DOI: 10.1111/cgf.13885

COMPUTER GRAPHICS *forum*
Volume 00 (2019), number 00 pp. 1–12

The *Matchstick* Model for Anisotropic Friction Cones

K. Erleben[1], M. Macklin[1,2], S. Andrews[3] and P. G. Kry[4]

[1]Department of Computer Science, University of Copenhagen, Copenhagen, Denmark
kenny@di.ku.dk, mmacklin@nvidia.com
[2]NVIDIA, Santa Clara, CA, USA
[3]École de technologie supérieure, Montreal, Quebec, Canada
sheldon.andrews@etsmtl.ca
[4]School of Computer Science, McGill University, Montreal, Quebec, Canada
kry@cs.mcgill.ca

Abstract

Inspired by frictional behaviour that is observed when sliding matchsticks against one another at different angles, we propose a phenomenological anisotropic friction model for structured surfaces. Our model interpolates isotropic and anisotropic elliptical Coulomb friction parameters for a pair of surfaces with perpendicular and parallel structure directions (e.g. the wood grain direction). We view our model as a special case of an abstract friction model that produces a cone based on state information, specifically the relationship between structure directions. We show how our model can be integrated into LCP and NCP-based simulators using different solvers with both explicit and fully implicit time-integration. The focus of our work is on symmetric friction cones, and we therefore demonstrate a variety of simulation scenarios where the friction structure directions play an important part in the resulting motions. Consequently, authoring of friction using our model is intuitive and we demonstrate that our model is compatible with standard authoring practices, such as texture mapping.

Keywords: physically based animation, animation, physically based modelling, modelling, natural phenomena animation

ACM CCS: • Computing methodologies → Physical simulation

1. Introduction

The isotropic Coulomb friction model is used by many rigid body simulations in the field of computer graphics [BET14]. Most of the work in this area has focused on the difficult problem of formulating and solving Coulomb frictional contact and its approximations. However, there has been comparatively less attention in computer graphics on more expressive anisotropic models of frictional contact. In this work, we focus on anisotropic Coulomb friction where the friction cone takes on an elliptical shape. This model is convenient for several reasons: it is intuitive for non-expert users, simple to compute, characterizes the frictional behaviour for a variety of materials and the mathematical description fits easily into existing numerical frameworks for computing contact forces.

We are missing a tool that allows realistic macroscopic descriptions of surfaces and frictional phenomena to be used in simulators. That is, we do not yet have the tools that can allow us to explore the suitability or accuracy of the usual friction model choices, or new models such as the one we present here. As such, our motivation for pursuing a friction modelling framework with additional degrees of freedom is to provide artistic control; the goal is not to match reality, but to gain a desired and plausible motion behaviour. Our model provides animators with artistic control over the friction behaviour so as to promote motions with the desired traits during simulation.

Many rigid body simulators use friction models expressed as cones, or as generic sets of feasible friction forces. Combining feasible set descriptions with extra constraints, such as the principle of maximum dissipation, allows us to compute the friction forces at a given instant in time. The benefit of such friction descriptions is that they permit an easy implementation within a simulator. One can define a projection operator, and iteratively project the friction force onto the closest feasible friction force. This property is the governing principle about which many friction models can be defined. In this paper, we design a friction model that we call the *Matchstick* model, which produces a cone based on the directions of surface structure on each object at the point of contact. Furthermore, we present a

Figure 1: *The Matchstick model allows for control of frictional behaviour in situations, such as tire ground contact, a hopper and soft robotic gripping, where each example shown here also uses different simulators with different contact solvers (Vortex, PROX and Flex).*

mathematical framework that supports the necessary operations for straightforward implementation of our model.

The main contributions of our work are as follows. We present a model with intuitive authoring and control that allows a range of frictional behaviours to be rapidly defined. Our phenomenological model has simulation state-dependent friction cones and supports both semi-implicit and fully implicit time-integration schemes. Finally, we describe methods for including our model in any simulator that supports non-smooth cones, such as PROX schemes and variants as demonstrated in Figure 1.

2. Related Work

Friction measurements in mechanical engineering have for a long time demonstrated the shortcomings of the standard isotropic Coulomb model used in Computer Graphics: Liley *et al.* [LGS*98] prove asymmetric anisotropic friction due to molecular tilt of a mica surface. Umbanhowar *et al.* [UVM*12] demonstrate direction-dependent surface friction properties to help design friction-induced velocity fields on a vibrating plate. Their experiments clearly show anisotropic behaviour due to microscale orthographic rough geometric features. Yu and Wang [YW12] show that very rough microscale geometry gives rise to anisotropic friction strongly correlated to the microscale structure. Further, fine roughness appears to give near isotropic behaviour. Hence, a certain sufficient roughness scale and directional structure is needed to get strong anisotropic response. This fits our scope of material modelling exactly. Walker and Leine [WL17] demonstrate the behaviour of non-convex cones and they propose a replacement approach for the principle of maximum dissipation based on a two-cone approach. One convex cone is used to pick the direction of the friction, and the other (possibly non-convex) cone is used to determine the magnitude. The Matchstick model produces elliptical cones, and the simulation method and modifications in our work are currently limited to convex cones obeying principle of maximum dissipation.

Table 1 presents a brief overview and comparison of relevant work on related friction models used in the field of computer graphics. In graphics, a wide collection of work has looked at different formulations and different methods for solving the frictional contact problem, largely using isotropic static Coulomb cones. Baraff's seminal work introduced linear complementarity problems to computer graphics [Bar94]. Fast frictional dynamics [KEP05] efficiently computes responses by merging cones across different contacts, while staggered projections [KSJP08] alternate between normal and tangent impulse solves for friction between elastic objects. Implicit contact models are likewise useful for thin models, such as cloth [OTSG09]. Exact Coulomb friction cones have been used for hair [DBDB11], and alternatives to Coulomb friction prove useful in the simulation of cloth at the thread level [CLO17]. Aggregate contact models to approximate frictional contact patches have been proposed for rigid bodies [BNT*15], elastic models [TMDO15] and at arbitrary resolution through voxelization of volume contacts [AFC*10].

The work addressing anisotropic friction modelling is sparse [BET14]. Among the exceptions is the use of the friction tensors by Pabst *et al.* [PTS09], which uses an additive model and has nine parameters to describe the material for planar dynamic friction. Our Matchstick model uses the angle between structure directions (much like the tensor eigenvectors) to interpolate the physics in a common frame, and only uses a small number of intuitive parameters. The friction tensors do not provide an immediate cone description, but give a direct equation for the dynamic friction force that does not rely on maximum dissipation. Since we express our model in terms of cones, it supports the principle of maximum dissipation.

Limit surfaces [GRP89] are a general concept that describe any cone shape that scales linearly with the normal force. Our model uses the same scaling concept, which means that our cones belong to an elliptical family of limit shapes. However, in comparison with limit surfaces, the 'shape' is dynamically dependent on the simulation state. Furthermore, we provide a method to determine the orientation of the cone in the world frame, whereas the limit surfaces concept does not provide a direct solution to this aspect. A pragmatic solution to this is to align the first axis of the contact frame with the relative sliding direction, as done in many physics engines and past work [BET14]. Instead, the Matchstick model uses the mean of the material structure directions to estimate the direction of least resistance and the friction cone intrinsically lives in this contact frame. In comparison with Pabst *et al.* [PTS09], their affine map is defined by a linear function, i.e. addition of tensors in a common frame to give the intrinsic representation of the friction model. See Appendix A for more details.

