Decomposing the Contact Linear Complementarity Problem into Separate Contact Regions

Olexiy Lazarevych
lazarevych@vision.ee.ethz.ch

Gabor Szekely
szekely@vision.ee.ethz.ch

Matthias Harders
mharders@vision.ee.ethz.ch

Computer Vision Laboratory, ETH Zurich
Sternwartstrasse 7, CH-8092 Zurich, Switzerland

ABSTRACT

We present a novel approach to handling frictional contacts for deformable body simulations. Our contact model allows to separate the contact area into a set of detached contact regions. For each of them a separate mixed linear complementarity problem (MLCP) is formulated. Parallel processing of these independent contact regions may considerably improve the performance of the contact handling routine. Moreover, the proposed contact model results in sparse matrix formulation of the corresponding MLCP in the individual contact regions. For solving the MLCPs we propose an iterative method which combines the projected conjugate gradient approach and the projected Gauss-Seidel method.

Keywords

Linear complementarity problem, contact force, deformable object.

1 INTRODUCTION

Contact handling of interacting solid objects is a common research topic, for instance in computer animation or surgical simulation. Physically plausible responses to collisions and contacts potentially enrich the animation, especially if frictional effects are taken into account. Contact response methods aim at computing a set of contact forces that prevent the simulated objects from interpenetrating, while taking into account friction.

Several approaches have been proposed in the field of computer graphics and simulations to handle contacts. The majority of these can be split into two classes: penalty-based and constraint methods (note that further approaches exist, e.g. impulse methods). Penalty methods compute virtual spring forces that drive the interacting objects apart. The values of these forces are usually considered to be proportional to a geometrical measure of the interpenetration of the interacting bodies [HTK*04, KMH*04, HVS*09]. Therefore, penalty based methods not only allow interpenetrations but essentially depend on them. Despite the lack of physical plausibility caused by this simplified contact model, they are still widely used because of the simplicity of their implementation and high computational efficiency.

In contrast, constraint methods aim at following the geometrical restrictions of non-penetration of the interacting objects based on their relative position and orientation [Bar89, DAK04, PPG04, Erl07]. The resulting system of equations can be solved by a large variety of methods among which the most preferable are fast iterative procedures. However, for complex systems which consist of many interacting objects the computation time of this approach becomes quickly prohibitive. Therefore, much effort is made to develop efficient algorithms [Bar96, GBF03, KEP05, KSJP08, OTSG09, HVS*09].

Contributions. We propose a new approach to resolving contacts for deformable objects by splitting the contact area into separate, independent regions. The deformation model together with the time-integration scheme we use allows the separate treatment of detached contact regions. Handling a number of local contacts instead of a single global contact system gives a significant gain in performance even without using parallel computation techniques. The proposed contact model results in a simple diagonal mass matrix as well as sparse constraint matrices.

In addition, we propose a novel iterative scheme for the mixed contact linear complementarity problems which combines a projected conjugate gradient method with the widely used projected Gauss-Seidel method. Although, the performance in our current implementation is not better than for the normal projected Gauss-Seidel method, our scheme demonstrated more stable convergence behavior and therefore was more reliable.

2 RELATED WORK

Constraint methods are widely used in computer graphics as well as in computational mechanics due to their physical correctness. The theoretical basis of the underlying mechanics and related contact problems are thoroughly discussed by Stronge [Str90] and Wriggers[Wri02]. Classical works in constraint based dynamics in computer graphics are by Baraff [Bar89, BW92] and Witkin [Wit97].

Constraint based approaches for contact problems usually employ Signorini's law [WP99] of unilateral contact resulting in the formulation of the contact linear complementarity problem (LCP) [AP97]. Lagrange multipliers belong to the most widely used solution approaches for this kind of problems [WP99]. The LCP formulation in contact handling is used for obtaining contact responses between rigid bodies [Cat05, Erl07] or deformable objects [DAK04, DDKA06, OG07], as well as in cloth simulations [VMT97, VT00, HB00].

General approaches to the LCP solution can be split into two classes: direct and iterative methods [CPS92]. Although direct methods, *e.g.* Lemke's algorithm, Danzig's method, and other pivoting techniques [Cot90, CPS92, Mur88] are designed to give precise solutions, they are computationally demanding and slow. Therefore, in computer graphics applications almost exclusively iterative methods are used. Iterative methods for the LCP follow the scheme similar to the one used to solve a linear system of equations [CPS92, Mur88]. Therefore, projected versions of well-known iterative methods such as Jacobi, successive overrelaxation, and its special case – Gauss-Seidel – are used [Cat05, Erl07]. They work very well for rigid body simulations, however, applied to deformable body collisions they become computationally very expensive. Attempts to find a compromise were presented in [PPG04, DDKA06].

Many researchers are working on optimization and improvement of the performance of these basic iterative methods in different application areas. Exploiting the sparsity of the matrices involved in computations is one of the basic optimization approaches which works for almost any underlying model of simulated objects [GL89]. Other more sophisticated algorithms consider the LCP formulation tightly linked with the dynamical model. Baraff and Witkin employed implicit integration methods for large time step simulations of cloth [BW98]. Otaduy et al. [OTSG09] proposed an iterative solver that includes two nested relaxation loops (based on the constraint anticipation introduced in [Bar96]).

