PROJECTED GAUSS–SEIDEL SUBSPACE MINIMIZATION METHOD FOR INTERACTIVE RIGID BODY DYNAMICS

Improving Animation Quality using a Projected Gauss–Seidel Subspace Minimization Method

Morten Silcowitz, Sarah Niebe and Kenny Erleben

eScience Center, Department of Computer Science, University of Copenhagen, Denmark

Abstract: In interactive physical simulation, contact forces are applied to prevent rigid bodies from penetrating and to control slipping between bodies. Accurate contact force determination is a computationally hard problem. Thus, in practice one trades accuracy for performance. This results in visual artifacts such as viscous or damped contact response. In this paper, we present a new approach to contact force determination. We formulate the contact force problem as a nonlinear complementarity problem, and discretize the problem to derive the Projected Gauss–Seidel method. We combine the Projected Gauss–Seidel method with a subspace minimization method. Our new method shows improved qualities and superior convergence properties for specific configurations.

1 INTRODUCTION

Most open source software for interactive real-time rigid body simulation use the widespread Projected Gauss–Seidel (PGS) method, examples are Bullet and Open Dynamics Engine. However, the PGS method is not always satisfactory, it suffers from two problems: linear convergence rate (Cottle et al., 1992) and inaccurate friction forces in stacks (Kaufman et al., 2008). Linear convergence results in viscous motion at contacts and loss of high frequency effects. The viscous appearance results in a time delay in contact responses and reduces plausibility (O'Sullivan et al., 2003). This has motivated us to develop a new numerical method, based on a nonlinear complementarity formulation of the contact force problem. The method combines the PGS method with a subspace minimization solver. The contribution is simple to implement, and existing PGS implementations can easily be extended using our ideas. Our method is compared to the PGS method for interactive simulation.

1.1 Previous Work

Rigid body simulation was introduced to the graphics community in the late 1980's (Hahn, 1988; Moore and Wilhelms, 1988), using penalty based and impulse based approaches to describe physical interactions. Penalty based simulation is not easily adopted to different simulations. Mirtich (Mirtich, 1996) presented an extended and improved impulse based formulation, however stacking was a problem and it suffered from creeping. These problems have since been rectified (Guendelman et al., 2003). Constraint based simulation (Baraff, 1994) has received much attention as an alternative to penalty based and impulse based simulation. Constraint based simulation can be divided into two groups: maximal coordinate and minimal coordinate methods (Featherstone, 1998). The focus of this paper is maximal coordinate methods, which are dominated by complementarity formulations. Alternatives to complementarity formulations are based on kinetic energy (Milenkovic and Schmidl, 2004) and motion space (Redon et al., 2003). However, the former solves a more general problem and is not attractive for performance reasons, the latter does not include frictional forces. Complementarity formulations are either acceleration based formulations (Trinkle et al., 2001) or velocity based formulations (Stewart and Trinkle, 1996). Acceleration based formulations can not handle collisions (Anitescu and Potra, 1997), in addition they suffer from indeterminacy and inconsistency (Stewart, 2000).

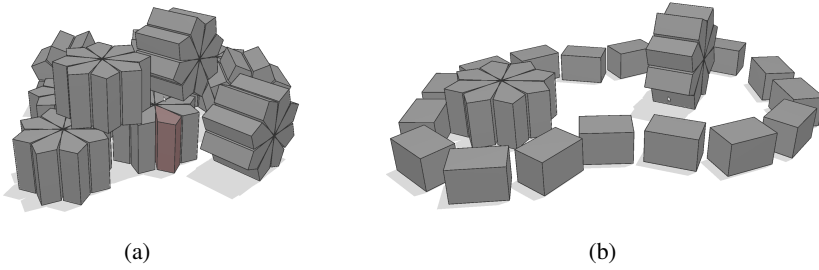(a)   (b)

Figure 1: Various configurations animated using the PGS–SM method. Configurations include composite rigid bodies, contact constraints, and hinge joint constraints with joint limits.

Velocity based formulations suffer from none of these problems, for this reason we use a velocity based formulation for this paper. The approach we present is based on reformulating the frictional problem as a nonlinear complementarity problem. The result is a slightly inaccurate model, with relatively few variables to solve for. This makes it advantageous in interactive simulations from a performance viewpoint.

Our work is inspired by (Morales et al., 2008; Arechavaleta et al., 2009). The novel contribution consists of tailoring the ideas to suit the NCP formulation, used for interactive rigid body dynamics. Further, we present optimizations to the subspace method to make the simulation fast enough for interactive usage. Prior work is limited to a linear complementarity problem formulation, missing evaluation of the ideas for general rigid body simulation. Application is limited to grasping, which is dominated by bilateral constraints and static friction. Here we evaluate with focus on both dynamic and static friction, as well as normal constraints.

