting on another fixed beam, elevated over the ground. The ends of the top beam are free, thus we have something similar to a lever/fulcrum. We drop a heavy deformable body (a cow shape) on one of the free ends, creating an impact. The intent is to have a stiff top beam, or it will not function well as a lever. However, we also want the computation time to be as fast as possible, creating a constraint on the stiffness property of the beam.

The simulation state is such that if we increase the stiffness significantly of the top beam, the numerical integration will diverge, and the result will be meaningless. We thus have to increase the time discretization resolution to achieve our intended results. Instead, we can approximate the top beam by a rigid body.

In figure 2, we see output frames from a simulation where the beam is deformable. We can clearly see that the stiffness is not enough to create a practical lever. In figure 3, the beam is approximated by a rigid body, and the stiffness of the beam can now also be increased (it is also used for contact/collision with other bodies). We can see that now the beam behaves like a lever.

The deformable model does not make a distinction in the quantitative simulation how bodies are defined, the “body” definition is not necessary at the quantitative level in fact. Thus, an element interaction inside a body and between two bodies is exactly the same interaction. This means that we can arbitrarily define a set of elements to be a rigid body. If the top beam is a deformable body, we can select one half of it, and make that a rigid body. In figure 4, the right half of the top beam is a rigid body. In this particular example, this part-body method has no specific purpose, only as a demonstration. In some applications, such as simulation of multi-material bodies (the soft tissue/bone tissue of a human body part for instance), it can be advantageous.

Finally, in figure 5, we have mapped the geometric boundary representations to their respective physical bodies [11], for the same part-body situation. While not always necessary, it can be important in geometric modeling applications, or in applications where visualization is important.

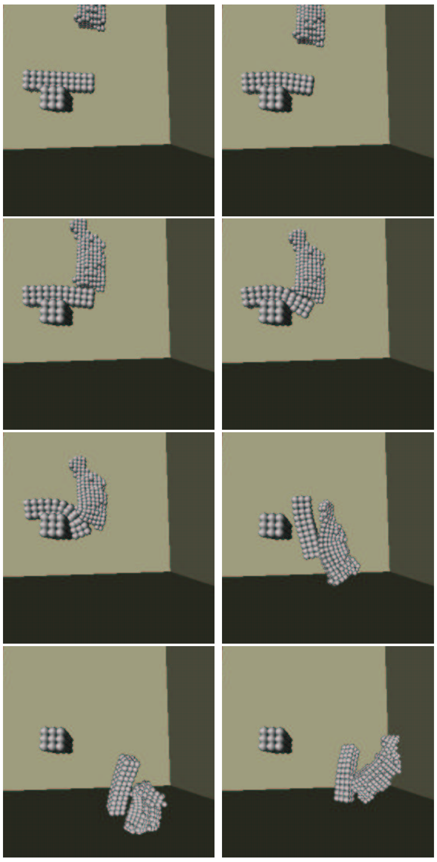

Fig. 2. The top beam is a deformable body.

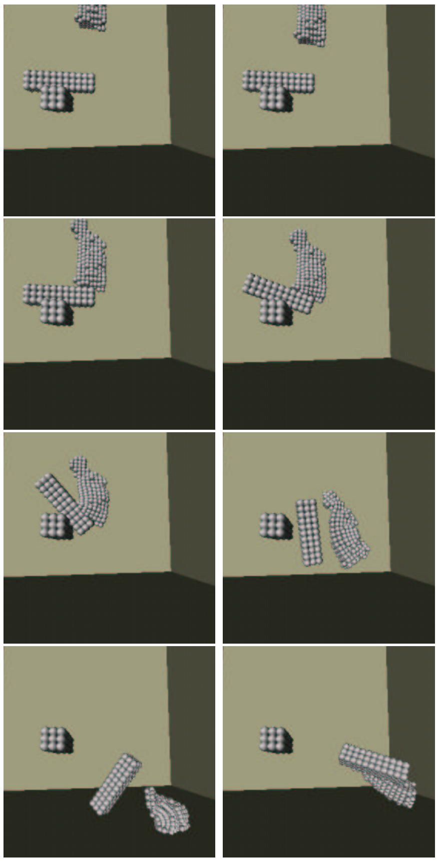

Fig. 3. The top beam is a rigid body.