Using the conjugate gradient method for general LCP was proposed by researchers in the area of computational mechanics, like Renouf and Alart [RA05], and Li et al. [LNZL08]. We explore the combination of the projected conjugate gradient approach with the projected Gauss-Seidel method.

3 DEFORMABLE CONTACT MODEL AND MLCP FORMULATION

In simulations of scenes with many interacting deformable objects, numerous pairs of objects or parts of the same object may be simultaneously in contact. The deformable nature of the simulated material provides non-instantaneous spreading of the contact forces from the contact area into the physical body. Therefore, simultaneous but spatially separated contacts may be considered independently as their effect spreads over the objects in contact during future simulation time steps. This is in contrast to rigid body simulations where all contacts have to be taken into account to correctly compute the reaction of the object. Following this reasoning we take advantage of considering spatially separated contacts between deformable objects independently. This should speed up the contact response computations in the simulations.

In our simulations deformable objects are represented as tetrahedralized meshes with mass points located in the nodes. Each object has a triangulated surface and contacts are treated between basic surface elements: point-triangle and edge-edge pairs. Point-edge and point-point contacts are treated as special cases of point-triangle contacts. For the sake of simplicity we omit edge-edge contacts and consider only point-triangle pairs in the further discussion.

Constraints Formulation

In the absence of friction the only constraint for the point-triangle collision is that contact points cannot penetrate planes of the corresponding contact triangles. Mathematically this can be described by the condition of non-negativity of the function $C(\mathbf{p}_0,\mathbf{p}_1,\mathbf{p}_2,\mathbf{p}_3)$ of the coordinates of the corresponding mass points.

$$C(\mathbf{p}_0,\mathbf{p}_1,\mathbf{p}_2,\mathbf{p}_3) = -((\mathbf{p}_1-\mathbf{p}_0)\times(\mathbf{p}_2-\mathbf{p}_0))\cdot(\mathbf{p}_3-\mathbf{p}_0) \tag{1}$$

The time derivative of this function gives the Jacobian matrix of the normal contact constraints.

$$\dot{C}(\mathbf{p}_0,\mathbf{p}_1,\mathbf{p}_2,\mathbf{p}_3) = \mathbf{J}_n\cdot\mathbf{u} \tag{2}$$

where $\mathbf{u} = \left[\mathbf{v}_0^T\,\mathbf{v}_1^T\,\mathbf{v}_2^T\,\mathbf{v}_3^T\right]^T$ is a generalized velocity vector of the corresponding points.

The principle of virtual work requires orthogonality of the constraint force and the constraint. Therefore, in the frictionless case for our model the constraint force is defined as

$$\mathbf{f}_n = \mathbf{J}_n^T\cdot\lambda_n \tag{3}$$

where the Lagrange multiplier λ_n is to be found.

According to Signorini's contact law [WP99] at a unilateral contact the following compementarity conditions have to be satisfied.

$$\mathbf{w}_n = \mathbf{J}_n \cdot \mathbf{u} \geq 0, \qquad \lambda_n \geq 0, \qquad \mathbf{w}_n \cdot \lambda_n = 0 \qquad (4)$$

The conditions (4) pose a linear complementarity problem (LCP) for a frictionless unilateral contact.

In general, if N mass-points are involved in contacts with K constraints, the Jacobian of the whole system is easily assembled from the Jacobians of each individual constraint. Therefore, the global Jacobian consists of K lines of blocks $\mathbf{J}_q^0$, $\mathbf{J}_q^1$, $\mathbf{J}_q^2$ and $\mathbf{J}_q^3$, where $q = 1,\ldots,K$. Note, that in each line only the entries corresponding to the mass-points involved in the q-th contact are non-zero. This way the Jacobian of the contact system has the dimension $K \times 3N$.

Separation of the Contact Regions

The time integration scheme of the simulations uses the net force of the internal, global (*e.g.* gravitational), and contact forces to compute position and velocity of each simulated contact point at the next time step. Thus, a force applied to a particular mass point in the current time step will influence its neighbors only in the next time step through internal deformation.

The nature of the time-integration scheme and the discretized model of simulated objects allows us to separate two contact areas if they do not have any common simulated mass points simultaneously involved in constraint equations of both contacts. As will be shown later, this way the amount of computations becomes significantly smaller and the convergence rate for each individual contact problem increases.

The separation of the contact areas is performed by analysis of the constraint matrix $\mathbf{J}_n$ which consists of the rows related to the normal contact constraints only. The element j_{ki} of the matrix is non-zero if and only if the i-th mass point is involved in the k-th constraint. Therefore, the area separating algorithm efficiently extracts sets of rows such that each pair of the sets does not have any non-zero elements in the same columns simultaneously. In terms of the contact graph of the current configuration which is encoded by the Jacobian, the region separation algorithm aims at finding a set of disconnected subgraphs.