2 THE NONLINEAR COMPLEMENTARITY PROBLEM FORMULATION

The frictional contact force problem can be stated as a linear complementarity problem (LCP) (Stewart and Trinkle, 1996). However, a different formulation is used in interactive physical simulations, we will derive this formulation. Without loss of generality, we will only consider a single contact point. The focus of this paper is on the contact force model, so the time stepping scheme and matrix layouts are based on the velocity-based formulation in (Erleben, 2007). We have the time discretized Newton–Euler equations,

$$\mathbf{M}\mathbf{v} - \mathbf{J}_n^T \lambda_n - \mathbf{J}_t^T \lambda_t = \mathbf{F}, \qquad (1)$$

where $\mathbf{J}_n$ is the Jacobian corresponding to normal constraints and $\mathbf{J}_t$ is the Jacobian corresponding to

the tangential contact impulses. $\mathbf{M}$ is the generalized mass matrix and $\mathbf{v}$ is the generalized velocity vector. We wish to solve for $\mathbf{v}$ in order to compute a position update. For readability we have, without loss of generality, abstracted the discretization details within the Lagrange multipliers λ_n, λ_t and generalized external impulses $\mathbf{F}$. Since the contact plane is two-dimensional, we span this plane by two orthogonal unit vectors, t_1 and t_2. Any vector in this plane can be written as a linear combination of these two vectors. Thus, $\mathbf{J}_t$ has only two rows corresponding to the two directions. From (1) we can obtain the generalized velocities,

$$\mathbf{v} = \mathbf{M}^{-1}\mathbf{F} + \mathbf{M}^{-1}\mathbf{J}_n^T \lambda_n + \mathbf{M}^{-1}\mathbf{J}_t^T \lambda_t. \qquad (2)$$

Let the Lagrange multipliers $\lambda = \begin{bmatrix} \lambda_n & \lambda_t^T \end{bmatrix}^T$ and contact Jacobian $\mathbf{J} = \begin{bmatrix} \mathbf{J}_n & \mathbf{J}_t \end{bmatrix}^T$, then we write the relative contact velocities $\mathbf{y} = \begin{bmatrix} y_n & \mathbf{y}_t^T \end{bmatrix}^T$ such that,

$$\mathbf{y} = \mathbf{J}\mathbf{v} = \underbrace{\mathbf{J}\mathbf{M}^{-1}\mathbf{J}^T}_{\mathbf{A}} \lambda + \underbrace{\mathbf{J}\mathbf{M}^{-1}\mathbf{F}}_{\mathbf{b}}. \qquad (3)$$

To compute the frictional component of the contact impulse, we need a model of friction. We base our model on Coulomb's friction law. In one dimension, Coulomb's friction law can be written as (Baraff, 1994),

$$y < 0 \Rightarrow \lambda_t = \mu\lambda_n, \qquad (4a)$$
$$y > 0 \Rightarrow \lambda_t = -\mu\lambda_n, \qquad (4b)$$
$$y = 0 \Rightarrow -\mu\lambda_n \leq \lambda_t \leq \mu\lambda_n. \qquad (4c)$$

For the full contact problem, we split $\mathbf{y}$ into positive and negative components,

$$\mathbf{y} = \mathbf{y}^+ - \mathbf{y}^-, \qquad (5)$$

where

$$\mathbf{y}^+ \geq 0, \quad \mathbf{y}^- \geq 0 \quad \text{and} \quad \left(\mathbf{y}^+\right)^T \left(\mathbf{y}^-\right) = 0. \qquad (6)$$

For the frictional impulses, we define the bounds $-l_t(\lambda) = u_t(\lambda) = \mu\lambda_n$ and for the normal impulse

$l_n(\lambda) = 0$ and $u_n(\lambda) = \infty$. Combining the bounds with (4), (5) and (6), we reach the final nonlinear complementarity problem (NCP) formulation,

$$\mathbf{y}^+ - \mathbf{y}^- = \mathbf{A}\lambda + \mathbf{b}, \tag{7a}$$

$$\mathbf{y}^+ \geq 0, \tag{7b}$$

$$\mathbf{y}^- \geq 0, \tag{7c}$$

$$u(\lambda) - \lambda \geq 0, \tag{7d}$$

$$\lambda - l(\lambda) \geq 0, \tag{7e}$$

$$\left(\mathbf{y}^+\right)^T \left(\lambda - l(\lambda)\right) = 0, \tag{7f}$$

$$\left(\mathbf{y}^-\right)^T \left(u(\lambda) - \lambda\right) = 0, \tag{7g}$$

$$\left(\mathbf{y}^+\right)^T \left(\mathbf{y}^-\right) = 0, \tag{7h}$$

where $l(\lambda) = [l_n(\lambda) \quad \mathbf{l}_t(\lambda)^T]^T$ and $u(\lambda) = [u_n(\lambda) \quad \mathbf{u}_t(\lambda)^T]^T$. The advantage of the NCP formulation is a much lower memory footprint than for the LCP formulation. The disadvantage is solving the friction problem as two decoupled one-dimensional Coulomb friction models.

