Nonsmooth Newton Method for Fischer Function Reformulation of Contact Force Problems for Interactive Rigid Body Simulation

M. Silcowitz   S. Niebe   K. Erleben

Department of Computer Science, University of Copenhagen, Denmark

Abstract

In interactive physical simulation, contact forces are applied to prevent rigid bodies from penetrating each other. Accurate contact force determination is a computationally hard problem. Thus, in practice one trades accuracy for performance. The result is visual artifacts such as viscous or damped contact response. In this paper, we present a new approach to contact force determination. We reformulate the contact force problem as a nonlinear root search problem, using a Fischer function. We solve this problem using a generalized Newton method. Our new Fischer–Newton method shows improved qualities for specific configurations where the most widespread alternative, the Projected Gauss-Seidel method, fails. Experiments show superior convergence properties of the exact Fischer–Newton method.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Physically based modeling
I.3.7 [Computer Graphics]: Animation G.1.6 [Mathematics of Computing]: Nonlinear programming

Keywords: Contact Force Problem, Complementarity Formulation, Newton Method, Fischer Function

1. Shortcomings of State-of-Art

Most open source software for interactive real-time rigid body simulation uses 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: the linear convergence rate [CPS92] and inaccurate friction forces in stacks [KSJP08]. The linear convergence results in viscous motion at contacts and loss of high frequency effects. The viscous appearance results in a time delay in the contact response and reduces plausibility [ODGK03]. This has motivated us to develop a new numerical method, based on a nonsmooth reformulation of the contact force problem. The reformulation transforms a nonlinear complementarity problem (NCP) formulation into a nonsmooth root search problem. Our method is compared to the PGS method for interactive simulation.

Rigid body simulation was introduced to the graphics community in the late 1980's [Hah88, MW88] using penalty based and impulse based approaches to describe the physical interactions. Penalty based simulation is not easily adopted to different simulations without parameter tuning. The impulse based approach was extended and improved [Mir96], however stacking was a problem and it suffered from creeping, these problems has since been rectified [GBF03]. Constraint based simulation has received much attention as an alternative [Bar94] to penalty and impulse based simulation. Constraint based simulation can be classified into two groups: maximal and minimal coordinate methods [Fea98]. The focus of this paper is maximal coordinates methods, which are dominated by complementarity formulations. There are alternatives to complementarity formulations, based on kinetic energy [MS01, MS04] and motion space [RKC03]. However, the former solves a more general problem and is not attractive for performance reasons, and the latter does not include frictional forces.

Complementarity formulations come in two flavors: acceleration based formulations [TTP01] and velocity based formulations [ST96]. Acceleration based formulations cannot handle collisions [AP97, CR98]. Further, acceleration based formulations suffer from indeterminacy and inconsis-
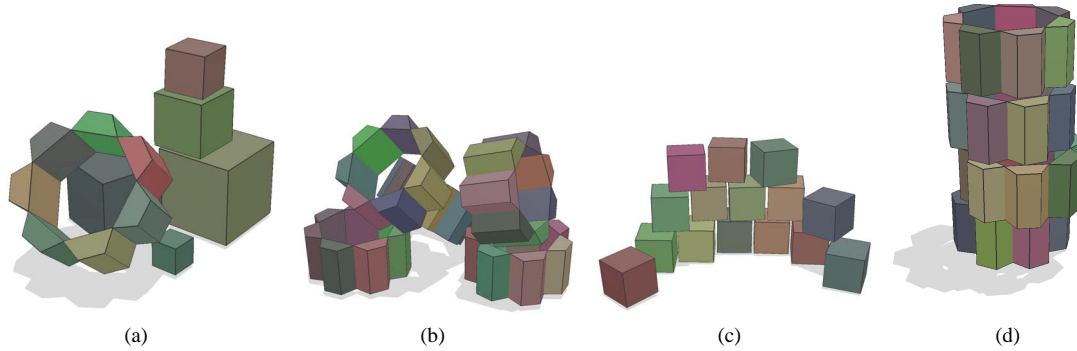
Figure 1: *Contact force computations using our Fischer–Newton method are more efficient for large mass ratios. In (a) we illustrate a random configuration of mixed objects, (b) a pile of non-convex gears, (c) a wall of boxes, and in (d) a pyramid of boxes.*

tency [Ste00]. The velocity based formulation suffers from none of these problems and is the formulation we use in this work. [KEP05] presented a velocity based method based on limit surfaces and an approximation of momentum conservation.

The approach we present here is more simple in the sense that we reformulate the frictional problem as a system of nonlinear equations, which we then solve using a Newton method.

2. The Fischer Function Reformulation

The problem of computing contact forces can be formulated as a linear complementarity problem (LCP). However, in interactive physical simulations a different formulation is used, both will be derived shortly. In the following we will compare the two formulations, and thereafter derive our reformulation. We will keep the notation general and simple to enhance readability. Without loss of generality, we will only consider a single contact point to make the differences more visible. The contact force model is the focus of our work and we refer the reader to [Erl07] for details on the time-stepping scheme and the assembly of the matrices used.

