

Rigid Body Adversarial Attacks

Supplementary Material

In this supplemental document, we provide an explicit formulation of the optimization problem solved in each timestep of the forward simulation, an overview of calculating its derivatives with the adjoint method, as well as further details regarding experiments.

A. Explicit Formulation of Forward Simulation Problem

To determine the optimization problem in the forward simulation, let us start with Newton's law: $F = Ma$. The BDF-2 scheme is given by:

$$3x_{n+1} - 4x_n + x_{n-1} = 2hf(t_{n+1}, x_{n+1}). \quad (10)$$

Writing Newton's law with the implicit formulation after time discretization gives us:

$$Ma_{n+1} = F_{n+1}.$$

Applying BDF-2 and substituting the forces with their corresponding differentiable potential energies, this expands to:

$$\frac{1}{2h}M(3v_{n+1} - 4v_n + v_{n-1}) = - \left. \frac{d\Phi}{dq} \right|_{q_{n+1}}$$

Now, use the BDF-2 formula on the velocities to obtain an equation in terms of positions:

$$\frac{1}{4h^2}M(3(3q_{n+1} - 4q_n + q_{n-1}) - 4(3q_n - 4q_{n-1} + q_{n-2}) + (3q_{n-1} - 4q_n - 2 + q_{n-3})) = - \left. \frac{d\Phi}{dq} \right|_{q_{n+1}}.$$

Simplifying, we get

$$9M(q_{n+1} - \frac{1}{9}(24q_n - 22q_{n-1} + 8q_{n-2} - q_{n-3})) = - 4h^2 \left. \frac{d\Phi}{dq} \right|_{q_{n+1}}.$$

Solving the forward step is then a root-finding problem, where we find the q_{n+1} that is a zero of:

$$f(q_{n+1}) = 9M(q_{n+1} - \frac{1}{9}(24q_n - 22q_{n-1} + 8q_{n-2} - q_{n-3})) + 4h^2 \left. \frac{d\Phi}{dq} \right|_{q_{n+1}}.$$

Let us introduce $\hat{q} = \frac{1}{9}(24q_n - 22q_{n-1} + 8q_{n-2} - q_{n-3})$, so that we can write:

$$f = 9M(q_{n+1} - \hat{q}) + 4h^2 \left. \frac{d\Phi}{dq} \right|_{q_{n+1}}.$$

Now, we can construct an energy E such that $f = \frac{dE}{dq_{n+1}}$, so finding the root of f is equivalent to minimizing the energy:

$$E = \frac{9}{2}(q_{n+1} - \hat{q})^T M(q_{n+1} - \hat{q}) + 4h^2\Phi + c,$$

which is in turn equivalent to minimizing:

$$E = \frac{1}{2}(q_{n+1} - \hat{q})^T M(q_{n+1} - \hat{q}) + \frac{4}{9}h^2\Phi. \quad (11)$$

Splitting the potential energy into its constituent components yields the following optimization problem to find the positions at the end of the timestep:

$$q_{t+1} = \underset{q}{\operatorname{argmin}} \frac{1}{2}(q - \hat{q})^T M(q - \hat{q}) + \frac{4}{9}h^2(\Phi_\Psi(q) + \Phi_g(q) + \Phi_c(q)), \quad (12)$$

where M is the FEM consistent mass matrix (see Eq. 12.55 of Rao [78] for tetrahedral definition), Φ_Ψ is the strain energy, Φ_g is the gravitational potential energy, and Φ_c is the contact potential energy.

For Φ_Ψ , we use the stable Neoohookean energy density given in Eq 6.9 of Kim and Eberle [50]:

$$\Phi_\Psi = \int \Psi_{SNH} dV, \quad (13)$$

$$\Psi_{SNH} = \frac{\mu}{2}(I_2 - 3) - \mu(I_3 - 1) + \frac{\lambda}{2}(I_3 - 1)^2,$$

where λ, μ are the first two Lamé parameters (functions of Y, ν). Note that when calculating this energy density, we must scale the stiffnesses by the cube of our material occupancy α .

