Rigid Body Contact Problems using Proximal Operators

Kenny Erleben
University of Copenhagen
kenny@di.ku.dk

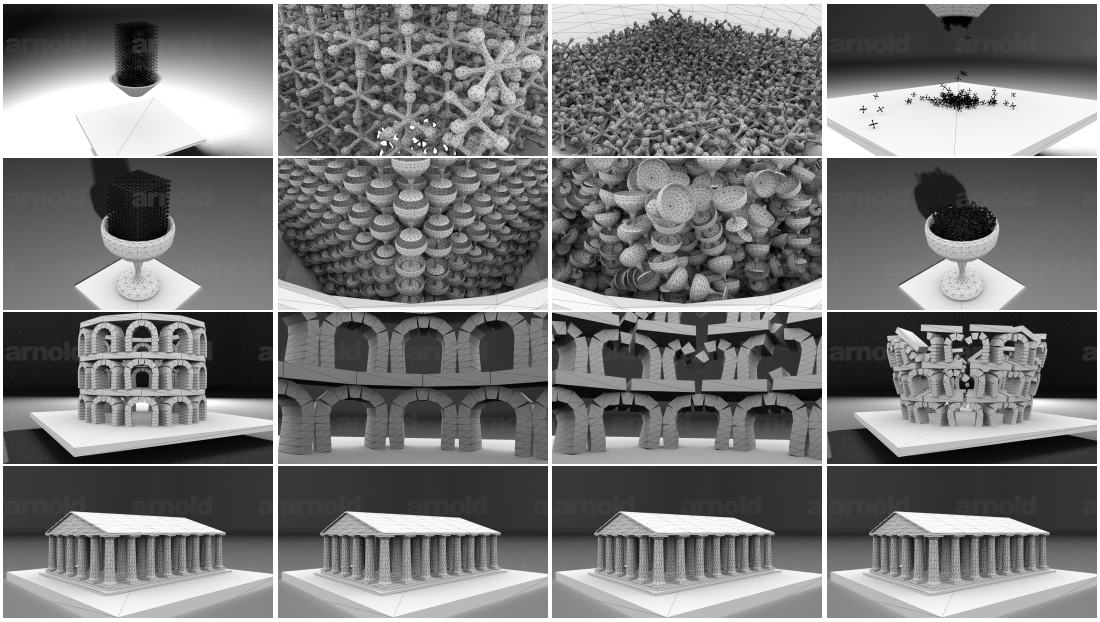

Figure 1: Examples of rigid body simulation using PROX schemes. Top row shows digital prototyping application studying jamming due to friction properties (Funnel), upper-middle: packing into a container (Chalices), middle-lower: building collapse due to poor design (Colosseum). Bottom demonstrates structured stacking (Temple).

ABSTRACT

Iterative methods are popular for solving contact force problems in rigid body dynamics. They are loved for their robustness and surrounded by mystery as to whether they converge or not. We provide a mathematical foundation for iterative (PROX) schemes based on proximal operators. This is a class of iterative Jacobi and blocked Gauss–Seidel variants that theoretically proven always converge and provides a flexible plug and play framework for exploring different friction laws. We provide a portfolio of experience for choosing r-Factor strategies for such schemes and we analyze the distribution of convergence behaviors. Our results indicate the Gauss-Seidel variant is superior in terms of delivering predictable convergence behaviour and hence should be preferred over Jacobi variants. Our results also suggest that Global r-Factor strategies are better for structured stacking scenarios and can achieve absolute convergence in more cases.

CCS CONCEPTS

• **Computing methodologies** → **Interactive simulation**; **Physical simulation**;

KEYWORDS

Contact Force Problems, Rigid Body Dynamics, Proximal Operators

ACM Reference format:
Kenny Erleben. 2017. Rigid Body Contact Problems using Proximal Operators. In *Proceedings of SCA '17, Los Angeles, CA, USA, July 28-30, 2017,* 12 pages.
DOI: 10.1145/3099564.3099575

SCA '17, Los Angeles, CA, USA

DOI: 10.1145/3099564.3099575

1 INTRODUCTION

Contact problems of rigid body dynamics are central to many applications in computer graphics, robotics and applied mechanics. These range from entertainment oriented contexts such as movies and games, to optimal control of robots, training simulators, risk assessments, cultural preservation, and digital prototyping as illustrated in Figure 1. One common objective for a simulator is to be both fast, scale to handle a large number of rigid bodies, deal with

non-convex shapes, accurately model the physics, and be robust as well as predictable in case of user interactions [Bender et al. 2014].

The Projected Gauss–Seidel (PGS) type of methods have become a de-facto standard for interactive entertainment. The robustness of PGS and Projected Succesive Over Relaxation (PSOR) variants are well known as is their poor convergence behaviour [Erleben 2007]. We provide a derivation of such iterative schemes using proximal operators. We name these schemes PROX methods to avoid confusion with the traditional PGS variants. We will present novel insight by taking a different path based on proximal operators than the usual PGS-type derivation such as the one done by Silcowitz-Hansen et al. [2010].

- The usual PGS complementarity based friction box-model variants are special cases of a PROX method with a fixed constant r-Factor (to be explained later). This is the reason why PGS can diverge and why the PROX method setting is theoretically guaranteed to always converge [Foerg et al. 2006; Parikh and Boyd 2014].
- PGS is tied to the box-model for performance reasons as it provides a small tight blocked memory footprint. PROX methods are in their very derivation not limited to such friction models.

We seek to provide a fundamental mathematical treatment to introduce and cement PROX methods to the computer graphics community. We claim that PGS variants are inferior and should be replaced by PROX methods as PROX schemes provide a theoretical guarantee of convergence and a flexible modeling of any convex multi-set friction law. For educational reasons and completeness in presentation, we show the model can express Newton style impact laws and post-stabilization. Further, we demonstrate

- Blocked r-Factor strategy is infeasible contrary to what other researchers have reported [Merlhiot 2007].
- Global and Local r-Factor strategies are both usable and provides complementary advantages; in terms of number of iterations and their ability to deal with structured stacks versus random piles.
- The variance of the distribution of convergence behaviour from Jacobi schemes are too large to guarantee predictable performance behavior. Hence, Gauss–Seidel variants are more predictable in their behavior.

The proximal operator model is derived directly from physical principles. One benefit of this model is that the iterative scheme for computing a solution more or less comes for free directly from the model it self. No extra discretization "hoops" are needed. Another benefit of this model is that it generalizes to general limit surfaces [Goyal et al. 1989] and can include friction torque and Coulomb–Contensou friction [Leine and Glocker 2003].

2 PREVIOUS WORK

For a general introduction to the field of rigid body simulation we refer to [Bender et al. 2014]. Here, we will focus mainly on work that is related to constraint-based iterative methods. Early work on Projected Gauss–Seidel type methods based on proximal operators exist in other fields [Jean and Moreau 1992; Jourdan et al. 1998] and our treatment of the proximal operator model is mostly educational. We provide new insight on adaptive r-Factors and present a different novel sub-solver for dealing with the projection onto friction cones.

The mathematical equivalence between the solution spaces of proximal operator models and complementarity formulations have been proven [Schindler et al. 2011]. Fixed constant r-Factor values for PROX schemes have been investigated and shown to be highly dependent on both the coefficient matrix $\mathbf{A}$ but also on the initial forces and velocities of the scene [Lu and Trinkle 2014; Schindler et al. 2011]. Our work is different in that we use an adaptive r-value. Remarkably, the dependency on initial values have been overlooked in other theoretical work [Foerg et al. 2006]. The adaptive r-value seems to cope with these issues, although we present no rigorous proofs of this.

In the fast frictional dynamics (FFD) method Gauss' principle of least constraint is solved using projections in a body-space setting [Kaufman et al. 2005]. Hence, one is really not solving for the normal force distribution, but rather projecting a post velocity of the bodies such that it agrees with the kinematic constraints. FFD is rather interesting as only one fully parallel sweep of projections over the bodies can resolve penetrating motion. PGS usually requires 5-30 sweeps where each sweep corresponds to one blocked sparse matrix vector (SpMV) multiplication [Erleben 2007; Tonge et al. 2012], followed by a simple box-projection on each block. FFD uses only one SpMV sweep over the bodies and performs a slightly more expensive projection for each body. It is evident that for some scenarios FFD can be quite fast. However, as the true normal force distribution is difficult to recover from resulting motions due to its ill-posed nature, it can be difficult to add a good model of friction to this scheme.

The staggered projections use a kinetic metric to formulate a method for solving the usual LCP contact problem as two staggered blocked projections [Kaufman et al. 2008]. The staggering combined with the kinetic metric implies that each blocked problem can be solved efficiently and accurately with a QP solver. This work was tied into a traditional isotropic planar Coulomb friction model and did not exploit the local nonlinear nature of each contact point as we do.

Nonlinear complementarity problem (NCP) contact models based on a Second-Order Cone Complementarity Problem (SOCCP) model have been investigated for hair fiber simulations [Daviet et al. 2011]. The SOCCP work relies on using a non-linear local model for friction solved using a modified Fischer–Burmeister (FB) Newton method. This makes the implementation of the friction model more coupled to the assembly of the Newton equation. In comparison the proximal operator approach offers a generalized projection viewpoint on the local solve that makes it more "plug-and-play" to implement new friction models. Our PROX methods share this local highly non-linear model aspect with the SOCCP work. Moreever, we are not aware of SOCCP having been applied to rigid bodies.

The PGS/PSOR complementarity formulations may similar to the proximal operator model replace the friction law with an analytical friction solution [Lacoursière 2007]. We show that PGS and PSOR are special PROX versions that use a fixed constant r-value instead of an adaptive r-Factor strategy.

The proximal operator model we adopt has commonalities to the cone complementarity problem (CCP) model that forms the

foundation for applying an accelerated projected gradient descent (APGD) method, a variant of Nesterov's method [Mazhar et al. 2015]. The CCP model results in an adhesion "component" from a convexification of the problem, it can be argued whether this is a desired model property or not as it is tied into a remodeling needed to provide a suitable model that will generate faster convergence rate. Our presentation does no such remodeling and retains the non-penetration constraints in traditional form.