First, a derivation of the LCP formulation for computing contact forces. Given the generalized mass matrix, M, the generalized velocity vector, u, and the generalized external force vector, F, we write the discretized Newton-Euler equation,

$$Mu - J_n^T \lambda_n - J_D^T \lambda_D = F, \tag{1}$$

where J_n is the Jacobian corresponding to normal constraints and J_D is the Jacobian corresponding to the tangential contact forces, λ_n and λ_D are the normal and frictional Lagrange multipliers. The actual discretization variables are contained within the Lagrange multipliers, this is a slight abstraction to keep the expression simple. Using (1), we isolate the generalized velocities,

$$u = M^{-1}F + M^{-1}J_n^T \lambda_n + M^{-1}J_D^T \lambda_D. \tag{2}$$

The non-penetration constraints on a velocity-based form can be written as

$$J_n u \geq 0, \tag{3a}$$
$$\lambda_n \geq 0, \tag{3b}$$
$$\lambda_n (J_n u) = 0. \tag{3c}$$

We use a scalar, β, as a measure of the most negative contact velocity along a given positive direction in the contact plane,

$$(\beta e + J_D u) \geq 0, \tag{4a}$$
$$\beta \geq 0, \tag{4b}$$

where e is a vector of ones. The contact velocity along the ith direction in the contact plane is equivalent to the dot product of the ith row of J_D and the generalized velocity, u. A positive span of tangent vectors ensures symmetric behavior. This is not explicit in the equations, but lies in the construction of the J_D-matrix. One can have any number of directional tangent vectors in the contact plane. A friction pyramid approximation corresponds to two orthogonal vectors in the contact plane, t_1 and t_2. In this case the Jacobian matrix J_D has four rows. The first two rows correspond to the directions t_1 and t_2 and the last two rows correspond to $-t_1$ and $-t_2$. We will bound the friction force by the friction polyhedra cone approximation,

$$\mu \lambda_n - e^T \lambda_D \geq 0 \quad \text{and} \quad \lambda_D \geq 0. \tag{5}$$

For the friction pyramid approximation one would have four components of λ_D. Finally contact velocity and frictional force are coupled through the complementarity conditions,

$$\lambda_D^T (\beta e + J_D u) = 0 \quad \text{and} \quad \beta \left(\mu \lambda_n - e^T \lambda_D \right) = 0. \tag{6}$$

Essentially, the first equation will try to pick a friction direction that is closest to the maximum dissipation direc-

tion. The last condition ensures that a non-zero friction force is found if slipping occurs. If we have a non-zero relative contact velocity then $\beta > 0$ and the last equation requires $\left(\mu\lambda_n - e^T\lambda_D\right) = 0$. This results in a friction force that lies on the boundary of the friction polyhedra cone approximation. This is in agreement with Coulomb's friction law, in the case of sliding friction. Finally, we can assemble the entire problem by substituting (2) into (3a) and (4a). Rewriting results in the matrix-vector formulation,

$$\underbrace{\begin{bmatrix} J_n M^{-1} J_n^T & J_n M^{-1} J_D^T & 0 \\ J_D M^{-1} J_n^T & J_D M^{-1} J_D^T & e \\ \mu & -e^T & 0 \end{bmatrix}}_{A_{\text{lcp}}} \underbrace{\begin{bmatrix} \lambda_n \\ \lambda_D \\ \beta \end{bmatrix}}_{x_{\text{lcp}}} + \underbrace{\begin{bmatrix} J_n M^{-1} F \\ J_D M^{-1} F \\ 0 \end{bmatrix}}_{b_{\text{lcp}}} \geq 0, \tag{7}$$

and we get the traditional LCP formulation,

$$A_{\text{lcp}} x_{\text{lcp}} + b_{\text{lcp}} \geq 0, \tag{8a}$$
$$x_{\text{lcp}} \geq 0, \tag{8b}$$
$$x_{\text{lcp}}^T (A_{\text{lcp}} x_{\text{lcp}} + b_{\text{lcp}}) = 0. \tag{8c}$$

Now we turn to the nonlinear formulation of the contact force computation. For the nonlinear case we have the Newton-Euler equations,

$$M u - J_n^T \lambda_n - J_t^T \lambda_t = F, \tag{9}$$

where J_n is the Jacobian corresponding to normal constraints. We do not use the J_D matrix from the LCP formulation instead we use J_t, which is the Jacobian corresponding to the tangential contact forces. Since the contact plane is two dimensional, we choose to 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, J_t has only two rows corresponding to the two directions. From (9) we can obtain the generalized velocities,

$$u = M^{-1} F + M^{-1} J_n^T \lambda_n + M^{-1} J_t^T \lambda_t. \tag{10}$$

Let the Lagrange multipliers $\lambda = \begin{bmatrix} \lambda_n & \lambda_t^T \end{bmatrix}^T$ and contact Jacobian $J = \begin{bmatrix} J_n & J_t \end{bmatrix}^T$, then we write the relative contact velocities as $y = \begin{bmatrix} y_n & y_t^T \end{bmatrix}^T$,

$$y = Ju = \underbrace{JM^{-1}J^T}_{A_{\text{ncp}}} \lambda + \underbrace{JM^{-1}F}_{b_{\text{ncp}}}. \tag{11}$$

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

$$y < 0 \Rightarrow \lambda_t = \mu\lambda_n, \tag{12a}$$
$$y > 0 \Rightarrow \lambda_t = -\mu\lambda_n, \tag{12b}$$
$$y = 0 \Rightarrow -\mu\lambda_n \leq \lambda_t \leq \mu\lambda_n. \tag{12c}$$

This can be proven by algebraic manipulation.