The gravitational potential energy is defined as:

$$\Phi_g = g^T Mq. \quad (14)$$

For the contact, we use the smoothly clamped barrier energy from Li et al. [57]:

$$\Phi_c = \begin{cases} -(d - \hat{d})^2 \log\left(\frac{d}{\hat{d}}\right) & 0 < d < \hat{d} \\ 0 & d \geq \hat{d} \end{cases}. \quad (15)$$

B. Simulation Derivatives

Since we are using reverse mode auto-differentiation, we are given the gradient of the final cost C with respect to the simulation step output (typically called GRAD_OUTPUT in Torch literature). At each timestep, we solve the minimization problem in Eq. 12, with the implied constraint $\frac{dE}{dq} = 0$. Thus, using the definition of the adjoint method (Eq. 7 from the main document), we find:

$$\begin{aligned} \left. \frac{\partial C}{\partial \hat{q}} \right|_{q_{t+1}} &= (\text{GRAD_OUTPUT}^T H^{-1}) M \\ \left. \frac{\partial C}{\partial \theta_{\{Y, \nu\}}} \right|_{q_{t+1}} &= -\frac{4}{9} h^2 (\text{GRAD_OUTPUT}^T H^{-1}) \frac{\partial^2 E_\Psi}{\partial q \partial \theta_{\{Y, \nu\}}} \\ \left. \frac{\partial C}{\partial \theta_{\{\rho, \alpha\}}} \right|_{q_{t+1}} &= -(\text{GRAD_OUTPUT}^T H^{-1}) \frac{\partial}{\partial \theta_{\{\rho, \alpha\}}} M(q - \hat{q} + \frac{4}{9} h^2 g), \end{aligned} \quad (16)$$

where we already have the per-timestep energy Hessian H ($\frac{\partial^2 E}{\partial q_{t+1}^2}$) from the forward simulation, and we can analytically compute the Jacobians.

C. Experiment Details

In this section we provide additional details regarding the setup for the experiments we have conducted. Unless specified otherwise, we use a timestep of 0.01s, an IPC barrier distance of 1e-3 m, gravitational acceleration of -9.8 m/s², a Young’s modulus range from 2.5 GPa to 650 GPa, a Poisson’s ratio range from 0.2 to 0.4, and a mass density range from 0.8 g/cc to 11.3 g/cc. In our examples, we use ADAM optimization parameters of $\beta_1 = 0.7$ and $\beta_2 = 0.95$. We run all simulations to convergence. By choosing a good value for the soft constraint coefficient, we are able to get very tight agreement for the moments of mass between the reference and adversarial objects, shown in Table 1.

C.1. Adversarial Ball

The adversarial ball scenario is constructed to mimic a standard basketball setup. The ball is constructed from a tetrahedral mesh of a sphere with radius 0.121m and is comprised of 1820 vertices and 9056 tetrahedra. The backboard assembly is constructed out of a plane and a rim, following NBA dimensions. The backboard assembly is placed at position [0, 0, 3.048] m, and is positionally constrained (treated as a collision mesh rather than simulated). The ball is placed at position [3.4, -3.287, 1.8495] m, and is released with velocity [-3.516, 3.516, 6.099] m/s.

While the ball collides off the planar backboard (xz plane) and has multiple collisions off of the rim, we construct the adversarial ball using just the first impact off the

backboard. To do this, we first simulate the trajectory of the ball from the release point described above, and capture its state just prior to the collision off the backboard. We then use this state as the initial condition for the 0.25s simulations of the ball bouncing off a plane positioned to match the backboard that we use to construct the adversarial object. The optimized adversarial ball is demonstrated from the initial conditions detailed above for a longer 1.5s simulation. In the black box attack demonstration, we observe a very large qualitative difference in the trajectories - in reference case, the ball successfully goes into the hoop and in the adversarial case, it bounces off the rim and out.

C.2. Adversarial Star

In the adversarial star example, we launch a star off of two perpendicular planes (located at $z = 0$ and $y = 0$). The star is of radius 1m, and is comprised of 440 vertices and 1506 tetrahedra. The star has an initial position of [0, 2.5, 2.0] m, and a velocity of [0, -10, -10] m/s. In this example, we disable gravity and use a timestep of $\frac{1}{30}$ s. Due to the thin profile of the star, we choose to optimize only the Young’s modulus, Poisson’s ratio, and mass density (i.e. we keep the star as a solid object, abstaining from any topology optimization). Additionally, we increase the minimum Young’s modulus to allow a range of 5GPa to 650GPa. To construct the adversarial example we simulate 0.3s, but the demonstration of the attack against the simulation uses a longer 1s simulation. In this experiment, we observe a roughly 8 degree angular separation between the reference and adversarial trajectories.