3 THE PROJECTED GAUSS–SEIDEL METHOD

The following is a derivation of the PGS method for solving the frictional contact force problem, stated as the NCP (7). Using a minimum map reformulation, the ith component of (7) can be written as

$$(\mathbf{A}\lambda + \mathbf{b})_i = \mathbf{y}_i^+ - \mathbf{y}_i^-, \tag{8a}$$

$$\min(\lambda_i - l_i, \mathbf{y}_i^+) = 0, \tag{8b}$$

$$\min(u_i - \lambda_i, \mathbf{y}_i^-) = 0. \tag{8c}$$

where $l_i = l_i(\lambda)$ and $u_i = u_i(\lambda)$. Note, when $\mathbf{y}_i^- > 0$ we have $\mathbf{y}_i^+ = 0$ which in turn means that $\lambda_i - l_i \geq 0$. In this case, (8b) is equivalent to

$$\min(\lambda_i - l_i, \mathbf{y}_i^+ - \mathbf{y}_i^-) = -(\mathbf{y}^-)_i. \tag{9}$$

If $\mathbf{y}_i^- = 0$ then $\lambda_i - l_i = 0$ and complementarity constraint (8b) is trivially satisfied. Substituting (9) for $\mathbf{y}_i^-$ in (8c) yields,

$$\min(u_i - \lambda_i, \max(l_i - \lambda_i, -(\mathbf{y}^+ - \mathbf{y}^-)_i)) = 0. \tag{10}$$

This is a more compact reformulation than (7) and eliminates the need for auxiliary variables $\mathbf{y}^+$ and $\mathbf{y}^-$. By adding λ_i we get a fixed point formulation

$$\min(u_i, \max(l_i, \lambda_i - (\mathbf{A}\lambda + \mathbf{b})_i)) = \lambda_i. \tag{11}$$

We introduce the splitting $\mathbf{A} = \mathbf{B} - \mathbf{C}$ and an iteration index k. Then we define $\mathbf{c}^k = \mathbf{b} - \mathbf{C}\lambda^k$, $l^k = l(\lambda^k)$ and $u^k = u(\lambda^k)$. Using this we have

$$\min(u_i^k, \max(l_i^k, (\lambda^{k+1} - \mathbf{B}\lambda^{k+1} - \mathbf{c}^k)_i)) = \lambda_i^{k+1}. \tag{12}$$

When $\lim_{k \to \infty} \lambda^k = \lambda^*$ then (12) is equivalent to (7). Next we perform a case-by-case analysis. Three cases are possible,

$$(\lambda^{k+1} - \mathbf{B}\lambda^{k+1} - \mathbf{c}^k)_i < l_i \Rightarrow \lambda_i^{k+1} = l_i, \tag{13a}$$

$$(\lambda^{k+1} - \mathbf{B}\lambda^{k+1} - \mathbf{c}^k)_i > u_i \Rightarrow \lambda_i^{k+1} = u_i, \tag{13b}$$

$$l_i \leq (\lambda^{k+1} - \mathbf{B}\lambda^{k+1} - \mathbf{c}^k)_i \leq u_i \Rightarrow$$
$$\lambda_i^{k+1} = (\lambda^{k+1} - \mathbf{B}\lambda^{k+1} - \mathbf{c}^k)_i. \tag{13c}$$

Case (13c) reduces to,

$$(\mathbf{B}\lambda^{k+1})_i = -\mathbf{c}_i^k, \tag{14}$$

which for a suitable choice of $\mathbf{B}$ and back substitution of $\mathbf{c}^k$ gives,

$$\lambda_i^{k+1} = (\mathbf{B}^{-1}(\mathbf{C}\lambda^k - \mathbf{b}))_i. \tag{15}$$

Thus, our iterative splitting method becomes,

$$\min(u_i^k, \max(l_i^k, (\mathbf{B}^{-1}(\mathbf{C}\lambda^k - \mathbf{b}))_i)) = \lambda_i^{k+1}. \tag{16}$$

This is termed a projection method. To realize this, let $\lambda' = \mathbf{B}^{-1}(\mathbf{C}\lambda^k - \mathbf{b})$ then,

$$\lambda^{k+1} = \min(\mathbf{u}^k, \max(\mathbf{l}^k, \lambda')), \tag{17}$$

is the $(k+1)$th iterate obtained by projecting the vector λ' onto the box given by $\mathbf{l}^k$ and $\mathbf{u}^k$. Using the splitting $\mathbf{B} = \mathbf{D} + \mathbf{L}$ and $\mathbf{C} = -\mathbf{U}$ results in the Projected Gauss–Seidel method. The resulting PGS method (17) can be efficiently implemented as a forward loop over all components and a component wise projection. To our knowledge no known convergence theorems exist for (16) in the case of variable bounds $l(\lambda)$ and $u(\lambda)$. However, for fixed constant bounds the formulation can be algebraically reduced to that of a LCP formulation. In general, LCP formulations can be shown to have linear convergence rate and unique solutions, when A is symmetric positive definite (Cottle et al., 1992). However, the $\mathbf{A}$ matrix equivalent of our frictional contact model is positive symmetric semi definite and uniqueness is no longer guaranteed, but existence of solutions are (Cottle et al., 1992).