Data-driven nonlinear friction models have been demonstrated for cloth animation [CFW13]. The data show structural state dependence on relative orientation and nonlinear normal force scaling. The model is similar to Pabst *et al.* [PTS09] except that three nonlinear functions are used to form a symmetric tensor, which is then used to determine the magnitude of the friction force using a quadratic form dependent on sliding velocity. The model can be seen as a

Table 1: *Overview of common planar dynamic friction models used in computer graphics.*

Model name	Shape	#Params	Contact frame orientation	Scaling	ANISO	PMD	INT
Isotropic Coulomb [Bar89]	Circle	1	None	Linear in λ_n	No	Yes	Yes(1)
Anisotropic Coulomb(2)	Ellipse	2	Fixed on one object or by sliding velocity	Linear in λ_n	Yes	Yes	Yes(1)
Limit Surfaces [GRP89]	Any cone	∞	Not specified	Linear in λ_n	Yes	Yes(3)	Yes
Friction Tensors [PTS09]	Affine map	9	Fixed to one of the structure fields	Linear in λ_n	Yes	No(4)	No(5)
Cloth Friction [CFW13]	Load force curve	> 6(6)	Fixed to material space	Nonlinear in λ_n, quadratic in $\mathbf{v}$	No	Yes	Yes
Matchstick	Ellipse	3	Mean of both structure field directions	Linear in λ_n	Yes	Yes	Yes

1. Many game engines either average friction coefficients assigned to objects use a material-pair look-up table.
2. Anisotropic Coulomb friction is used in many game engines, where the sliding direction is used to determine the first major principal axis of the contact coordinate system. However, the sliding and least direction of friction are not always parallel in reality.
3. Supports non-convex cones giving rise to non-uniqueness even when the principle of maximum dissipation is applied.
4. Only under restriction of symmetric positive definite friction tensors is the principle of maximum dissipation fulfilled.
5. Two separate friction tensors combine to give the intrinsic friction map.
6. Three nonlinear functions are fit to data using linear regression. Hence, more than six parameters seem reasonable as otherwise the three nonlinear functions would be no better than a linear fitting.

ANISO denotes aniosotropic, PMD denotes principle of maximum dissipation, INT denotes intrinsicly defined in the contact frame, λ_n is the normal force and $\mathbf{v}$ is the contact velocity.

specialized version of the earlier work by reducing the choice of parameter selection to fitting functions to data.

A link between frictional behaviour and the simulation solver algorithm has been observed, particularly for long kinematic chains and ill-conditioned linear systems [EATK18]. This indicates that in order to be useful, our friction model should be agnostic to the underlying solver algorithm. Although the approach proposed by Pabst *et al.* [PTS09] is tied to the semi-implicit time-stepping method of Bridson *et al.* [BFA02], we demonstrate the compatibility of our Matchstick model using several solver types such as both semi-implicit and fully implicit time-stepping methods. These include a pivoting algorithm with direct solver, a nonlinear implicit Newton-type solver [MEM19] and a PROX-based iterative Gauss–Seidel scheme.

Daviet *et al.* [DBDB11] solve isotropic Coloumb friction for hair dynamics with a Gauss–Seidel scheme, which is efficient due to a scaling that makes impulse and velocity cones self-dual. The sliding direction is an unknown and solved at each time step with a fully-implicit treatment. Our model can likewise be solved in a fully implicit manner given that our anisotropic cones depend on structure directions rather than sliding velocity. However, it is less obvious how the scaling approach of Daviet *et al.* can be adapted to work for anisotropic friction without distorting space and not preserving 'angles'. A somewhat similar distortion of angles is observed for blocked r-factors [Erl17]. In contrast, Macklin *et al.* [MEM19] show similarity between r-factors and preconditioners that create self-dual isotropic cones; however, adaptive r-factors can adapt locally to the anisotropy and rescale cones just in the direction needed. For certain specific conditions, a cone description of the friction tensor model can be recovered at added computational cost as we prove in Appendix A, and with additional restrictions the maximum principle of dissipation can be fulfilled.

Modelling restitution for rigid body impact using bounce maps [WSJP17] shares some similarity to using structure maps. We too consider the structural dependency of friction behaviour as mapped to surfaces of objects. However, unlike bounce maps, which consider the restitution as a two body function, we treat the physical coefficients depending on the pair-wise material types and separate only the structure (micro-geometry) into our maps. Given these similarities, we speculate that restitution could be modelled in a similar fashion. Recently, Costes *et al.* [CDA*18] propose a holistic approach using texture maps to represent surface properties for haptic simulation. Their material format includes friction, and our proposed friction model is compatible with their format since we demonstrate that it can be represented by a texture.

We consider the use of an active measurement facility by Pai *et al.* [PDJ*01] to be an excellent early effort to capture and model friction behaviour. Contact friction textures are recorded with a robotic probe under the assumption of a symmetric isotropic Coulomb model. Recent work by Dreßel *et al.* [DEKA19] revisits this idea by dropping these assumptions, and provides an open data set that clearly demonstrates curved trajectories and rotational alignment of cubed rigid bodies while sliding on inclined wood planes. Our work reproduces these behaviours, which is only possible if state dependency and anisotropy are included in the models.

3. Friction Cone Modelling

The idea is to define a friction cone generator that we initially require to be able to generate a convex pointed cone. The cone will be generated on the fly within a typical solver allowing the cone to change shape depending on the kinematic state.

Without loss of generality, we will study two objects that we label A and B. We assume that they have materials assigned such that the materials can be characterized by a single material structure direction associated with any surface point on the objects. The directions can be interpreted as fibre directions or as micro-scale geometry features, such as grooves. While some materials, e.g. cloth with warp and weft directions, can be seen as having multiple

structure directions, we limit our investigation to the case of a single structure direction.

Let $\mathbf{p} \in \mathbb{R}^3$ be a single point of contact between A and B, with unit normal $\mathbf{n}$ pointing from A to B, and with relative contact velocity $\mathbf{v}$, given by,

$$\mathbf{v} \equiv \mathbf{J} \begin{bmatrix} \mathbf{u}_\mathsf{A} \\ \omega_\mathsf{A} \\ \mathbf{u}_\mathsf{B} \\ \omega_\mathsf{B} \end{bmatrix}, \tag{1}$$

where $\mathbf{u}_\mathsf{A}, \mathbf{u}_\mathsf{B} \in \mathbb{R}^3$ are linear velocities of the bodies and $\omega_\mathsf{A}, \omega_\mathsf{B} \in \mathbb{R}^3$ are the angular velocities. The contact constraint Jacobian matrix is given by $\mathbf{J}$. The relative velocity can include both a normal and tangential component, $\mathbf{v}_n \in \mathbb{R}$ and $\mathbf{v}_t \in \mathbb{R}^2$.

The contact point frame at position $\mathbf{p}$ has orientation with respect to the world coordinate frame given by

$$\mathbf{C} \equiv \begin{bmatrix} \mathbf{n} & \mathbf{t} & \mathbf{b} \end{bmatrix}. \tag{2}$$

Here, $\mathbf{t}$ and $\mathbf{b}$ span the tangent plane of the contact point and are given in the world coordinate system. The choice of $\mathbf{t}$ determines the friction cone orientation and we use the mean of the structure field directions for this, as explained in detail later.

A contact frame is needed for assembling the Jacobian matrix $\mathbf{J}$ properly. For a one point contact with planar friction, the Jacobian will be:

$$\mathbf{J} \equiv \mathbf{C}^T \begin{bmatrix} -\mathbf{I}_{3\times3} & -\boldsymbol{r}_\mathsf{A}^\times & \mathbf{I}_{3\times3} & \boldsymbol{r}_\mathsf{B}^\times \end{bmatrix}, \tag{3}$$

where $\mathbf{I}_{3\times3}$ is the identify matrix, $\mathbf{r}_\mathsf{A}$ and $\mathbf{r}_\mathsf{B}$ are the contact arms from the respective object centres to the contact position and $\mathbf{r}^\times$ is the skew symmetric matrix that computes the $\mathbf{r}$ cross product.

In above definitions, we have by convention chosen to order normal components before frictional components. The reason for this is that most iterative solves tend to solve normal constraints before friction constraints. This is done because the friction part depends strongly on the normal part, while strong dependence in the opposite order is rare. This is merely a convention and our model works with any ordering and is independent of solving normal force before friction forces.

