Benchmarking Rigid Body Contact Models

Michelle Guo MGUO95@CS.STANFORD.EDU
Stanford University, California, United States

Yifeng Jiang YIFENGJ@STANFORD.EDU
Stanford University, California, United States

Andrew Everett Spielberg AESPIELBERG@SEAS.HARVARD.EDU
Harvard University, Massachusetts, United States

Jiajun Wu JIAJUNWU@CS.STANFORD.EDU
Stanford University, California, United States

Karen Liu KARENLIU@CS.STANFORD.EDU
Stanford University, California, United States

Editors: N. Matni, M. Morari, G. J. Pappas

Abstract

As robots are increasingly deployed in contact-rich tasks, there has been increased interest in models of contact that are more accurate than those of untuned simulations. These methods typically rely on simulators that have been system-identified, full dynamical models that are learned, or a combination of both approaches. These methods have typically targeted scenes with well-behaved physical parameters and a single body; however, wider ranges of phenomena are important for many real-world settings and serve as stress-tests that probe the strengths and weaknesses of these methods. In this study, we present a large synthesized dataset with diverse scenes, including objects with varying materials and geometries, or multiple objects involved in inter-body collisions. We use this dataset, to compare and contrast recent approaches in a systematic and unified way. Our empirical evaluations show that while some analytical methods work well in some settings and learned (and hybrid) methods work well in others, no existing method excels in all situations, and all tend to struggle as geometric complexity and the number of scene bodies increase. Our findings call for the collection of more diverse real-world contact datasets for better evaluation of future models. Supplemental videos can be viewed at the project website.[1]

Keywords: Benchmarking, Dataset, Physics Simulation, Dynamics Learning, Contact Modeling

1. Introduction

The physical phenomenon of rigid-body collision and contact is extremely common, yet extremely complex. Accurate predictive modeling of contact dynamics is not only a scientific pursuit itself, but is also of practical importance for improving model-based control techniques in settings in which intelligent robots must efficiently interact with their environment. Since conventional models used in simulations (e.g., Coulomb friction) fail to fully capture the complexity of contact phenomena, research in recent years has focused on data-driven methods for improving their accuracy. The wide spectrum of novel methods developed range from system identification with analytical physics models (Acosta et al., 2022) to fully learning-based methods of varying model architectures (Pizzuto

[1] Project Website: https://sites.google.com/view/rigid-contact/

and Mistry, 2021; Allen et al., 2022), as well as hybrid methods integrating learned and analytical components (Jiang et al., 2022; Pfrommer et al., 2021).

While these research efforts have expanded from simpler dynamical settings such as planar pushing to full object tumbling and sliding in 3D, most methods rely on contact data consisting of a single rigid body colliding with flat ground. Further, that single body is often studied across very few geometric and material variations. How well do state-of-the-art contact modeling methods perform in more diverse scenarios, beyond those previously reported?

To answer this question, we need a diverse array of benchmarking tasks, with different numbers of objects potentially in contact, and with varying geometric complexity and material properties. With the long-term goal of building diverse and accurate benchmarks with contact data from real-world observations, in this paper, we focus on first building a set of more diverse benchmarks in analytical simulators. Simulation provides a first-step at exploring modeling challenges and opportunities without the variability or cumbersome expense of real-world data collection. In total, we generate 12,000 synthetic trajectories of colliding objects with 17 unique objects.

Based on these new tasks, we evaluate several representative data-driven contact modeling methods (Acosta et al., 2022; Allen et al., 2022; Jiang et al., 2022) to study how each varying input system parameter (material, geometry, and number of bodies) impacts the performance of each method. Importantly, for system identification and hybrid methods involving analytical solvers, we ensure the simulators used in modeling and testing are different from those used to collect the synthetic data, so that all methods, regardless of the amount of learning they employ, fairly assume no prior knowledge of the ground-truth contact dynamics.

Our empirical evaluations show that the accuracy of the evaluated methods generally drops with less damped materials, more complex geometry, and more bodies in contact. The results call for more comprehensive and diverse real-world contact datasets to better evaluate future contact models. All benchmarks and implementations will be open-sourced to facilitate future research in the contact modeling community. In summary, we contribute:

- A benchmark dataset containing multiple objects per scene, with varying geometry, material, and number of collisions.
- Empirical evaluation of multiple existing rigid contact models on the benchmark dataset.
- Analysis of the strengths and weaknesses of each method in each scenario.