4 THE PROJECTED GAUSS–SEIDEL SUBSPACE MINIMIZATION METHOD

We will present a Projected Gauss–Seidel Subspace Minimization (PGS–SM) method, specifically tailored for the nonlinear complementarity problem formulation of the contact force problem. Unlike previous work, which is limited to the linear complementarity problem formulation, our method is more general and further specialized for interactive usage. The PGS–SM method is an iterative method, each iteration consisting of two phases. The first phase estimates a set of active constraints, using the standard PGS method. The second phase solves accurately for the active constraints, potentially further reducing the set of active constraints for the next iteration. In the following we will describe the details of the PGS–SM method.

4.1 Determining Index Sets

We define three index sets corresponding to our choice of active constraints in (17)

$$\mathcal{L} \equiv \{i | \mathbf{y}_i > 0\} \tag{18a}$$
$$\mathcal{U} \equiv \{i | \mathbf{y}_i < 0\} \tag{18b}$$
$$\mathcal{A} \equiv \{i | \mathbf{y}_i = 0\} \tag{18c}$$

assuming $\mathbf{l}_i \leq 0 < \mathbf{u}_i$ for all i. The definition in (18) is based on the $\mathbf{y}$-vector. However, one may just as well use the λ-vector, thus having

$$\mathcal{L} \equiv \{i | \lambda_i = \mathbf{l}_i\} \tag{19a}$$
$$\mathcal{U} \equiv \{i | \lambda_i = \mathbf{u}_i\} \tag{19b}$$
$$\mathcal{A} \equiv \{i | \mathbf{l}_i < \lambda_i < \mathbf{u}_i\} \tag{19c}$$

When the PGS method terminates, we know λ is feasible (although not the correct solution). However, $\mathbf{y}$ may be infeasible due to the projection on λ made by the PGS method. This votes in favor of using (19) rather than (18). In our initial test trials, no hybrid or mixed schemes seemed to be worth the effort. Therefore, we use the classification defined in (19) for the PGS–SM method.

4.2 Posing the Reduced Problem

Next we use a permutation of the indexes, creating the imaginary partitioning of the system of linear equations (3)

$$\begin{bmatrix} \mathbf{y}_{\mathcal{A}} \\ \mathbf{y}_{L} \\ \mathbf{y}_{\mathcal{U}} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{\mathcal{AA}} & \mathbf{A}_{\mathcal{AL}} & \mathbf{A}_{\mathcal{AU}} \\ \mathbf{A}_{L\mathcal{A}} & \mathbf{A}_{LL} & \mathbf{A}_{L\mathcal{U}} \\ \mathbf{A}_{\mathcal{UA}} & \mathbf{A}_{\mathcal{UL}} & \mathbf{A}_{\mathcal{UU}} \end{bmatrix} \begin{bmatrix} \lambda_{\mathcal{A}} \\ \lambda_{L} \\ \lambda_{\mathcal{U}} \end{bmatrix} + \begin{bmatrix} \mathbf{b}_{\mathcal{A}} \\ \mathbf{b}_{L} \\ \mathbf{b}_{\mathcal{U}} \end{bmatrix}. \tag{20}$$

Utilizing that $\forall i \in \mathcal{A} \Rightarrow \mathbf{y}_i = 0$, as well as $\forall i \in \mathcal{L} \Rightarrow \lambda_i = \mathbf{l}_i$ and $\forall i \in \mathcal{U} \Rightarrow \lambda_i = \mathbf{u}_i$, we get

$$\begin{bmatrix} \mathbf{0} \\ \mathbf{y}_{L} \\ \mathbf{y}_{\mathcal{U}} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{\mathcal{AA}} & \mathbf{A}_{\mathcal{AL}} & \mathbf{A}_{\mathcal{AU}} \\ \mathbf{A}_{L\mathcal{A}} & \mathbf{A}_{LL} & \mathbf{A}_{L\mathcal{U}} \\ \mathbf{A}_{\mathcal{UA}} & \mathbf{A}_{\mathcal{UL}} & \mathbf{A}_{\mathcal{UU}} \end{bmatrix} \begin{bmatrix} \lambda_{\mathcal{A}} \\ \mathbf{l}_{L} \\ \mathbf{u}_{\mathcal{U}} \end{bmatrix} + \begin{bmatrix} \mathbf{b}_{\mathcal{A}} \\ \mathbf{b}_{L} \\ \mathbf{b}_{\mathcal{U}} \end{bmatrix}. \tag{21}$$