Currently, a basic sequential algorithm is used to assign each contact to a contact region. Contacts corresponding to a line of the Jacobian $\mathbf{J}_n$ are assigned to a particular region, such that any two different contact regions do not have contacts that share a simulated mass point. Thus, contacts that involve the same mass point belong to the same contact region. The outline of the contact region separation is presented in Algorithm 1. Here, $Contact[i][j]$ contains the index of the j-th point on the i-th contact, $i = 1,\ldots,K$, $j = 1,\ldots,4$ and $\{Contact[i]\}$ is the set of points that belong to the i-th contact. $Area[i]$ contains the index of the detached region to which the point i belongs. Note that more advanced, *e.g.* parallel, algorithms could be applied in this stage. Moreover, it should be mentioned that we consider contacts of deformable objects which usually are maintained over a number of successive simulation time steps, even in dynamic scenes. Thus, information about contact regions could be stored and updated on successive time steps as required.

Algorithm 1 Contact region separation

```
nextIndex ← 1
CheckedPointSet ⇐ ∅
for i = 1 to K do
    if Area[i] not assigned then
        Area[i] ← nextIndex++
        CheckedPointSet ⇐ {Contact[i]}
        for j = i+1 to K do
            if Area[j] is assigned then
                continue
            endif
            if {Contact[j]} ∩ {Area[i]} ≠ ∅ then
                Area[j] = Area[i]
            endif
        endfor
    else
        for l = 1 to 4 do
            if Contact[i][l] ∉ CheckedPointSet then
                for j = i+1 to K do
                    if Area[j] > 0 then
                        continue
                    endif
                    if {Contact[j]} ∩ {Area[i]} ≠ ∅
                    then
                        Area[j] = Area[i]
                    endif
                endfor
                CheckedPointSet ⇒ Contact[j][l]
            endif
        endfor
    endif
endfor
```

Including Frictional Contact

Classically the frictional part of the contact force lying in the plane of the contact triangle is introduced having two components along two orthogonal vectors $\mathbf{e}_1$ and $\mathbf{e}_2$ [Bar94]. In the frame of our contact model the part of the Jacobian responsible for friction is

$$\begin{bmatrix} \mathbf{J}_{e_1} \\ \mathbf{J}_{e_2} \end{bmatrix} = \begin{bmatrix} -\mathbf{e}_1^T & \alpha\,\mathbf{e}_1^T & \beta\,\mathbf{e}_1^T & \gamma\,\mathbf{e}_1^T \\ -\mathbf{e}_2^T & \alpha\,\mathbf{e}_2^T & \beta\,\mathbf{e}_2^T & \gamma\,\mathbf{e}_2^T \end{bmatrix} \qquad (5)$$

where (α,β,γ) are barycentric coordinates of the contact point at the time of collision.

Coulomb's friction model is often approximated by a 4-sided [Bar94] (in general, k-sided [KEP05, DDKA06]) pyramid with faces parallel to the orthogonal vectors $\mathbf{e}_1$ and $\mathbf{e}_2$. This friction model leads to the following conditions to be satisfied at the contact.

$$\begin{aligned} \mathbf{J}_{e_i}\cdot\mathbf{u} > 0 \quad &\Rightarrow \quad \lambda_{e_i} = -\mu\lambda_n \\ \mathbf{J}_{e_i}\cdot\mathbf{u} < 0 \quad &\Rightarrow \quad \lambda_{e_i} = \mu\lambda_n \\ \mathbf{J}_{e_i}\cdot\mathbf{u} = 0 \quad &\Rightarrow \quad \lambda_{e_i} \in [-\mu\lambda_n; \mu\lambda_n] \end{aligned} \tag{6}$$

where $i = 1, 2$ and μ is the friction coefficient.

In addition, we also tested a friction cone model which more precisely follows Coulomb's law. We project the solution onto the friction cone domain. If the tangential component of the contact force is larger than $\mu\lambda_n$ then we scale the friction components to fit the friction cone without changing the direction of the friction force.

$$||\lambda_{e_1}\mathbf{e}_1 + \lambda_{e_2}\mathbf{e}_2|| > \mu\lambda_n \Rightarrow \begin{cases} \lambda_{e_1} \leftarrow \frac{\lambda_{e_1}\cdot\mu\lambda_n}{||\lambda_{e_1}\mathbf{e}_1+\lambda_{e_2}\mathbf{e}_2||} \\ \lambda_{e_2} \leftarrow \frac{\lambda_{e_2}\cdot\mu\lambda_n}{||\lambda_{e_1}\mathbf{e}_1+\lambda_{e_2}\mathbf{e}_2||} \end{cases} \tag{7}$$

For a single point-triangle frictional contact the complementary conditions (4) together with (6) or (7) have to be satisfied. The general Jacobian of the system is built in the same way as in the frictionless case. The dimension of the matrix is $3K \times 3N$.

Dynamics Formulation

After separating the contact area into detached contact regions we formulate and solve the dynamic equations for each of the regions independently. In the following discussion we consider a part of the simulated system which corresponds to a particular contact region C. This part consists of the mass points involved in the contacts of that specific region. The simulated system obeys the following equation of motion.

$$\mathbf{M}_C\cdot\mathbf{u}_C = \mathbf{J}_C^T\cdot\lambda_C + \mathbf{f}_C \tag{8}$$

where $\mathbf{M}_C$ is the mass matrix of the system, $\lambda_C = (\lambda_{n,j_1}\lambda_{e_1,j_1}\lambda_{e_1,j_1}\ldots\lambda_{n,j_k}\lambda_{e_1,j_k}\lambda_{e_1,j_k})^T$ – the generalized vector of contact forces for the region, and $\mathbf{f}_C = (\mathbf{f}_1^T\mathbf{f}_2^T\ldots\mathbf{f}_l^T)^T$ – the generalized vector of non-contact forces acting on each mass point. k and l are the number of constraints and mass points of the contact region C, respectively.

