dvc3D: a three dimensional physical simulation tool for rigid bodies with contacts and Coulomb friction

Binh Nguyen*, **Jeffrey C. Trinkle#**

* Computer Science Department
Rensselaer Polytechnic Insitute
Troy, NY 12180, USA
e-mail: nguyeb2@cs.rpi.edu

# Computer Science Department
Rensselaer Polytechnic Insitute
Troy, NY 12180, USA
e-mail: trink@cs.rpi.edu

ABSTRACT

In this paper, we introduce a physical simulation library for three-dimensional for multi-rigid-bodies dynamics with contacts and Coulomb friction named dvc3d. The library focuses on time stepping simulation methods that have been justified in theories to converge to the underlying physical models as step size goes to zero. We further verify dvc3D's simulation accuracy by comparing with experimental results. We also discuss a recent addition of a new contact model that is capable of handling locally non-convex configuration spaces that are usually incorrectly modeled in other methods. Then, we follow by the discussion of dvc3D's two types of solvers: accurate but slow direct methods that are suitable for small scientific simulation scenes or fast but less accurate iterative approaches that can handle much larger systems of bodies.

Keywords: physical simulation, library, tool, software, contact model, nonconvex, iterative method, direct method, simulation verification.

1 Introduction

For robotic systems to automatically plan and execute manipulation tasks involving intermittent contact and friction, the ability to accurately predict the motions of manipulated objects is invaluable. Due to the intermittency of contact and the presence of stick-slip frictional behavior, dynamic models of these problems are inherently nonsmooth and usually do not have closed-form solutions. Thus, they must rely on simulation techniques.

With such importance, unsurprisingly, robotics software tools often make use of physical simulation through certain physics engines. For example, Webots [7], a development environment used to model, program and simulate mobile robots employ the opensource Open Dynamics Engine [12]. Microsoft Robotics Studio, a Windows-based environment for hobbyist, academic and commercial developers to create robotics applications for a variety of hardware platforms relies on the commercial physics engine Nvidia PhysX. Because many of the current physics engines are developed for gaming and virtual reality applications, they usually focus on the performance at the expense of accuracy. As the result of it, the simulation results of these libraries often do not match with the ones inferred from the underlying physical model. Figure 1 depicts simulations of a stack of ten boxes initially placed at the height of 15 units above the ground using the Bullet physics engine [5], the most popular opensource physics engine. Under the assumption made in the physical model, the expected outcome should be that the stack stays stably in contact with the ground unlike Bullet's outcomes. Moreover, Bullet exhibits indeterministic behavior by having three different results given the same initial state. Most of the gaming physics engines also suffer from these problems to some degrees.

dvc3d is different. The methods implemented in the library are numerically stable and provably convergent. In particular, it has been proven that when the Stewart-Trinkle (ST) method is applied to the dynamic system, as the step size goes to zero, the trajectories produced converge to the exact solution of the original physical model [15]. Figure 2 is an example showing that dvc3D's simulation result correctly predicts the system behavior.

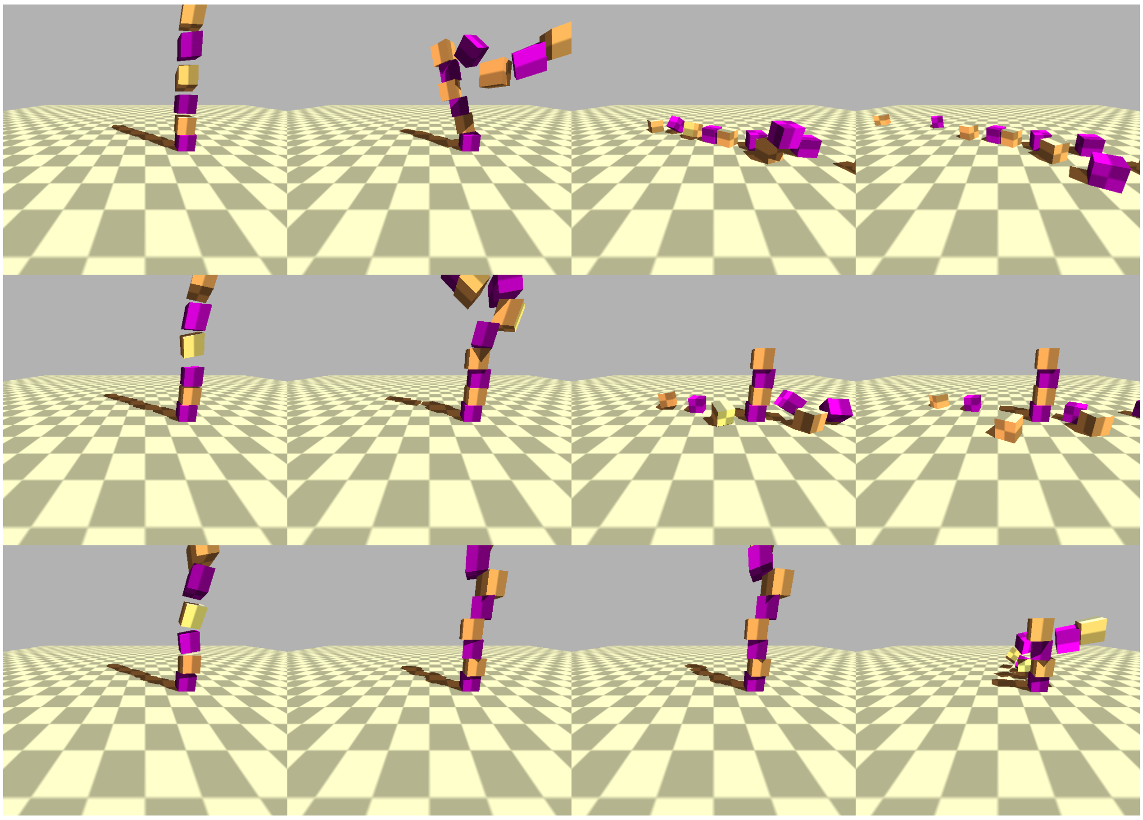


Figure 1: Box stack scene in Bullet in three runs with the same initial condition and model parameters. The row depicts different runs and columns correspond to time t=2,4,6,8 s.