To solve this system for $\mathbf{y}_L$, $\mathbf{y}_{\mathcal{U}}$ and $\lambda_{\mathcal{A}}$, we first compute $\lambda_{\mathcal{A}}$ from

$$\mathbf{A}_{\mathcal{AA}}\lambda_{\mathcal{A}} = -(\mathbf{b}_{\mathcal{A}} + \mathbf{A}_{\mathcal{AL}}\mathbf{l}_{L} + \mathbf{A}_{\mathcal{AU}}\mathbf{u}_{\mathcal{U}}). \tag{22}$$

Observe that $\mathbf{A}_{\mathcal{AA}}$ is a symmetric principal submatrix of $\mathbf{A}$. Knowing $\lambda_{\mathcal{A}}$, we can easily compute $\mathbf{y}_L$ and $\mathbf{y}_{\mathcal{U}}$,

$$\mathbf{y}_{L} \leftarrow \mathbf{A}_{L\mathcal{A}}\lambda_{\mathcal{A}} + \mathbf{A}_{LL}\mathbf{l}_{L} + \mathbf{A}_{L\mathcal{U}}\mathbf{u}_{\mathcal{U}} + \mathbf{b}_{L}, \tag{23a}$$
$$\mathbf{y}_{\mathcal{U}} \leftarrow \mathbf{A}_{\mathcal{UA}}\lambda_{\mathcal{A}} + \mathbf{A}_{\mathcal{UL}}\mathbf{l}_{L} + \mathbf{A}_{\mathcal{UU}}\mathbf{u}_{\mathcal{U}} + \mathbf{b}_{\mathcal{U}}. \tag{23b}$$

Finally, we verify that $\mathbf{y}_L < 0$, $\mathbf{y}_{\mathcal{U}} > 0$ and $\mathbf{l}_{\mathcal{A}} \leq \lambda_{\mathcal{A}} \leq \mathbf{u}_{\mathcal{A}}$. If this holds, we have found a solution. Rather than performing the tests explicitly, it is more simple to perform a projection on the reduced problem

$$\lambda_{\mathcal{A}} \leftarrow \min(\mathbf{u}_{\mathcal{A}}, \max(\mathbf{l}_{\mathcal{A}}, \lambda_{\mathcal{A}})) \tag{24}$$

We assemble the full solution $\lambda \leftarrow \begin{bmatrix} \lambda_{\mathcal{A}}^T & \mathbf{l}_{L}^T & \mathbf{u}_{\mathcal{U}}^T \end{bmatrix}^T$, before reestimating the index sets for the next iteration. Observe that the projection on the reduced problem will either leave the active set unchanged or reduce it further.

4.3 The Complete Algorithm

The resulting algorithm can be outlined as

```
1 :   while not convergence do
2 :     λ ← run PGS for at least k_pgs iterations
3 :     if termination criteria is passed then
4 :       return λ
5 :     endif
6 :     for k = 1 to k_sm
7 :       L ≡ {i|λ_i = l_i}
8 :       U ≡ {i|λ_i = u_i}
9 :       A ≡ {i|l_i < λ_i < u_i}
10:        solve: A_AA λ_A = −(b_A + A_AL l_L + A_AU u_U)
11:        y_L ← A_LA λ_A + A_LL l_L + A_LU u_U + b_L,
12:        y_U ← A_UA λ_A + A_UL l_L + A_UU u_U + b_U
```

```
13 :    update: (l, u)
14 :    λ_𝒜 ← min(u_𝒜, max(l_𝒜, λ_𝒜))
15 :    λ ← [λ_𝒜^T  l_ℒ^T  u_𝒰^T]^T
16 :    if termination criteria is passed then
17 :      return λ
18 :    endif
19 :   next k
20 : end while
```

An absolute termination criteria could be applied

$$\psi(\lambda) < \varepsilon_{abs} \qquad (25)$$

where ψ is a merit function to the solution of (4), and ε_{abs} is a user specified value. An alternative termination criteria could be to monitor if the set $\mathcal{A}$ has changed from previous iteration,

$$\mathcal{A}(\lambda^{k+1}) = \mathcal{A}(\lambda^k). \qquad (26)$$

A third termination criteria could be testing for stagnation

$$\psi(\lambda^{k+1}) - \psi(\lambda^k) < \psi(\lambda^k)\varepsilon_{rel}. \qquad (27)$$

for some user specified value $\varepsilon_{rel} > 0$. Other merit-functions could be used in place of ψ. Examples include natural merit functions of the Fischer reformulation (Silcowitz et al., 2009) or the minimum map reformulation (Erleben and Ortiz, 2008). We prefer the Fischer reformulation, as it seems to be more global in the inclusion of boundary information (Billups, 1995). Finally, to ensure interactive performance, one could use an absolute termination criteria on the number of iterations. Using such a criteria, the algorithm may not perform iterations enough to reach an accurate solution, we observed this behavior in a few cases. To counter this, a fall back to the best iterate found while iterating could be employed. This would ensure that the PGS–SM method behaves no worse than the PGS method would have done.