For any given contact frame, we can write the material structure direction of each surface in coordinates of the tangent plane of the common frame as unit vectors $\mathbf{s}_\mathsf{A}$ and $\mathbf{s}_\mathsf{B}$. These structure directions depend on the point of contact on each surface, and in a typical implementation can be stored in a texture or generated procedurally.

We can then consider a simple parameterized friction cone generator, $\mathcal{G}_{\mathsf{A}\leftrightarrow\mathsf{B}}$, specific to a pair of materials A and B. The generator computes a friction cone that describes the set of allowable friction forces based on a set of local parameters. We denote the cone by the symbol $\mathcal{F}_{\mathsf{A}\leftrightarrow\mathsf{B}}$. We will ease notation and not explicitly write the material pairs. The generator in general abstract notion can be defined as:

$$\mathcal{F} \equiv \mathcal{G}(\lambda_n, \mathbf{v}, \mathbf{s}_\mathsf{A}(\mathbf{p}), \mathbf{s}_\mathsf{B}(\mathbf{p})). \tag{4}$$

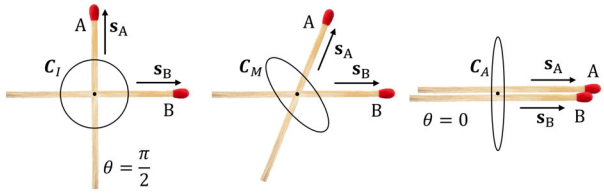


Figure 2: *We use the angle between matchsticks to determine the friction cone. The Matchstick model interpolates between isotropic* $\mathbf{C}_I$ *and anisotropic* $\mathbf{C}_A$ *extremes (see Equations* 6*,* 9 *and* 12*).*

The normal force magnitude is given by λ_n. The above parameterization could be extended with even more parameters to account for many other dependencies, which is an interesting avenue for future work. We will abuse notation and simply write $\mathcal{F}$ to indicate any parametric friction cone model to enhance the readability of equations.

3.1. The Matchstick model

We derive the Matchstick model from several observations and propose a novel model that interpolates between isotropic and extreme anisotropic Coulomb behaviours using the minimum angle between the structure directions (Figure 2).

1. Some materials have a visually noticeable simple structure, for instance a matchstick has a directional material structure (fibre) direction. Probing with a fingertip, sliding along the structure has less resistance than sliding orthogonal to the structure direction. Hence, the structure direction describes the direction of least resistance of the material.
2. Friction (resistance) appears to be isotropic for a pair of matchsticks when moving in different direction but keeping structure directions orthogonal.
3. Friction (resistance) appears to be an extremely anisotropic for a pair of matchsticks when sliding in different directions while keeping structure directions parallel. In general, one observes very low resistance in common structure directions and very large resistance in orthogonal direction.
4. Friction (resistance) appears to be anisotropic, but not as extreme, for a pair of matchsticks when sliding in different directions while keeping structure directions at oblique angles. As the angle approaches orthogonality, the behaviour becomes isotropic, while the behaviour becomes maximally anisotropic as the angle goes to zero.

With the simplifying assumption that we can ignore the sign of the structure vector (i.e. if the friction only depends on the orientation and not the direction), then the Matchstick model interpolation parameter is based on the angle θ and is defined as:

$$d \equiv 1 - \frac{2}{\pi} \underbrace{\cos^{-1} |\mathbf{s}_\mathsf{A} \cdot \mathbf{s}_\mathsf{B}|}_{\equiv\theta}. \tag{5}$$

Let the friction force in the world frame be given by $\mathbf{f}$ and let the coefficient of friction for a planar isotropic Coulomb friction cone be μ, equal to the tangent and binormal direction coefficients for

the isotropic cone, i.e. $\mu_t^I = \mu_b^I = \mu$, then the isotropic cone can be written as:

$$\mathbf{f}^T\mathbf{R}^T\underbrace{\begin{bmatrix}\frac{1}{\mu}^2 & 0\\ 0 & \frac{1}{\mu}^2\end{bmatrix}}_{\equiv\mathbf{C}_I}\mathbf{R}\mathbf{f} = \lambda_n^2. \tag{6}$$

Here, $\mathbf{R}$ can be any rotation matrix for an isotropic cone, but for consistency we define $\mathbf{R}$ as the 2D rotation of $(\mathbf{s}_\mathsf{A} + \mathbf{s}_\mathsf{B})$ onto the $\mathbf{t}$ axis, assuming that $\mathbf{s}_\mathsf{A} \cdot \mathbf{s}_\mathsf{B}$ is positive (again, given that we only need the orientation, we can swap the sign of one of the vectors to ensure positive dot product). This implies that

$$\mathbf{t} \equiv \frac{\mathbf{s}_\mathsf{A}+\mathbf{s}_\mathsf{B}}{\|\mathbf{s}_\mathsf{A}+\mathbf{s}_\mathsf{B}\|}, \tag{7}$$

$$\mathbf{b} \equiv \mathbf{n} \times \mathbf{t}. \tag{8}$$

We only need a 2D rotation matrix $\mathbf{R}$ to explain the model. However, the $\mathbf{t}$ and $\mathbf{b}$ column vectors for the contact coordinate frame are needed when assembling the Jacobian matrix as shown in Equation (3). By construction, $\mathbf{t}$ and $\mathbf{b}$ are the major and minor axis of the anisotropic ellipse cone, respectively.

We define the coefficients of friction for a planar anisotropic Coulomb friction cone as $\mu_t^A \leq \mu_b^A$, where the A superscript denotes anisotropic, and can define the anisotropic cone by

$$\mathbf{f}^T\mathbf{R}^T\underbrace{\begin{bmatrix}\frac{1}{\mu_t^A}^2 & 0\\ 0 & \frac{1}{\mu_b^A}^2\end{bmatrix}}_{\equiv\mathbf{C}_A}\mathbf{R}\mathbf{f} = \lambda_n^2. \tag{9}$$

In our model, the friction cone generator computes actual coefficients through spherical linear interpolation,

$$\mu_t \equiv d\,\mu_t^A + (1-d)\,\mu, \tag{10}$$

$$\mu_b \equiv d\,\mu_b^A + (1-d)\,\mu, \tag{11}$$

and thus, the actual Coulomb cone will be given by

$$\mathbf{f}^T\mathbf{R}^T\underbrace{\begin{bmatrix}\frac{1}{\mu_t}^2 & 0\\ 0 & \frac{1}{\mu_b}^2\end{bmatrix}}_{\equiv\mathbf{C}_M}\mathbf{R}\mathbf{f} = \lambda_n^2. \tag{12}$$

We give an outline of the Matchstick friction model generator in Algorithm 1. Finally, note that the Matchstick friction cone can be written as:

$$\mathcal{F}_M \equiv \left\{\lambda_f \,|\, \left(\frac{\lambda_t^2}{\mu_t^2} + \frac{\lambda_b^2}{\mu_b^2}\right) \leq \lambda_n^2\right\}. \tag{13}$$

The model has small memory footprint and fast computational complexity for both generating the cone and also for using the cone at run time. Immediate benefit of our model is that it allows artistic modelling of the isotropic and extreme anisotropic behaviours, and automatically determines the orientation and shape of the anisotropic friction cone.

4. Simulator Integration

Iterative methods for contact force computations are the natural choice for arbitrary friction cones. We will first consider the class of methods based on proximal operators [Erl17]. In methods based on proximal operators, the next feasible friction force iterate λ_f^{k+1} is given by projecting the current friction force guess λ_f^k onto the friction cone, $\mathcal{F}$,

$$\lambda_f^{k+1} \leftarrow \text{prox}_{\mathcal{F}}\left(\lambda_f^k - r\,\mathbf{v}_f^k\right). \tag{14}$$

Algorithm 1: MATCHSTICKFRICTIONGENERATOR The generator gives both the world orientation of the friction cone as well as the coefficients of friction, which is an advantage when working with an analytic cone that is fully described by these parameters.