2 dvc3d mathematical model

dvc3d underlying physic model consists of five components: the Newton-Euler equation, a kinematic map (to relate orientation parameters to angular velocity variables), equality constraints (to model joint connections), a normal contact condition (to model intermittent contact behavior), and a dry friction law satisfying the Maximum Work Principle.

But first, we need to introduce the notion of complementarity problems, which will be used to model the intermittency of contact and friction stick-slip.

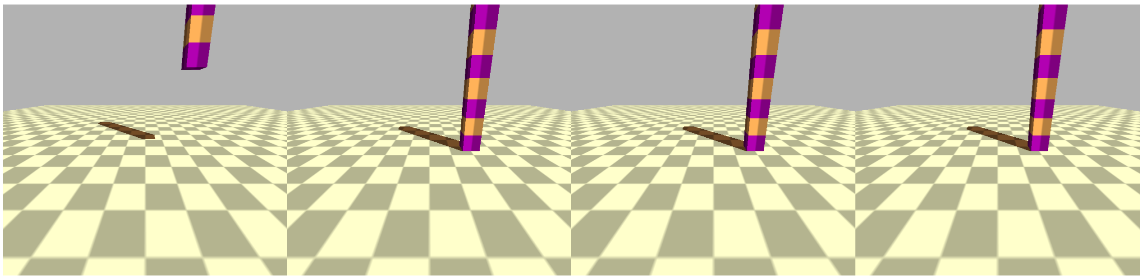
Definition 1 (Differential Complementarity Problem(DCP))**.** *Let* $h(\mathbf{u},\mathbf{v}) : \mathbb{R}^{n1} \times \mathbb{R}^{n2} \leftarrow \mathbb{R}^{n1}$ *and* $f(\mathbf{u},\mathbf{v}) : \mathbb{R}^{n1} \times \mathbb{R}^{n2} \leftarrow \mathbb{R}^{n2}$ *be given vector functions of* $\mathbf{u} \in \mathbb{R}^{n1}$ *and* $\mathbf{v} \in \mathbb{R}^{n2}$*. Find* $\mathbf{u}$ *and* $\mathbf{v}$ *satisfying:*

$$\dot{\mathbf{u}} = h(\mathbf{u},\mathbf{v}), \mathbf{u}\,\textit{free}$$
$$0 \leq \mathbf{v} \perp f(\mathbf{u},\mathbf{v}) \geq 0$$
$$\textit{which equivalents to:} \mathbf{v} \geq 0, f(\mathbf{u},\mathbf{v}) \geq 0, \mathbf{v}^T f(\mathbf{u},\mathbf{v}) = 0.$$

Definition 2 (Mixed Linear Complementarity Problem(Mixed LCP))**.** *Let* $\mathbf{A} \in \mathbb{R}^{n_1 \times n_1}, \mathbf{B} \in \mathbb{R}^{n_2 \times n_2}, \mathbf{C} \in \mathbb{R}^{n_1 \times n_2}$*, and* $\mathbf{D} \in \mathbb{R}^{n_2 \times n_1}$ *be given matrices. let* $\mathbf{a} \in \mathbb{R}^{n_1}$ *and* $\mathbf{b} \in \mathbb{R}^{n_2}$ *be given vectors. Find* $\mathbf{u}$ *and* $\mathbf{v}$ *satisfying:*

$$0 = \mathbf{a} + \mathbf{A}\mathbf{u} + \mathbf{C}\mathbf{v} \tag{1}$$
$$0 \leq \mathbf{v} \perp b + \mathbf{D}\mathbf{u} + \mathbf{B}\mathbf{v} \geq 0 \tag{2}$$

More information of these complementarity problems can be found in [4, 11].



(a) t=0s (b) t=3s (c) t=7s (d) t=10s

Figure 2: The same box stack scene as in figure reffig:BoxStackBullet running in dvc3d

2.1 Instantaneous formulation of constrained dynamics

Consider a system of n rigid bodies. Let $\mathbf{q_j} \in \mathbb{R}^{n_q}$, $n_q \in [6, 7]$, be the position and orientation in inertia frame and $\boldsymbol{\nu}_j$ be the generalized velocity of the center of mass of body j-th. Define the generalized coordinates and velocity of the system as $\mathbf{q}$ and $\boldsymbol{\nu}$ by concatenating all the individual body configuration and velocity vectors $\mathbf{q_j}$ and $\boldsymbol{\nu}_j$. Time will be denoted by t.