To our knowledge we provide studies of PROX methods with general non-convex shapes and friction torque that was not investigated with the CCP or SOCCP works.

The proximal operator model for rigid body contact force problems opens up for a large class of numerical methods to be applied, collectively referred to as proximal algorithms [Parikh and Boyd 2014]. In this work we explore Jacobi and Gauss–Seidel variants of a kind of proximal gradient method. We note that the Gauss–Seidel variant can be seen as a blocked coordinate descent approach of the proximal gradient method. Proximal operators have also found their usage in Projective dynamics [Bouaziz et al. 2014; Narain et al. 2016] and recently primal dual optimization for fluids [Inglis et al. 2017] where the alternating direction method of multipliers (ADMM) and Chebyshev method [Wang 2015] are demonstrated to work well. The first method is a primal dual proximal algorithm and the later can be seen as a variant of an accelerated proximal gradient method. This can be interpreted as a movement in the field towards methods based on solving several local highly non-linear models possible in parallel and then combine their solutions globally. Hence, we claim there is a trend pushing toward using proximal operators in the field of computer graphics as a general modeling approach. Here we take the full-step in regards to rigid body contact problems and provide a foundation.

Compared to prior work on proximal operator models for contact problems [Alart and Curnier 1991; Foerg et al. 2006; Jean and Moreau 1992; Merlhiot 2007; Niebe 2013; Studer 2008], we present a unifying formalism not just for the mechanics of a rigid body system, but also for other sub-tasks such as impact and stabilization. This is important when constructing a simulator. Past work have used a Coulomb–Contensou friction model as we do [Leine and Glocker 2003]. However, we provide numerical details that allow for an efficient proximal point solve.

3 THE PROXIMAL OPERATOR MODEL

We will present a contact force model for a single contact point. The notation makes use of a proximal operator, $\text{prox}_C(\mathbf{z})$. The proximal point of a convex set C to a point $\mathbf{z}$ is the point in C that minimizes the distance to $\mathbf{z}$,

$$\text{prox}_C(\mathbf{z}) \equiv \arg\min_{\mathbf{x}\in C} \| \mathbf{z}-\mathbf{x} \|^2, \quad \mathbf{z}\in\mathbb{R}^n. \tag{1}$$

this is called the proximal operator [Parikh and Boyd 2014]. Given the normal contact velocity $\mathbf{v}_N \in \mathbb{R}$ then the non-penetration constraints can be stated as

$$\mathbf{v}_N \geq 0, \quad \lambda_N \geq 0, \quad \text{and} \quad \mathbf{v}_N\lambda_N = 0, \tag{2}$$

or equivalently the fixed point relationship have the same solution as the non-penetration constraint [Jourdan et al. 1998]

$$\lambda_N = \text{prox}_{\mathcal{N}}\left(\lambda_N - r_N\mathbf{v}_N\right) \quad \text{for} \quad r_N > 0 \tag{3}$$

where $\mathcal{N} = \{\gamma \in \mathbb{R} \,|\, \gamma \geq 0\}$, $\lambda_N \in \mathbb{R}$ is the magnitude of the normal force, and r_N is a mathematical scalar variable named an r-Factor. The equation holds for all $r_N > 0$ and the exact r-value will have practical impact on convergence. Given the coefficient of friction $\mu > 0$ and the magnitude of normal force $\lambda_N \geq 0$, then the planar friction force $\lambda_F \in \mathbb{R}^2$ is bounded by the friction cone,

$$\lambda_F \in \mathcal{F}(\mu\lambda_N) \tag{4}$$

where the friction cone could be defined as

$$\mathcal{F}(\mu\lambda_N) \equiv \left\{ \gamma \in \mathbb{R}^2 \,\middle|\, \| \gamma \| \leq \mu\lambda_N \right\}. \tag{5}$$

This is the typical isotropic Coulomb friction law. For the proximal operator model to work we only need $\mathcal{F}$ to be a convex set. According to principle of maximum dissipation the friction force should dissipate as much work as possible from the system. Given the friction force $\lambda_F \in \mathcal{F}(\mu\lambda_N)$ and the tangential contact velocity, $\mathbf{v}_F \in \mathbb{R}^2$, then the dissipation power, P_{λ_F}, is,

$$P_{\lambda_F} = \lambda_F^T\mathbf{v}_F. \tag{6}$$

Note that dissipation implies that $P_{\lambda_F} < 0$. According to the principle of maximum dissipation the power, P_{γ_F}, done by any other possible friction force, $\gamma_F \in \mathcal{F}(\mu\lambda_N)$, has to be larger than or equal to P_{λ_F}. From this we have

$$P_{\lambda_F} \leq P_{\gamma_F}, \tag{7a}$$

$$\lambda_F^T\mathbf{v}_F \leq \gamma_F^T\mathbf{v}_F, \tag{7b}$$

$$0 \leq \gamma_F^T\mathbf{v}_F - \lambda_F^T\mathbf{v}_F = \left(\gamma_F - \lambda_F\right)^T\mathbf{v}_F. \tag{7c}$$

Resulting in the condition

$$\forall\gamma_F \in \mathcal{F}(\mu\lambda_N) \quad \text{and} \quad \left(\gamma_F - \lambda_F\right)^T\mathbf{v}_F \geq 0. \tag{8}$$

A solution to this variational inequality is equivalent to the fixed point of the proximal operator,

$$\lambda_F = \text{prox}_{\mathcal{F}(\mu\lambda_N)}\left(\lambda_F - r_F\mathbf{v}_F\right) \quad \text{for} \quad r_F > 0. \tag{9}$$

Since $\mathbf{v} = [\mathbf{v}_N^T\mathbf{v}_F^T]^T$ can be written as a linear combination of $\lambda = [\lambda_N^T\lambda_F^T]^T$ we have $\mathbf{v} = \mathbf{A}\lambda + \mathbf{b}$. The equation holds for all values of the variable r_F but the actual value used will have impact on convergence. Substituting $\mathbf{v}$ into the proximal operator lead to a fixed point problem,

$$\underbrace{\begin{bmatrix}\lambda_N\\ \lambda_F\end{bmatrix}}_{\lambda} = \underbrace{\begin{bmatrix}\text{prox}_{\mathcal{N}}\left(\lambda_N - r_N\left(\mathbf{A}_{NN}\lambda_N + \mathbf{A}_{NF}\lambda_F + \mathbf{b}_N\right)\right)\\ \text{prox}_{\mathcal{F}(\mu\lambda_N)}\left(\lambda_F - r_F\left(\mathbf{A}_{FN}\lambda_N + \mathbf{A}_{FF}\lambda_F + \mathbf{b}_F\right)\right)\end{bmatrix}}_{\mathbf{F}(\lambda)} \tag{10}$$

Now one could compute the iterates, $\lambda^{k+1} = \mathbf{F}(\lambda^k)$. More details are given in Section 4. The iteration sequence converges locally if the spectral radius of the Jacobian,

$$\rho\left(\frac{\partial\mathbf{F}(\lambda)}{\partial\lambda}\right) < 1 \tag{11}$$

remains limited and smaller than one. This requirement can be realized by making a suitable choice of the parameters r_N and r_F [Foerg et al. 2006; Niebe 2013; Parikh and Boyd 2014; Studer 2008].

3.1 Multiple Contact Points

We will now extend our notation to multiple contact points, add impact modeling, and generalize the friction model. The Newton–Euler equations and kinematic maps are given by

$$\mathbf{M}\dot{\mathbf{u}} = \mathbf{h} + \mathbf{J}^T\lambda, , \tag{12a}$$

$$\dot{\mathbf{q}} = \mathbf{u} \tag{12b}$$

where $\mathbf{u}$ is the generalized velocity vector, $\mathbf{q}$ is generalized position vector, $\mathbf{M}$ is the mass matrix, $\mathbf{J}$ is the contact Jacobian, and $\mathbf{h}$ holds external and gyroscopic force terms, see Appendix A for details. The relative contact velocity is computed from

$$\mathbf{v} = \mathbf{J}\mathbf{u}\,. \tag{13}$$

Using the Augmented Lagrangian method, the contact constraints are formulated using proximal operators,

$$\forall i \quad \lambda_{N_i} = \text{prox}_{\mathcal{N}_i}\left(\lambda_{N_i} - r_{N_i}\left(\mathbf{v}^+_{N_i} + \varepsilon_{N_i}\mathbf{v}^-_{N_i}\right)\right)\,, \tag{14a}$$

$$\forall i \quad \lambda_{F_i} = \text{prox}_{\mathcal{F}_i}\left(\lambda_{F_i} - r_{F_i}\left(\mathbf{v}^+_{F_i} + \varepsilon_{F_i}\mathbf{v}^-_{F_i}\right)\right) \tag{14b}$$

where sub-index i refers the contact point index and

$$\mathcal{N}_i \equiv \{\gamma \in \mathbb{R}\,|\,\gamma \geq 0\}\,, \tag{15a}$$

$$\mathcal{F}_i \equiv \left\{\gamma \in \mathbb{R}^3 \,\middle|\, \left(\frac{\gamma_S}{a}\right)^2 + \left(\frac{\gamma_T}{b}\right)^2 + \left(\frac{\gamma_\tau}{c}\right)^2 \leq 1\right\} \tag{15b}$$

and

$$a = \mu_{S_i}\lambda_{N_i}, \quad b = \mu_{T_i}\lambda_{N_i}, \quad \text{and} \quad c = \mu_{\tau_i}\lambda_{N_i}\,. \tag{16}$$

Above we changed $\mathcal{F}$ to be the Coulomb–Contensou friction model, a generalization over the previous definition in (5). We used $\mathbf{v}^-$ and $\mathbf{v}^+$ to denote pre- and post-impact contact velocities. We have also applied a Newton impact law. Usually one has $\varepsilon_{N_i} \in [0,1]$ and $\varepsilon_{F_i} = 0$. Further, one may choose $r = r_{N_i} = r_{F_i} > 0$. Although other r-Factor strategies can be used. Using a time-discretization on the differential equations we have

$$\mathbf{u}^{t+\Delta t} = \mathbf{u}^t + \Delta t\mathbf{M}^{-1}\mathbf{h} + \mathbf{M}^{-1}\mathbf{J}^T\lambda\,. \tag{17}$$