5 EXPERIMENTS

We have compared the PGS–SM method to the standard PGS method. For testing the PGS–SM method, we have selected various test cases which we believe to be challenging. The test cases are shown in Figure 2. The test cases include bilateral hinge joints with joint limits, large mass-ratios, inclined plane setups to provoke static friction handling, stacked configurations of different sizes with both box and gear geometries.

Convergence rates for all the test cases are shown in Figure 3. In order to ease comparison, great care is taken to measure the time usage of both methods in units of PGS iterations.

For the tests, we use the iteration limits $k_{pgs} = 25$ and $k_{sm} = 5$. Further we use an error tolerance of $\varepsilon_{abs} = 10^{-15}$. For the reduced problem we use a non-preconditioned Conjugate Gradient (CG) method with a maximum iteration count equal to the number of variables, and an error tolerance on the residual of $\varepsilon_{residual} = 10^{-15}$. The algorithms were implemented in Java using JOGL, and the tests were run on a Lenovo T61 2.0Ghz machine.

As observed in Figure 3, the PGS–SM method behaves rather well for small configurations and configurations with joints. For larger configurations, we obtain convergences similar to the PGS method.

The supplementary video shows interactive simulations of an articulated snake-like figure, comparing the animation quality of the PGS–SM method to the PGS method. All test cases run at interactive frame rates, 25 fps or above. We have observed a different quality in the motion simulated by the PGS–SM method. It is our hypothesis that the PGS–SM method seems to favor static friction over dynamic friction. Our subjective impression is that the PGS–SM method delivers a more plausible animation quality.

The presented algorithm is capable of very accurate computations, compared to the PGS method. However, we have observed problematic instances where simulation blow-up was noticed. The simulation blow-ups appear to occur regardless of how accurate the subspace problem is solved. We observed blow-ups even when using a singular value decomposition pseudo inverse of the reduced problem (22).

In general, if bounds are fixed the problem reduces to a LCP formulation. Applying a simple diagonalization to the LCP, using an eigenvalue decomposition of $\mathbf{A}$, one can easily show that a solution to the problem always exists when $\mathbf{A}$ is positive semi definite. However, when bounds are variable the nonlinear nature of the problem makes it hard to say anything conclusive about existence of a solution. The accuracy of the system is thus clearly affected, when attempting to solve a system that has no solution. The effect can be observed in the behavior of the PGS method. By increasing the number of iterations, the PGS method will converge to a positive merit value. This indicates convergence to a local minimizer of the merit function, and not a global minimizer.
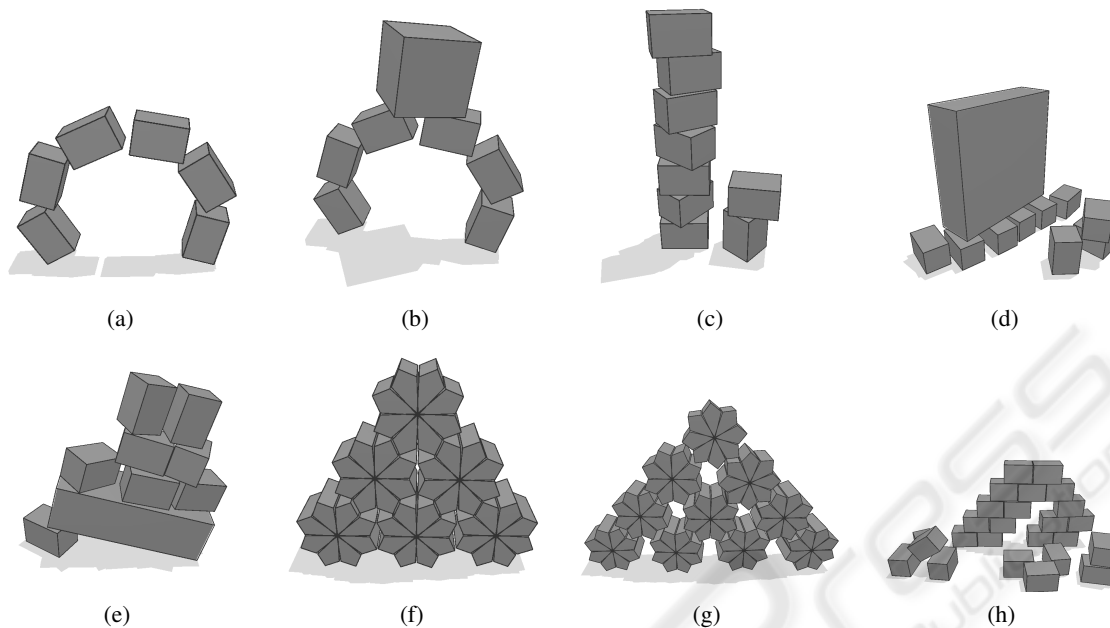
Figure 2: Illustrated test cases used for the PGS–SM method: (a) An arched snake composed of boxes and hinge joints with limits, (b) A heavy box placed upon an arched snake, (c) A large stack of boxes of equal mass, (d) A heavy box resting on lighter smaller boxes, (e) Boxes resting on an inclined surface resulting in static friction forces, (f) A small pyramid of gears, (g) A medium-scale pyramid of gears, (h) A large configuration of boxes stacked in a friction inducing maner.