C.3. Adversarial Bunny

In the adversarial bunny example, a bunny is bounced off of the ground towards a bin. Similarly to the adversarial ball, the adversarial bunny is constructed using just the first planar contact. The bunny mesh consists of 699 vertices and 2274 tetrahedra. To construct the adversarial example, the bunny is given an initial position of [0, 1.5, 1.5] m, and an initial velocity of [0, -7.75, -10] m/s. It is simulated for 0.2s during which it bounces off of the xy plane. To demonstrate the attack, the bunny has an initial position of [0, 2.0, 2.0] m, an initial velocity of [0, -7.75, -10] m/s, and the bin is placed at [0, -13.75, 0] m. These simulations are run for 2.5s. Similarly to the adversarial ball example, we have a major qualitative difference in the simulation result, with the reference bunny successfully going into the bin and the adversarial bunny failing to do so.

C.4. Adversarial Cubes

The adversarial cubes example differs from the previous experiments as there are multiple bodies being simulated, there is non-zero friction, and the simulation used to construct the adversarial example differs greatly from the simu-

Table 1. Comparison of reference and adversarial moments of mass for all of our examples. In all cases, the moments closely match.

<i>Object</i>	m_0	$m_{1,x}$	$m_{1,y}$	$m_{1,z}$	$m_{2,xx}$	$m_{2,yy}$	$m_{2,zz}$	$m_{2,xy}$	$m_{2,xz}$	$m_{2,yz}$
Ball (ref)	1.825e+01	-3.215e-05	-4.682e-05	3.930e-05	1.058e-01	1.059e-01	1.055e-01	7.475e-08	-1.050e-06	2.079e-06
Ball (adv)	1.825e+01	-3.215e-05	-4.682e-05	3.930e-05	1.058e-01	1.059e-01	1.055e-01	7.462e-08	-1.050e-06	2.080e-06
Star (ref)	8.811e+02	3.314e-03	-3.201e-03	1.016e-02	2.178e+02	1.155e+02	1.155e+02	-5.076e-04	1.560e-03	-4.151e-03
Star (adv)	8.811e+02	3.311e-03	-3.200e-03	1.016e-02	2.178e+02	1.155e+02	1.155e+02	-5.259e-04	1.565e-03	-4.142e-03
Bunny (ref)	3.901e+03	-2.403e-13	3.411e-13	-4.263e-13	1.382e+03	8.974e+02	1.028e+03	9.705e-01	4.553e+01	2.682e+02
Bunny (adv)	3.901e+03	-1.920e-07	2.939e-08	2.131e-06	1.382e+03	8.974e+02	1.028e+03	9.705e-01	4.553e+01	2.682e+02
Cube (ref)	3.124e-01	9.922e-07	3.564e-07	2.571e-07	1.301e-04	1.301e-04	1.301e-04	1.164e-08	3.443e-08	-1.833e-08
Cube (adv)	3.124e-01	9.922e-07	3.565e-07	2.571e-07	1.301e-04	1.301e-04	1.301e-04	1.164e-08	3.443e-08	-1.833e-08
Bat (ref)	4.132e+00	3.082e-05	-2.229e-05	2.287e+00	1.453e+00	1.453e+00	1.713e-03	2.258e-07	-1.887e-05	2.489e-05
Bat(adv)	4.132e+00	-8.139e-05	-4.273e-05	2.287e+00	1.453e+00	1.453e+00	1.930e-03	3.406e-06	-2.030e-05	2.410e-05

lation used to evaluate it. To construct the adversarial cube, we use a simplified setup where the cube of side length 5cm is placed at $[0, 0, 0.05]$ m, is given an initial velocity of $[0, 0, -0.1]$ m/s. The cube bounces off the xy plane, and again off a plane above it at $z = 0.1$. The cube mesh contains 2167 vertices and 10164 tetrahedra.

The evaluation of this example involves three identical cubes stacked (offset) atop each other, and a fourth cube strikes the stack from above (inspired by Figure 6 of Twigg and James [94]). The three cubes are placed at: $[0, 0, 0.026]$ m, $[0, -0.0225, 0.65]$ m, and $[0, 0, 0.128]$ m. The fourth cube is rotated 45 degrees in the x -axis, and is given an initial position of $[0, -0.0225, 0.65]$ m and an initial velocity of $[0, 0, -3.0]$ m/s. These simulations use a timestep of 1e-3s and a friction coefficient of 0.4. In this experiment, the stack of reference cubes successfully stays upright post collision, and the top cube in the stack of adversarial cubes falls over.