This paper is structured as follows. Section 2 provides a brief review of related work on rigid body contact models. Sections 3 and 4 describe our experimental setup, including the dataset generation procedure and model implementation details. Experimental results are reported in Section 5.

2. Related Work

Rigid-body contact is challenging to model due to its non-smooth nature (Parmar et al., 2021; Bianchini et al., 2022). Within each discrete mode, contact forces are stochastic with large variability. Below, we review related work on modeling contact phenomena, with a focus on methods that utilize observed contact data, collected for specific objects of interest, to fit (i.e., learn) better models.

Analytical Models. Pure analytical contact models use physics equations and solve for contact forces that satisfy contact constraints (e.g., non-penetration, Coulomb friction cone constraints) without fitting models or parameters on collected data. Existing physics simulators employ different strategies to tackle this theoretically NP-hard problem, including penalty-based methods (Freeman et al., 2021), convex relaxations (Todorov, 2014), and linear complementarity problems (LCPs) (Baraff, 1989; Stewart and Trinkle, 1996; Anitescu and Potra, 2002).

System Identification. The accuracy of analytical (or hybrid) models can be improved by fitting the model parameters to match observed data; this process is commonly referred to as system identification (SysID). If the data is distributed similarly to deployment scenarios, fitted models can more accurately predict unseen contact motions (Acosta et al., 2022; Jatavallabhula et al., 2021; Chebotar et al., 2019; Le Lidec et al., 2021). System parameters may have physical meaning (e.g., friction coefficient, coefficient of restitution), or may be parameters in the solvers analytical models. Therefore, the goal of SysID is usually only to improve prediction accuracy, rather than to uncover the true underlying physical parameters.

Learning-Based Models. Another class of methods completely forgo analytical physics models and allow for more freedom in fitting complex contact behavior through the use of machine learning. Gaussian processes and fully-connected neural networks have been explored (Fazeli et al., 2017; Pizzuto and Mistry, 2021). Contrary to previous evidence (Parmar et al., 2021; Bianchini et al., 2022), graph neural network (GNN) frameworks were recently demonstrated Allen et al. (2022) as being able to emulated highly discontinuous rigid-body contact.

Hybrid Models. Researchers have investigated hybrid methods that *combine* analytical physics simulators with machine learning (Ajay et al., 2018; Fazeli et al., 2020; Heiden et al., 2021). For example, ContactNets (Pfrommer et al., 2021) proposed learning contact Jacobians to be incorporated into an analytical simulator and further introduced a real-world dataset featuring single cube tosses. Jiang et al. (2022) demonstrated an analytical routine for the computing the contact normal force that uses body-level friction wrenches output from a learned neural network.

In this paper, we evaluate three representative methods for data-driven contact modeling, one from each of the above categories: system identification, learning-based models, and hybrid models. We extend the evaluation tasks to include diverse scenarios involving multiple objects with varying gemeotry and material properties.

3. Benchmark Datasets

We present datasets of rigid bodies in motion involving collisions. Informally, one can think of these as varying "tosses" in the spirit of Pfrommer et al. (2021). These tosses feature objects at random initial positions and velocities, resulting in diverse trajectories and observed contact dynamics. For our analysis, these objects are varied in a controlled manner over shape, material, and number of objects in simultaneous contact.

Our benchmark datasets are summarized in Table 1. Each dataset contains scene variations (i.e., *scene variant*) where a single physical parameter is modified while other parameters remain constant. In total, our dataset consists of seventeen objects comprised of seven unique geometries and eleven materials. Each scene consists of one or more objects and a ground plane. For each scene variant in each dataset, we randomly draw 275 tosses for training, 165 for validation, and 110 for testing. Each toss is approximately 0.95 seconds of motion, consisting of 140 image frames.

We use the Bullet simulator (Coumans, 2015) to simulate our object tossing scenes. The simulator parameters are reported in Table 2a. All objects have a mass of 0.37 kg, and are scaled such that the axis-aligned bounding box measures 10.48 centimeters wide. The inertia of each object is determined with uniform density assumption.