Write down all five parts of the physic model, we will have a system of nonlinear DCP:

$$\begin{aligned}
\mathbf{M}(\mathbf{q})\dot{\boldsymbol{\nu}} &= \mathbf{W}_{\mathrm{n}}(\mathbf{q})\boldsymbol{\lambda}_{\mathrm{n}} + \mathbf{W}_{\mathrm{b}}(\mathbf{q})\boldsymbol{\lambda}_{\mathrm{b}} + \mathbf{W}_{\mathrm{t}}(\mathbf{q})\boldsymbol{\lambda}_{\mathrm{t}} + \mathbf{W}_{\mathrm{o}}(\mathbf{q})\boldsymbol{\lambda}_{\mathrm{o}} + \boldsymbol{\lambda}_{\mathrm{app}}(\mathbf{q}, t) + \boldsymbol{\lambda}_{\mathrm{vp}}(\mathbf{q}, \boldsymbol{\nu}) && (3)\\
\dot{\mathbf{q}} &= \mathbf{G}(\mathbf{q})\boldsymbol{\nu} && (4)\\
0 &= \mathbf{W}_{\mathrm{b}}(\mathbf{q}, t)\boldsymbol{\nu} + \frac{\partial \Phi(\mathbf{q}, t)}{\partial t} && (5)\\
0 \le \boldsymbol{\lambda}_{\mathrm{n}} &\perp \boldsymbol{\psi}_{\mathrm{n}}(\mathbf{q}, t) \ge 0 && (6)\\
0 &= (\mathbf{U}\boldsymbol{\lambda}_{\mathrm{n}})\circ(\mathbf{v}_{\mathrm{t}}) + \boldsymbol{\lambda}_{\mathrm{t}}\circ\boldsymbol{\sigma} && (7)\\
0 &= (\mathbf{U}\boldsymbol{\lambda}_{\mathrm{n}})\circ(\mathbf{v}_{\mathrm{o}}) + \boldsymbol{\lambda}_{\mathrm{o}}\circ\boldsymbol{\sigma} && (8)\\
0 \le \boldsymbol{\sigma} &\perp (\mathbf{U}\boldsymbol{\lambda}_{\mathrm{n}})\circ(\mathbf{U}\boldsymbol{\lambda}_{\mathrm{n}}) - \boldsymbol{\lambda}_{\mathrm{t}}\circ\boldsymbol{\lambda}_{\mathrm{t}} - \boldsymbol{\lambda}_{\mathrm{o}}\circ\boldsymbol{\lambda}_{\mathrm{o}} \ge 0 && (9)
\end{aligned}$$

Equation (3) is the Newton-Euler equations of the system, where $\mathbf{M}(\mathbf{q})$ is the mass/inertia matrix of the part, $\boldsymbol{\lambda}_{\mathrm{vp}}$ is the vector of Coriolis and centripetal forces, and $\mathbf{W}_{\mathrm{n}}$, $\mathbf{W}_{\mathrm{t}}$, and $\mathbf{W}_{\mathrm{o}}$ are dependent[1] on $\mathbf{q}$ and map the normal and frictional wrench magnitudes to the body reference frame.

$$\mathbf{W}_{\mathrm{n}} = \begin{bmatrix} \hat{\mathbf{n}}_i(\mathbf{q}) \\ \mathbf{r}_i(\mathbf{q}) \times \hat{\mathbf{n}}_i(\mathbf{q}) \end{bmatrix} \qquad \mathbf{W}_{\mathrm{t}} = \begin{bmatrix} \hat{\mathbf{t}}_i(\mathbf{q}) \\ \mathbf{r}_i(\mathbf{q}) \times \hat{\mathbf{t}}_i(\mathbf{q}) \end{bmatrix} \qquad \mathbf{W}_{\mathrm{o}} = \begin{bmatrix} \hat{\mathbf{o}}_i(\mathbf{q}) \\ \mathbf{r}_i(\mathbf{q}) \times \hat{\mathbf{o}}_i(\mathbf{q}) \end{bmatrix}$$

where $\hat{\mathbf{n}}_i$ is the unit vector normal to the tangent plane at contact i, $\hat{\mathbf{t}}_i$ and $\hat{\mathbf{o}}_i$ are two orthogonal unit vectors both orthogonal to $\hat{\mathbf{n}}_i$ (*i.e.* basis vectors of the tangent plane) at contact i, and $\mathbf{r}_i$ is a vector from the center of gravity of the part to contact i.

Equation (4) is the kinematic map of the system where $\mathbf{G}$ is the matrix mapping the generalized velocity of the body to the time derivative of the position and orientation. $\mathbf{G}(\mathbf{q}) = \begin{bmatrix} \mathbf{I}_{3\times3} & 0_{3\times3} \\ 0_{4\times3} & \mathbf{J}_{4\times3}(\mathbf{q}) \end{bmatrix}$ where $\mathbf{I}_{3\times3}$ is the identity matrix of given size and $\mathbf{J}(\mathbf{q}) = \frac{1}{2}\begin{bmatrix} -e_x & -e_y & -e_z \\ e_s & e_z & -e_y \\ -e_z & e_s & e_x \\ e_y & -e_x & e_s \end{bmatrix}$.

Equation (5) is the concatenation of all joint constraints of the system. $\mathbf{W}_{\mathrm{b}}$ denotes the Jacobian matrix of all the joints.

Equation (6) is the nonpenetration constraint for all contacts written as a complementarity condition where

[1] For brevity we have not written the vectors with respect to a frame.

$\boldsymbol{\psi}_{\mathrm{n}}$ is a concatenated vector of all the signed distance functions for each contact i. Note that there is no closed form expression for $\psi_{in}(\mathbf{q}, t)$

Equations (7)–(9) represent Coulomb's friction law, written compactly for all contacts, where $\mathbf{U}$ is a diagonal matrix with i^{th} diagonal element equal to μ_i, the coefficient of friction at contact i, σ_i is a Lagrange multiplier arising from the conversion of the maximum dissipation condition from its "argmax" form into the inequality form given above, $\mathbf{v}_{\mathrm{t}}$ and $\mathbf{v}_{\mathrm{o}}$ are the concatenated vectors of sliding velocities for all contacts, and $\circ$ connotes the Hadamard product.