Both $\mathbf{M}$ and $\mathbf{J}$ depends on $\mathbf{q}$, $\mathbf{h}$ depends on both $\mathbf{q}$ and $\mathbf{u}$ due to gyroscopic force terms. For now we ignore these dependancies and will specify them when we introduce the time-stepping method in Section 4.3. The pre-impact contact velocities are given by $\mathbf{v}^- = \mathbf{J}\mathbf{u}^t$ and the post-impact velocities are given by $\mathbf{v}^+ = \mathbf{J}\mathbf{u}^{t+\Delta t}$. Multiplying equation (17) by $\mathbf{J}$ from the left yields,

$$\mathbf{v}^+ = \mathbf{J}\mathbf{M}^{-1}\mathbf{J}^T\lambda + \mathbf{J}\mathbf{u}^t + \Delta t\mathbf{J}\mathbf{M}^{-1}\mathbf{h}\,. \tag{18}$$

Defining

$$\mathbf{z} = \lambda - \mathbf{R}\left(\mathbf{v}^+ + \mathbf{E}\mathbf{v}^-\right)\,, \tag{19a}$$

$$= \lambda - \mathbf{R}\left(\mathbf{J}\mathbf{M}^{-1}\mathbf{J}^T\lambda + \mathbf{J}\mathbf{u}^t + \Delta t\mathbf{J}\mathbf{M}^{-1}\mathbf{h} + \mathbf{E}\mathbf{J}\mathbf{u}^t\right) \tag{19b}$$

where $\mathbf{R}$ and $\mathbf{E}$ are diagonal matrices containing r-Factors and ε-coefficients. Now the proximal operators read

$$\forall i \quad \lambda_{N_i} = \text{prox}_{\mathcal{N}_i}\left(\mathbf{z}_{N_i}\right)\,, \tag{20a}$$

$$\forall i \quad \lambda_{F_i} = \text{prox}_{\mathcal{F}_i}\left(\mathbf{z}_{F_i}\right)\,. \tag{20b}$$

The solutions of these are given by

$$\forall i \quad \lambda_{N_i} = \max\left(0, \mathbf{z}_{N_i}\right) \tag{21a}$$

$$\forall i \quad \lambda_{F_i} = \begin{cases} \mathbf{z}_{F_i} & ;\mathbf{z}_{F_i} \in \mathcal{F}_i \\ \min_{\gamma\in\mathcal{F}_i} \parallel \mathbf{z}_{F_i} - \gamma \parallel^2 & ;\text{otherwise} \end{cases} \tag{21b}$$

The last case computes the closest point in the $\mathcal{F}_i$ to the point $\mathbf{z}_{F_i}$. This can be solved as described in Section 4.1.

The linear complementarity problem formulations use a polyhedra friction cone approximation combined with a discrete version of principle of maximum dissipation [Bender et al. 2014]. This leads to an inner friction cone approximation, which can be made more accurate by adding more faces to the friction cone approximation. The drawback is that one Lagrange multiplier is needed for each cone face of the friction cone. Further, one auxiliary variable is needed to select the discrete direction closest to direction of maximum dissipation. To overcome the large number of variables needed in the linear complementarity formulation some physics engines ignore the coupling between the friction directions and each friction direction is modeled as a boxed-constraint with variable lower and upper bounds [Silcowitz-Hansen et al. 2010]. The result is an outer friction cone approximation and no direct modeling of the direction of maximum dissipation. The proximal operator model we have described uses no approximation of the friction cone and explicitly picks the direction of maximum dissipation. Further, the proximal operator model can easily be extended to support any kind of contact model as long as the friction cone can be described as a convex set. Theoretically, one may use limit surfaces in the model [Goyal et al. 1989]. In this work, we settle for a Coulomb law combined with frictional torque.

4 THE ITERATIVE PROX METHODS

A Jacobi scheme can be used to solve the fixed point problem in (20). This consists of first computing the iterate,

$$\mathbf{z}^k = \lambda^k - \mathbf{R}\left(\underbrace{\mathbf{J}\mathbf{M}^{-1}\mathbf{J}^T}_{\mathbf{A}}\lambda^k + \underbrace{\mathbf{J}\mathbf{u}^t + \Delta t\mathbf{J}\mathbf{M}^{-1}\mathbf{h} + \mathbf{E}\mathbf{J}\mathbf{u}^t}_{\mathbf{b}}\right), \tag{22a}$$

$$= \lambda^k - \mathbf{R}\left(\mathbf{A}\lambda^k + \mathbf{b}\right)\,. \tag{22b}$$

Next one solves for the next iterate of λ^{k+1} using,

$$\forall i \quad \lambda^{k+1}_{N_i} = \text{prox}_{\mathcal{N}_i}\left(\mathbf{z}^k_{N_i}\right)\,, \tag{23a}$$

$$\forall i \quad \lambda^{k+1}_{F_i} = \text{prox}_{\mathcal{F}^k_i}\left(\mathbf{z}^k_{F_i}\right) \tag{23b}$$

where $\mathcal{F}^k_i$ is defined using the value of the k^{th} iterate, $a = \mu_{S_i}\lambda^k_{N_i}$, $b = \mu_{T_i}\lambda^k_{N_i}$ and $c = \mu_{N_i}\lambda^k_{N_i}$. As it stands the scheme is inherently easily parallelized, although we do not exploit parallelism in this work.

A Gauss–Seidel scheme can be created from the Jacobi scheme. The idea is to always use the most updated λ-values in any computation. Let $\mathbf{z}$ denote the most updated value at all times. The normal and friction solves for the i^{th} contact is now computed using,

$$\lambda^{k+1}_{N_i} = \text{prox}_{\mathcal{N}_i}\left(\mathbf{z}_{N_i}\right)\,, \tag{24a}$$

$$\lambda^{k+1}_{F_i} = \text{prox}_{\mathcal{F}^{k+1}_i}\left(\mathbf{z}_{F_i}\right)\,. \tag{24b}$$

Observe that $\mathcal{F}_i^{k+1}$ is used instead of $\mathcal{F}_i^k$. After having solved for $\lambda_{N_i}^{k+1}$ and $\lambda_{F_i}^{k+1}$ then all $\mathbf{z}$ dependent entries must be updated before moving on to the next contact. For this update we exploit the factorization technique from [Erleben 2007]. The idea is to write $\mathbf{z}$ as $\mathbf{z} \equiv \lambda + \mathbf{R}\,(\mathbf{J}\mathbf{w} + \mathbf{b})$ and use $\mathbf{w} \equiv \mathbf{M}^{-1}\mathbf{J}^T\lambda$. For convenience we introduce the index set of all the bodies, $\mathcal{B}$, and the index set of the i^{th} contact point, $\mathcal{I} \equiv \{N_i, F_i\}$. Now we can find the most updated value of $\mathbf{z}_\mathcal{I}$ before computing the contact forces,

$$\mathbf{z}_\mathcal{I} = \lambda_\mathcal{I}^k - \mathbf{R}_i\,(\mathbf{J}_{\mathcal{I}\mathcal{B}}\,\mathbf{w} + \mathbf{b}_\mathcal{I})\ . \tag{25}$$

After having computed the contact forces we can update $\mathbf{w}$ so its value is ready for the next contact,

$$\mathbf{w} = \left(\mathbf{M}^{-1}\mathbf{J}^T\right)_{\mathcal{B},\mathcal{I}}\left(\lambda_\mathcal{I}^{k+1} - \lambda_\mathcal{I}^k\right)\ . \tag{26}$$

In this work we compare the Jacobi variant against the Gauss–Seidel variant. We have summarized the complete Algorithms in 1 and 2. We will keep on iterating using the PROX scheme until the residual has absolutely or relatively converged, or we exceed a maximum iteration count. A residual can defined as

$$\mathbf{r}^{k+1} = \lambda^{k+1} - \lambda^k \tag{27}$$

We are missing two intrinsic parts of the PROX scheme, one is how the closest points on an ellipsoid is computed and the other is how to control the r-Factors.

Data: $\mathcal{K}$: indices of all contacts, $\mathcal{B}$ indices of all bodies, $\mathbf{J}, \mathbf{M}, \mathbf{b}, \mathbf{R}, \lambda^0, \nu$
Result: λ^k

$\left(k, \lambda^k, r^k\right) \leftarrow \left(0, \lambda^0, \infty\right)$;
while *not converged* **do**
  $\mathbf{w} \leftarrow \mathbf{M}^{-1}\,\mathbf{J}^T\lambda^k$;
  $\mathbf{z} \leftarrow \lambda^k - \mathbf{R}\,(\mathbf{J}\,\mathbf{w} + \mathbf{b})$;
  foreach $i \in \mathcal{K}$ **do**
    $\mathcal{I} \equiv \{n, f\} \leftarrow$ indices of block N_i, F_i;
    $\lambda_n^{k+1} \leftarrow \text{prox}_{\mathcal{N}_i}\,(\mathbf{z}_n)$;
    $\lambda_f^{k+1} \leftarrow \text{prox}_{\mathcal{F}_i(\lambda_n^k)}\left(\mathbf{z}_f\right)$;
  end
  $r^{k+1} = \| \lambda^{k+1} - \lambda^k \|_\infty$;
  if $r^{k+1} > r^k$ **then**
    $\mathbf{R} \leftarrow \nu\mathbf{R}$;
  else
    $\left(\lambda^k, r^k, k\right) \leftarrow \left(\lambda^{k+1}, r^{k+1}, k+1\right)$;
  end
end

Algorithm 1: The PROX Jacobi variant with adaptive r-Factor strategy. For efficiency the matrix product $\mathbf{M}^{-1}\mathbf{J}^T$ may be precomputed and stored in transposed form.