Initial state. The initial state is uniformly sampled from the distributions detailed in Table 2b. For multi-body scenes, to avoid spawning overlapping objects, we adopt the routine from Greff et al. (2022), which spawns objects iteratively. For each additional object, we sample a rotation, compute the axis-aligned 3D bounding box of the object in its orientation, and sample the position from the spawn region minus the bounding box. Sampling is repeated until the current object does not overlap with any existing objects in the scene, for a maximum of 100 trials. We bias the velocity such that the bodies move towards the center of the scene by subtracting a constant vector in line with the object position with respect to center from the sampled velocity vector.

4. Evaluated Contact Models

To cover a wider spectrum of data-driven contact modeling, we select one method from each category: system identification, learned, and hybrid. For each method, we follow the traditional train-test evaluation framework. For a single scene variant, a contact model is fit on the training set. The best model is selected according to its performance on a validation set. Then, the best contact model is applied to a held-out test set, and evaluation metrics are reported.

4.1. Evaluation Metrics

We report metrics following Pfrommer et al. (2021), but extended for multiple bodies. For each object k in each frame with ground truth pose (translation and rotation matrix) $(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{R}}_k)$ and predicted pose $(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{R}}_k)$, we compute:

1. *positional error*, defined as $\frac{1}{w_k}\|\bar{\boldsymbol{x}}_k - \hat{\boldsymbol{x}}_k\|_2$, where w_k is the width of the k^{th} object,
2. *rotational error*, defined as $|\text{angle}(\bar{\boldsymbol{R}}_k, \hat{\boldsymbol{R}}_k)|$, where $\text{angle}(\boldsymbol{R}_1, \boldsymbol{R}_2)$ is the angle of rotation between $\boldsymbol{R}_1$ and $\boldsymbol{R}_2$, and
3. *floor penetration*, defined as $\max_j \left(\max\left(0, -z_{j,k}\right)\right)$ where $z_{j,k}$ is the z coordinate of the j^{th} mesh node of the k^{th} object.

For each trajectory in the test set, we compute frame-level metrics on each object. Metrics are averaged over trajectories and timesteps (and over objects for the MultiCubes dataset).

Table 1: **Datasets.** Variants of the object tossing datasets. For each scene variant in one dataset, we vary a single physical or scene parameter while keeping all others fixed. Varied parameters include friction coefficient, coefficient of restitution, geometry, and number of bodies.

Dataset	**Friction**	**Restitution**	**Geometry**	**No. of Bodies**
FrictionCubes	0, 0.25, 0.5, 0.75, 1.0	0.0	cube	1
BouncyCubes	0.0	0, 0.25, 0.5, 0.75, 1.0	cube	1
Polyhedra	0.18	0.125	4, 6, 8, 12, 14, 20, 26-face polyhedra	1
MultiCubes	0.18	0.125	cube	1, 2, 3, 4, 5

Table 2: **Implementation details and simulator settings.** Parameters to simulate ground truth, initial state distribution, and optimization settings for system identification.

(a) Bullet parameters used to simulate the ground truth

Parameter	Setting
Timestep	0.00676
gravity	(0, 0, -9.8)
restitution velocity threshold	0.0
floor friction	1.0
floor restitution	1.0

(b) Initial state distribution of the benchmark datasets

State Component	Bounds
position (low)	(-0.4, -0.4, 0.13)
position (high)	(0.4, 0.4, 0.15)
rotation	SO(3)
velocity (low)	(-1.2, -1.2, 0)
velocity (high)	(1.2, 1.2, 0)
angular velocity (low)	(0, 0, 0)
angular velocity (high)	(0, 0, 0)

(c) Optimization domains for system identification in MuJoCo

Parameter	Domain
per-object friction	[0, 1]
floor friction	[0, 1]
floor contact stiffness	[1e2, 1e4]
floor contact damping	[0, 1e3]

4.2. System Identification

We use system identification to fit contact parameters of an analytical simulator to maximally reproduce training trajectories. The identified contact parameters $\boldsymbol{\theta}^*$ are the solution to the optimization problem $\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}\in\Theta} L(\boldsymbol{\theta})$, where Θ is the optimization domain of the contact parameters and $L(\boldsymbol{\theta})$ is an objective function evaluating the predicted tosses for the contact parameters $\boldsymbol{\theta}$.