Proof Coulomb's friction law is defined as,

$$\mu\lambda_n - \sqrt{\lambda_t^2} \geq 0, \tag{13a}$$
$$\|y\| \left(\mu\lambda_n - \sqrt{\lambda_t^2}\right) = 0, \tag{13b}$$
$$\|y\| \sqrt{\lambda_t^2} = -y\lambda_t. \tag{13c}$$

The first equation yields a maximum bound on the friction force. The second equation models sticking and slipping friction, while the last ensures that friction is opposing motion in the case of slipping friction. We perform a case-by-case analysis. If $y \neq 0$ then from (13b) we must have,

$$\mu\lambda_n - \sqrt{\lambda_t^2} = 0. \tag{14}$$

If $y > 0$ then from (13c) we must have $\lambda_t < 0$ and if $y < 0$ we have that $\lambda_t > 0$. So putting it together,

$$y < 0 \Rightarrow \lambda_t = \mu\lambda_n \quad \text{and} \quad y > 0 \Rightarrow \lambda_t = -\mu\lambda_n. \tag{15}$$

If $y = 0$ then (13b) and (13c) are trivially fulfilled and from (13a) we have,

$$-\mu\lambda_n \leq \lambda_t \leq \mu\lambda_n. \tag{16}$$

This concludes the proof. $\square$

We split y into positive and negative components,

$$y = y^+ - y^-, \tag{17}$$

where

$$y^+ \geq 0, \quad y^- \geq 0 \quad \text{and} \quad \left(y^+\right)^T \left(y^-\right) = 0. \tag{18}$$

In case of friction we define the bounds $-l_t(\lambda) = u_t(\lambda) = \mu\lambda_n$ and for normal force $l_n(\lambda) = 0$ and $u_n(\lambda) = \infty$. Using the bounds (12), (17) and (18), yields the final NCP formulation,

$$y^+ - y^- = A_{\text{ncp}}\lambda + b_{\text{ncp}}, \tag{19a}$$
$$y^+ \geq 0, \tag{19b}$$
$$y^- \geq 0, \tag{19c}$$
$$u(\lambda) - \lambda \geq 0, \tag{19d}$$
$$\lambda - l(\lambda) \geq 0, \tag{19e}$$
$$\left(y^+\right)^T (\lambda - l(\lambda)) = 0, \tag{19f}$$
$$\left(y^-\right)^T (u(\lambda) - \lambda) = 0, \tag{19g}$$
$$\left(y^+\right)^T \left(y^-\right) = 0. \tag{19h}$$

The advantages are

- There is no need for the auxiliary variable β
- There is no need for a positive span of tangent vectors, we just use two orthogonal directions in the contact plane.
- For two tangent directions J_t is one fourth the size of J_D.

Thus the NCP formulation has a much lower memory footprint. To make the connection to the LCP formulation more clear one could define,

$$J_D u = \begin{bmatrix} J_t u \\ -J_t u \end{bmatrix}, \lambda_D = \begin{bmatrix} \lambda_t^+ \\ \lambda_t^- \end{bmatrix}, \quad \text{and} \quad \beta = \max_j |y_j|. \quad (20)$$

The main problem with the nonlinear formulation as presented, is that t_1 and t_2 are independent. The disadvantage is that it solves the friction problem as two decoupled one-dimensional Coulomb friction models. This results in an outer friction pyramid approximation, where the LCP formulation uses an inner diamond shape approximation. Thus the NCP formulation overestimates the magnitude of the friction force whereas the LCP formulation underestimates the magnitude of the friction force. Using the NCP formulation, the corner directions of the friction pyramid are favored in case of unaligned contact velocities. The LCP formulation favors the tangent direction that best approximates the direction of maximum dissipation. The NCP formulation is attractive for interactive simulations where performance is of greater importance than accuracy. In practice the problem of accuracy can be remedied by choosing t_1 parallel to the sliding direction. Using such a heuristic would make both the LCP and NCP formulations accurate in the case of sliding friction.

We have shown that the formulation (12) is equivalent to the coupled complementarity problems given in (19). We can reformulate those complementarity problems using a Fischer function. The Fischer function, $\phi : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}$ is defined as,

$$\phi(a, b) = \sqrt{a^2 + b^2} - a - b. \quad (21)$$

Notice that solutions of the complementarity problem corresponds to the roots of the Fischer function,

$$a \geq 0, \quad b \geq 0, \quad a \cdot b = 0, \quad \text{iff} \quad \phi(a, b) = 0. \quad (22)$$

This allows us to rewrite a complementarity problem into a root search problem, which again allows us to compute solutions using a Newton method. The real valued function $\phi(a, b)$ is trivially extended to a vector function. Using the Fischer function, (19) can be written as,

$$\phi\left(y^+, \lambda - l(\lambda)\right) = 0, \quad (23a)$$

$$\phi\left(y^-, u(\lambda) - \lambda\right) = 0. \quad (23b)$$

Observe, for any arbitrary positive scalar k, (23a) is equivalent to,

$$\phi\left(ky^+, \lambda - l(\lambda)\right) = 0. \quad (24)$$

Further, there always exists a positive scalar k such that,

$$ky^+ = \phi\left(y^- - y^+, u(\lambda) - \lambda\right). \quad (25)$$

This is proven by a case-by-case analysis. If $y^+ = 0$ then (25) reduces to (23b) which trivially holds for any value of k. If $y_i^+ > 0$, then we must have $y_i^- = 0$ and (25) reduces to,

$$ky_i^+ = \underbrace{\sqrt{\left(y_i^+\right)^2 + (u(\lambda)_i - \lambda_i)^2} - (u(\lambda)_i - \lambda_i)}_{c} + y_i^+. \quad (26)$$

Using the triangle inequality, we always have $c > 0$ so $k = c/y^+ + 1 > 1$. Thus, for any given $y^+ > 0$ we can always find a $k > 1$ such that (25) holds. We can now substitute (25) into (24) and obtain the Fischer reformulation of our problem,

$$\phi(\lambda - l(\lambda), \phi(-y, u(\lambda) - \lambda)) = 0. \quad (27)$$

Notice our formulation differs from the mixed complementarity problem formulation [Bil95]. We are using variable bounds and not fixed bounds. Our derivation do not rely on explicitly enforcing $u(\lambda) \geq l(\lambda)$ at all times, instead this holds implicitly for any solution of the problem.