We employ the forward Euler integration scheme to relate the unknown general velocity at time $t+\Delta t$ to the known velocity at the previous time step t. For deformable object collisions we employ Newton's rule for changes of the normal component of velocity after the collision [Str90], *i.e.* $\frac{v_{reflected}}{v_{incident}} = \kappa$.

$$\mathbf{u}_C(t+\Delta t) = (1+\kappa)\mathbf{u}_C(t) + \mathbf{M}_C^{-1}\mathbf{J}_C^T\cdot\lambda_C\Delta t + \mathbf{M}_C^{-1}\cdot\mathbf{f}_C \tag{9}$$

By pre-multiplying (9) with $\mathbf{J}_C$ we connect the dynamics equation with the complementarity conditions (4) and (6) discussed above.

$$\mathbf{w}_C = \mathbf{J}_C\cdot\mathbf{u}_C(t+\Delta t) = \mathbf{A}\cdot\lambda_C + \mathbf{b} \tag{10}$$

where

$$\mathbf{A} = \mathbf{J}_C\mathbf{M}_C^{-1}\mathbf{J}_C^T \tag{11}$$

$$\mathbf{b} = (1+\kappa)\mathbf{J}\cdot\mathbf{u}_C(t) + \mathbf{J}_C\cdot\mathbf{M}_C^{-1}\cdot\mathbf{f}_C \tag{12}$$

Note, that we included the factor Δt into λ_C and therefore λ_C is no longer the force but the impulse vector.

The above equations (11) and (12) together with general complementarity condition (6) or (7) constitute the MLCP that has to be solved for the values of the contact force components λ_C.

Unlike the usual formulation of the dynamics equations we explicitly consider only mass-points involved in each contact. Therefore, the generalized velocity vector does not include the angular velocity of the contact triangle and the mass matrix does not include 3×3 blocks corresponding to inertia tensors. This formulation provides a strictly diagonal form of the matrix $\mathbf{M}$ allowing optimized matrix multiplications.

Each line of the constraint matrix $\mathbf{J}_C$ consists of four 3×3 blocks. However, if the matrix $\mathbf{J}_C$ is stored in a suitable reduced format [GL89, Cat05], the calculations of $\mathbf{J}_C\mathbf{M}_C^{-1}\mathbf{J}_C^T$ can be done very efficiently in linear time.

4 ITERATIVE METHODS FOR LCP

Here, we leave aside the underlying dynamics and consider iterative methods for solution of the LCP(**A**,**b**)

$$\begin{aligned} \mathbf{A}\cdot\lambda - \mathbf{b} &> 0 \\ \lambda &> 0 \\ (\mathbf{A}\cdot\lambda - \mathbf{b})\cdot\lambda &= 0 \end{aligned} \tag{13}$$

Projected Gauss-Seidel Iterative Method

A general splitting scheme for iterative LCP solving is described in [CPS92]. By splitting the matrix **A** of the LCP(**A**,**b**) in different ways, iterative schemes similar to those for systems of linear equations are obtained. The projected Gauss-Seidel method is derived by splitting $\mathbf{A} = \mathbf{L}+\mathbf{D}+\mathbf{U}$, where **L** , **D** and **U** are the strictly lower, diagonal, and strictly upper matrix components of **A**.

According to the iterative scheme for solving the LCP(**A**,**b**) [CPS92] each iteration cycle consists of two steps. In the first a new approximation of the solution is found

$$\lambda_{k+\frac{1}{2}} = (\mathbf{L}+\mathbf{D})^{-1}\cdot(\mathbf{b} - \mathbf{U}\cdot\lambda_k) \tag{14}$$

In the second step this approximation is projected onto the set of feasible solutions.

$$\lambda_{k+1} = \max\left\{0, \lambda_{k+\frac{1}{2}}\right\} \tag{15}$$

Although, the projected Gauss-Seidel method demonstrates only first-order convergence, its computational efficiency and implementation simplicity have made it a common choice for many constraint based collision response methods in computer animation, *e.g.* [Cat05, DDKA06, Erl07, OTSG09].

Projected Conjugate Gradient Method

The conjugate gradient method [She94] can also be adapted for solving the LCP(**A**,**b**) [RA05]. The original conjugate gradient method has been widely used for optimization problems as well as for the solution of systems of linear and non-linear equations. For a linear system the method converges after at most n iterations, where n is the order of the system. If the method is applied to a non-linear system it gives successive approximations and is stopped if a particular condition is fulfilled, *e.g.* the residual $\mathbf{r}_{i+1}$ is less than some predefined threshold. The general scheme of the conjugated gradient method as well as its detailed analysis can be found in [She94]. Nevertheless, some specific remarks related to the application to LCP are given below.