The orthogonal sliding velocity components v_{it} and v_{io} can be written as:

$$v_{i\mathrm{t}} = \mathbf{W}_{i\mathrm{t}}^T \boldsymbol{\nu} + \frac{\partial \psi_{\mathrm{t}}}{\partial t} \qquad\qquad v_{i\mathrm{o}} = \mathbf{W}_{i\mathrm{o}}^T \boldsymbol{\nu} + \frac{\partial \psi_{\mathrm{o}}}{\partial t} \tag{10}$$

2.2 Mixed LCP Formulation

Usually, we cannot solve DCP problem directly. Instead, we have to rely on the time stepping method that approximates $\dot{\boldsymbol{\nu}} \approx (\boldsymbol{\nu}^{\ell+1} - \boldsymbol{\nu}^{\ell})/h$ and $\dot{\mathbf{q}} \approx (\mathbf{q}^{\ell+1} - \mathbf{q}^{\ell})/h$. We also linearized quadratic friction equations (7)-(9) in order to get the system in the form of mixed LCP. These derivations have been discussed in [16, 17, 2]. We will skip straight to the final mixed LCP form of the system. Readers can refer to the above citations for the details.

$$\begin{aligned}
\mathbf{M}\boldsymbol{\nu}^{\ell+1} &= \mathbf{M}\boldsymbol{\nu}^{\ell} + h(\mathbf{W}_{\mathrm{n}}\boldsymbol{\lambda}_{\mathrm{n}}^{\ell+1} + \mathbf{W}_{\mathrm{b}}\boldsymbol{\lambda}_{\mathrm{b}}^{\ell+1} + \mathbf{W}_{\mathrm{f}}\boldsymbol{\lambda}_{\mathrm{f}}^{\ell+1} + \boldsymbol{\lambda}_{\mathrm{app}}^{\ell} + \boldsymbol{\lambda}_{\mathrm{vp}}^{\ell}) \\
\mathbf{q}^{\ell+1} &= \mathbf{q}^{\ell} + h\mathbf{G}\boldsymbol{\nu}^{\ell+1} \\
0 &= \mathbf{W}_{\mathrm{b}}^{\mathrm{T}}\boldsymbol{\nu}^{\ell+1} + \frac{\Phi^{\ell}}{h} + \frac{\partial \Phi^{\ell}}{\partial t} \\
0 \le \boldsymbol{\lambda}_{\mathrm{n}}^{\ell+1} &\perp \Psi_{\mathrm{n}}(\mathbf{q}^{\ell+1}) \ge 0 \\
0 \le \boldsymbol{\lambda}_{\mathrm{f}}^{\ell+1} &\perp \mathbf{v}_{\mathrm{f}}^{\ell+1} + \mathbf{E}\boldsymbol{\sigma}^{\ell+1} \ge 0 \\
0 \le \boldsymbol{\sigma}^{\ell+1} &\perp \mathbf{U}\boldsymbol{\lambda}_{\mathrm{n}}^{\ell+1} - \mathbf{E}^T\boldsymbol{\lambda}_{\mathrm{f}}^{\ell+1} \ge 0
\end{aligned} \tag{11}$$

This is the formulation current dvc3d formulates and solves every time step. Note that quadratic Coulomb friction now is linearized as the two last equations.

3 dvc3d library overview

This section outlines the main aspects of dvc3d.

3.1 Programming Interface

dvc3d is written in Object Oriented C++ with flexible design. Current dvc3d library uses the same programming interface as the Bullet physics library [5].We chose to use Bullet's interface because it is well designed, easy to use and flexible enough not to restrict the development of our methods. Also, Bullet's users can be benefited from the interface sharing between the two libraries.

dvc3d supports many types of collision shapes through Bullet: box, sphere, cylinder, cone, capsule, compound, convex, height field and concave triangle mesh. It also supports popular joint type: hinge, point-to-point, slider and generic 6DOF. More information on how to use them are available at [5].

3.2 dvc3d implementation notes

Current dvc3d 1.0 formulates and solves the set of equations (11) using PATH library [6] at each time step. These equations are also known as Stewart-Trinkle formulation, which first was described in [16]. This formulation has been proven to converge to the solution of original DCP model in [14, 15]. We were also able to verify simulation results of this formulation with experimental data in [13, 3, 2, 1].