3. The Nonsmooth Newton Method

Numerical solutions to complementarity problems can be computed using well known iterative methods. In [CPS92], the authors describe matrix splitting methods similar to well known relaxation methods for linear systems: Jacobi, Gauss–Seidel, and successive over–relaxation (SOR). These methods enforce the complementarity constraints by doing a projection following a relaxation. Projected Gauss–Seidel solvers have previously been used [Mor99, Jea99] although in a blocked version. Krylov subspace methods like the Conjugate Projected Gradient (CPG) can be used in a similar manner [RAD05] or together with an active set method [Mur88]. The CPG has quadratic convergence rate for frictionless contacts, but erratic convergence rate when friction is considered [RA04]. Two other types of iterative algorithms are based on Newton and Interior Point (IP) methods. The theoretical convergence rates of the Newton and IP methods are quadratic, which is a clear improvement over relaxation schemes. Complexity can be reduced even further if one considers a multilevel preconditioner for the Newton equation [Ort07]. Direct methods are costly and often limited to LCP formulations. When using incremental matrix factorization, Lemke's and Keller's methods need n iterations resulting in a total complexity of $O(n^3)$ [Lac03]. We choose the Newton method due to the quadratic convergence rate and do not consider IP methods due to the nature of root search problems.

In the following, we will describe our Newton based method, which we have named the Fischer–Newton (FN) method to reflect the use of a Fischer reformulation.

We will now extend our model to support multiple contacts. Let the normal and frictional Lagrange multipliers of the ith contact point $\lambda_i = \begin{bmatrix} \lambda_{n,i} & \lambda_{t_1,i} & \lambda_{t_2,i} \end{bmatrix}^T$. For the ith contact point we have from (27),

$$\phi_i(\lambda) = \phi(\lambda_i - l_i(\lambda_i), \phi(-y_i, u_i(\lambda_i) - \lambda_i)) = 0, \quad (28)$$

where $l_i(\lambda_i) = \begin{bmatrix} 0 & -\mu_i\lambda_{n,i} & -\mu_i\lambda_{n,i} \end{bmatrix}^T$ and $u_i(\lambda_i) = \begin{bmatrix} \infty & \mu_i\lambda_{n,i} & \mu_i\lambda_{n,i} \end{bmatrix}^T$ are the normal and frictional lower and upper bounds. The relative contact point velocity y_i is an affine function of the agglomerated Lagrange multiplier vector $\lambda = \begin{bmatrix} \lambda_1^T & \lambda_2^T & \dots & \lambda_N^T \end{bmatrix}^T$. Now we can agglomerate all contact problems (28) into one vector valued problem, given by the vector valued function $\Phi : \mathbb{R}^n \to \mathbb{R}^n$ with $n = 3N$,

$$\Phi(\lambda) = \begin{bmatrix} \phi_1(\lambda) \\ \vdots \\ \phi_N(\lambda) \end{bmatrix}, \tag{29}$$

which is almost everywhere Frechét differentiable. We define the set Ω_Φ to be the set of all λ for which Φ is not Frechét differentiable. The B-subdifferential of Φ at λ is,

$$\partial_B\Phi = \{H : \mathbb{R}^n \to \mathbb{R}^n | H = \lim J_\Phi(\lambda_i) : \lambda_i \to \lambda, \lambda \notin \Omega_\Phi\}, \tag{30}$$

where J_Φ is the usual Jacobian. Using the B-subdifferential, the generalized Newton equation is,

$$H_k\Delta\lambda_k = -\Phi(\lambda_k), \tag{31}$$

where H_k is any nonsingular element in $\partial_B\Phi$ [Bil95]. The Newton update is then

$$\lambda_{k+1} = \lambda_k + \Delta\lambda_k. \tag{32}$$

The convergence behavior of the numerical method is described by the following Theorem.

Theorem 3.1 (Qi & Sun [QS98, Theorem 2.2]) Suppose that $\Phi(\lambda^*) = 0$ and that all $H \in \partial_B\Phi(\lambda^*)$ are nonsingular. Then the generalized Newton method (31) is Q-superlinearly convergent in a neighborhood of λ^* if Φ is semismooth at λ^*, and quadratically convergent at λ^* if Φ is strongly semismooth at λ^*.

We know from [Bil95, Theorem 3.2.8] that Φ is strongly semismooth everywhere when l and u are fixed bounds. This is easily extended to hold for our version, where the bounds are functions of λ. The only limitation we must impose on $l(\lambda)$ and $u(\lambda)$ is that they are both semismooth functions. This follows since any sum, product or composite of semismooth functions is also semismooth [Bil95]. Figure 2 shows that quadratic convergence is not only a theoretical property but is achievable in practise as well. The test case used for Figure 2 is a small 6×6 problem. It takes 0.0033 seconds on average to compute the solution. For interactive usage – 30 frames per second – using 10 % of the time for contact force solving [YFR06], this restricts the FN method to problems of small dimensions.