4.1 Closest Point on Ellipsoid

For isotropic friction models or omission of torque effects the ellipsoid collapses to a 2D ellipse and finding the closest point can be done analytically by finding the largest positive real root of a 4^{th} order polynomial. However, we consider the Coulomb–Contensou

Data: $\mathcal{K}$: indices of all contacts, $\mathcal{B}$ indices of all bodies, $\mathbf{J}, \mathbf{M}, \mathbf{b}, \mathbf{R}, \lambda^0, \nu$
Result: λ^k

$\left(k, \lambda^k, r^k\right) \leftarrow \left(0, \lambda^0, \infty\right)$;
while *not converged* **do**
  $\mathbf{w} \leftarrow \mathbf{M}^{-1}\,\mathbf{J}^T\lambda^k$;
  foreach $i \in \mathcal{K}$ **do**
    $\mathcal{I} \equiv \{n, f\} \leftarrow$ indices of block N_i, F_i;
    $\mathbf{z}_\mathcal{I} \leftarrow \lambda_\mathcal{I}^k - \mathbf{R}_i\,(\mathbf{J}_{\mathcal{I}\mathcal{B}}\,\mathbf{w} + \mathbf{b}_\mathcal{I})$;
    $\lambda_n^{k+1} \leftarrow \text{prox}_{\mathcal{N}_i}\,(\mathbf{z}_n)$;
    $\lambda_f^{k+1} \leftarrow \text{prox}_{\mathcal{F}_i(\lambda_n^{k+1})}\left(\mathbf{z}_f\right)$;
    $\mathbf{w} \leftarrow \left(\mathbf{M}^{-1}\mathbf{J}^T\right)_{\mathcal{B},\mathcal{I}}\left(\lambda_\mathcal{I}^{k+1} - \lambda_\mathcal{I}^k\right)$;
  end
  $r^{k+1} = \| \lambda^{k+1} - \lambda^k \|_\infty$;
  if $r^{k+1} > r^k$ **then**
    $\mathbf{R} \leftarrow \nu\mathbf{R}$;
  else
    $\left(\lambda^k, r^k, k\right) \leftarrow \left(\lambda^{k+1}, r^{k+1}, k+1\right)$;
  end
end

Algorithm 2: The PROX Gauss–Seidel variant with adaptive r-Factor strategy. The product $\mathbf{M}^{-1}\mathbf{J}^T$ may be precomputed as for the Jacobi variant.

law. That means our ellipsoid models anisotropic spatial friction with torque effects and corresponds to computing the roots of 6^{th} order polynomial. Hence, we seek a robust and efficient numerical method. For general convex sets one may use the Minikowsky difference to transform the problem into that of finding the closest point between origin and a convex set. The GJK algorithm is an example of such an algorithm exploiting these ideas. The advantage of a GJK approach is that it is generally applicable to any friction model as long as the limit surfaces defines a convex set. However, in our case the friction model is an ellipsoid and the generality of GJK is uncalled for. We exploit the corresponding 6^{th} order polynomial and apply a root search method to search for the root.

Let $a, b, c > 0$ be given. These parameters defines an ellipsoid surface. The ellipsoid surface consists of all points $\mathbf{x} \in \mathbb{R}^3$ where

$$f(\mathbf{x}) = \mathbf{x}^T\mathbf{K}\mathbf{x} - 1 = 0 \tag{28}$$

and $\mathbf{K}$ is a positive diagonal matrix given by

$$\mathbf{K} = \begin{bmatrix} \frac{1}{a^2} & 0 & 0 \\ 0 & \frac{1}{b^2} & 0 \\ 0 & 0 & \frac{1}{c^2} \end{bmatrix}. \tag{29}$$

Given a point $\mathbf{z} \in \mathbb{R}^3$ outside the ellipsoid, $f(\mathbf{z}) > 0$, then we wish to find the closest point, $\mathbf{x}^*$ on the ellipsoid surface to $\mathbf{z}$,

$$\mathbf{x}^* = \arg\min_x \frac{1}{2}\,\| \mathbf{x} - \mathbf{z} \|^2 \quad \text{subject to} \quad f(\mathbf{x}) = 0 \tag{30}$$

The first order optimality conditions yields

$$\mathbf{x}^* - \mathbf{z} - \lambda^*\nabla f(\mathbf{x}^*) = 0. \tag{31}$$

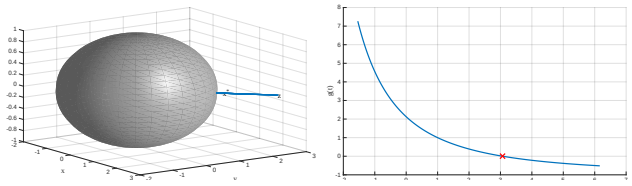

Figure 2: Left shows the spatial search path from z to x*and right shows the corresponding function $g(t)$ with the root solution marked by a red dot.

Using $\nabla f(\mathbf{x}^*) = 2\mathbf{K}\mathbf{x}^*$ and $t = 2\lambda^*$ we have

$$\mathbf{x}^* = (t\mathbf{K} + \mathbf{I})^{-1}\mathbf{z}\,. \tag{32}$$

Further, we must have that $\mathbf{x}^*$ lies on the surface of the ellipsoid,

$$f(\mathbf{x}^*) = f((t\mathbf{K} + \mathbf{I})^{-1}\mathbf{z}) = 0\,. \tag{33}$$

Defining

$$g(t) \equiv f((t\mathbf{K} + \mathbf{I})^{-1}\,\mathbf{z})\,, \tag{34}$$

$$= \frac{a^2 z_1^2}{(a^2+t)^2} + \frac{b^2 z_2^2}{(b^2+t)^2} + \frac{c^2 z_3^2}{(c^2+t)^2} - 1\,. \tag{35}$$

We observe that our problem of finding the closest point has been reformulated into the problem of finding a root of $g(t) = 0$. From geometry we know that the curve given by $g(t)$ can pierce an non-degenerate ellipsoid in at most two points. If $f(\mathbf{z}) > 0$ then we are seeking the intersection point with $t > 0$. If $f(\mathbf{z}) \leq 0$ then $\mathbf{z}$ is already in the ellipsoid and we simply can return $\mathbf{z}$ as our solution.

For $t > 0$ we have $\frac{d}{dt}g(t) < 0$. However, $g(t)$ is very steep for small t-values and very flat for larger t-values as seen in Figure 2. Thus, we find a binary search numerical method to be a good choice for finding the positive root of $g(t)$. Initially we used a Newton–Raphson method, we found this to diverge or find negative roots when the root approaches the flat parts of the g-function. An initial bracketing technique is adopted for our binary search method. The minimum t-value for our search interval is given by $t_{\min} = 0$ the maximum t-value for the search interval is guessed as

$$t_{\text{guess}} = \max\{a, b, c\} \parallel \mathbf{z} \parallel\ . \tag{36}$$

The maximum t-value is then given by the minimum non-negative integer, k where $g(t_{\max}) < 0$ and $t_{\max} \equiv \alpha^k t_{\text{guess}}$, where $\alpha > 1$ is the scalar expansion coefficient of the interval. In our implementation we use $\alpha = 1.5$, we made no attempts to tune the expansion coefficient. In order to make the binary search method more robust in the sense of having better precision the problem is scaled to be within the unit-cube. That is a scaling factor is computed as

$$s = \frac{1}{\max\{1, a, b, c, z_1, z_2, z_3\}} \tag{37}$$

and then re-define the problem as follows.

$$\mathbf{z} \leftarrow s\mathbf{z}, \quad \text{and} \quad \{a, b, c\} \leftarrow \{sa, sb, sc\}\,. \tag{38}$$

However, when the scheme has converged one must remember to convert the solution back to the unscaled problem.

4.2 The r-Factor Strategies

It has been proven that if the r-value is sufficiently low then the fixed point schemes will converge [Foerg et al. 2006; Studer 2008]. Convergence speed is expected to be worse for small r-values than larger r-values. Hence algorithmically, we apply an adaptive back-tracking approach for adjusting the r-values. Regardless of chosen strategy, if we detect divergence while iterating, ie. the norm of the residual $\parallel \mathbf{r}^{k+1} \parallel > \parallel \mathbf{r}^k \parallel$ then we drop updating λ^{k+1}. Instead we roll-back to λ^k and reduce all r-values by a user specified fraction ν. For Global and Blocked strategies we use $\nu = 0.5$ and for Local strategy we use $\nu = 0.9$. These values are not critical but was tuned experimentally.

For the Global r-Factor strategy we simply initialize all r-values of all constraints to a single global constant value. Ideally based on theory [Foerg et al. 2006] one should initialize this to the reciprocal of the sum of the absolute values of the minimum and maximum eigenvalues of $\mathbf{A}$. However, these are not easily obtained and we found that simply hardwiring the initial r-value to say 5 works fine for our examples.

For the Local r-Factor strategy one would initially choose the r-value of the i^{th} variable to be

$$r_i = \frac{1}{\mathbf{A}_{ii}}\,. \tag{39}$$

Comparing this choice to the algebraic form of the variable updates used in PGS and PSOR we immediately recognize this as using a PGS-strategy for setting the initial value. If one used $r_i = \frac{\gamma}{\mathbf{A}_{ii}}$ where γ is the successive over-relaxation coefficient then the PSOR variant is achieved. Hence, one may replace $\mathcal{F}_i$ with a box-model and keeping Local r-values as fixed constants and the PROX scheme will deteriorate into PGS/PSOR variants.

The last strategy we consider is called Blocked. The idea is to replace the scalar r-values with small blocked matrices build from the diagonal blocks of the $\mathbf{A}$-matrix. Let us consider the k^{th} block corresponding to the k^{th} contact, then the r-Factor strategy uses the block $\mathbf{R}_k \in \mathbb{R}^{4\times 4}$,

$$\mathbf{R}_k = \begin{bmatrix} \frac{1}{\mathbf{A}_{ii}} & 0 \\ 0 & \mathbf{A}^{-1}_{s:\tau,s:\tau} \end{bmatrix} \tag{40}$$

where $n = 4k$, $s = n+1$, $t = n+2$, $\tau = n+3$. The Blocked strategy have been reported to be working quite well [Merlhiot 2007].