Data: Structure directions $\mathbf{s}_\mathsf{A}$, $\mathbf{s}_\mathsf{B}$, contact normal $\mathbf{n}$, isotropic friction μ, extreme friction coefficients μ_t^A, μ_b^A.
Result: Contact plane vectors $\mathbf{t}$, $\mathbf{b}$, and coefficients of friction μ_t, μ_b.

```
if s_A · s_B < 0 then
    s_B ← −s_B
end
θ ← cos^{-1}(s_A · s_B)
d ← 1 − 2θ/π
μ_t ← d μ_t^A + (1 − d) μ
μ_b ← d μ_b^A + (1 − d) μ
t ← (s_A + s_B) / ||s_A + s_B||
b ← n × t
```

Algorithm 2: PROXGAUSSSEIDEL The PROX Gauss-Seidel variant with an adaptive r-Factor strategy and parametric friction cones. The product $\mathbf{M}^{-1}\mathbf{J}^T$ may be precomputed.

Data: Indices of all contacts $\mathcal{K}$, indices of all bodies B, and $\mathbf{J}$, $\mathbf{M}$, $\mathbf{b}$, r, λ^0, ν.
Result: λ^k

```
(k, λ^k, ε^k) ← (0, λ^0, ∞)
while not converged do
    w ← M^{-1} J^T λ^k
    foreach i ∈ K do
        I ≡ {n, f} ← indices of block N_i, F_i
        z_I ← λ_I^k − r (J_{I,B} w + b_I)
        λ_n^{k+1} ← prox_{N_i}(z_n)
        F ← G(λ_n^{k+1}, ...)
        λ_f^{k+1} ← prox_F(z_f)
        w ← w + (M^{-1} J^T)_{B,I} (λ_I^{k+1} − λ_I^k)
    end
    ε^{k+1} = || λ^{k+1} − λ^k ||_∞
    if ε^{k+1} > ε^k then
        r ← νr
    else
        (λ^k, ε^k, k) ← (λ^{k+1}, ε^{k+1}, k + 1)
    end
end
```

Here, k is the iteration index and r is a scalar relaxation parameter known as the r-factor. This is how our parametric model can be used in a typical sweeping process. First, Equation (4) is used to instantiate the current friction cone, $\mathcal{F}$, and then that cone is used in the proximal point update given by Equation (14).

Algorithm 2 illustrates how a PROX-based block Gauss–Seidel variant is modified to accommodate our friction models. Observe that the only change is the addition of line 8 in the algorithm, before the friction proximal step. This line instantiates the friction model using Equation (4).

4.1. Linear complementarity problem (LCP) based simulators

For LCP-based approaches, one may discretize a parameterized cone by shooting rays from the origin of the limit surface in various directions and use the limit surface points to build a polygonal approximation to the generated cone. Each facet of the polyhedral cone will match one complementary constraint in the LCP model. While it is trivial to generate the polygonal facets, the main drawback is that one may need many facets to obtain a good approximation. The memory footprint of the LCP has quadratic scaling with the number of constraints and the solver time will suffer accordingly. Hence, nonlinear complementary formulations can be more attractive for parametric cones.

4.2. Nonlinear complementarity problem (NCP) based simulators

We will now outline how to use a parameterized $\mathcal{F}$ in a Newton-type framework using non-smooth functions, such as the Fischer–Burmeister function [MEM19]. Without loss of generality, assume that we have any type of complementary function, $\psi(a,b) : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}$ such that

$$0 \leq a \perp b \geq 0 \quad \Leftrightarrow \quad \psi(a,b) = 0. \tag{15}$$

Using an implicit limit surface models of the cone, we let $\phi_{\mathcal{F}}$ be the corresponding implicit function of $\mathcal{F}$,

$$\phi_{\mathcal{F}} \equiv \mathbf{f}^T \mathbf{R}^T \mathbf{C}_M \mathbf{R}\mathbf{f} - \lambda_n^2. \tag{16}$$

Then, by the principle of maximal dissipation, we can write:

$$\nabla \phi_{\mathcal{F}}(\mathbf{f}) = -\beta\, \mathbf{v}_t, \tag{17}$$

where $\beta \geq 0$ is an auxiliary scalar variable. We can now restate the model with the help of the complementary function,

$$\psi(\beta, -\phi_{\mathcal{F}}(\mathbf{f})) = 0, \tag{18}$$

$$\psi(\mathbf{v}_t^T \mathbf{v}_t, \mathbf{w}^T \mathbf{w}) = 0, \tag{19}$$

where we now introduce $\mathbf{w} = \nabla \phi_{\mathcal{F}}(\mathbf{f}) + \beta\, \mathbf{v}_t$. The above model gives us a root search problem and can be solved with a Newton type of method. For this purpose, we must obtain the generalized Jacobian of these equations. The differential becomes

$$d\psi(\beta, -\phi(\mathbf{f})) = \partial_a \psi\, d\beta - \partial_b \psi\, \nabla_\phi^T d\mathbf{f}, \tag{20}$$

$$d\psi(\mathbf{v}_t^T \mathbf{v}_t, \mathbf{w}^T \mathbf{w}) = 2\partial_a \psi\, \mathbf{v}_t^T d\mathbf{v}_t + 2\partial_b \psi\, \mathbf{w}^T d\mathbf{w}, \tag{21}$$

where

$$d\mathbf{w} = \mathbf{v}_t\, d\beta + \nabla^2 \phi\, d\mathbf{f} + \beta\, d\mathbf{v}_t. \tag{22}$$

Here, we use ∇^2 to denote the Hessian of ϕ. Assembling all parts, we can write:

$$\begin{bmatrix} d\psi(\beta, -\phi(\mathbf{f})) \\ d\psi(\mathbf{v}_t^T \mathbf{v}_t, \mathbf{w}^T \mathbf{w}) \end{bmatrix} = \mathbf{J}_\psi \begin{bmatrix} d\beta \\ d\mathbf{f} \\ d\mathbf{v}_t \end{bmatrix}, \tag{23}$$

where $\mathbf{J}_\psi$ is the Jacobian one will need for implementing a Newton method, and is computed as:

$$\mathbf{J}_\psi \equiv \begin{bmatrix} \partial_a \psi & -\partial_b \psi\, \nabla_\phi^T & 0 \\ 2\big(\partial_b \psi\, \mathbf{w}^T \mathbf{v}_t\big) & 2\big(\partial_b \psi\, \mathbf{w}^T \nabla^2 \phi\big) & 2\big(\partial_a \psi\, \mathbf{v}_t^T + \partial_b \psi\, \mathbf{w}^T \beta\big) \end{bmatrix}.$$

4.3. Pivoting solver

Pivoting methods for solving frictional contact attempt to find a partitioning of the system into active and inactive variables. These labels indicate whether a variable is within bounds, and thus unknown, or if boundary conditions are violated by its current value, and hence it is determined by a projection onto the limit surface, or a linear approximation of it, and the variable is treated as a known entity.

The pivoting method used in our experiments initializes the limit surfaces using an estimate of the non-interpenetration forces, λ_n, by a preliminary step that finds a solution to the multibody system excluding friction. The friction cones are then updated according to Equation (13), and a direct solver is used to find a solution of the system combining normal forces and friction forces. The labelling of active and inactive variables is revised, and subsequent iterations can be used to refine the limit surfaces. Further details on the block Bard-type algorithm used in our experiments can be found in the Vortex Dynamics documentation [CM 17].