Solving the generalized Newton equation exact is an operation that has a time complexity of $O(n^3)$, a cheaper alternative should improve the overall performance of the FN method.
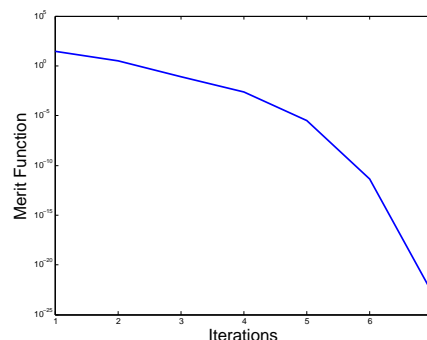

Figure 2: *The Fischer–Newton method on a well behaved problem displaying quadratic convergence. Average time used on solution is 0.0033 seconds for this test case.*

Quasi Nonlinear Conjugate Gradient Method for Interactive Frame Rates The conjugate gradient (CG) method is the usual choice for iteratively solving equation systems like (31) approximately. It is guaranteed to converge to the solution in at most n iterations. Unfortunately, this only holds for symmetric and positive definite matrices. We cannot guarantee that this holds for H_k. However, it is appealing to use one of the other CG like methods due to the nice properties. The biconjugate gradient method does not require symmetry of the matrix, instead it uses the matrix and its transposed to compute a search direction. For this reason it yields the same result as the CG for symmetric matrices, but at twice the cost. For interactive usage the extra cost is undesired, so a third CG like method is considered. The standard CG method can be considered as the minimization of a quadratic function,

$$\min_\lambda \underbrace{\frac{1}{2}\lambda^T H_k\lambda - \Phi(\lambda)^T\lambda}_{\gamma(\lambda)}, \tag{33}$$

where the gradient is $\nabla\gamma(\lambda) = H_k\lambda - \Phi(\lambda)$ when H_k is symmetric. The gradient is used in the computation of the new search direction. The nonlinear conjugate gradient (NCG) method makes no such assumption, instead it just minimizes the general function $f(\lambda)$. To do this we need the gradient $\nabla f(\lambda)$ for the computation of new search directions. However, such a gradient is computationally expensive and the aim of interactivity forces us to take the less expensive choice. Instead of using the exact ∇f we use a finite difference approximation,

$$\nabla f(\lambda) \approx \frac{f(\lambda+h) - f(\lambda)}{h}. \tag{34}$$

Thus, our method can be perceived as a quasi Fletcher–Reeves [NW99] method where we use finite difference approximations for the gradient and a line-search similar to the exact minimizer used in the Conjugate Gradient method.
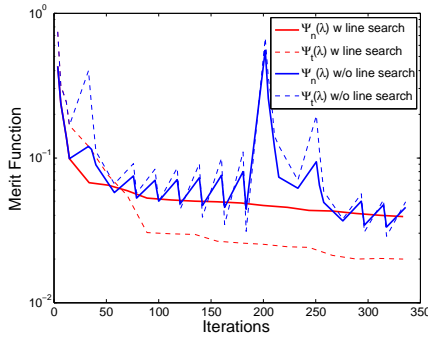
Figure 3: *Motivation for including a line search method. The test case is a pyramid configuration, having 103 contact points resulting in 309 variables. Using line search, a smoother convergence is achieved. The frictional component has the additional benefit of an improved accuracy.*

Globalization using Line Search The Newton method is not globally convergent from an arbitrary starting point. It has to be initialized within a small enough neighborhood of the solution. To achieve global convergence, a line search method is often used. Experiments indicate that it is beneficial to use a line search method when including friction, see Figure 3. We use a backtracking line search with an Armijo condition to ensure sufficient decrease and that the chosen step length is not too small [NW99]. The line search uses the natural merit function of $\Phi(\lambda)$ as a measure of convergence. The natural merit function is defined as,

$$\Psi(\lambda) = \frac{1}{2}||\Phi(\lambda)||^2. \qquad (35)$$

The gradient is,

$$\nabla\Psi(\lambda) = H_k^T \Phi(\lambda). \qquad (36)$$

The Armijo condition is used as a termination criteria for the line search method. Applied to $\Psi(\lambda)$, the Armijo condition is,

$$\Psi(\lambda + \Delta\lambda) \leq \Psi(\lambda) + c\alpha\nabla\Psi(\lambda)^T\Delta\lambda, \qquad (37)$$

where $c \in (0,1)$. The object of the line search method is to find an α such that (37) is satisfied. To avoid computing $\nabla\Psi(\lambda)$ we insert the definition from (36) into (37) and make a substitution using (31),

$$\Psi(\lambda + \Delta\lambda) \leq (1 - 2\alpha c)\Psi(\lambda). \qquad (38)$$

The step length is used when updating λ_{k+1} such that

$$\lambda_{k+1} = \lambda_k + \alpha\Delta\lambda_k. \qquad (39)$$

Semi-Staggered Warm Starting to Accelerate Convergence The frictional upper and lower bounds depend on the magnitude of the normal forces. Intuition suggests that given a good estimate of the normal forces, the computation of frictional forces will be more feasible. The idea is
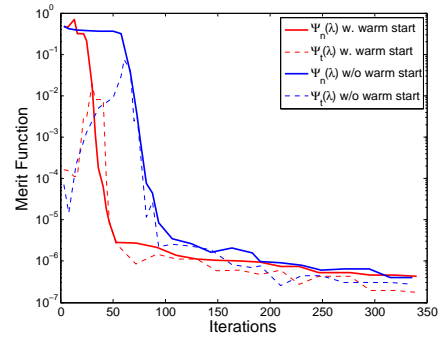

Figure 4: *Two convergence plots for the large mass ratio test case showing how warm starting the Fischer–Newton method provides improved convergence.*

similar to a staggered approach [KSJP08]. However, instead of continuously iterating, we simply start the FN method with the frictional components disabled and use an absolute threshold termination criteria. Once convergence is detected, the frictional components are enabled and the full contact force problem is solved until convergence or a maximum allowed iteration count is reached. Figure 4 illustrates the added convergence benefit from this semi-staggered warm starting technique.