5.1 Stability Improvements

Stability can be improved by adding minor changes to the presented algorithm. Such a change could be the use of a relative termination criteria for the PGS method similar to (27), thus forcing the method to iterate long enough to improve the estimate of the active set. In our experience, this can be very beneficial although it counteracts interactive performance.

Another strategy that seems to improve stability, is to add numerical regularization to the sub-problem. The matrix $\mathbf{A}_{\mathcal{A}\mathcal{A}}$ is replaced with $\mathbf{A}'_{\mathcal{A}\mathcal{A}} = \mathbf{A}_{\mathcal{A}\mathcal{A}} + \gamma \mathbf{I}$ for some positive scalar γ. The regularization makes $\mathbf{A}'_{\mathcal{A}\mathcal{A}}$ positive definite, which improves the performance of the CG method. The result of the regularization is observed as a damping in the contact forces. In our opinion, it severely affects the realism of the simulation but works quite robustly. In our experience, it seems that one can get away with only dampening the entries corresponding to friction forces.

Rather than regularizing the $\mathbf{A}$-matrix, one could regularize the bounds. We have experienced positive results when applying a lazy evaluation of the bounds inside the subspace solver loop. Thus, having slightly relaxed bounds appear to add some freedom in reaching proper friction forces. On the downside, it appears to make the solver favor static friction solutions. We leave this idea for future work.

One final variation we will mention, is the staggered approach to the contact force problem. The approach is conceptually similar to (Kaufman et al., 2008). The idea is an iteration-like approach. First solve for normal forces assuming fixed given friction forces, and secondly solve for frictional forces assuming fixed given normal forces. The advantage of the staggered approach is that each normal and friction sub-problem has constant bounds, thus the NCP formulation is trivially reduced to a boxed MLCP, equivalent to a LCP. Given the properties of the $\mathbf{A}$-matrix this guarantees solutions exist for the sub-problems. However, whether the sequence of sub-problems will converge in the staggered approach is hard to say. We have not observed any conclusive results on using a staggered approach.

6 CONCLUSIONS

A Projected Gauss–Seidel subspace minimization (PGS–SM) method has been presented, evaluated and compared to the Projected Gauss–Seidel (PGS) method for interactive rigid body dynamics. The PGS–SM method is stable for small sized configurations with large mass ratios, static friction and bilateral joints subject to limits. For medium and larger sized configurations, the PGS–SM method deterio-
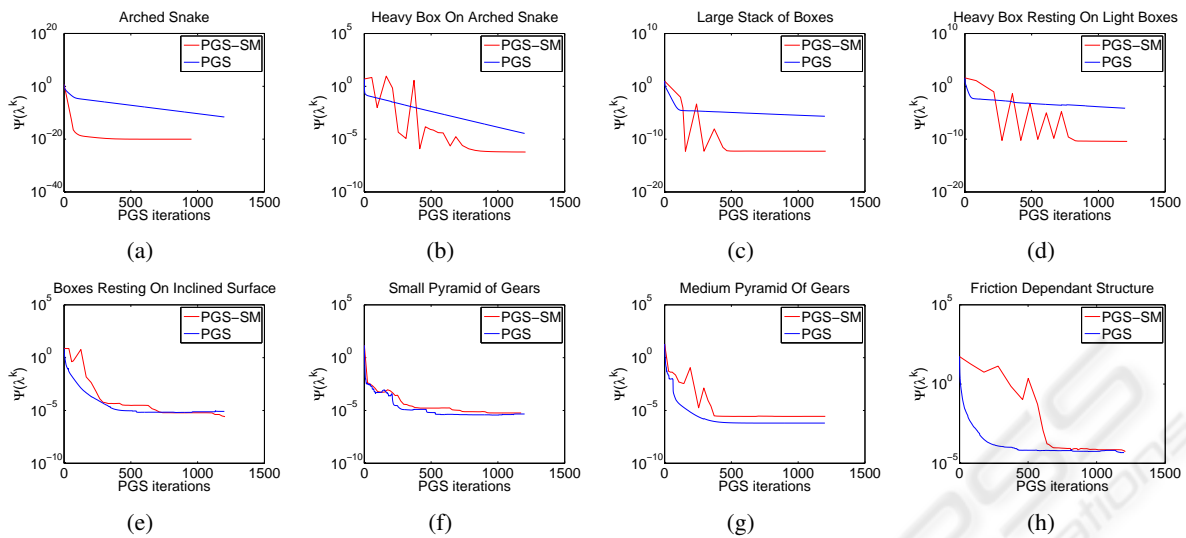
Figure 3: Corresponding convergence plots for the test cases in Figure 2. Observe the jaggyness in the PGS–SM plots in (b), (c), (d), and (g). The spikes indicates that the PGS–SM method guessed a wrong active set. This can cause the merit function to rise abruptly. The ψ funciton is the Fischer function from (Silcowitz et al., 2009).