4.4. Implementation details

Notice that the friction cone generator $\mathcal{G}$ returns both a friction cone $\mathcal{F}_M$ and a cone orientation $\mathbf{C} \equiv \mathbf{n}\ \mathbf{t}\ \mathbf{b}$ as indicated in Algorithm 1. However, $\mathbf{C}$ is needed for the assembly of the contact Jacobian, as is evident from Equation (3). Hence, one may wish to invoke the generator when assembling the contact Jacobian or split the generator implementation into two sub-routines. The choice is intimately related to how the time-discretization of the friction cone is implemented. Imagine that the cone generator parametrically depends on the sliding velocity $\mathbf{v}_t$ and that a full implicit scheme is wanted, in which case, the generator truly needs to be invoked every time before calling a proximal operator. However, in most cases, the positions and orientations are not updated inside an algorithm as shown in Algorithm 2. In this case, both the cone orientation and limit surface can be computed outside the solver for improved computational efficiency. For the Newton-type solver, we outlined that one must update cone orientations continuously as positions and velocities are solved in a fully coupled way in a full implicit time-integration method.

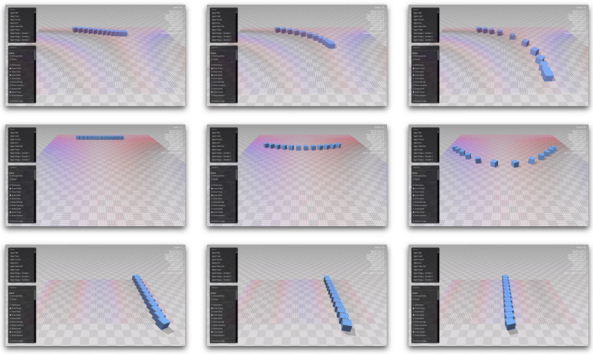


Figure 3: *Procedurally generated structural directions provide intuitive interactive control. These didactic examples demonstrate channelling (top), spreading (middle) and slaloming (bottom).*

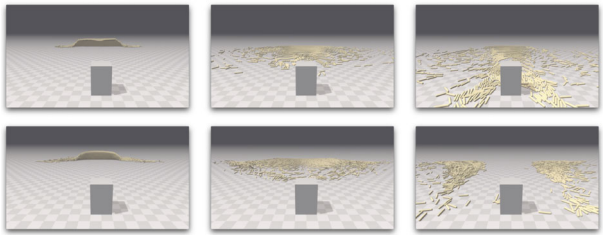


Figure 4: *A structure field on a ravine slope can be used to steer a log slide towards (top) or around (bottom) a cube-shaped building.*

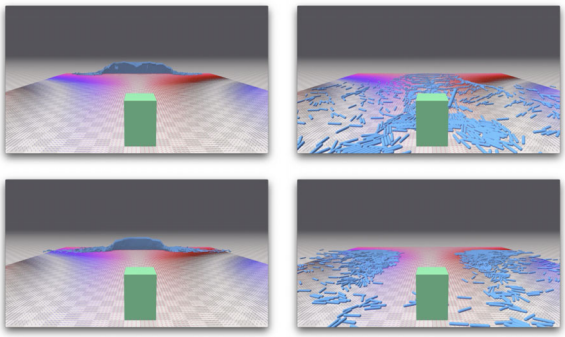


Figure 5: *Visualization of the ravine example structure fields. Logs collide with the box (top) and spread prior to collision (bottom).*

We note that in our implementation, an additional drilling torque is included in the friction model that introduces an angular moment about the contact normal based on an additional friction coefficient, μ_τ. The value of this coefficient is computed similarly to μ_t and μ_b. Inclusion of this torque is optional from a modelling perspective, but does result in a higher order limit surface and thus requires a third diagonal term in $\mathbf{C}$ as $\frac{1}{\mu_\tau}^2$ and promoting $\mathbf{R}$ to a 3D rotation matrix. The proximal operator for this surface can then be solved numerically, as outlined in [Erl17], whereas omitting this additional term gives a planar surface and the projection can be solved analytically.

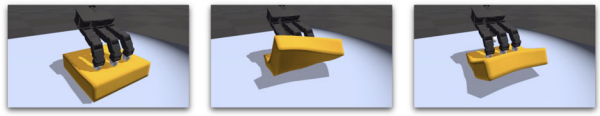


Figure 6: *Grasping with a robotic gripper demonstrates how our model affects the friction behaviour during interaction with a deformable object.*

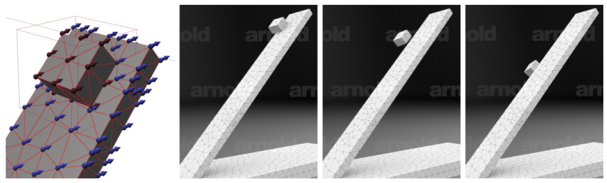


Figure 7: *A box on an inclined plane shows how oblique structure directions allow control of the sliding trajectory.*

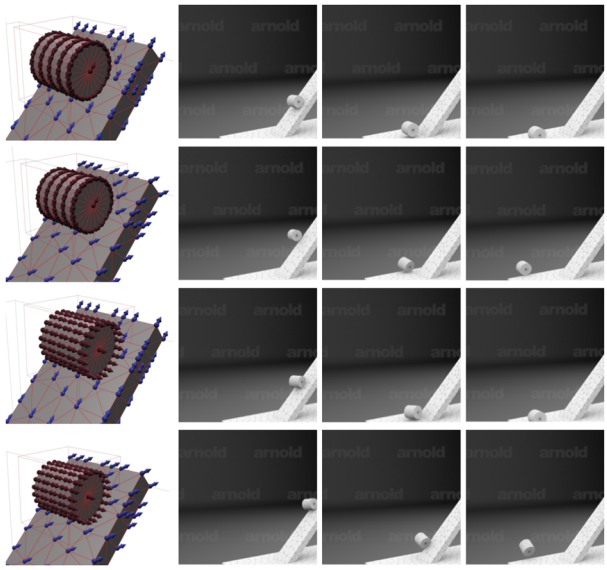


Figure 8: *Structure directions of a cylinder and inclined plank are visualized on the left. These produce different rolling and sliding behaviours (top to bottom): cylinder-concentric with plank-aligned produces stable rolling; cylinder-concentric with plank-oblique causes the cylinder to skid off the plank; orthogonal cylinder and plank directions yield isotropic friction cone behaviour; and cylinder-axial with plank-oblique also produces skidding.*

5. Results

To demonstrate the agility of the new model and its ability to be included in very different simulation paradigms, we decided to implement the Matchstick model into three existing simulators: Vortex Dynamics, NVIDIA Flex and PROX [Ken]. Respectively, these frameworks use an LCP-based formulation with direct solver, NCP-based formulation with fully implicit Newton solver and iterative Gauss–Seidel type scheme [Erl17]. Figures 3–6 use Flex [MEM19],

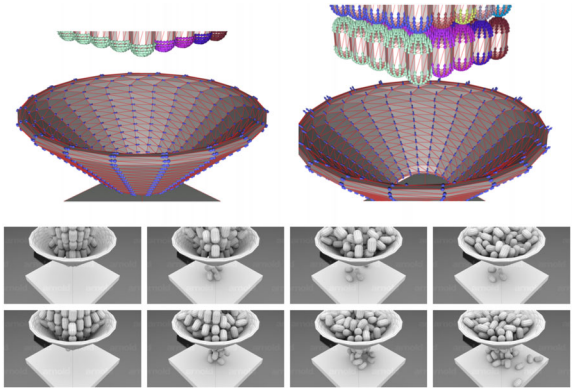


Figure 9: *Pills in a hopper flow or jam depending on structure directions. Top left shows concentric structure directions (promotes jamming), while top right shows radial/axial structure directions (promotes flowing). Middle and bottom rows show resulting simulations of jamming and flowing, respectively. Note that the anisotropic simulations result in different orientations of the capsules. This kind of behaviour difference cannot be created using only isotropic friction to cause the jam.*

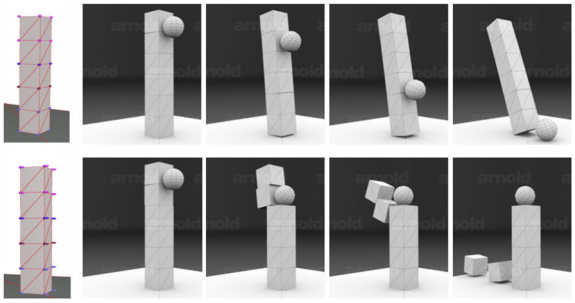


Figure 10: *A zero-structure field sphere projectile causes a tower to fall or break depending on structure directions. Top, structure directions orthogonal to impact permit large friction forces, and the impact causes the tower to tip and fall. Bottom, structure directions parallel to impact permit only small friction forces, and the sphere impact only knocks the top blocks off the stack.*

Figures 7–11 use PROX and Figures 12 and 13 use Vortex. This choice of solvers shows examples of semi-implicit time-integration and fully implicitly time-integration, as well as global versus local coupled schemes, and fully coupled normal and friction forces too.