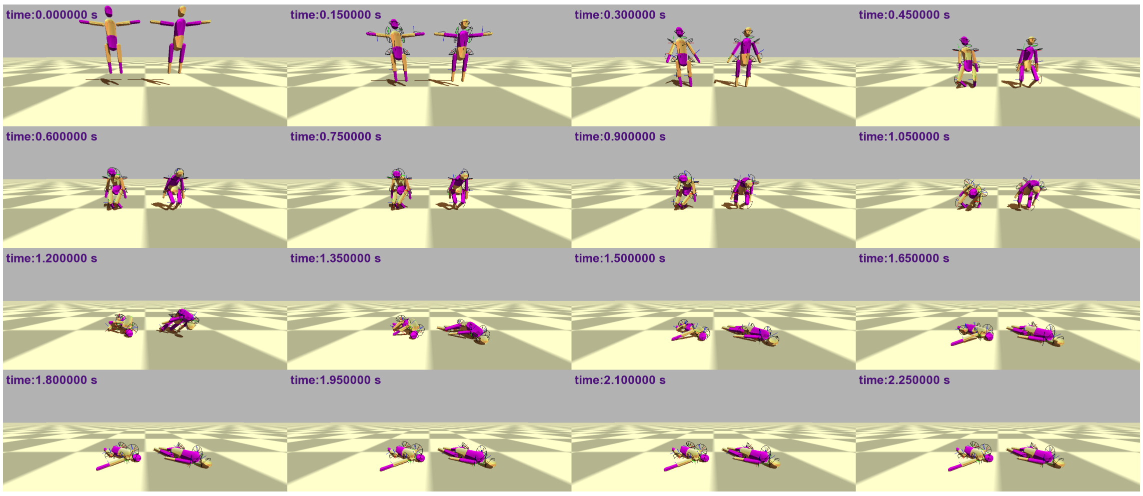


Figure 3: Simulation of two ragdolls in dvc3d version 1.0.

dvc3d is the first and only general 3D physical simulation library that correctly implements the Stewart-Trinkle (ST) method. One of the main challenges in developing the method is that it requires a different set of contacts from popular physics engines. In general, in order to prevent penetration, ST needs the pairs of geometric features that might touch in the next time step, which are usually not reported in current available collision detection libraries. In fact, the ability to prevent instead of just correct penetrations is one of major strengths of ST formulation. We developed a technique to use Bullet's collision detection routines to generate "future" contact sets needed by ST in dvc3d. Basically, the interpolated states of bodies are used in collision detection inputs instead of current ones. The set of contacts returned is also the one could be in collision in the future. This technique is similar to continuous collision detection but without the cost of root findings to find the times of impacts.

4 Locally nonconvex contact model

This section describes a new contact model and formulation that we will make available in dvc3d. For brevity, all illustrated examples in this section are in two dimensions, but the extension to the spatial case is straight forward. Also, because of the space constraint, we cannot provide detailed proofs and analysis. Interested readers could find additional details in our recent paper [10]

4.1 Locally nonconvex effect in simple contact

The common contact model in current methods is the simple one between a point and a face (or edge in planar case). An active constraint associated with this simple contact model keeps the distance ψ_{in} between its point and face from becoming negative. Usually for the ith active contact, the constraint has the form:

$$0 \leq \lambda_{in} \perp \psi_{in}(\mathbf{q}, t) \geq 0 \tag{12}$$

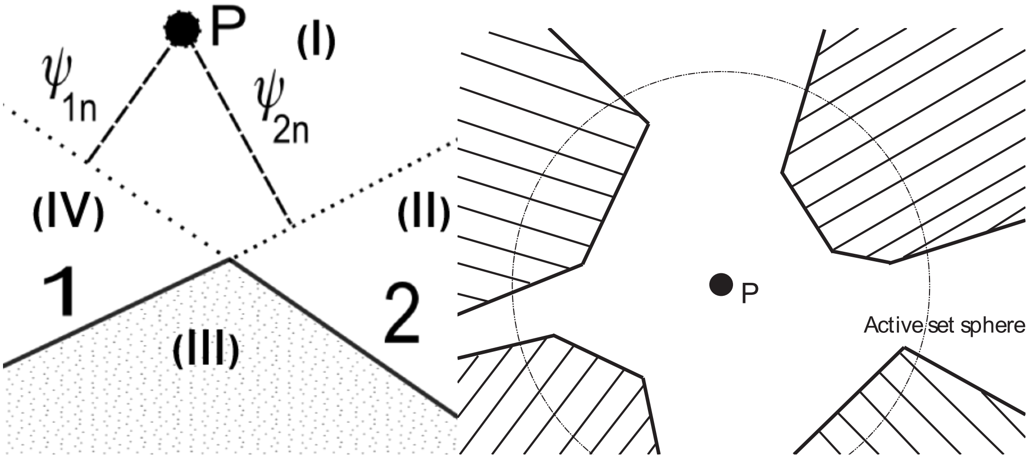
where ψ_{in} is a signed distance function or *gap function* for the ith contact with the property $\psi_{in}(\mathbf{q}, t) > 0$ implies separation, $\psi_{in}(\mathbf{q}, t) = 0$ implies touching, and $\psi_{in}(\mathbf{q}, t) < 0$ for interpenetration. Note that in general, there is no closed-form expression for $\psi_{in}(\mathbf{q}, t)$ so usually approximation values are used instead. λ_{in} is the force or impulse needed to prevent ψ_{in} from becoming negative.

Compactly, we can write the non-penetration constraint for all contacts as

$$0 \leq \boldsymbol{\lambda}_{\mathrm{n}} \perp \boldsymbol{\psi}_{\mathrm{n}}(\mathbf{q}, t) \geq 0 \tag{13}$$

where $\boldsymbol{\psi}_{\mathrm{n}}$ and $\boldsymbol{\lambda}_{\mathrm{n}}$ are the concatenated vectors of all the signed distance functions and normal forces (impulses) respectively. Equation (13) over-constrains the system at those contacts where the local configuration space is non-convex. It can be illustrated in two simple cases below.

In figure 4a, if we write the constraints in the form (13) then the result must satisfy: $\psi_{1\mathrm{n}} \geq 0$ and $\psi_{2\mathrm{n}} \geq 0$ which means in the next time step, the particle P cannot leave the cone (I) while physically, it should be allowed to pass through cone (II) and (IV).