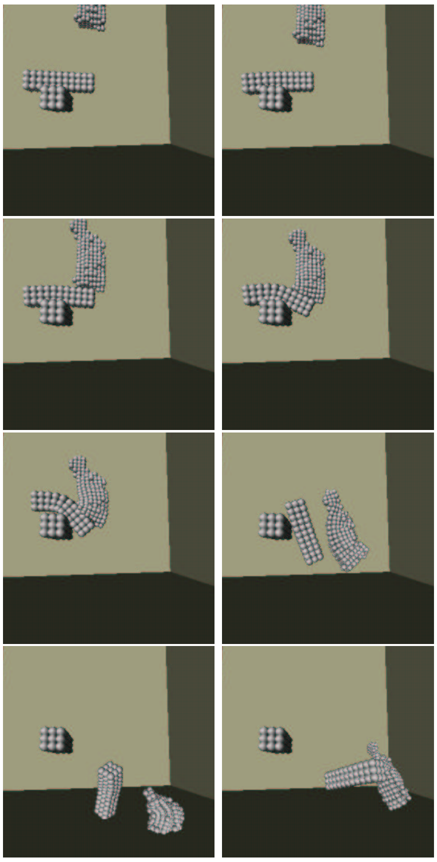

Fig. 4. The right half of the top beam is a rigid body.

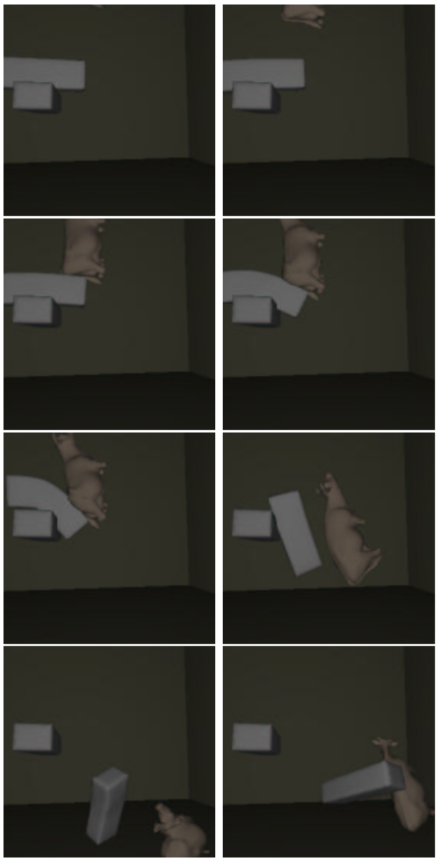

Fig. 5. The original geometric boundary representation is mapped to the bodies.

6.3 Possible Application Fields

Conceptual Mechanical Design Work has been done on integrating multi-resolution quantitative deformable body simulation in a conceptual design system [12]. This rigid body interface could ease simulation in certain design fields or situations, such as the lever example shown, or even further in mechanisms with certain deformable components.

Conceptual Geometric Modeling Interactive mechanics simulation can be used as a tool in geometric modeling, to replace or assist control point manipulation in NURBS for example. The rigid body interface can ease creating virtual manipulation "tools" in the simulation [11] [13], and can also be used to localize deformation (by making the rest of the target body rigid).

Computer Animation/Visualization In any situation where deformable body simulation is used, the rigid body interface can be used for optimization of stiff bodies.

6.4 Integration with Constraints

Many rigid body models and simulation systems include constraint functionality. With the rigid body interface, this functionality can be used in the deformable body model without further modifications. With the part-body method, we can even impose constraints on individual elements. This application has not been tested however.

7 The Future

Baraff and Witkin show in [2] how to efficiently solve a mass-spring type system of differential equations using the backward Euler method, the implicit variant of the Euler method. This method is unconditionally stable [9], and thus not sensitive to stiffness. It overcomes the so far most serious limitation of mass-spring systems, which has been that the maximum step size is constrained by the stiffness of the system, produced by for example material stiffness or damping.

While the solution described by Baraff and Witkin is specific to their particular application (cloth simulation in animation), we expect that the ideas can be carried over to a more general solution method which could be used for arbitrary mass-spring systems, specifically the one described in this article. We have created a preliminary implementation which confirms this.

One consequence of this advance is that the advantages of using a rigid body model over a mass-spring model are diminished. It is still unclear exactly how far the implicit integration method can be carried. While it theoretically allows arbitrarily stiff materials, there is presumably a relation between stiffness and computation time for the numerical algorithms which the method requires. It remains to be seen how much of the application space occupied by the rigid body model the mass-spring model with implicit integration will displace.