Using the Flex solver, Figure 3 shows simple cases where the behaviour of sliding boxes can be altered easily by generating a varying structure field for the plane. On a sloped plane, Figures 4 and 5 show control of a log slide in a ravine. We also demonstrate changing structure directions on a soft object with the robotic grasping example shown in Figure 6. As seen in the Supplementary Movie, interactive changes to the structure fields immediately change the behaviour as the Allegro gripper strokes the soft gel-like material.

Using the PROX solver, Figure 7 shows a didactic example of a box on an inclined plane. Likewise, Figure 8 illustrates cases of a rolling cylinder that changes behaviour when sliding occurs on an

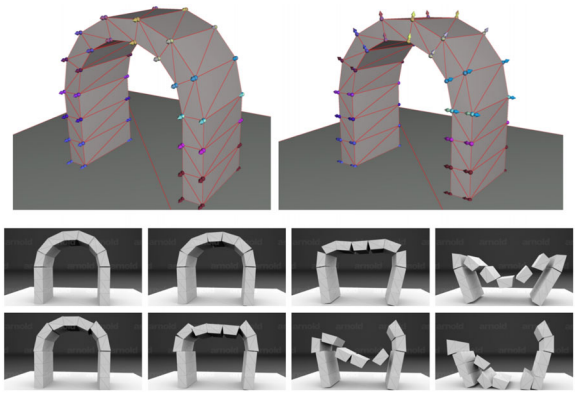


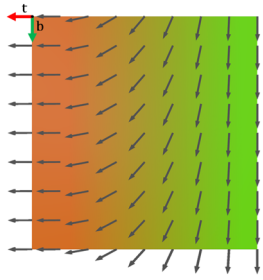
Figure 11: *Structure directions for the destruction of an arch. Top left, the design provides only weak radial friction. Top right, the directions provide strong radial friction. Middle row, with strong radial friction, the arch breaks with the pillars being pushed outwards. Bottom row, with weak radial friction, the arch falls faster due to sliding between the upper stones.*

inclined plane. Structure fields can be used for flow rate control or jamming effects, as shown in Figure 9. Artistic structure directions may be used to control the desired behaviour for destruction of masonry structures, as demonstrated in Figures 10 and 11.

Using the Vortex solver, Figure 12 shows a variation of the controlled channelling behaviour seen in Figure 3, and Figure 13 shows an example of ground structure directions used to influence the skidding behaviour of a vehicle.

In our work, we either procedurally generate structure fields for mesh vertices and interpolated these values for contact points, or we use texture maps to store the structure fields. The texture maps allow for a traditional artistic way of modelling structure directions with an imaging inspired approach.

5.1. Structure field direction textures



The structure directions used by the Matchstick model can be conveniently stored as a texture map. The inline figure on the right shows an example of one such texture map used in our experiments, along with structure directions shown in grey. Here, the red and green channels of the image are used to store the direction information. The process we use to compute the structure direction from a texture is similar to normal mapping, which is a common computer graphics technique, and we briefly explain this process below.

Each vertex in the object's polygonal mesh stores a normal and a tangent vector that are used to define surface characteristics at the vertex. These are specified during the modelling process. When a collision is detected between two objects, the barycentric

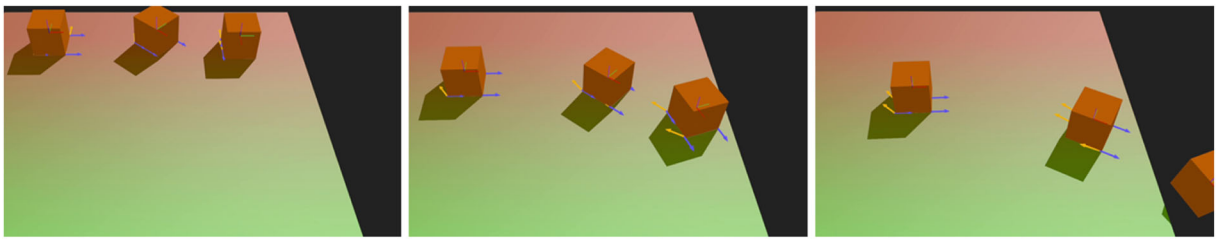


Figure 12: *Three boxes with different starting orientations slide down an inclined plane with a Matchstick friction texture. The path of the boxes is determined by the principal structure direction, which has lower friction compared to the lateral direction. As a result, all boxes exit the plane near the same location.*

Figure 13: *A moving vehicle brakes hard on a terrain with a friction texture. The region of the terrain with the orange texture has a high friction coefficient in the direction lateral to the tires. However, when the vehicle skids onto the green region, the structure direction changes suddenly and the vehicle begins to fishtail due to friction anisotropy.*

coordinates at the contact point are used to interpolate between the normal and tangent vectors at each vertex in the colliding triangle. This gives a normal $\mathbf{n}$ and tangent $\mathbf{t}$ vector at the contact point, and this is done for both objects. We compute the binormal vector as $\mathbf{b} = \mathbf{n} \times \mathbf{t}$, which gives an orthogonal set of basis vectors. The barycentric coordinates are also used to interpolate the UV texture coordinates, which are likewise specified at each vertex. Using the interpolated texture coordinates u, v, we perform a lookup in the friction texture I such that

$$\mathbf{c} = I(u, v),$$

where $\mathbf{c} = r, g, b$ is a tuple with red, green and blue colour values in the range $[0, 1]$. The structural direction $\mathbf{s}_i$ at the surface of body i may then be computed using the red and green channels, such that

$$\mathbf{s}_i = (2r - 1)\mathbf{t} + (2g - 1)\mathbf{b}.$$

Figures 12 and 13 show examples of friction textures being used to encode the structure directions. The textures were created using a typical image editing software application.

6. Discussion and Limitations

We note that there is a need for accurate normal force distributions. Most stationary point method solvers just pick one out of multiple solutions; however, we want the solution that is closest to the real contact force distribution. Iterative methods, such as the one outline in Algorithm 2, tend to average out force distributions when no friction is present. But, there is no guarantee on which of multiple solutions this type of method will choose once converged. Adding compliance and friction to the dynamics is one solution that can help, and we note that Newton-type methods may ultimately be more appropriate due to improved convergence behaviours.

Inaccuracy in solving proximal operators can produce imprecise frictional forces, generating torque effects even when no torque effects should be present. This is not a problem specific for our model, but shared by any simulator that uses a projection onto an elliptical cone with a large aspect ratio.

The Matchstick model is limited to classes of materials that can be described by a structural direction with low frictional resistance. Our new model includes all the behaviours that can be obtained with the usual Coulomb friction modelling done in rigid body simulations. However, in the extension of a simulation to an anistropic friction model, our Matchstick model offers an intuitive solution to defining highly anisotropic friction cones. Note that our model supports setting one of the structural directions to zero, in which case only the object with the non-zero structure determines the anisotropy. This will make the object with the zero structural field appear isotropic, and the non-zero object anisotropic.