4. Newton Method in Action

We have constructed three test cases to test different properties of the FN method. All three test cases use simple box geometries to aid the visual detection of physical anomalies, which might be hard to detect if more complex geometries were used.

Further the Jacobian – and subsequently the Newton equation – inherits a lot of the properties of the A_{ncp} matrix. We expect that our method struggles when applied to problems where A_{ncp} is badly conditioned. The three test cases are therefore designed to "stress" the A_{ncp} matrix.

Large Mass Ratios This is a small configuration consisting of only three objects, two boxes and a floor. A heavy box is placed on top of a lighter box. The desired effect is for the boxes to remain still with no additional movement after the initial settlement. This setup results in a A_{ncp} matrix which has an unfortunate distribution of eigenvalues, as seen in Figure 7(a). This gives an ill-conditioned A_{ncp} matrix, making the Newton equation singular. In addition to "stressing" our method, this setup is notoriously difficult for the PGS method to handle. The PGS method has a tendency to undershoot the magnitudes of the contact forces, resulting in severe penetrations.

Sparse Structure For the sparse structure, a stack of identical boxes are simulated. This setup poses some of the same
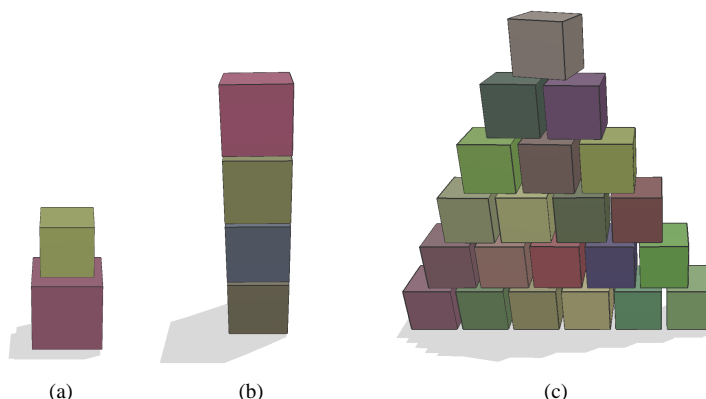
Figure 5: *The three test cases (a) A heavy box placed on top of a light box (b) A stack of equal sized boxes of equal density (c) A dense structured stack of equal sized boxes of equal density.*

difficulties as the large mass ratio setup. The bottommost box suffers from the accumulated effect of the remaining boxes on top, acting as one large mass. Furthermore, all contact normals are parallel, which results in an over-determined system.

Dense Structure When the complexity of the physical system increases, people become less competent at noticing dynamic anomalies [ODGK03]. Thus structured stacking is very helpful for visual confirmation of complex configurations. The dense structure covers the above mentioned difficulties. Further this setup tests scalability.

4.1. The Qualities of the Newton Method

Figure 6 shows the fill patterns for the three test cases, visualized by a gray scale image. It confirms that the A_{ncp} is symmetric and that sparsity increases with configuration size. The diagonal entries appear to be larger than off diagonal entries, although not enough for the matrix to be diagonally dominant. Studying the eigenvalue spectrums in Figure 7, reveals multiplicity greater than one for some non-zero eigenvalues and a large number of zero eigenvalues, roughly 60 % for all three cases. The poor eigenvalue properties are a direct result from over-determined systems and large mass ratios. By this analysis, we expect non-unique solutions to exist and to observe convergence to local minima.

The plots in Figure 8 show the convergence properties of the natural merit function (35) when using the FN and PGS methods. The cost of FN iterations are presented in units of PGS iterations in order to aid the comparison. The convergence is divided into normal and frictional components. The division helps to visualize the effect of the semi-staggered warm start heuristic and whether convergence problem are due to frictional or normal components. Figure 8(a) shows an improvement in both convergence and accuracy in several orders of magnitude, when using the FN method. As the complexity of the scene increases the benefit of using the FN method decreases. In Figure 8(b) the FN method remains superior on both convergence and accuracy for the frictional contribution. When given enough iterations, the PGS method almost achieves the same level of accuracy. Figure 8(c) shows the convergence results for the most complex test case. Both methods converge to local minima for this test case, but in terms of accuracy the minimizer chosen by the FN method is poorer than the one chosen by the PGS method. However, the FN method converged using fewer iterations than PGS.

4.2. Interactive Setting

Figure 10 shows image sequences of a comparison of the handling of large mass ratios, the FN method versus the PGS method. As the images show, the fidelity and consistency break down for the PGS method, where the FN method produces plausible results. In Figure 9 still frames from a larger configuration are shown, the FN method is used to compute contact forces. The FN method is still capable of delivering interactive frame rates for small to medium sized configurations.

5. Conclusion and Discussion

We have presented a novel Fischer reformulation of a contact force problem for interactive physical simulation. To solve the problem we have developed a novel Fisher–Newton (FN) type method. We have evaluated and compared the new method against the Projected Gauss–Seidel (PGS) method.