rates into convergence behavior similar to the PGS method. Still, the PGS–SM method shows qualitatively different appearance in the simulations. For larger configurations, the PGS–SM method may be subject to simulation instability. In our opinion, our investigations indicate a more fundamental problem with the nonlinear complementarity problem (NCP) formulation of the contact force problem. We speculate that existence of solution is vital when accurate computations are performed. The minimum norm nature of the PGS method handles such cases robustly, although not very accurately.

Future work may include investigation into the nature of the NCP formulation, addressing existence of solutions. A more practical viewpoint would be exploring various iterative solvers for the reduced problem, as well as regularization ideas for the NCP formulation. In particular, we find the lazy evaluation of friction bounds appealing.

REFERENCES

Anitescu, M. and Potra, F. A. (1997). Formulating dynamic multi-rigid-body contact problems with friction as solvable linear complementarity problems. *Nonlinear Dynamics. An International Journal of Nonlinear Dynamics and Chaos in Engineering Systems*.

Arechavaleta, G., E.Lopez-Damian, and Morales, J. (2009). On the use of iterative lcp solvers for dry frictional contacts in grasping. In *International Conference on Advanced Robotics 2009, ICAR 2009*.

Baraff, D. (1994). Fast contact force computation for non-penetrating rigid bodies. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*.

Billups, S. C. (1995). *Algorithms for complementarity problems and generalized equations*. PhD thesis, University of Wisconsin at Madison.

Cottle, R., Pang, J.-S., and Stone, R. E. (1992). *The Linear Complementarity Problem*. Academic Press.

Erleben, K. (2007). Velocity-based shock propagation for multibody dynamics animation. *ACM Trans. Graph.*, 26(2).

Erleben, K. and Ortiz, R. (2008). A Non-smooth Newton Method for Multibody Dynamics. In *American Institute of Physics Conference Series*.

Featherstone, R. (1998). *Robot Dynamics Algorithms*. Kluwer Academic Publishers, second printing edition.

Guendelman, E., Bridson, R., and Fedkiw, R. (2003). Non-convex rigid bodies with stacking. *ACM Trans. Graph.*

Hahn, J. K. (1988). Realistic animation of rigid bodies. In *SIGGRAPH '88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques*.

Kaufman, D. M., Sueda, S., James, D. L., and Pai, D. K. (2008). Staggered projections for frictional contact in multibody systems. *ACM Trans. Graph.*, 27(5).

Milenkovic, V. J. and Schmidl, H. (2004). A fast impulsive contact suite for rigid body simulation. *IEEE Transactions on Visualization and Computer Graphics*, 10(2).

Mirtich, B. V. (1996). *Impulse-based dynamic simulation of rigid body systems*. PhD thesis, University of California, Berkeley.

Moore, M. and Wilhelms, J. (1988). Collision detection and response for computer animation. In *SIGGRAPH*

'88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques.

Morales, J. L., Nocedal, J., and Smelyanskiy, M. (2008). An algorithm for the fast solution of symmetric linear complementarity problems. *Numer. Math.*, 111(2).

O'Sullivan, C., Dingliana, J., Giang, T., and Kaiser, M. K. (2003). Evaluating the visual fidelity of physically based animations. *ACM Trans. Graph.*, 22(3).

Redon, S., Kheddar, A., and Coquillart, S. (2003). Gauss least constraints principle and rigid body simulations. In *In proceedings of IEEE International Conference on Robotics and Automation*.

Silcowitz, M., Niebe, S., and Erleben, K. (2009). Nonsmooth Newton Method for Fischer Function Reformulation of Contact Force Problems for Interactive Rigid Body Simulation. In *VRIPHYS 09: Sixth Workshop in Virtual Reality Interactions and Physical Simulations*, pages 105–114. Eurographics Association.

Stewart, D. E. (2000). Rigid-body dynamics with friction and impact. *SIAM Review*.

Stewart, D. E. and Trinkle, J. C. (1996). An implicit timestepping scheme for rigid body dynamics with inelastic collisions and coulomb friction. *International Journal of Numerical Methods in Engineering*.

Trinkle, J. C., Tzitzoutis, J., and Pang, J.-S. (2001). Dynamic multi-rigid-body systems with concurrent distributed contacts: Theory and examples. *Philosophical Trans. on Mathematical, Physical, and Engineering Sciences*.