There are also consequences for the implementation of the interface described in this article. For the case of explicit integration, the rigid and deformable models can be kept completely separate. Implicit integration requires the solution of a nonlinear equation system involving the equations of motion of all elements. This would presumably require a certain degree of integration between the rigid and deformable models.

8 Conclusion

We have created an interface between a volumetric mass-spring model, and a rigid body model, to produce a merged model. We performed an informal experiment where we represented a beam in several ways to examine the possibilities of the interface. Results show that the interface can successfully be used for approximating stiff deformable bodies by rigid bodies in deformable body mechanical systems.

References

[1] Baraff, D., Witkin, A., “Dynamic Simulation of Non-Penetrating Flexible Bodies”. In *Computer Graphics* (Proc. SIGGRAPH) volume 26, pp. 303-308 (1992).

[2] Baraff, D., Witkin, A., “Large Steps in Cloth Simulation”. In *Computer Graphics* (Proc. SIGGRAPH 98) pp. 43-54 (1998).

[3] Baraff, D., Witkin, A., “Partitioned Dynamics”. Tech. Report CMU-RI-TR-97-33, Robotics Institute, Carnegie Mellon University. (1997).

[4] Barenbrug, B., “Designing a Class Library for Interactive Simulation of Rigid Body Dynamics”. Ph. D. thesis, Technical University of Eindhoven (2000).

[5] Chen, “Physically-based Animation of Volumetric Objects”. Proceedings of *IEEE Computer Animation ’98*, pp. 154-160 (1998).

[6] Dongarra, J. Pozo, R., Walker, D., “LAPACK++: A Design Overview of Object-Oriented Extensions for High Performance Linear Algebra”. Proceedings of Supercomputing ‘93, pages 162-171, IEEE Computer Society Press, (1993).

[7] Ganovelli, F., Cignoni, P., Montani, C., Scopigno, R., “A Multiresolution Model for Soft Objects Supporting Interactive Cuts and Lacerations”. *Eurographics 2000*, Volume 19, Number 3 (2000).

[8] Goldstein, H., “Classical Mechanics”, Addison-Wesley, Reading, MA. (1950).

[9] Heath, M., “Scientific Computing”. The McGraw-Hill Companies, Inc. (1997).

[10] Horváth, I., Kuczogi, G., Staub, G., “Spatial Behavioural Simulation of Mechanical Objects”. *Proceedings of TMCE ’98 Tools and Methods of Concurrent Engineering Symposium*, pp. 221-223 (1998).

[11] Jansson, J., Vergeest, J. S. M., “A General Mechanics Model for Systems of Deformable Solids”. Proceedings International Symposium On Tools and Methods for Concurrent Engineering 2000 (2000).

[12] Jansson, J., Horváth, I., “Behavioral Simulation of Incomplete Representations in Conceptual Design”. Proceedings European Simulation Multi-Conference 2001 (2001) (To be published).

[13] Jansson, J., Horváth, I., Vergeest, J. S. M., “Implementation and Analysis of a Mechanics Simulation Module for Use in a Conceptual Design System”. Proceedings ASME Design Engineering Technical Conferences 2000 (2000).

[14] Kang, H., Kak, A., “Deforming Virtual Objects Interactively in Accordance with an Elastic Model”. *Computer Aided Design* (1996).

[15] Keller, H., Stolz, H., Ziegler, A., Bräunl, T., “Virtual Mechanics - Simulation and Animation of Rigid Body Systems”. *Computer Science Report* No. 8/93 (1993). `http://www.ee.uwa.edu.au/~braunl/aero/ftp/docu.english.ps.gz`

[16] Kleppner, D., Kolenkow, R., “An Introduction to Mechanics”, pp. 91. McGraw-Hill (1978).

[17] O’Brien, J., Zordan, V., Hodgins, J., “Combining Active and Passive Simulations for Secondary Motion”. *IEEE Computer Graphics and Applications* Vol. 20, No. 4, pp. 86-96 (2000).

[18] Provot, X. “Deformation Constraints in a Mass-Spring Model to Describe Rigid Cloth Behavior”. Proceedings *Graphics Interface* ’95, pp. 147-154 (1995).

[19] Terzopoulous, D., Platt, J., Barr, A., Fleischer, K. “Elastically Deformable Models”. SIGGRAPH’87 pp. 205-214 (1987).