(a) Locally non-convex C-space

(b) A more complicated scene resembling a robot moving through a room

Figure 4: Two cases with locally nonconvex contacts

In the case of figure 4b, formulation (13) is infeasible as there is no point in space that has non negative distances with all lines that support the active edges (the ones that intersect with the circle interior in the figure).

4.2 New locally nonconvex contact model

This new contact model begins with the assertion that we need at most one impulse to prevent penetration between a vertex and a convex shape. Thus, the new contact between two bodies is defined by:

- A vertex in the first body.
- A polygonal convex subset of the second body's geometry.

We have shown that for the case of locally non-convex configuration spaces, treating all non-penetration constraints conjunctively will lead to error. For the case shown in figure 4a, we should not constrain *both* of $\psi_{1\mathrm{n}}$ and $\psi_{2\mathrm{n}}$ but only one of them to be nonnegative or equivalently:

$$\max(\psi_{1\mathrm{n}}, \psi_{2\mathrm{n}}) \geq 0 \tag{14}$$

We leverage the existing Linear Complementarity framework to encode equation (14).

Lemma 4.1. *Given* $a, b \in \mathbb{R}$*,* $b = \max(a, 0) \Longleftrightarrow 0 \leq b - a \perp b \geq 0$

Lemma 4.2. *Given* $a, b \in \mathbb{R}$*,* $b = |\min(a, 0)| \Longleftrightarrow 0 \leq b + a \perp b \geq 0$

Using these two lemmas, we can transform the constraint (14) into linear complementarity conditions as follow:

$$\max(\psi_{1\mathrm{n}}, \psi_{2\mathrm{n}}) = \psi_{2\mathrm{n}} + \max(\psi_{1\mathrm{n}} - \psi_{2\mathrm{n}}, 0) \tag{15}$$

Then define the variable c as follows:

$$c = \max(\psi_{1\mathrm{n}} - \psi_{2\mathrm{n}}, 0) \tag{16}$$

Using lemma 4.1 equation (16) can be written as a linear complementarity condition:

$$0 \leq c - (\psi_{1\mathrm{n}} - \psi_{2\mathrm{n}}) \perp c \geq 0 \tag{17}$$

Then, the correct non-penetration constraint at the contact along edge 1 is:

$$0 \le c + \psi_{\mathrm{2n}} \perp \lambda_{\mathrm{1n}} \ge 0 \tag{18}$$

This constraint basically means that when the term $c + \psi_{\mathrm{2n}}$, which is equivalent to $\max(\psi_{\mathrm{1n}}, \psi_{\mathrm{2n}})$, becomes negative, the normal force (impulse) λ_{1n} along edge 1 will be positive to prevent the penetration.

Similarly, non-penetration constraint along edge 2 is:

$$0 \le c + \psi_{\mathrm{2n}} \perp \lambda_{\mathrm{2n}} \ge 0 \tag{19}$$

Each of equations (18) and (19) correctly models the obstacle geometry. Having both in a formulation could be redundant. The only different between the two constraints is the direction of the normal force when there is penetration. If no penetration happens in the next step, the effect of having any or both in the formulation would be the same.

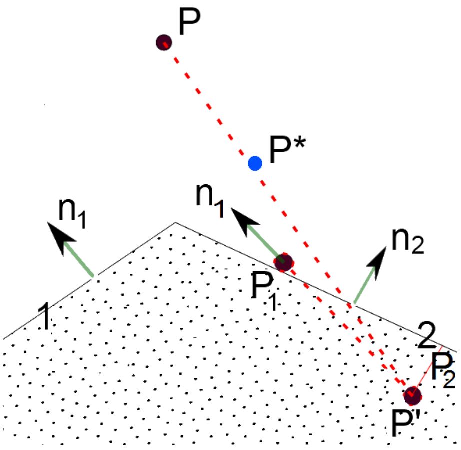


Figure 5: Different outcomes of equations (18) and (19) when penetration occurs in the next time step.

Figure 5 shows the results of using locally nonconvex constraint along different sides of the simple obstacle. Initially, the vertex in the contact is at position P, next time step, it would be at P'(in penetration) if there is no active contact. In the case we activate locally nonconvex (P,[1,2]) along edge 1(normal n_1), then the vertex will end up at P_1. If we activate the one along edge 2(normal n_2), it will be at P_2 (arguably the best result). Note that both P_1 and P_2 are better than the result of ST, where the vertex would be at location P^*.

In general, the contact has the form $(P, [1, 2, ...n])$. The extended formulation will become:

$$\begin{aligned}
0 \le & \quad c_2 - \psi_{\mathrm{2n}} + \psi_{\mathrm{1n}} \perp \ c_2 \ge 0 \\
0 \le & \quad c_3 - \psi_{\mathrm{3n}} + c_2 + \psi_{\mathrm{1n}} \perp \ c_3 \ge 0 \\
& \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
0 \le & c_n - \psi_{nn} + c_{n-1} + \cdots + c_2 + \psi_{\mathrm{1n}} \perp \ c_n \ge 0 \\
0 \le & \quad c_n + c_{n-1} + \cdots + c_2 + \psi_{\mathrm{1n}} \perp \lambda_{un} \ge 0
\end{aligned} \tag{20}$$

The last equation is equivalent to:

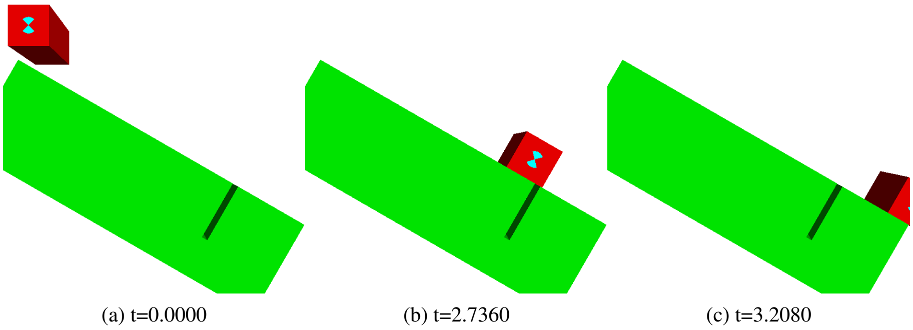
$$0 \le \max(\psi_{\mathrm{1n}}, \psi_{\mathrm{2n}}, \cdots, \psi_{nn}) \perp \lambda_{un} \ge 0 \tag{21}$$

The physical intuition here is that when the maximum predicted distance of the contact vertex to contact faces progresses to negative value, a normal force along the chosen face u will kick in to prevent the penetration.

This LCP formulation can be solved using the same ST solver PATH library[6]. The number of LCP conditions required is equal the number of edges, which is also equal ST method. Thus, this new contact model formulation is more accurate and almost as fast as ST.

4.3 Simulation results

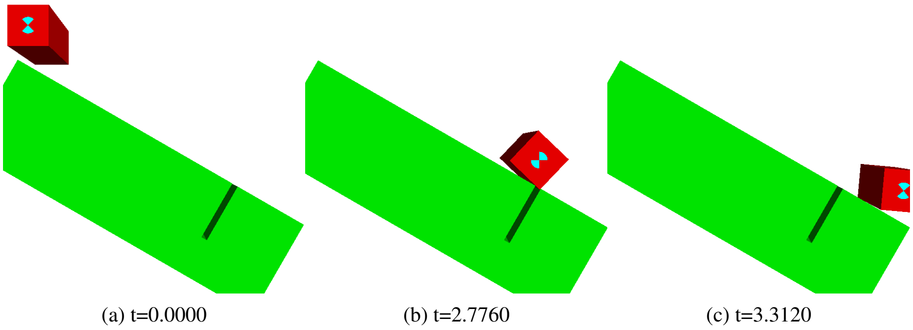
We simulate a box falling then sliding down a cracked ramp. Note that the crack is small so physically, the box should be able to slide to the end of the ramp.


(a) t=0.0000

(b) t=2.7360

(c) t=3.2080

Figure 6: New method, time step $h = 0.008$


(a) t=0.0000

(b) t=2.7760

(c) t=3.3120

Figure 7: Old method, time step $h = 0.008$.

With the old contact model, the interaction with the crack caused the block to tumble. This example is motivated by the fact that Computer-Aided Design (CAD) software sometimes produces flawed models.

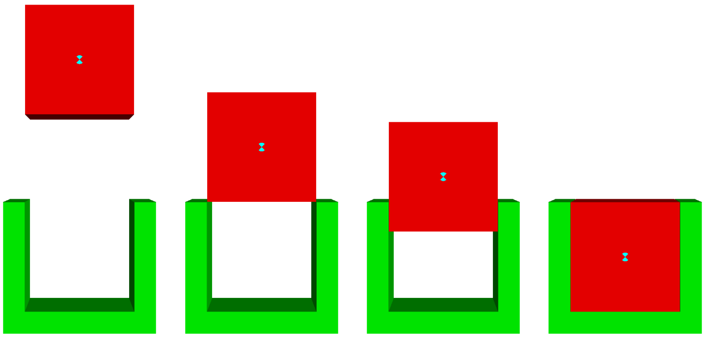
In figure 8 and 9, we simulate a box falls under gravity into a hole with a minimal clearance below (box width = 9.99, hole width = 10) using new and old contact model respectively. Physically, the box should be able to fall through until making contact with the hole's floor.

5 Future works

In the future, we will fully integrate the new contact model formulation in dvc3d along with current ST method.

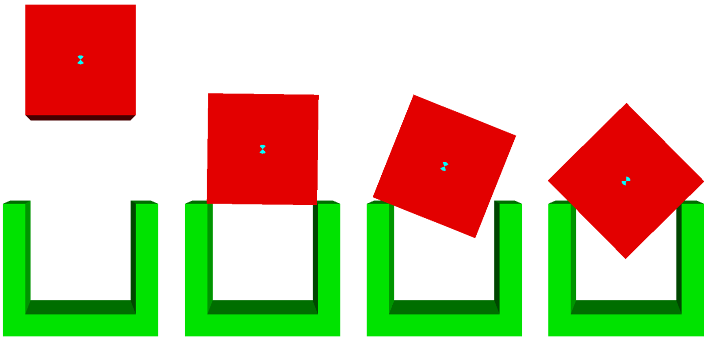
One of our major focus is to make dvc3d's solver faster and parallelable. We are developing a new class of iterative methods that have linear expected running time instead of PATH [6] $O(n^3)$ average. The new methods will also rely on convex quadratic programming (QP) which is easier to be solved in parallel and has many more available solvers than LCP. In essence, these methods convert the original Mixed LCP problem (11) into two smaller subproblems. First subproblem assumes friction forces are known then try to solve for normal forces. It can be reformulated as a convex QP.

$$\begin{aligned} \underset{\boldsymbol{\nu}_1^{\ell+1}}{\text{minimize}} \quad & \frac{1}{2}{\boldsymbol{\nu}_1^{\ell+1}}^T \mathbf{M}\boldsymbol{\nu}_1^{\ell+1} + \mathbf{c}^T\boldsymbol{\nu}_1^{\ell+1} \\ \text{subject to} \quad & \mathbf{W}^T\boldsymbol{\nu}_1^{\ell+1} \le \mathbf{b}. \end{aligned}$$



(a) t=0.0000 (b) t=1.2720 (c) t=1.4720 (d) t=2.0000

Figure 8: New method, time step = 0.0125.