Simulator. We use MuJoCo (Todorov et al., 2012) as our choice of simulator. Unlike Bullet, which uses hard collision constraints, MuJoCo uses a relaxed, convex approximation of rigid contact (Anitescu, 2006; Todorov, 2014). The discrepancy in contact modeling between the ground truth trajectories (simulated with Bullet) and predicted trajectories (simulated with MuJoCo) attempts to replicate the simulation-to-reality gap, making comparisons with purely learned methods more fair. The simulation frequency and gravity vector are set to match those in Table 2a.

Optimization. We use NGOpt (Meunier et al., 2021), a gradient-free optimizer (often used for meta-optimization), to optimize the contact parameters of each object in the scene. We use the open source implementation[2] provided by Nevergrad (Rapin and Teytaud, 2018). The optimization domain and parameters are provided in Table 2c. We use a budget of 10,000 iterations. The identified contact parameters are those that minimize the objective function, which is a frame-level configuration error averaged over timesteps and trajectories in the batch. The frame-level configuration error is extended from Acosta et al. (2022) to multiple objects, defined as,

$$L(\bar{\boldsymbol{q}}, \hat{\boldsymbol{q}}) = \sum_{k=1}^{K} \left(\frac{2}{w_k} \|\bar{\boldsymbol{x}}_k - \hat{\boldsymbol{x}}_k\|_2^2 + \text{angle}(\bar{\boldsymbol{R}}_k, \hat{\boldsymbol{R}}_k)^2 \right), \tag{1}$$

where K is the number of objects in the scene. The positional loss weighting per-object normalizes the losses across objects, relative to their physical size. Since gradient-free optimizers tend to provide few guarantees for local optimality, and in some cases can diverge, for each scene, we select the model across ten runs that performed the best on the validation set.

4.3. Graph Neural Network

In order to evaluate pure learning-based methods, we use GNNs as our model representation, inspired by recent performant research (Allen et al., 2022). GNNs use the mesh vertices of each object

[2] https://github.com/facebookresearch/nevergrad

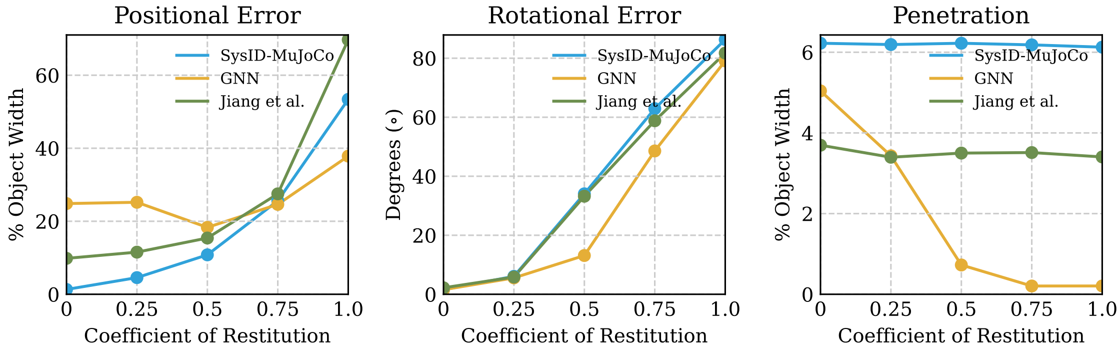


Figure 1: **Effect of restitution.** Average positional error, rotational error, and ground penetration for cubes with different coefficients of restitution from the BOUNCYCUBES dataset.

as the input graph nodes, and object mesh edges as graph edges. Input node features include velocities and distances to boundaries. Edge features consist of relative displacement between nodes, and their magnitudes. The training objective is the mean squared error over vertex-level accelerations.

We use an open source implementation[3] by Sanchez-Gonzalez et al. (2020). We apply random walk noise with scale 3×10^{-4} and five previous velocities of history as input to the model. For each scene variant, a single GNN model is trained on the training set. Each model was trained with a batch size of 64 and a learning rate of 10^{-4} for up to one million iterations. The model iteration with the lowest rollout error on the validation set is selected as the final model, to be evaluated on the test set. Since the model directly outputs vertex positions (which have no guarantee of matching the resting mesh shape of the simulated object), we follow Allen et al. (2022) and project the output vertices onto the resting mesh using shape matching Müller et al. (2005). This is only applied as a post-processing step, after an entire rollout is generated by the model.