C.5. Adversarial Bat

The adversarial bat example contains two different simulated bodies - the bat which contains the degrees of freedom, and the ball whose trajectory is optimized. This case also differs from the previous examples in that it is a directed attack rather than undirected. The optimization objective is to get the ball as close as possible to the center (i.e. $x = 0$). This gives us an optimization cost term of $\|q_{\text{adv}}(t_{\text{end}})_x\|^2$. The bat consists of 2214 vertices and 9098 tetrahedra. The swing of the bat is encoded by choosing the 14 vertices on the base of the bat to be used as Dirichlet boundary conditions; at each timestep of the simulation, the energy minimization problem has a constraint that pins the boundary condition vertices to their positions corresponding to the swing. We solve this using the standard extension to Newton’s method for feasible start nonlinear optimization problems with equality constraints. In this way, we give the bat a constant angular velocity of 1.25π rad/s, and placed with its base at the origin with an initial orientation of -55 degrees about the z axis. The ball is given an initial position of $[0.75, 1.25, 0.245]$ m, with an initial velocity of $[0, -6, 0]$ m/s. To construct the adversarial bat, we follow

the same procedure as with the adversarial ball example, running the simulation forward to capture the state shortly before contact, and using that state as the initial conditions for the simulation in the optimization loop. For constructing the adversarial example, the simulation duration is 0.25 s, and for demonstrating it is 2.0 s. As with the star, we increase the minimum Young’s modulus to 25 GPa to keep the bat’s motion perceptually rigid. In this experiment, we observe a roughly 5 degree angular difference between the rigid body and adversarial trajectories.

D. Baseline Simulations

Different simulation tools work very differently under the hood, for instance, the use of barrier functions vs linear complementarity problem for contact modelling, choice of integrator, etc. One might wonder to what extent the trajectory difference of the adversarial object is due to the construction of the object itself as opposed to the differences in the simulation tools used. We investigate these concerns by running baseline simulations in our deformable simulator (POLYFEM) with extremely stiff, uniform material objects.

Ultimately, both rigid body and deformable simulators are meant to model real world phenomena. Thus, by taking measures such as choosing appropriate simulation parameters and using highly stiff materials, one would expect that the simulated results match across simulators. For the baseline material parameters, we use a Young’s modulus of 1e13 Pa, a Poisson’s ratio of 0.28, and a mass density of 2.5 g/cc. We choose to run the baseline simulation using the same timesteps and contact parameters used in the general experiments.

The results of these baseline simulations are show in figures 10, 11, 12, 13, and 14. We see that the baseline simulations match the rigid body simulations reasonably well. In the three experiments that have a qualitative “success / failure” outcomes (adversarial ball, bunny, cubes), the baseline simulations match the rigid body simulations. For the star and bat examples, the baseline simulation trajectories is about two degrees off from the rigid body reference. While this difference is certainly nontrivial, we note that in both

cases the baseline simulation is closer to the rigid body reference than it is to the adversarial examples.

We believe the main sources of the inaccuracies between the simulators are our large timesteps, the internal mesh contact sampling, and the use of single value coefficient of restitution in the rigid body simulators (standard in rigid body simulation). It may be possible to get much better agreement by decreasing the timestep, exploring different levels of mesh refinement, and using more advanced restitution models such as Wang et al. [96].