Surface descriptions in simulators using our Matchstick friction model must be extended with information about structure directions such that these can be extracted at points of contacts. From an implementation perspective, this is no more complicated than applying textures to surfaces. This is straightforward in computer graphic applications, and also allows artistic modelling, where artists can paint structure directions on surfaces, or directions can be obtained through image processing of photographs of real materials. Material descriptions are no longer simple coefficients of friction. Instead, friction cones are instantiated and evaluated during simulation as they depend on the kinematic state of the system.

7. Conclusion and Future Work

We have presented a new phenomenological anisotropic friction model for structured surfaces. We proved that our novel friction modelling can be incorporated into simulators that support convex cones. We note that our model is particularly convenient for PROX schemes and variants thereof. Our results include a wide range of examples using different simulators and scenes ranging from small didactic cases illustrating the intuitive nature of the Matchstick model to more complex scenarios of digital prototyping, jamming, destruction physics, masonry structures, ravines, skidding cars and more. This provides evidence that using the Matchstick model to control friction behaviour is intuitive. The behaviours we obtain throughout these examples were designed with minimal effort in tuning structure fields. Even for robot hand grasping scenario, we can interactively play with structure fields generated on the fly to explore the consequences of stroking a gel object with robotic fingers. Furthermore, we have shown how normal mapping techniques are a convenient way to generate the material structure fields of different objects.

The novelty of considering cones as state-dependent high-dimensional parametric functions that can change dynamically opens up a doorway for a multitude of modelling possibilities not yet seen in the field of computer graphics. We believe that it will be straightforward to extend our work to address friction models with other dependencies, such as sliding velocity (Stribeck effect) or nonlinear scaling with normal forces as employed in the recent work on cloth simulation [CFW13]. We note that anisotropic friction is generally in demand for cloth simulation [LDN*18].

Our car skidding example is highly motivating from an animation system viewpoint. We note that Berry *et al.* [BBM*17] demonstrate usability improvement in a driving system by using splines to sketch trajectories of cars. We speculate that it may be possible to produce similar trajectories using our friction model.

Acknowledgements

The authors gratefully acknowledge the financial support of the Natural Sciences and Engineering Research Council of Canada (NSERC) and Fonds de Recherche du Québec Nature et technologies (FRQNT). We also thank CM Labs Simulations for providing the truck example.

Appendix A: Friction Tensors

This appendix contains a detailed analysis of the friction tensors model. Pabst *et al.* [PTS09] define dynamic friction by a direct evaluation of

$$\mathbf{f} = -\lambda_n \, \kappa \left(\mathbf{Q}_\mathsf{A} + \mathbf{R}\mathbf{Q}_\mathsf{B}\mathbf{R}^T\right)\hat{\mathbf{v}}_t, \tag{A.1}$$

where

$$\hat{\mathbf{v}}_t \equiv \tfrac{\mathbf{v}_t}{\|\mathbf{v}_t\|}, \tag{A.2}$$

$$\phi \equiv \cos^{-1}(\mathbf{s}_\mathsf{A} \cdot \mathbf{s}_\mathsf{B}), \tag{A.3}$$

$$\mathbf{R} \equiv \mathbf{R}(\mathbf{n}, \phi) \tag{A.4}$$

where $\mathbf{R}(\mathbf{n}, \phi)$ is the rotation around $\mathbf{n}$ with angle ϕ. Material parameters are described by one scalar and two tensors: $\kappa \in \mathbb{R}_+$ and $\mathbf{Q}_\mathsf{A}, \mathbf{Q}_\mathsf{B} \in \mathbb{R}^{2\times 2}$. For planar friction, this gives a total of nine parameters to describe.

We will now examine if a cone description can be extracted from the model by Pabst *et al.* We define the linear operator $\mathbf{L}$ as follows:

$$\mathbf{f} = -\lambda_n \underbrace{\kappa \left(\mathbf{Q}_\mathsf{A} + \mathbf{R}\mathbf{Q}_\mathsf{B}\mathbf{R}^T\right)}_{\equiv \mathbf{L}} \hat{\mathbf{v}}_t, \tag{A.5}$$

$$= -\lambda_n \, \mathbf{L}\, \hat{\mathbf{v}}_t. \tag{A.6}$$

Under the assumption that $\mathbf{L}$ is non-singular, we may now write:

$$\mathbf{f} = -\lambda_n \, \mathbf{L}\, \hat{\mathbf{v}}_t, \tag{A.7}$$

$$\mathbf{L}^{-1}\mathbf{f} = -\lambda_n \, \hat{\mathbf{v}}_t, \tag{A.8}$$

$$\| \, \mathbf{L}^{-1}\mathbf{f} \, \|^2 = \lambda_n^2, \tag{A.9}$$

$$\mathbf{f}^T \underbrace{\left(\mathbf{L}^{-T}\mathbf{L}^{-1}\right)}_{\equiv \mathbf{C}_P} \mathbf{f} = \lambda_n^2. \tag{A.10}$$

We observe that this is the same as an elliptical-cone Coulomb friction model. An eigenvalue decomposition will recover the usual cone description.

$$\mathbf{C}_P \equiv \mathbf{R}^T \begin{bmatrix} \frac{1}{\mu_t^2} & 0 \\ 0 & \frac{1}{\mu_b^2} \end{bmatrix} \mathbf{R}. \tag{A.11}$$

Next, we will investigate dissipation of the model. We start by writing up the instantaneous power

$$\mathbf{v}_t \cdot \mathbf{f} = -\lambda \frac{\mathbf{v}_t^T \mathbf{L} \mathbf{v}_t}{\| \, \mathbf{v}_t \, \|}. \tag{A.12}$$

We observe that if $\mathbf{Q}_\mathsf{A}$ and $\mathbf{Q}_\mathsf{B}$ are symmetric positive definite, then we have sufficient conditions to state that $\mathbf{L}$ will always be symmetric and positive definite. In this case, friction force is always dissipating.

Recall that principle of maximum dissipation is equivalent with $-\hat{\mathbf{v}}_t \in \mathcal{N}_{\mathbf{C}_P}(\mathbf{f})$ that is the negative sliding velocity must be in the normal cone of the friction cone if $\mathbf{f}$ is the maximal dissipate force. We will use an implicit function to express the friction cone of Pabst *et al.*,

$$\psi(\mathbf{f}) \equiv \mathbf{f}^T \left(\mathbf{L}^{-T}\mathbf{L}^{-1}\right)\mathbf{f} - \lambda_n^2. \tag{A.13}$$

To prove if $\mathbf{f}$ is maximal dissipating, we wish to show that there exists some value $\beta > 0$ such that

$$-\beta \, \hat{\mathbf{v}}_t = \nabla_\mathbf{f} \, \psi(\mathbf{f}). \tag{A.14}$$

We find

$$d_{\mathbf{f}}\,\psi = d\mathbf{f}^T\left(\mathbf{L}^{-T}\mathbf{L}^{-1}\right)\mathbf{f} + \mathbf{f}^T\left(\mathbf{L}^{-T}\mathbf{L}^{-1}\right)d\mathbf{f} \tag{A.15}$$

$$= \underbrace{2\mathbf{f}^T\left(\mathbf{L}^{-T}\mathbf{L}^{-1}\right)}_{\equiv(\nabla_{\mathbf{f}}\,\psi)^T} d\mathbf{f}. \tag{A.16}$$

Hence,

$$-\beta\,\hat{\mathbf{v}}_t = 2\left(\mathbf{L}^{-T}\mathbf{L}^{-1}\right)\mathbf{f}. \tag{A.17}$$

We redefine $\beta \leftarrow \frac{\beta}{2\lambda_n}$ and recall $\mathbf{f} = -\lambda_n\mathbf{L}\hat{\mathbf{v}}_t$, then we have

$$\beta\,\hat{\mathbf{v}}_t = \mathbf{L}^{-T}\,\hat{\mathbf{v}}_t. \tag{A.18}$$

This shows that Pabst *et al.* model is only maximal dissipating when the direction of sliding is an eigenvector of $\mathbf{L}^{-T}$. This proves that there can be single cases (two for planar sliding), where the model is maximal dissipating but in general $\mathbf{v}_t$ can have any direction. Hence, there are infinitely many cases where the principle of maximum dissipation will not hold.