Note r-Factors are not the same as numerical damping or regularization as known from LCP methods [Cottle et al. 1992]. For those the coefficient matrix is changed like $\mathbf{A} \leftarrow \mathbf{I}\rho + \mathbf{A}$ for some damping parameter $\rho > 0$. This changes the model and adds compliance to the contact. The r-Factors does not change the model, the solutions hold for all r-values. The specific r-value only affect the convergence constants.

4.3 Time-Stepping Methods

The PROX schemes can be used with different time-stepping methods. We favor the mid-point scheme of Moreau [Moreau 1999]. It consists of using an explicit half-step position update

$$\mathbf{q}^{t+\frac{1}{2}} = \mathbf{q}^t + \frac{\Delta t}{2}\mathbf{u}^t\,. \tag{41}$$

Then one computes $\mathbf{M} = \mathbf{M}(\mathbf{q}^{t+\frac{1}{2}}), \mathbf{J} = \mathbf{J}(\mathbf{q}^{t+\frac{1}{2}})$ and $\mathbf{h} = \mathbf{h}(\mathbf{q}^{t+\frac{1}{2}}, \mathbf{u}^t)$ and solves

$$\lambda = \text{PROX}\left(\mathbf{q}^{t+\frac{1}{2}}, \mathbf{u}^t, \Delta t, \ldots\right), \tag{42a}$$

$$\mathbf{u}^{t+1} = u^t + \Delta t \mathbf{M}^{-1}\mathbf{h} + M^{-1}\mathbf{J}^T\lambda, \tag{42b}$$

$$\mathbf{q}^{t+1} = \mathbf{q}^{t+\frac{1}{2}} + \frac{\Delta t}{2}\mathbf{u}^{t+1}, \tag{42c}$$

where PROX (. . .) denotes invokation of the PROX scheme developed above, see Algorithm 1 and 2. Previous work on interactive simulation have used a simple semi-implicit scheme [Erleben 2007]. We favor the Moreau variant due to the mid-point evaluation adds some "softness" into how contact is detected compared to the simpler semi-implicit scheme.

4.4 Constraint Stabilization

Our velocity based formulation solves constraints on a velocity level. This means that numerical drift will occur on position level constraints. Regardless of how many iterations used there will be some inaccuracy in the constraint forces. Stabilization can be done in various ways. A Baumgarte type of stabilization can be done by adding a penalty term to the kinematic constraints. The approach we derive here is similar in spirit to the one by [Baraff 1993] and [Cline and Pai 2003]. Given a finite displacement, $\Delta\mathbf{q} = \mathbf{q}^{\text{corrected}} - \mathbf{q}^{\text{initial}}$, of the bodies then this would yield the first order gap displacement,

$$\Delta\mathbf{g} = \mathbf{J}\Delta\mathbf{q}, \tag{43}$$

where the gap displacement is given by $\Delta\mathbf{g} = \mathbf{g}^{\text{corrected}} - \mathbf{g}^{initial}$. From first-order physics we have the equation of motion,

$$\mathbf{M}\Delta\mathbf{q} = \mathbf{J}^T\lambda\,. \tag{44}$$

In this formulation λ is the contact displacements and $\mathbf{J}^T$ transforms the contact displacements into body displacements. Finally, body displacements are weighted by the body masses. The effect is that linear contact displacements are distributed in a physical plausible way as linear and rotational displacements of the bodies. From the equation of motion we derive,

$$\Delta\mathbf{q} = \mathbf{M}^{-1}\mathbf{J}^T\lambda. \tag{45}$$

Next we substitute into our kinematic gap equation,

$$\Delta\mathbf{g} = \mathbf{J}\mathbf{M}^{-1}\mathbf{J}^T\lambda. \tag{46}$$

We seek a body displacement such that the gap function becomes non-negative.

$$\mathbf{g}^{\text{corrected}} = \Delta\mathbf{g} + \mathbf{g}^{initial} \geq 0\,. \tag{47}$$

Contact displacements can only "push" bodies appart which means $\lambda \geq 0$. Lastly we can not have a contact displacement at a separated contact. Thus, letting $\mathbf{g} = \mathbf{g}^{initial}$, we wish to find a body displacement such that

$$(\Delta\mathbf{g} + \mathbf{g}) \geq 0, \lambda \geq 0 \quad \text{and} \quad \lambda^T(\Delta\mathbf{g} + \mathbf{g}) = 0\,. \tag{48}$$

Substitution of (46) yields

$$\left(\mathbf{J}\mathbf{M}^{-1}\mathbf{J}^T\lambda + \mathbf{g}\right) \geq 0, \lambda \geq 0 \quad \text{and} \quad \lambda^T\left(\mathbf{J}\mathbf{M}^{-1}\mathbf{J}^T\lambda + \mathbf{g}\right) = 0\,. \tag{49}$$

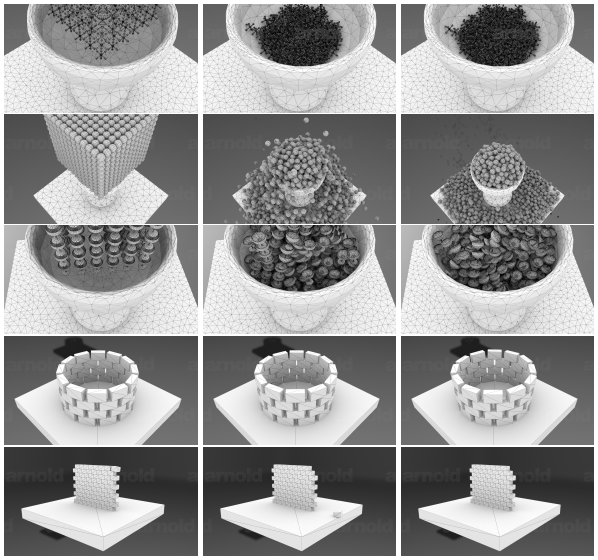

Figure 3: Test scenes used for r-Factor strategy and Jacobi vs. Gauss–Seidel schemes studies. Top shows the Jacks scene, Second row shows Spheres scene, Third row shows Glasses scene, and Fourth and Fifth rows show the Tower and Wall scenes. From left to right frames are show at 0.5s, 2.5s and 5s.

A linear complementarity problem, which can be recast as a proximal operator model,

$$\lambda = \text{prox}_{\mathcal{N}}\left(\lambda - r\left(\mathbf{J}\mathbf{M}^{-1}\mathbf{J}^T\lambda + \mathbf{g}\right)\right), \tag{50}$$

for all $r > 0$ where $\mathcal{N} = \mathcal{N}_1 \times \cdots \times \mathcal{N}_n$. Having solved for λ a position update can be performed.

$$\mathbf{q}^{\text{corrected}} = \mathbf{q}^{\text{initial}} + \mathbf{M}^{-1}\mathbf{J}^T\lambda\,. \tag{51}$$

This can be seen as an instantaneous change of the body positions disregarding velocities, external forces or velocity dependent forces. The scheme is of course based on a linearization of the true gap function as such it may not be able to resolve the errors with a single step and several steps have to be taken.

5 RESULTS

Basic scene statistics of the test scenes used are found in Table 1. Observe that the mesh shapes create a great number of contact points even though the number of objects corresponds to small to medium size scenes. Our simulator uses discrete collision detection (DCD) with a grid-based algorithm for broad phase collision detection and a kDOP BVH algorithm for narrow phase collision detection. Contact points are generated using a closest points algorithm for the Temple scene whereas the other scenes apply either the most opposing face method of [Niebe 2013] or the growth distance based technique from [Silcowitz et al. 2010]. Box shapes are handled with the improved box test from [Erleben 2005]. Deterministic ordering and filtering of redundant contacts are done with lexicographic sorting. For all the tests we present we wish to stress the PROX scheme solver and observe possible artefacts coming

Name	Objects (#)	Vertices (#)	Tetrahedra (#)
Tower	49	441	588
Wall	81	-	-
Glasses	362	590727	1825958
Jacks	362	600447	1947278
Temple	639	16990	32954
Colosseum	649	80438	244956
Spheres	1730	264915	775953
Funnel	2002	1596701	4676267
Chalices	4004	1102561	3365707

Table 1: Statistics for our test scenes. Wall uses box shapes and the remaining scenes use tetrahedral meshes. Each Jack shape is roughly 35 kg, A sphere is about 68 kg, a small glass is 91 kg, Colosseum and temple stones are approx. 4000 kg each. Friction coefficients are: $\mu_S = 0.75$, $\mu_t = 0.75$ and $\mu_\tau = 0.01$. Test scenes have varying number of contact points. Funnel have approximately 10^6 contacts. The other larger scenes have 10^5 and the smaller ones have no more than 10^2 contacts.

from this scheme only. Hence, we have turned of post stabilization, and use quite aggressive thresholds and a quite large maximum iteration count. There are visual artefacts arising due to using DCD with time-step sizes of 0.01 seconds and not applying stabilization at all. We do note it is remarkable how little drift we observe due to missing stabilization. Artefacts are caused by tunneling and ad-hoc rules for determining contact normals. These are not related to the constraint sovler and hence we leave these two problems for future work.

Figure 4 compares quantitive measures for different r-Factor strategies. In all test cases we use the same merit function, the infinity norm of equation (27), to measure convergence toward a fixed point. We specify a maximum iteration count of 1000 and absolute and relative stopping thresholds of 0.001. We measured convergence rates of the first 450 simulation steps of five test scenes and display results using quartiles to illustrate the distribution of behaviour. Hence, the results are based on 2250 convergence plots in total. The scenes: Glasses, Jacks, and Spheres were selected to demonstrate variation in shape complexity and Tower and Wall scenes should demonstrate behaviour in structured stacks versus behaviour in random piles.

Our results show clearly that the Blocked r-Factor strategy is the overall worst strategy for our test portfolio and setups. Global and Local both appear to be competitive and have complementary traits. Global r-Factor appears to give better accuracy and less overall divergence detection than the Local r-Factor strategy. It is noticeable more effective for the cases of structured stacks. Local r-Factor strategy seems inferior in terms of number of observed absolute convergence cases and clearly have more divergence detections particular for structured stacks. However, the Local strategy appears to get the job done in less iterations compared to the Global r-Factor strategy.