The FN method was able to compute solutions comparable to the PGS method, both in terms of solution accuracy, and in terms of computational effort. Regarding small sized configurations, the FN method demonstrated superior abilities in handling large mass ratios. To a lesser extend, the FN method showed good abilities when applied to smaller stacks
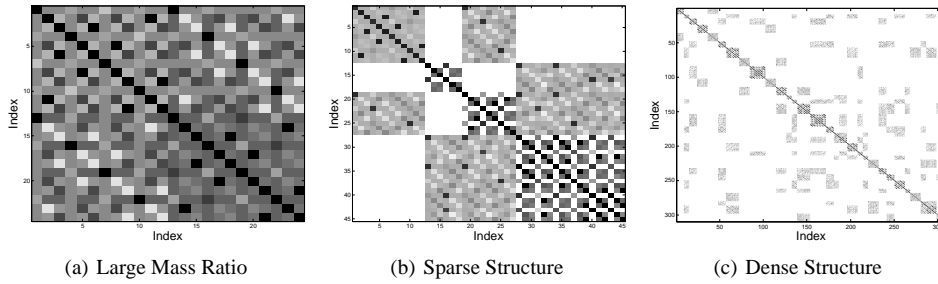
(a) Large Mass Ratio   (b) Sparse Structure   (c) Dense Structure

Figure 6: *Fill patterns for the A_{ncp} matrices of the three test cases. The color corresponds to magnitude of values, white is zero and black is $\max(A_{ncp})$. Observe mostly black diagonal entries and increasing sparsity as configuration size grows.*
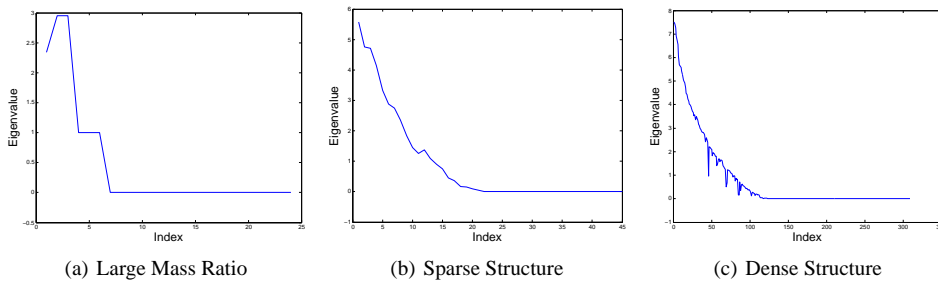

(a) Large Mass Ratio   (b) Sparse Structure   (c) Dense Structure

Figure 7: *Distribution of eigenvalues in A_{ncp} for the three test configurations. Observe a large amount of zero-valued eigenvalues, multiplicity and poor distribution.*
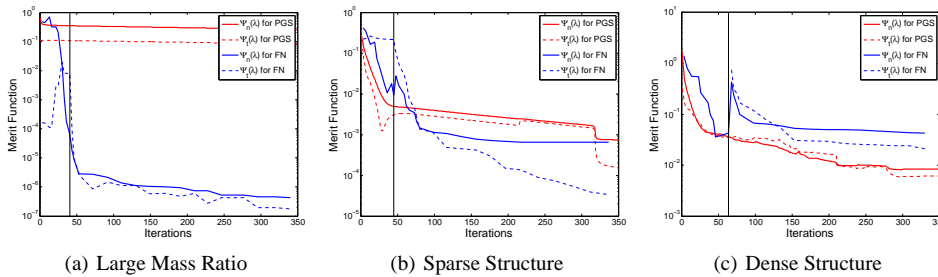

(a) Large Mass Ratio   (b) Sparse Structure   (c) Dense Structure

Figure 8: *Comparative convergence plots of the Fischer–Newton method versus the Projected Gauss–Seidel method. Notice the improved convergence properties of the Fischer–Newton method for the cases of large mass ratios and sparse structure. The vertical line indicates the point where friction is enabled in the warm start approach.*

Figure 9: *Still frames from an interactive simulation of a medium sized configuration, with approximately 280 contact points yielding 840 variables. The Fischer–Newton method is implemented in Java and uses 6 milliseconds on average per frame. The program utilizes one core on a 2.1 GHz CPU Duo Core.*
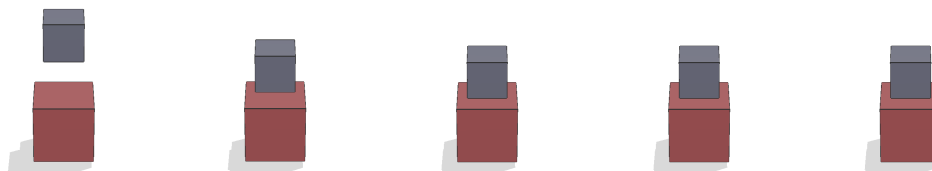
(a) The Fischer–Newton method. Newton iteration limit 10, sub-system iteration limit 15, line-search iteration limit 7
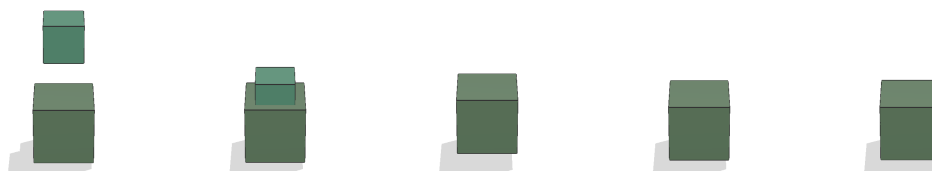
(b) The Projected Gauss–Seidel method. 150 iterations per frame

Figure 10: *Still frames from two comparative movie strips showing the success of the Fischer–Newton method in opposition to the failure of the Projected Gauss–Seidel method. Captures was done every 30 frames. Even with a 150 iterations per frame, the Projected Gauss-Seidel method fails to support the upper heavy box.*

of 3-4 objects. The FN method proved to have convergence problems when applied to setup where large friction forces were present. Inclusion of friction results in greater amount of over-determinacy. This over-determinacy is inherited by the Newton equation. We speculate that the increased over-determinacy is the cause of the inability to handle more complex and frictionally dependent problems. One may apply matrix damping to counter the numerical problems coming from over-determinacy. However, it results in a damping of the solution and little is gained in accuracy compared to using the computationally cheaper PGS method. Thus, future work on Newton methods for contact force problems should focus on how to deal efficiently with the over-determinacy in the contact force problem.