The expression for calculating the conjugate direction

$$\mathbf{d}_{i+1} = \mathbf{r}_{i+1} + \beta_{i+1}\mathbf{d}_i \tag{16}$$

usually takes the value of the coefficient β_{i+1} from Fletcher-Reeves' formula.

$$\beta_{i+1} = \frac{\mathbf{r}_{i+1}^T \mathbf{r}_{i+1}}{\mathbf{r}_i^T \mathbf{r}_i} \tag{17}$$

However, another possible approach is to calculate β_{i+1} using Polak-Ribiere's formula.

$$\beta_{i+1} = \frac{\mathbf{r}_{i+1}^T (\mathbf{r}_{i+1} - \mathbf{r}_i)}{\mathbf{r}_i^T \mathbf{r}_i} \tag{18}$$

Analysis of both approaches in our computations showed that the Fletcher-Reeves method converged if the initial approximation was sufficiently close to the solution, whereas the Polak-Ribiere method sometimes resulted in an infinite loop. However, the latter often converged faster.

To adapt the conjugate-gradient algorithm to our specific MLCP(**A**,**b**) formulation, we add an additional projection step (15) to the general scheme. Another important modification we introduce concerns the residual. Given the current solution λ_{i+1} of the MLCP(**A**,**b**) we denote the set of feasible $\mathbf{w} = \mathbf{A}\cdot\lambda - \mathbf{b}$ as $W(\lambda_{i+1})$. Since we are interested only in solutions lying in the feasible domain, we modify the intermediate residual $\tilde{\mathbf{r}}$ by projecting its value onto the set $W(\lambda_{i+1})$.

$$\mathbf{r}_{i+1} = \mathbf{Proj}(\tilde{\mathbf{r}}_{i+1}, W(\lambda_{i+1})) \tag{19}$$

This way, the direction for searching the solution on the current iteration step is lying in the feasible domain. Moreover, if the current solution is close to the real solution then the projected residual $\mathbf{r}_{i+1}$ is close to zero, which may not be the case for $\tilde{\mathbf{r}}_{i+1}$.

We did not carry out a rigorous theoretical investigation of the convergence of the obtained projected conjugate gradient-like method, but we thoroughly tested it experimentally. The complete algorithm for the projected conjugate gradient method is summarized in Algorithm 2.

Algorithm 2 Projected conjugate gradient algorithm

$\mathbf{d}_0 \leftarrow \mathbf{b} - \mathbf{A}\cdot\lambda_0$
$\mathbf{r}_0 \leftarrow \mathbf{b} - \mathbf{A}\cdot\lambda_0$
for $i = 0$ to i_{max} **do**
  $\alpha_i \leftarrow \frac{\mathbf{r}_i^T \mathbf{r}_i}{\mathbf{d}_i \mathbf{A} \mathbf{d}_i}$
  $\tilde{\lambda}_{i+1} \leftarrow \lambda_i + \alpha_i \lambda_i$
  $\tilde{\mathbf{r}}_{i+1} \leftarrow \mathbf{r}_i - \alpha_i \cdot \mathbf{A}\cdot\mathbf{d}_i$
  $\lambda_{i+1} \leftarrow \mathbf{Proj}_{contact}(\tilde{\lambda}_{i+1})$
  $\mathbf{r}_{i+1} \leftarrow \mathbf{Proj}(\tilde{\mathbf{r}}_{i+1}, W(\lambda_{i+1}))$
  if *error is small*[1] **then**
    exit
  endif
  if *Polak-Ribiere* **then**
    $\beta_{i+1} \leftarrow \frac{\mathbf{r}_{i+1}^T(\mathbf{r}_{i+1}-\mathbf{r}_i)}{\mathbf{r}_i^T \mathbf{r}_i}$
  else
    $\beta_{i+1} \leftarrow \frac{\mathbf{r}_{i+1}^T \mathbf{r}_{i+1}}{\mathbf{r}_i^T \mathbf{r}_i}$
  endif
  $\mathbf{d}_{i+1} \leftarrow \mathbf{r}_{i+1} + \beta_{i+1}\mathbf{d}_i$
endfor

Combined Iterative Method and Termination Criteria

In order to improve the iterative search for the solution of the MLCP(**A**,**b**) we combine the projected conjugate gradient and the projected Gauss-Seidel methods. One of the advantages of using the projected conjugate gradient is its fast convergence rate during the first iteration steps. The conjugate direction is chosen for optimal convergence, and therefore this method has a clear advantage over the projected Gauss-Seidel approach at this stage. However, the convergence rate decreases while approaching the solution and the projected Gauss-Seidel method becomes more preferable. Following this consideration we perform several steps of the projected conjugate gradient method and then use the resulting solution as the initial approximation of the projected Gauss-Seidel algorithm.

As termination criteria of the iterative loops we check the values of the successive approximations of the solution $||\lambda_{i+1} - \lambda_i||$ as well as the value of the projected residual $||\mathbf{r}_{i+1}||$. If either $||\lambda_{i+1} - \lambda_i|| \le \varepsilon$ or $||\mathbf{r}_{i+1}|| \le \delta$

[1] The details of the exit criterion are discussed in the following section.

is fulfilled then the corresponding iterative loop is terminated. The error thresholds ε_{cg}, δ_{cg} and ε_{gs}, δ_{gs} for the conjugate gradient and Gauss-Seidel iterative loops respectively can be set to different values (obviously, $\varepsilon_{cg} \geq \varepsilon_{gs}$ and $\delta_{cg} \geq \delta_{gs}$).