In conclusion, the model by Pabst *et al.* cannot efficiently plugged into an existing simulator based on the concept of cones as it requires $\mathbf{L}$ to be non-singular and needs the computation of $\mathbf{C}_p$ and its eigenvalue decomposition. Further, when $\mathbf{L}$ is symmetric positive definite, the model is always dissipating but not necessarily maximal dissipating.

References

[AFC*10] Allard J., Faure F., Courtecuisse H., Falipou F., Duriez C., Kry P. G.: Volume contact constraints at arbitrary resolution. *ACM Transactions on Graphics 29*, 4 (July 2010), 82:1–82:10.

[Bar89] Baraff D.: Analytical methods for dynamic simulation of non-penetrating rigid bodies. *SIGGRAPH Computer Graphics 23*, 3 (1989), 223–232.

[Bar94] Baraff D.: Fast contact force computation for nonpenetrating rigid bodies. In *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1994), SIGGRAPH '94, ACM, pp. 23–34.

[BBM*17] Berry B., Bianchi D., Marshall S., Nguyen G., Oakley P., Rangaswamy S., Tai F.: Crazy eight: The making of a race sequence in Disney/Pixar's "Cars 3". In *ACM SIGGRAPH 2017 Production Sessions* (New York, NY, USA, 2017), SIGGRAPH '17, ACM, pp. 6–6.

[BET14] Bender J., Erleben K., Trinkle J.: Interactive simulation of rigid body dynamics in computer graphics. *Computer Graphics Forum 33*, 1 (2014), 246–270.

[BFA02] Bridson R., Fedkiw R., Anderson J.: Robust treatment of collisions, contact and friction for cloth animation. *ACM Transactions on Graphics 21*, 3 (July 2002), 594–603.

[BNT*15] Bouchard C., Nesme M., Tournier M., Wang B., Faure F., Kry P. G.: 6D frictional contact for rigid bodies. In *Proceedings of the 41st Graphics Interface Conference* (Toronto, Ontario, Canada, 2015), GI '15, Canadian Information Processing Society, pp. 105–114.

[CDA*18] Costes A., Danieau F., Argelaguet F., Lécuyer A., Guillotel P.: Haptic material: A holistic approach for haptic texture mapping. In *International Conference on Human Haptic Sensing and Touch Enabled Computer Applications* (2018), Springer, pp. 37–45.

[CFW13] Chen Z., Feng R., Wang H.: Modeling friction and air effects between cloth and deformable bodies. *ACM Transactions on Graphics 32*, 4 (July 2013), 88:1–88:8.

[CLO17] Cirio G., Lopez-Moreno J., Otaduy M. A.: Yarn-level cloth simulation with sliding persistent contacts. *IEEE Transactions on Visualization and Computer Graphics 23*, 2 (Feb 2017), 1152–1162.

[CM 17] CM Labs Simulations: *Theory Guide: Vortex Software's Multibody Dynamics Engine*. Tech. rep., 2017.

[DBDB11] Daviet G., Bertails-Descoubes F., Boissieux L.: A hybrid iterative solver for robustly capturing Coulomb friction in hair dynamics. *ACM Transactions on Graphics 30*, 6 (Dec. 2011), 139:1–139:12.

[DEKA19] Dreßel K., Erleben K., Kry P. G., Andrews S.: Automated acquisition of anisotropic friction. In *16th Conference on Computer and Robot Vision (CRV 2019)* (2019).

[EATK18] Enzenhöfer A., Andrews, S., Teichmann, M., Kövecses, J.: Comparison of mixed linear complementarity problem solvers for multibody simulations with contact. In *Workshop on Virtual Reality Interaction and Physical Simulation* (2018), The Eurographics Association, pp. 11–20.

[Erl17] Erleben, K.: Rigid body contact problems using proximal operators. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2017), SCA '17, ACM, pp. 13:1–13:12.

[Erl19] Erleben, K.: PROX Matchstick supplementary code repository (https://github.com/erleben/matchstick), Accessed 2019.

[GRP89] Goyal, S., Ruina, A., Papadopoulos, J.: Limit surface and moment function descriptions of planar sliding. In *Proceedings of the 1989 IEEE International Conference on Robotics and Automation (Vol. 2)* (Scottsdale, AZ, 1989), pp. 794–799.

[KEP05] Kaufman D. M., Edmunds T., Pai D. K.: Fast frictional dynamics for rigid bodies. *ACM Transactions on Graphics 24*, 3 (July 2005), 946–956.

[KSJP08] Kaufman D. M., Sueda S., James D. L., Pai D. K.: Staggered projections for frictional contact in multibody systems. *ACM Transactions on Graphics 27*, 5 (Dec. 2008), 164:1–164:11.

[LDN*18] Li J., Daviet G., Narain R., Bertails-Descoubes F., Overby M., Brown G. E., Boissieux L.: An implicit frictional contact solver for adaptive cloth simulation. *ACM Transactions on Graphics 37*, 4 (July 2018), 52:1–52:15.

[LGS*98] Liley M., Gourdon D., Stamou D., Meseth U., Fischer T., Lautz C., Stahlberg H., Vogel H., Burnham N., Duschl C.: Friction anisotropy and asymmetry of a compliant monolayer induced by a small molecular tilt. *Science 280*, 5361 (1998), 273–275.

[MEM19] Macklin, M., Erleben, K., Müller, M., Chentanez, N., Jeschke, S., Makoviychuk, V.: Non-smooth Newton methods for deformable multi-body dynamics. *ACM Transactions on Graphics 38*, 5 (2019), 140:1–140:20.

[OTSG09] Otaduy M. A., Tamstorf R., Steinemann D., Gross M.: Implicit contact handling for deformable objects. *Computer Graphics Forum (Proceedings of Eurographics) 28*, 2 (Apr 2009), 559–568.

[PDJ*01] Pai D. K., Doel K. v. d., James D. L., Lang J., Lloyd J. E., Richmond J. L., Yau S. H.: Scanning physical interaction behavior of 3D objects. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 2001), SIGGRAPH '01, ACM, pp. 87–96.

[PTS09] Pabst S., Thomaszewski B., Straßer W.: Anisotropic friction for deformable surfaces and solids. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2009), SCA '09, ACM, pp. 149–154.

[TMDO15] Talvas A., Marchal M., Duriez C., Otaduy M. A.: Aggregate constraints for virtual manipulation with soft fingers. *IEEE Transactions on Visualization and Computer Graphics 21*, 4 (2015), 452–461.

[UVM*12] Umbanhowar P., Vose T. H., Mitani A., Hirai S., Lynch K. M.: The effect of anisotropic friction on vibratory velocity fields. In *2012 IEEE International Conference on Robotics and Automation* (May 2012), pp. 2584–2591.

[WL17] Walker S. V., Leine R. I.: Anisotropic dry friction with non-convex force reservoirs: Modeling and experiments. In *9th European Nonlinear Dynamics Conference (ENOC 2017)* (2017), pp. 1–7.

[WSJP17] Wang J.-H., Setaluri R., James D. L., Pai D. K.: Bounce maps: An improved restitution model for real-time rigid-body impact. *ACM Transactions on Graphics 36*, 4 (2017), 150.

[YW12] Yu C., Wang Q. J.: Friction anisotropy with respect to topographic orientation. *Scientific Reports 2* (2012), 988.

Supporting Information

Additional supporting information may be found online in the Supporting Information section at the end of the article.

Video S1