Let us examine how the different r-Factor strategies impact the performance of the overall simulator. Our measurements of the "Jacks" scene are shown in Figure 5. The simulations were done on a Intel(R) Xeon(R) CPU E5-2697 v2 2.70GHz using a single thread on a single core. The measurements suggest that Global and Local r-Factor strategies are comparable in performance. If we investigate the "Spheres" scene we would expect Local r-Factor strategy to become more expensive as it detects divergence in more cases. However, as Figure 6 shows the two strategies are still performance-wise competitive. The explanation is that the Local r-Factor strategy needs fewer overall solver iterations to reach a state of relative convergence compared to the Global r-Factor strategy. The measurements of the Local r-Factor strategy appear to show less variation hence one may argue that it yields a more predictable performance for this particular scene.

Another interesting question is whether the r-Factor strategies have influence on the computed motion. To shed some light on this aspect we have plotted the energies of the Jack scene in Figure 7. The Blocked and Local strategies seem to agree on a "bump" around solver iteration 120. Whereas the Global strategy appears to produce a much more smooth energy plot. Suggesting a difference in motions. Whether this is perception-wise noticeable is questionable. However, from a digital prototyping viewpoint it could be important to clarify which behavior is more correct.

We compare how Gauss–Seidel and Jacobi schemes behave. Figure 8 compares convergence rates for the Jacks and Spheres test scenes. Overall Gauss–Seidel appear much more predictable in its convergence behavior than the Jacobi scheme. Further, the number of times divergence was detected is much larger for Jacobi scheme than the Gauss–Seidel scheme as shown in Figure 9.

6 CONCLUSION AND DISCUSSION

We have presented a proximal operator model of contact force problems for rigid bodies and outlined a class of iterative methods, the PROX schemes. We demonstrated how PROX schemes converge and how their behaviour varies under different r-Factor strategies. Our results indicate the Gauss–Seidel variant is superior in terms of delivering less varied convergence behaviour and hence should in our opinion be preferred over Jacobi variants. Our results strongly suggest that Global r-Factors are better for structured stacking scenarios and can achieve absolute convergence when Local strategy only achieves relative convergence. On the other hand the Local strategy seems to need lesser iterations and is performance-wise competitive with the Global strategy. Perception-wise we have not observed major significance between the Local and Global strategies. However, kinetic and potential energy monitoring do suggest a significant difference that could have importance if the predicted outcome of the simulation is important. We have presented a specific tailored novel numerical method for the proximal operator of the Coulomb–Contensou friction law [Leine and Glocker 2003] and outlined the generality by which the proximal operator model can incorporate any convex multiset friction models in future work.

There are many paths we have not explored when we have open up the door to proximal algorithms [Parikh and Boyd 2014]. There are some obvious methods to study in future work such as accelerated gradient methods or generalized Conjugate Gradient methods [Silcowitz-Hansen et al. 2010]. Further, the connection to proximal operators in projective dynamics and fluids may hint that ADMM could become an interesting alternative if extending to

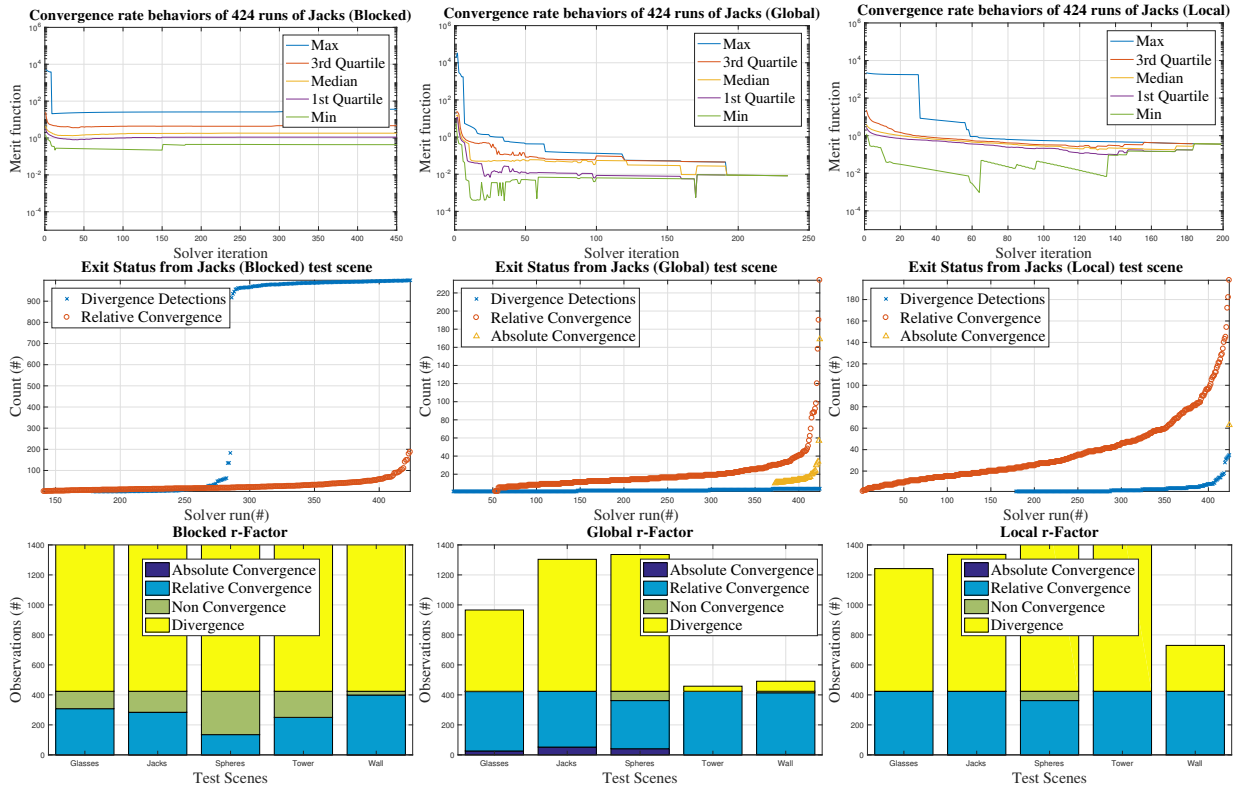

Figure 4: Top row shows detailed comparison of "Jacks" test scene using different r-Factor strategies. Left column is Blocked r-Factor strategy, middle column is Global r-Factor strategy and right column shows Local r-Factor strategy. Botom row shows histograms of five other test scenes, we trimmed the histograms at 1400 observations to make visual comparison easier. Observe that Blocked r-Factor strategy appears to detect divergence many more times than the other strategies and often results in non-convergence. Comparing the Global r-Factor strategy against the Local r-Factor strategy suggests that Global strategy gives overall better merit values and absolute convergence in more cases than the Local r-Factor strategy. However, the Local strategy uses less iterations than the Global strategy and slightly fewer cases of non-convergence.

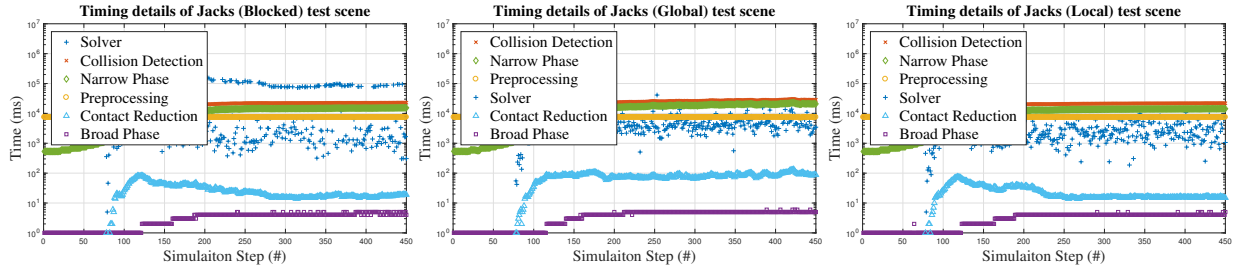

Figure 5: We compare the overall performance characteristic of the simulator as a function of different r-Factor strategies. On the left we show Blocked, middle shows Global, and right shows Local r-Factor strategy. The Blocked r-Factor shows computation times 1s larger than the other two (and mind it has not converged in any step). The Local and Global strategies seem comparable in solver times. This is supported by the fact that for this test scene they both detected divergence in nearly the same number of times. The Local r-Factor strategy uses more storage and a few more FLOPS than the Global strategy. The Local strategy uses less iterations and is therefore performance-wise competitive to the Global r-Factor strategy.

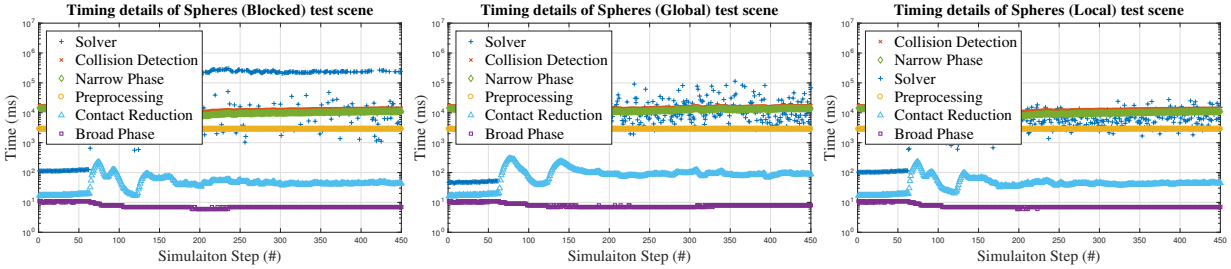

Figure 6: We compare performance of the "Spheres" test scene for different r-Factor strategies (left Blocked, middle Global, and right Local). Due to many more divergence detections of the Local r-Factor strategy we expected this strategy to give worse performance results. However, it is very similar to the Global r-Factor strategy.