Taking into account the physical meaning of the solution λ – in our case this is the contact impulse or force – it is reasonable to require a certain precision for each component of λ which is related to the accuracy of the computer simulation. Therefore, along with above criteria we also use

$$||\lambda_{i+1} - \lambda_i||_\infty \leq \varepsilon_\infty \tag{20}$$

as well as

$$||\mathbf{r}_{i+1}||_\infty \leq \delta_\infty \tag{21}$$

In some cases the convergence rate of both iterative methods is slow. This is presumably a consequence of the numerical properties of the matrix $\mathbf{A}$ and the limited numerical accuracy. For instance, for the projected Gauss-Seidel the convergence rate is small if $||\mathbf{L}+\mathbf{D}||$ is close to 1 [CPS92, Mur88]. In such cases the successive approximations of the solution may oscillate or even diverge. In order to prevent infinite loops we restrict the number of iteration within both phases of the combined method. The termination of the projected conjugate gradient loop is enforced after $2n$ iterations, where n is the size of the system in consideration, and the projected Gauss-Seidel loop is halted after a predefined number of N_{max} iterations.

In order to improve the precision in cases of forced termination we store the best solution approximation showing the smallest residual $\mathbf{r}$. The value is used as the outcome of the corresponding phase of the method, if it is better then the last approximation. Thus, we guarantee that the best approximation obtained in the conjugate gradient phase is taken for initializing the Gauss-Seidel phase. The final solution will correspond to the smallest residual among all of the obtained approximations. It should also be noted that according to the experimental results the portion of the cases with poor convergence, *i.e.* cases for which the iterative process did not terminate within the maximum number of iterations, is quite small – ranging from 0 to 0.9%. On the contrary, using a pure projected Gauss-Seidel method for the same simulating scenarios gave up to 3% cases with poor convergence.

5 RESULTS

In order to compare the performance of the proposed method for separated and non-separated contact treatment, several scenes were simulated.

Separated vs. Non-Separated Contact Region Handling

A scene of balls breaking a pyramid of bowling skittles with friction was used to test methods in a dynamic simulation without any resting states because of the absence of gravity. A scene of balls stacking in a bucket under gravity was used to test the methods in mostly static conditions. The number of contacts varies from 1 to ~ 45 for the dynamic scene and from 1 to ~ 80 for the static scene. Note that all objects in the simulations are (slightly) deformable.

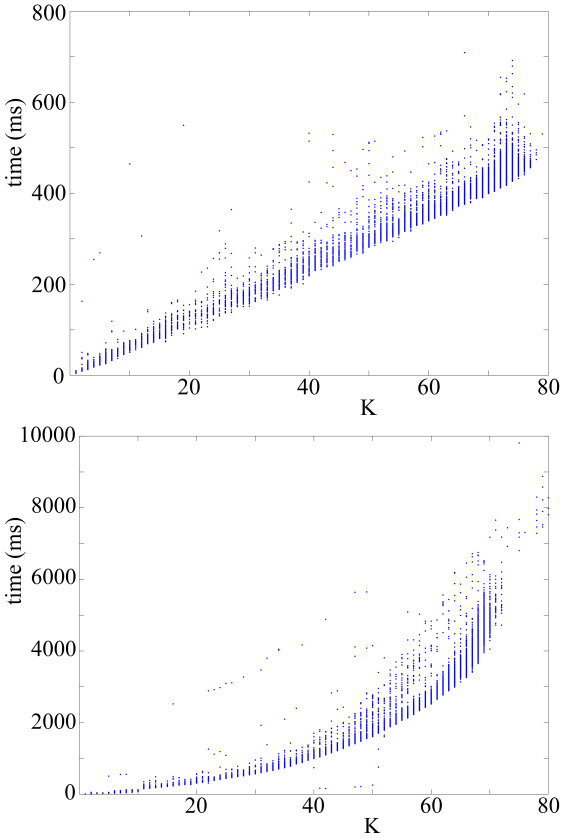

Figure 1: Static scene: Number of contacts K vs. computation time for separate (above) and non-separate (below) contact handling (the latter plot can be omitted)

The advantage of the separation of the contact area into independent regions becomes apparent for MLCPs with larger numbers of contacts. The benefit is even present if the processing of the independent regions is performed sequentially for a method of complexity $O(n^2)$. The average total computation time is $\sim 2.5 - 3$ times less for the dynamic, and $\sim 7 - 8$ times less for the static scene.

Figure 1 shows the dependency of the computation time on the number of contacts. In case of non-separated contact handling the time increases much faster than for separated contact handling. Moreover, since the independent contact regions in the latter approach have similar sizes, an almost linear growth is obtained. Note that a further possible improvement could be achieved by processing the detached contact regions in parallel.