4.4. Hybrid Contact Model

We evaluate a representative hybrid contact model similar to Jiang et al. (2022), where the body-level friction wrench is inferred with a neural network, followed by a frictionless LCP solve which provides normal contact forces. Given observed data, we use inverse contact dynamics to compute the body-level contact wrench per transition. The friction wrench is extracted from the contact wrench and is used as supervision to train a regressor for predicting the friction wrench. For the regressor model, we use a three-layer MLP with 512, 360, and 180 hidden nodes at each layer, respectively. The input features and loss function follow Jiang et al. (2022). We train the model up to 1,000 epochs with a batch size of 512 and learning rate of 5×10^{-4} until the validation loss fails to improve for twelve epochs. The model with the best validation loss is selected as the final model. We implemented this method in Python with Nimble Physics (Werling et al., 2021). As this method assumes a single rigid body contacting a plane, it is only evaluated in single rigid body experiments.

5. Evaluation Results

Section 5.1 evaluates the aforementioned baseline methods on each of our benchmarking datasets and scene variants. In Section 5.2, we conduct ablation studies to investigate in further detail.

[3] https://github.com/deepmind/deepmind-research/tree/master/learning_to_simulate

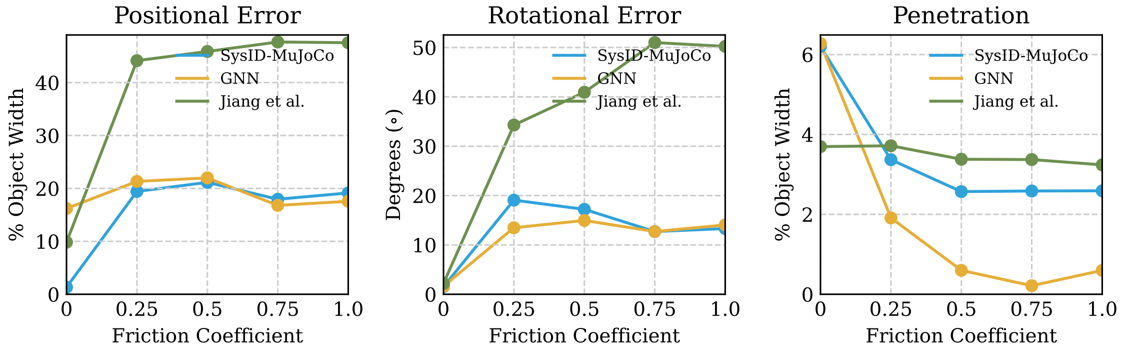


Figure 2: **Effect of friction.** Average positional error, rotational error, and ground penetration for cubes with different friction coefficients from the FRICTIONCUBES dataset.

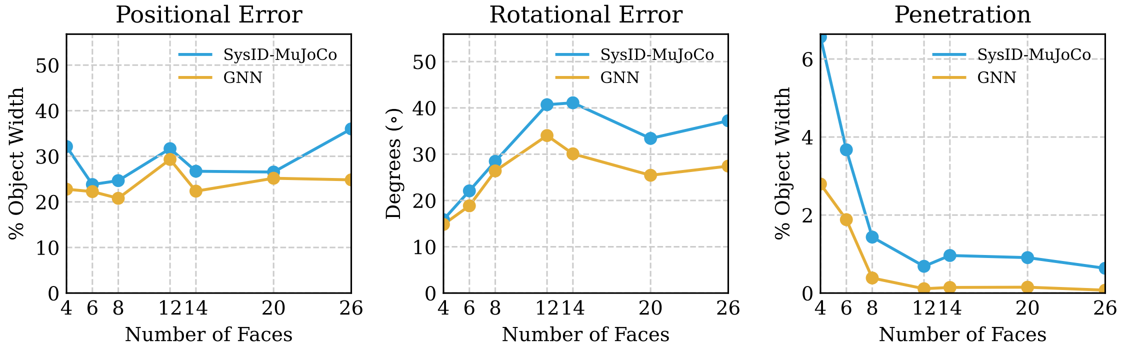


Figure 3: **Effect of geometry.** Average positional error, rotational error, and ground penetration for objects with different numbers of faces from the POLYHEDRA dataset.

5.1. Benchmark

Restitution. In Figure 1, we show performance as a function of restitution on the BOUNCYCUBES dataset. The positional error steadily increases for the models with analytical components, while staying fairly constant for GNN. The rotational error increases quickly for all methods. We hypothesize that elastic collision is more difficult to model, likely due to more energy in the system, suggesting that less damped materials are important for future methods to evaluate on.