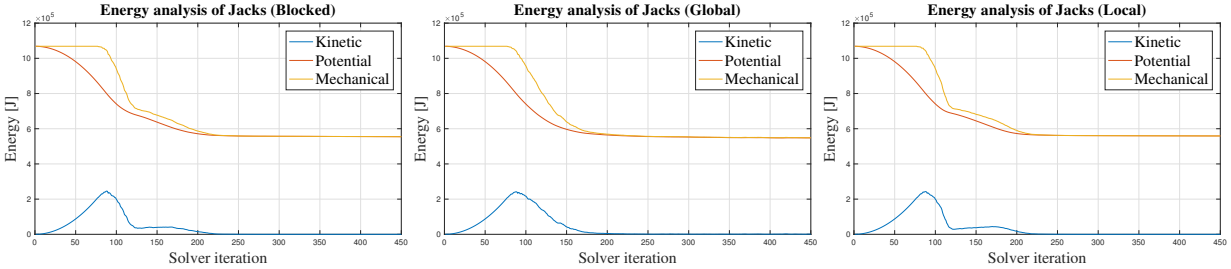

Figure 7: We compare energy plots of the "Jacks" test scene for different r-Factor strategies (left Blocked, middle Global, and right Local). The kinetic energy reveals a difference around step 100 between Local and Global suggesting differences in motions. Blocked appear more similar to Local than Global.

include deformable models. Currently projective dynamics [Bouaziz et al. 2014; Narain et al. 2016] focus on constitutive equations and non-penetration constraints, and only apply ad-hoc updates of velocities during collisions to model friction and restitution.

Our work did not exploit any kind of parallelization. However, we speculate that prior work on parallelization of the box-model PGS methods can be heavily reused. Overall, when we study our motions by visual inspection then it does not appear to be the constraint solver itself who is the bottleneck for increasing the visual quality. Rather tunneling artefacts and normal generation seem to be major sources for visual artefacts.

REFERENCES

Pierre Alart and Alain Curnier. 1991. A Mixed Formulation for Frictional Contact Problems Prone to Newton Like Solution Methods. *Comput. Methods Appl. Mech. Eng.* 92, 3 (Nov. 1991), 353–375.

David Baraff. 1993. Issues in computing contact forces for nonpenetrating rigid bodies. *Algorithmica. An International Journal in Computer Science* 10, 2-4 (1993), 292–352. Computational robotics: the geometric theory of manipulation, planning, and control.

Jan Bender, Kenny Erleben, and Jeff Trinkle. 2014. Interactive Simulation of Rigid Body Dynamics in Computer Graphics. *Computer Graphics Forum* 33, 1 (2014), 246–270.

Sofien Bouaziz, Sebastian Martin, Tiantian Liu, Ladislav Kavan, and Mark Pauly. 2014. Projective Dynamics: Fusing Constraint Projections for Fast Simulation. *ACM Trans. Graph.* 33, 4 (July 2014), 154:1–154:11.

Michael B. Cline and Dinesh K. Pai. 2003. Post-stabilization for rigid body simulation with contact and constraints. In *2003 IEEE International Conference on Robotics and Automation*, Vol. 3. 3744–3751.

Richard Cottle, Jong-Shi Pang, and Richard E. Stone. 1992. *The Linear Complementarity Problem.* Academic Press.

Gilles Daviet, Florence Bertails-Descoubes, and Laurence Boissieux. 2011. A hybrid iterative solver for robustly capturing coulomb friction in hair dynamics. *ACM Trans. Graph.* 30, 6, Article 139 (Dec. 2011), 12 pages.

Kenny Erleben. 2005. *Stable, robust, and versatile multibody dynamics animation.* Ph.D. Dissertation. Department of Computer Science, University of Copenhagen, Denmark.

Kenny Erleben. 2007. Velocity-based Shock Propagation for Multibody Dynamics Animation. *ACM Trans. Graph.* 26, 2, Article 12 (June 2007).

Martin Foerg, Thomas Geier, Lutz Neumann, and Heinz Ulbrich. 2006. R-Factor Strategies for the Augmented Lagrangian Approach in Multi-Body Contact Mechanics. In *Proc. of III European Conference on Computational Mechanics.*

Suresh Goyal, Andy Ruina, and Jim Papadopoulos. 1989. Limit Surface and Moment Funktion Descriptions of Planar Sliding. In *Proc. of the 1989 IEEE International Conference on Robotics and Automation (Vol. 2).* Scottsdale, AZ, 794–799.

Tiffany Inglis, Marie-Lena Eckert, James Gregson, and Nils Thuerey. 2017. Primal-Dual Optimization for Fluids. *Computer Graphics Forum* (2017).

Michel Jean and Jean Jacques Moreau. 1992. Unilaterality and dry friction in the dynamics of rigid bodies collections. In *Proc. of Contact Mech. Int. Symp.*, A. Curnier (Ed.). 31–48.

Franck Jourdan, Pierre Alart, and Michel Jean. 1998. A Gauss-Seidel like algorithm to solve frictional contact problems. *Computer Methods in Applied Mechanics and Engineering* 155, 1 (1998), 31 – 47.

Danny M. Kaufman, Timothy Edmunds, and Dinesh K. Pai. 2005. Fast Frictional Dynamics for Rigid Bodies. *ACM Trans. Graph.* 24, 3 (July 2005), 946–956.

Danny M. Kaufman, Shinjiro Sueda, Doug L. James, and Dinesh K. Pai. 2008. Staggered Projections for Frictional Contact in Multibody Systems. *ACM Trans. Graph.* 27, 5, Article 164 (Dec. 2008), 11 pages.

Claude Lacoursière. 2007. *Ghosts and machines : regularized variational methods for interactive simulations of multibodies with dry frictional contacts.* Ph.D. Dissertation. Umeå University, Computing Science.

Remco I. Leine and Christoph Glocker. 2003. A set-valued force law for spatial coulomb-contensou friction. *European Journal of Mechanics - A/Solids* 22, 2 (2003), 193–216.

Ying Lu and Jeff Trinkle. 2014. On the Convergence of Fixed-point Iteration in Solving Complementarity Problems Arising in Robot Locomotion and Manipulation. In *IEEE International Conference on Intelligent Robots and Systems* (Chicago).

Hammad Mazhar, Toby Heyn, Dan Negrut, and Alessandro Tasora. 2015. Using Nesterov's Method to Accelerate Multibody Dynamics with Friction and Contact. *ACM Trans. Graph.* 34, 3, Article 32 (May 2015), 14 pages.

Xavier Merlhiot. 2007. A robust, efficient and time-stepping compatible collision detection method for non-smooth contact between rigid bodies of arbitrary shape. In *Proceedings of the Multibody Dynamics 2007 ECCOMAS Thematic Conference.*

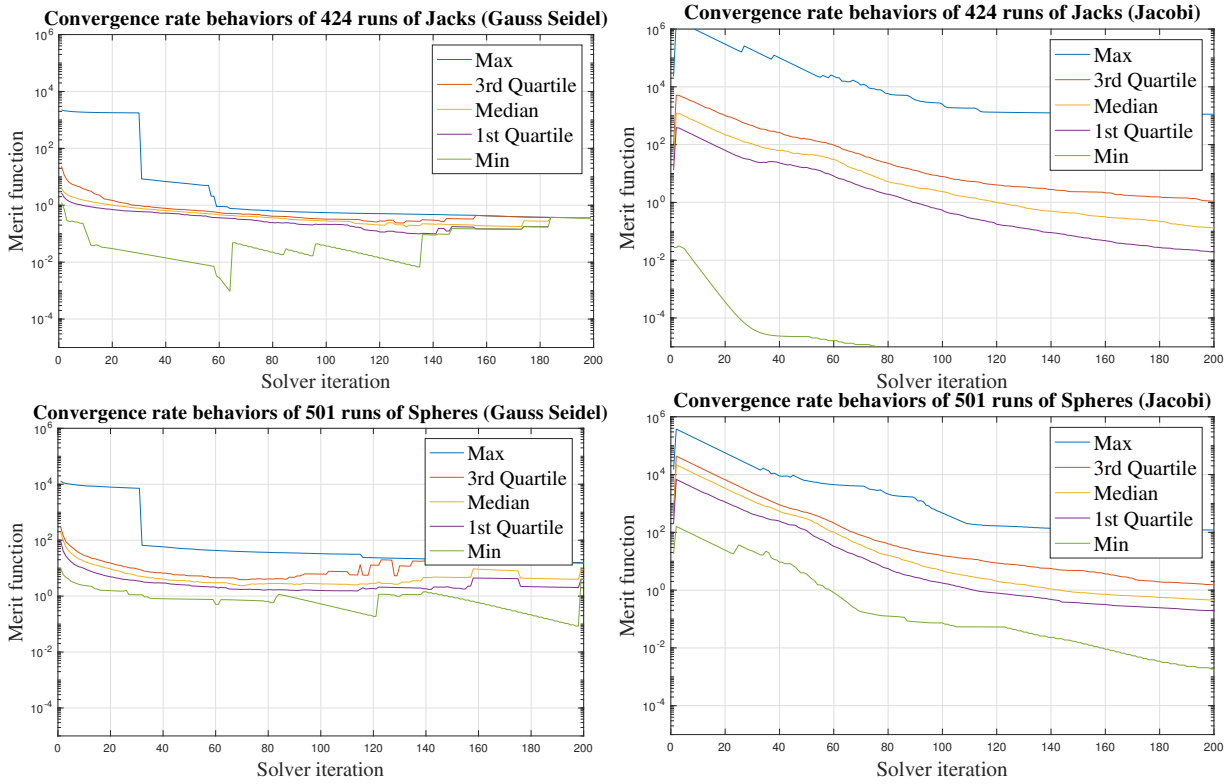

Figure 8: We compare convergence rates of Gauss–Seidel (left column) against Jacobi (right column) schemes. The Jacobi scheme have a much larger variation in convergence behaviour. Hence, Gauss–Seidel will have a much more predictable outcome. The Gauss–Seidel scheme tends to much more quickly reduce the error. Jacobi needs approximately 100 iterations more than Gauss–Seidel to reach the same error level when considering the median.