Friction Handling

Simple static scenes of deformable objects placed on an inclined plane were used to verify the correctness of the friction handling. Experiments showed that the critical inclination angle of the plane corresponds to the friction coefficient between objects and the plane with high accuracy. Moreover, the number of separate contact areas between objects and the plane had no influence on the result. It was the same for global and separated contact area handling.

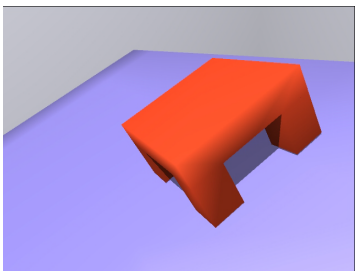

Figure 2: A table on the inclined plane

When simulating the sliding of a deformable plastic table on a plane (Figure 2), even a typical behavior found in reality could be reproduced. If the friction coefficient exceeds the critical value for the given inclination, a deformable table still can move downwards with its legs sliding in turns (*i.e.* the front legs slide while the back ones remain still, then the front legs stop and the deformation tension transfers to the back legs which start to slide until the opposite deformation tension cause them to stop and the cycle repeats). This phenomenon is a distinctive feature of certain objects made of plastic and can be easily observed in reality. It also has been described in related work dealing with contact friction [KSJP08].

Finally, both friction models were tested in more complex scenes – the 4-sided pyramid and the friction cone. The combined MLCP solving method demonstrated a considerably better performance when using the friction cone model – the convergence time decreased by $\sim 20-40\%$.

6 DISCUSSION AND CONCLUSION

We have presented an algorithm for the separation of detached contact regions in a simulated scene consisting of deformable objects. The experimental results demonstrated considerable gain in performance by using this approach. Moreover, the separate handling of the contact regions allows further acceleration by parallelization.

The presented contact model is based on simple constraint conditions and directly considers the mass points of the discretized deformable objects. This approach provides a simple diagonal mass matrix of the system which does not contain blocks related to the inertia tensors unlike most of previously proposed models. The simplicity of the mass matrix combined with the sparsity of the constraint matrix potentially allows efficient implementation of matrix computations by employing known patterns of **M** and **J**. Therefore, no auxiliary routines or modifications, *e.g.* iterative constraint anticipation [OTSG09], are needed.

We also presented an iterative method for the solution of the contact MLCP which combines the projected conjugate gradient and the widely used projected Gauss-Seidel methods.

7 ACKNOWLEDGEMENTS

This work has been performed within the frame of the EU project PASSPORT ICT-223894 and the Swiss CTI project ArthroS.

REFERENCES

[AP97] ANITESCU M., POTRA F.: Formulating dynamic multi-rigid-body contact problems with friction as solvable linear complementarity problems. *Nonlinear Dynamics 14*, 3 (1997), 231–247.

[Bar89] BARAFF D.: Analytical methods for dynamic simulation of non-penetrating rigid bodies. In *Computer Graphics, SIGGRAPH89* (1989), vol. 23, pp. 223–232.

[Bar94] BARAFF D.: Fast contact force computation for nonpenetrating rigid bodies. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques* (1994), ACM, pp. 23–34.

[Bar96] BARAFF D.: Linear-time dynamics using lagrange multipliers. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques* (1996), ACM, pp. 137–146.

[BW92] BARAFF D., WITKIN A.: Dynamic simulation of non-penetrating flexible bodies. *Computer Graphics (Proc. Siggraph) 26*, 2 (1992), 303–308.

[BW98] BARAFF D., WITKIN A.: Large steps in cloth simulation. In *Computer Graphics Proceedings, Annual Conference Series* (1998), SIGGRAPH, pp. 43–54.

[Cat05] CATTO E.: Iterative dynamics with temporal coherence. *Online Paper* (2005).

[Cot90] COTTLE R. W.: The principal pivoting method revisited. *Math. Program. 48* (1990), 369–385.

[CPS92] COTTLE R., PANG J. S., STONE R. E.: *The Linear Complementarity problem*. Academic Press, 1992.

[DAK04] DURIEZ C., ANDRIOT C., KHEDDAR A.: Signorini's contact model for deformable objects in haptic simulations. In *IROS, 2004. Proceedings.* (2004), vol. 4, pp. 3232–3237 vol.4.

[DDKA06] DURIEZ C., DUBOIS F., KHEDDAR A., ANDRIOT C.: Realistic haptic rendering of interacting deformable objects in virtual environments. *IEEE Transactions on Visualization and Computer Graphics 12*, 1 (2006), 36–47.

[Erl07] ERLEBEN K.: Velocity-based shock propagation for multibody dynamics animation. *ACM Trans. Graph. 26*, 2 (2007), 12.

[GBF03] GUENDELMAN E., BRIDSON R., FEDKIW R.: Nonconvex rigid bodies with stacking. *ACM Transaction on Graphics 22*, 3 (2003), 871–878.

[GL89] GOLUB G. H., LOAN C. F. V.: *Matrix Computations*, second ed. Baltimore, MD, USA, 1989.

[GMS04] GIROD B., MAGNOR M. A., SEIDEL H.-P. (Eds.):. *Proceedings of the Vision, Modeling, and Visualization Conference 2004* (2004), Aka GmbH.