References

[AP97] ANITESCU M., POTRA F. A.: 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 14*, 3 (1997), 231–247.

[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* (New York, NY, USA, 1994), ACM, pp. 23–34.

[Bil95] BILLUPS S. C.: *Algorithms for complementarity problems and generalized equations.* PhD thesis, University of Wisconsin at Madison, Madison, WI, USA, 1995.

[CPS92] COTTLE R., PANG J.-S., STONE R. E.: *The Linear Complementarity Problem.* Computer Science and Scientific Computing. Academic Press, February 1992.

[CR98] CHATTERJEE A., RUINA A.: A new algebraic rigid body collision law based on impulse space considerations. *Journal of Applied Mechanics 65*, 4 (1998), 939–951.

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

[Fea98] FEATHERSTONE R.: *Robot Dynamics Algorithms*, second printing ed. Kluwer Academic Publishers, 1998.

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

[Hah88] HAHN J. K.: Realistic animation of rigid bodies. In *SIGGRAPH '88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1988), ACM, pp. 299–308.

[Jea99] JEAN M.: The non-smooth contact dynamics method. *Computer Methods in Applied Mechanics and Engineering 177*, 3–4 (July 1999), 235–257.

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

[KSJP08] KAUFMAN D. M., SUEDA S., JAMES D. L., PAI D. K.: Staggered projections for frictional contact in multibody systems. *ACM Trans. Graph. 27*, 5 (2008), 1–11.

[Lac03] LACOURSIERE C.: Splitting methods for dry frictional contact problems in rigid multibody systems: Preliminary performance results. In *The Annual SIGRAD Conference* (November 2003), Ollila M., (Ed.), no. 10 in Linkøping Electronic Conference Proceedings.

[Mir96] MIRTICH B. V.: *Impulse-based dynamic simulation of rigid body systems.* PhD thesis, University of California, Berkeley, 1996.

[Mor99] MOREAU J. J.: Numerical aspects of the sweeping process. *Computer Methods in Applied Mechanics and Engineering 177*, 3–4 (July 1999), 329–349.

[MS01] MILENKOVIC V. J., SCHMIDL H.: Optimization-based animation. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 2001), ACM, pp. 37–46.

[MS04] MILENKOVIC V. J., SCHMIDL H.: A fast impulsive contact suite for rigid body simulation. *IEEE Transactions on Visualization and Computer Graphics 10*, 2 (2004), 189–197.

[Mur88] MURTY K. G.: *Linear Complementarity, Linear and Nonlinear Programming.* Helderman-Verlag, 1988.

[MW88] MOORE M., WILHELMS J.: Collision detection and response for computer animation. In *SIGGRAPH '88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1988), ACM, pp. 289–298.

[NW99] NOCEDAL J., WRIGHT S. J.: *Numerical optimization.* Springer Series in Operations Research. Springer-Verlag, New York, 1999.

[ODGK03] O'SULLIVAN C., DINGLIANA J., GIANG T., KAISER M. K.: Evaluating the visual fidelity of physically based animations. *ACM Trans. Graph. 22*, 3 (2003), 527–536.

[Ort07] ORTIZ R.: *Newton/AMG algorithm for solving complementarity problems arising in rigid body dynamics with frictional impacts.* PhD thesis, University of Iowa, July 2007.

[QS98] QI L., SUN D.: Nonsmooth equations and smoothing Newton methods. *Progress in Optimization: Contributions from Australasia, A. Eberhard, B. Glover, R. Hill and D. Ralph eds., Kluwer Academic Publisher, Boston* (1998).

[RA04] RENOUF M., ALART P.: Gradient type algorithms for 2d/3d frictionless/frictional multicontact problems. In *Proceedings of European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS)* (2004).

[RAD05] RENOUF M., ACARY V., DUMONT G.: Comparison of algorithms for collisions, contact and friction in view of real-time applications in multibody dynamics. In *Proceedings (International Conference on Advances in Computational Multibody Dynamics. ECCOMAS Thematic Conference)* (2005).

[RKC03] REDON S., KHEDDAR A., COQUILLART S.: Gauss least constraints principle and rigid body simulations. In *In proceedings of IEEE International Conference on Robotics and Automation* (2003).

[ST96] STEWART D. E., TRINKLE J. C.: An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and coulomb friction. *International Journal of Numerical Methods in Engineering 39*, 15 (1996), 2673–2691.

[Ste00] STEWART D. E.: Rigid-body dynamics with friction and impact. *SIAM Review 42*, 1 (2000), 3–39.

[TTP01] TRINKLE J. C., TZITZOUTIS J., PANG J.-S.: Dynamic multi-rigid-body systems with concurrent distributed contacts: Theory and examples. *Philosophical Trans. on Mathematical, Physical, and Engineering Sciences 359*, 1789 (December 2001), 2575–2593.

[YFR06] YEH T. Y., FALOUTSOS P., REINMAN G.: Enabling real-time physics simulation in future interactive entertainment. In *Sandbox '06: Proceedings of the 2006 ACM SIGGRAPH symposium on Videogames* (New York, NY, USA, 2006), ACM, pp. 71–81.