(a) t=0.0000 (b) t=1.2800 (c) t=1.4000 (d) t=1.6640

Figure 9: Old method, time step = 0.0125

The same applies to subproblem two which assumes normal forces are known then solve for friction forces which also end up to be a convex QP. Then the algorithm to approximate solution for equation (11) is:

```
$\boldsymbol{\lambda}_{\mathrm{t}}^0 = 0; \boldsymbol{\lambda}_{\mathrm{o}}^0 = 0; \boldsymbol{\nu}_1^{\ell+1} = \epsilon + 1; \boldsymbol{\nu}_2^{\ell+1} = 0;$
$i = 1;$
while $|\boldsymbol{\nu}_1^{\ell+1} - \boldsymbol{\nu}_2^{\ell+1}| > \epsilon$ do
    $\boldsymbol{\nu}_1^{\ell+1}, \boldsymbol{\lambda}_{\mathrm{n}}^i \leftarrow$ Solve subproblem1($\boldsymbol{\lambda}_{\mathrm{t}}^{i-1}, \boldsymbol{\lambda}_{\mathrm{o}}^{i-1}$);
    $\boldsymbol{\nu}_2^{\ell+1}, \boldsymbol{\lambda}_{\mathrm{t}}^i, \boldsymbol{\lambda}_{\mathrm{o}}^i \leftarrow$ Solve subproblem2($\boldsymbol{\lambda}_{\mathrm{n}}^i$);
    $i \leftarrow i + 1;$
end
```

The approach discussed here was inspired from the ideas proposed in the paper [8].

6 Conclusion

In this paper, we presented the theoretical foundation, some implementation details of dvc3d version 1.0, an accurate physical simulation library. The methods used in the library were proven in theories and tested in experiments to match the expected results inferred from underlying physical models. We also briefly introduced a new, more accurate contact model that can correctly represent contacts locally nonconvex regions of configuration spaces. We are in the process of implementing this model in dvc3d. Then, our efforts to develop a class of fast and parallelable solvers for dvc3d were mentioned. These solvers would enable dvc3d to be used in much larger scenes with a high number of rigid bodies. Readers can download dvc3d version 1.0 at [9].

7 Acknowledgements

This work was supported by the National Science Foundation under grant CCF-0729161 and the Vietnam Education Foundation (www.vef.gov).

REFERENCES

[1] BERARD, S., NGUYEN, B., ANDERSON, K., AND TRINKLE, J. Sources of error in a rigid body simulation of rigid parts on a vibrating rigid plate. In *ASME Journal of Computational and Nonlinear Dynamcis* (2010). to appear.

[2] BERARD, S., NGUYEN, B., AND TRINKLE, J. Sources of error in a rigid body simulation of rigid parts on a vibrating rigid plate. In *ACM Symposium on Applied Computing* (March 2009).

[3] BERARD, S., TRINKLE, J., NGUYEN, B., ROGHANI, B., KUMAR, V., AND FINK, J. daVinci code: A multi-model simulation and analysis tool for multi-body systems. In *IEEE International Conference on Robotics and Automation* (April 2007), pp. 2588–2593.

[4] COTTLE, R. W., PANG, J., AND STONE, R. E. *The Linear Complementarity Problem*. Academic Press, 1992.

[5] COUMANS, E. Bullet physics library. http://bulletphysics.org, 2006.

[6] FERRIS, M. C., AND MUNSON, T. S. Path mcp solver. http://www.cs.wisc.edu/cpnet/cpnetsoftware, 1996.

[7] MICHEL, O., AND INC, C. Webots. http://www.cyberbotics.com, 2004.

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

[9] NGUYEN, B., AND TRINKLE, J. dvc3d: an accurate physics engine. http://www.cs.rpi.edu/~nguyeb2/doku.php?id=dvc3d, 2010.

[10] NGUYEN, B., AND TRINKLE, J. Modeling non-convex configuration space using linear complementarity problems. In *IEEE International Conference on Robotics and Automation* (May 2010). to appear.

[11] PANG, J.-S., AND FACCHINEI, F. *Finite-Dimensional Variational Inequalities and complementarity Problems (I)*. Springer Verlag, New York, 2003.

[12] SMITH, R. Open dynamics engine. http://www.ode.org, 2001.

[13] SONG, P., TRINKLE, J., KUMAR, V., AND PANG, J.-S. Design of part feeding and assembly processes with dynamics. In *Proceedings, IEEE International Conference on Robots and Automation* (April 2004).

[14] STEWART, D. Convergence of a time-stepping scheme for rigid body dynamics and resolution of painlevé's problems. In *Archive Rational Mechanics and Analysis* (1998). vol. 145 (issue 3), pp. 215-260.

[15] STEWART, D. Rigid-body dynamics with friction and impact. In *SIAM Review* (2003). vol. 42 (issue 1), pp. 3-39.

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

[17] TRINKLE, J. Formulations of multibody dynamics as complementarity problems. In *Proceedings, ASME International Design Engineering Technical Conferences* (September 2003). VIB-48342 (no page numbers in CD proceedings).