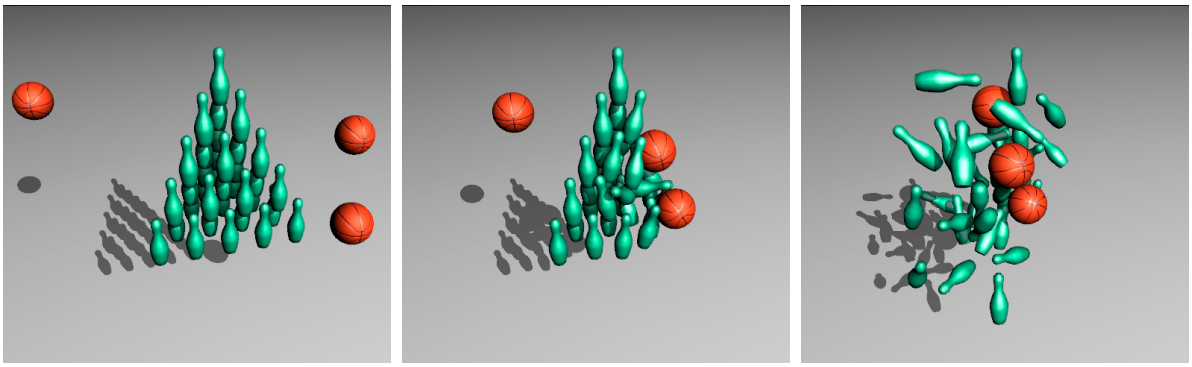

Figure 3: Dynamic scene

Figure 4: Static scene

[HB00] HOUSE D. H., BREEN D. E. (Eds.): *Cloth modeling and animation*. A. K. Peters, Ltd., 2000.

[HTK*04] HEIDELBERGER B., TESCHNER M., KEISER R., MÜLLER M., GROSS M. H.: Consistent penetration depth estimation for deformable collision response. In Girod et al. [GMS04], pp. 339–346.

[HVS*09] HARMON D., VOUGA E., SMITH B., TAMSTORF R., GRINSPUN E.: Asynchronous contact mechanics. *ACM Trans. Graph. 28*, 3 (2009), 1–12.

[KEP05] KAUFMAN D., EDMUNDS T., PAI D.: Fast frictional dynamics for rigid bodies. *ACM Trans. Graph. 24*, 3 (2005), 946–956.

[KMH*04] KEISER R., MÜLLER M., HEIDELBERGER B., TESCHNER M., GROSS M. H.: Contact handling for deformable point-based objects. In Girod et al. [GMS04], pp. 315–322.

[KSJP08] KAUFMAN D., SUEDA S., JAMES D., PAI D.: Staggered projections for frictional contact in multibody systems. In *ACM SIGGRAPH Asia 2008 papers* (2008), ACM, pp. 1–11.

[LNZL08] LI D.-H., NIE Y.-Y., ZENG J.-P., LI Q.-N.: Conjugate gradient method for the linear complementarity problem with s-matrix. *Mathematical and Computer Modelling 48*, 5-6 (2008), 918–928.

[Mur88] MURTY K. G.: *Linear Complementarity, Linear and Nonlinear Programming*, vol. 3 of *Sigma Series in Applied Mathematics*. Heldermann Verlag, 1988.

[OG07] OTADUY M. A., GROSS M.: Transparent rendering of tool contact with compliant environments. In *WHC '07: Proceedings of the Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems* (2007), IEEE Computer Society, pp. 225–230.

[OTSG09] OTADUY M., TAMSTORF R., STEINEMANN D., GROSS M.: Implicit contact handling for deformable objects. *Computer Graphics Forum (Proc. of Eurographics) 28*, 2 (2009).

[PPG04] PAULY M., PAI D., GUIBAS L.: Quasi-rigid objects in contact. In *SCA '04: Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2004), Eurographics Association, pp. 109–119.

[RA05] RENOUF M., ALART P.: Conjugate gradient type algorithms for frictional multi-contact problems: applications to granular materials. *Computer Methods in Applied Mechanics and Engineering 194*, 18-20 (2005), 2019–2041.

[She94] SHEWCHUK J. R.: *An Introduction to the Conjugate Gradient Method Without the Agonizing Pain*. Tech. rep., 1994.

[Str90] STRONGE W.: Rigid body collisions with friction. *Proceedings: Mathematical and Physical Sciences 431*, 1881 (1990), 169–181.

[VMT97] VOLINO P., MAGNENAT-THALMANN N.: Developing simulation techniques for an interactive clothing system. In *VSMM '97: Proceedings of the 1997 International Conference on Virtual Systems and MultiMedia* (1997), IEEE Computer Society, p. 109.

[VT00] VOLINO P., THALMANN N. M.: Accurate collision response on polygonal meshes. In *CA '00: Proceedings of the Computer Animation* (2000), IEEE Computer Society, p. 154.

[Wit97] WITKIN A.: Physically based modeling: Principles and practice. In *Computer Graphics* (1997), pp. 11–21.

[WP99] WRIGGERS P., PANATIOTOPOULOS P. (Eds.): *New Developments in Contact Problems*. SpringerWienNewYork, 1999.

[Wri02] WRIGGERS P.: *Computational Contact Mechanics*. John Wiley & Sons Ltd., 2002.