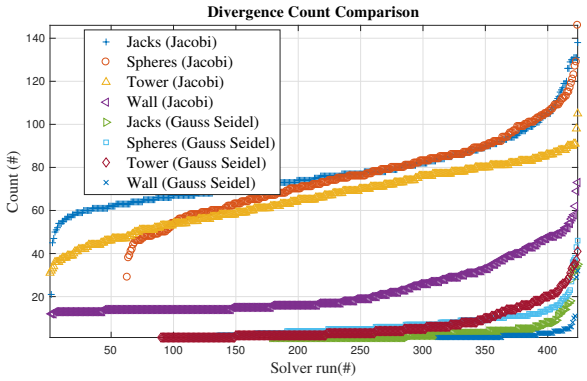

Figure 9: We compare number of times divergence is detected between Gauss–Seidel and Jacobi schemes. It is obvious that Jacobi detect divergence far more often than the Gauss–Seidel method. Hence Jacobi will in each solver run roll-back the solution 60-100 times more than Gauss–Seidel.

Jean Jacques Moreau. 1999. Numerical aspects of the sweeping process. *Computer Methods in Applied Mechanics and Engineering* 177, 3–4 (July 1999), 329–349.

Rahul Narain, Matthew Overby, and George E. Brown. 2016. ADMM ⊇ Projective Dynamics: Fast Simulation of General Constitutive Models. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '16)*. Eurographics Association, 21–28.

Sarah Maria Niebe. 2013. *Rigid Bodies in Contact: and Everything in Between.* Ph.D. Dissertation. Department of Computer Science, Faculty of Science, University of Copenhagen.

Neal Parikh and Stephen Boyd. 2014. Proximal Algorithms. *Found. Trends Optim.* 1, 3 (Jan. 2014), 127–239.

Thorsten Schindler, Binh Nguyen, and Jeff Trinkle. 2011. Understanding the difference between prox and complementarity formulations for simulation of systems with contact. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 1433–1438.

Morten Silcowitz, Sarah Niebe, and Kenny Erleben. 2010. Contact Point Generation for Convex Polytopes in Interactive Rigid Body Dynamics. Poster at SCA 10'. (2010).

Morten Silcowitz-Hansen, Sarah Niebe, and Kenny Erleben. 2010. A Nonsmooth Nonlinear Conjugate Gradient Method for Interactive Contact Force Problems. *Vis. Comput.* 26, 6-8 (June 2010), 893–901.

Christian Walter Studer. 2008. *Augmented Time-stepping Integration of Non-smooth Dynamical Systems.* Ph.D. Dissertation. ETH Zurich.

Richard Tonge, Feodor Benevolenski, and Andrey Voroshilov. 2012. Mass Splitting for Jitter-free Parallel Rigid Body Simulation. *ACM Trans. Graph.* 31, 4, Article 105 (July 2012), 8 pages.

Huamin Wang. 2015. A Chebyshev Semi-iterative Approach for Accelerating Projective and Position-based Dynamics. *ACM Trans. Graph.* 34, 6 (Oct. 2015), 246:1–246:9.

A ASSEMBLING THE MATRICES

We will explain how to construct the matrices $\mathbf{M}$ and $\mathbf{J}$ and the vector $\mathbf{h}$. Consider the k^{th} contact point. The contact normal is given by $\mathbf{n}_k$ and the two orthonormal vectors spanning the contact plane are $\mathbf{t}_k$ and $\mathbf{s}_k$. The indices of the two bodies meeting at the

contact are i and j, where we assume that $i < j$. We will adopt the convention that $\mathbf{n}_k$ is pointing from body i to body j. The vector arms from the center of mass of the bodies to the point of contact is given by $\mathbf{r}_{k_i}$ and $\mathbf{r}_{k_j}$ respectively. The relative contact point velocity can now be written as follows

$$\begin{bmatrix} v_{N_k} \\ v_{T_k} \\ v_{S_k} \\ v_{\tau_k} \end{bmatrix} = \begin{bmatrix} -\mathbf{C}_k^T & -\left(\boldsymbol{r}_{\boldsymbol{k_i}}{}^{\times}\mathbf{C}_k\right)^T & \mathbf{C}_k^T & \left(\boldsymbol{r}_{\boldsymbol{k_j}}{}^{\times}\mathbf{C}_k\right)^T \\ \mathbf{0}^T & -\mathbf{n}_k^T & \mathbf{0}^T & \mathbf{n}_k^T \end{bmatrix} \begin{bmatrix} \mathbf{v}_i \\ \omega_i \\ \mathbf{v}_j \\ \omega_j \end{bmatrix} \tag{52}$$

where

$$\mathbf{C}_k = \begin{bmatrix} \mathbf{n}_k & \mathbf{t}_k & \mathbf{s}_k \end{bmatrix}. \tag{53}$$

The tangent plane directions $\mathbf{t}_k$ and $\mathbf{s}_k$ are often computed by picking $\mathbf{t}_k$ to be in the direction of sliding and letting $\mathbf{s}_k$ be orthogonal to $\mathbf{t}_k$. The skew-symmetric cross product matrix of a vector $r = [x, y, z]$ is defined as

$$\boldsymbol{r}^{\times} = \begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix}. \tag{54}$$

Defining the blocked notation

$$\mathbf{v}_k = \begin{bmatrix} v_{N_k} \\ v_{T_k} \\ v_{S_k} \\ v_{\tau_k} \end{bmatrix}, \mathbf{u}_i = \begin{bmatrix} \mathbf{v}_i \\ \omega_i \end{bmatrix}, \mathbf{u}_j = \begin{bmatrix} \mathbf{v}_j \\ \omega_j \end{bmatrix}, \tag{55}$$

and

$$\mathbf{J}_{k,i} = \begin{bmatrix} -\mathbf{C}_k^T & -\left(\boldsymbol{r}_{\boldsymbol{k_i}}{}^{\times}\mathbf{C}_k\right)^T \\ \mathbf{0}^T & -\mathbf{n}_k^T \end{bmatrix}, \mathbf{J}_{k,j} = \begin{bmatrix} \mathbf{C}_k^T & \left(\boldsymbol{r}_{\boldsymbol{k_j}}{}^{\times}\mathbf{C}_k\right)^T \\ \mathbf{0}^T & \mathbf{n}_k^T \end{bmatrix} \tag{56}$$

and assembling a full system of all K contacts and N bodies we have

$$\underbrace{\begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_k \\ \vdots \\ \mathbf{v}_K \end{bmatrix}}_{\mathbf{v}} = \underbrace{\begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ 0\cdots 0 & \mathbf{J}_{k,i} & 0\cdots 0 & \mathbf{J}_{k,j} & 0\cdots 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}}_{\mathbf{J}} \underbrace{\begin{bmatrix} \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_i \\ \vdots \\ \mathbf{u}_j \\ \vdots \\ \mathbf{u}_N \end{bmatrix}}_{\mathbf{u}}. \tag{57}$$

The i^{th} rigid body has the center of mass position $\mathbf{r}_i$ and the orientation given by the unit quaternion $Q_i \equiv (\phi_i, \psi_i, \theta_i, \xi_i)$ together with the mass m_i and local body-frame inertia tensor $\mathbf{I}_{\text{BF}_i}$. The local body frame inertia tensor must be updated to reflect the world-frame inertia tensor

$$\mathbf{I}_{\text{WF}_i} = \mathbf{R}_i \mathbf{I}_{\text{BF}_i} \mathbf{R}_i^T \tag{58}$$

where $\mathbf{R}_i$ is the rotation matrix corresponding to Q_i. Defining the mass-block notation

$$\mathbf{M}_i = \begin{bmatrix} m_i \mathbf{1}_{3\times 3} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{\text{WF}_i} \end{bmatrix}. \tag{59}$$

Now the assembled mass matrix reads

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \mathbf{M}_N \end{bmatrix}. \tag{60}$$

Let $\mathbf{f}_{\text{ext}_i}$ and τ_{ext_i} be the force and torque accumulators of all external force types acting on the i^{th} body. Using the block-notation

$$\mathbf{h}_i = \begin{bmatrix} \mathbf{f}_{\text{ext}_i} \\ \tau_{\text{ext}_i} - \omega_i \times \mathbf{I}_{\text{WF}} \omega_i \end{bmatrix}, \tag{61}$$

for the external and gyroscopic forces we assemble the $\mathbf{h}$-vector,

$$\mathbf{h} = \begin{bmatrix} \mathbf{h}_1^T & \vdots & \mathbf{h}_N^T \end{bmatrix}^T. \tag{62}$$

Defining the blocked notation

$$\mathbf{q}_i = \begin{bmatrix} \mathbf{r}_i^T & Q_i^T \end{bmatrix}^T \tag{63}$$

then the global assembled version reads

$$\mathbf{q} = \begin{bmatrix} \mathbf{q}_1^T & \cdots & \mathbf{q}_N^T \end{bmatrix}^T. \tag{64}$$

Notice that the dimension of $\mathbf{q}$ and $\mathbf{u}$ mis-match and this causes a problem for the generalized kinematic relation, $\dot{\mathbf{q}} = \mathbf{u}$. The problem is solved by introducing the blocked matrix

$$\boldsymbol{\Gamma}_i = \frac{1}{2} \begin{bmatrix} -\psi_i & -\theta_i & -\xi_i \\ \phi_i & \xi_i & -\theta_i \\ -\xi_i & \phi_i & \psi_i \\ \theta_i & -\psi_i & \phi_i \end{bmatrix} \tag{65}$$

where $Q_i \equiv (\phi_i, \psi_i, \theta_i, \xi_i)$ and defining

$$\boldsymbol{\Gamma} = \begin{bmatrix} \mathbf{1}_{3\times 3} & & & & \mathbf{0} \\ & \boldsymbol{\Gamma}_i & & & \\ & & \ddots & & \\ & & & \mathbf{1}_{3\times 3} & \\ \mathbf{0} & & & & \boldsymbol{\Gamma}_N \end{bmatrix}. \tag{66}$$

The discretized kinematic relation leading to the position update is now given by

$$\mathbf{q}^{t+\Delta t} = \mathbf{q}^t + \Delta t \boldsymbol{\Gamma} \mathbf{u} \tag{67}$$

which must always be followed by a normalizing of all Q_i to keep them a proper rotation.