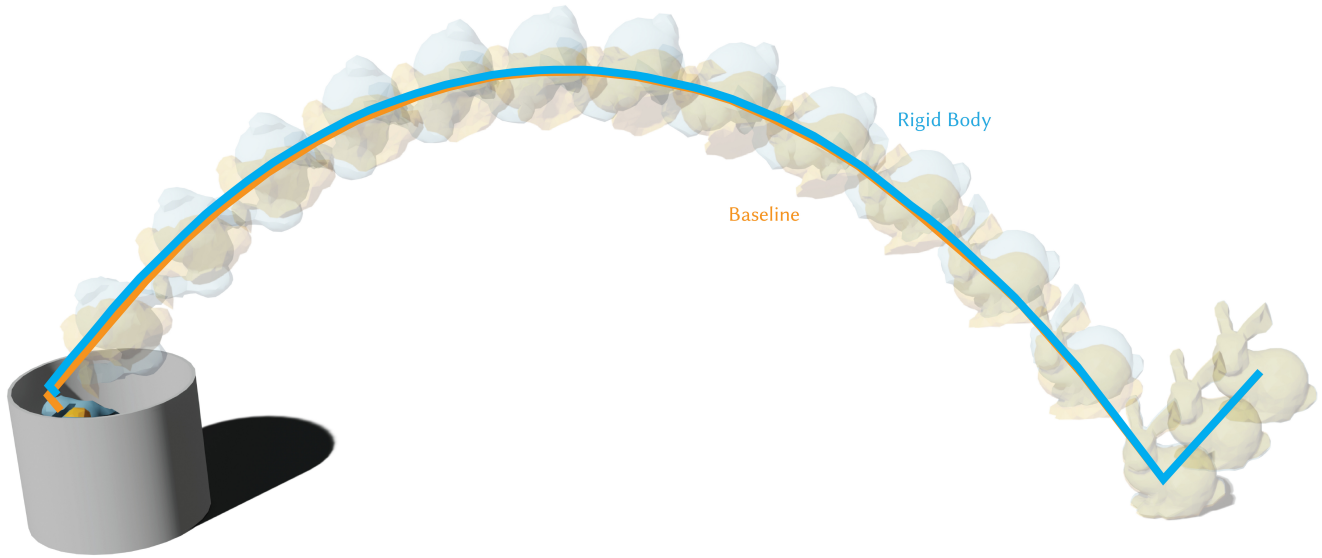


Figure 10. Following the setup from Fig. 7, the baseline simulation (gold) matches the rigid body simulation's (blue) defining characteristic of successfully going into the bin. Note that the trajectories are in very close agreement.

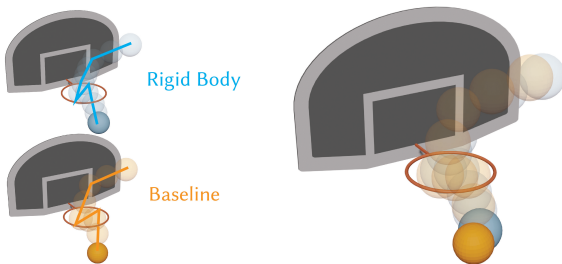


Figure 11. Following the setup from Fig. 1, the baseline simulation (gold) matches the rigid body simulation's (blue) defining characteristic of successfully going into the hoop.

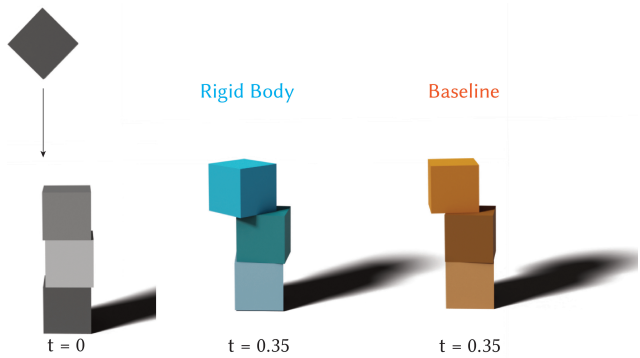


Figure 13. Here, we simulate a collision between a block and a stack of blocks using a stiff deformable simulation (gold). The post collision state looks similar to that of the corresponding rigid body simulation (blue), with the stacks left intact after the collision.

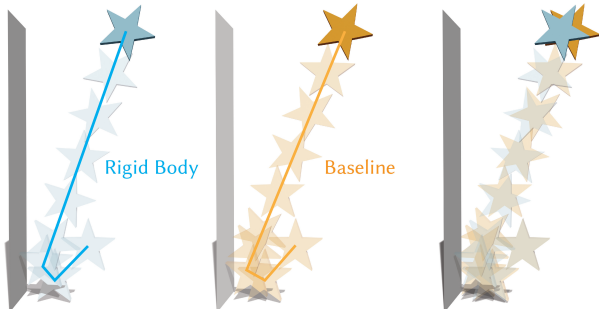


Figure 12. The trajectory of the baseline simulation (gold) corresponding to the star example from Fig 3 is close to that of the rigid body simulation (blue), with a roughly 2 degree angular difference.

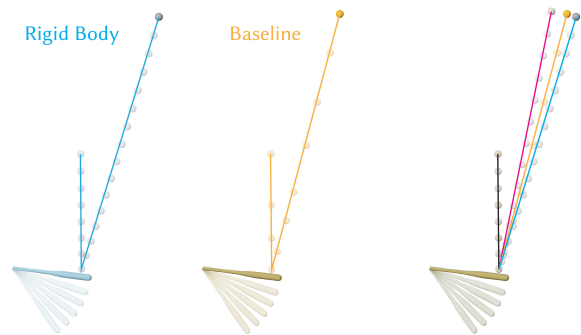


Figure 14. The baseline simulation (gold) of the bat example from Fig. 9 shows poorer agreement than the other examples. We see a roughly 2 degree angular difference between the rigid body (blue) and baseline trajectories.