Friction. We analyze the effect of friction on model performance in Figure 2 with the FRICTIONCUBES dataset. In general, system identification (SysID-MuJoCo) and GNN outperform Hybrid (Jiang et al., 2022). In the extreme case of friction $\mu = 0$, all methods exhibit lower errors compared to higher μ, implying that $\mu = 0$ is the simplest scene variant for all methods. For realistic values of $\mu > 0$, we do not observe a trend in performance across different friction coefficients.

Geometry. Figure 3 reports results on the POLYHEDRA dataset. Renderings of final-frame predicted poses and ground truth can be found in Figure 4. Our results indicate that the number of geometric faces does not affect positional error, while rotational error worsens as the number of faces increases, but plateaus as the polyhedra asymptotically approach the geometry of a sphere. With lower discrete curvature at each vertex and more compact vertex/face sets, there is more dense information in z for both analytical and learning-based methods to use, decreasing penetration. The

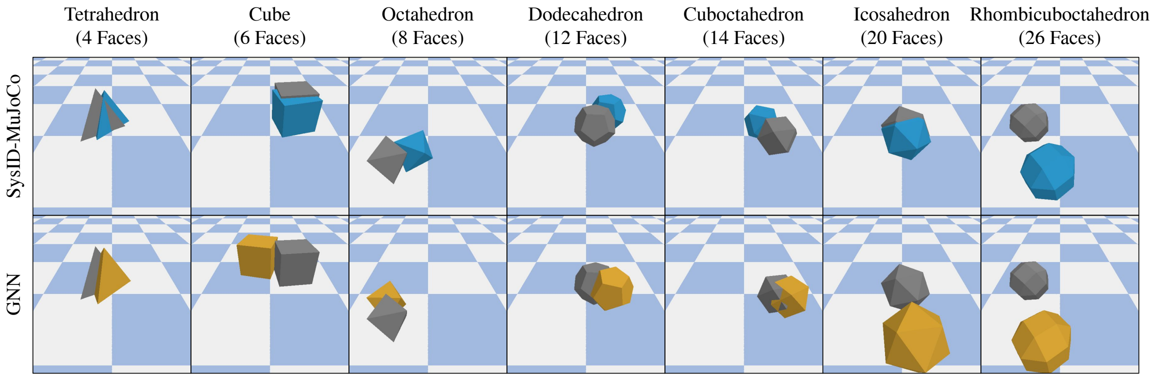


Figure 4: **Rendering of different number of faces.** Ground truth versus predicted final-frame poses on the POLYHEDRA dataset. Gray denotes ground truth. Blue denotes the predicted pose from SysID-MuJoCo. Gold denotes the predicted pose from GNN.

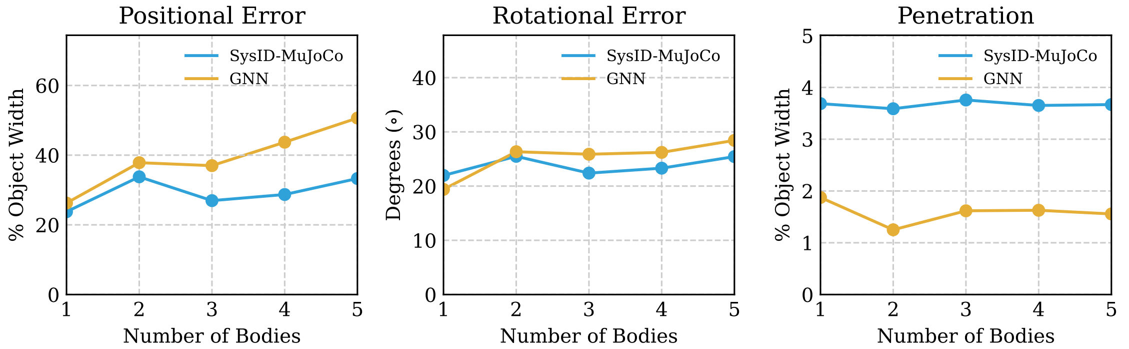


Figure 5: **Effect of number of bodies.** Average positional error, rotational error, and ground penetration for different numbers of bodies from the MULTICUBES dataset.

result is that the polyhedra quickly approach a rolling motion as vertex count increases, as opposed to a tumbling motion that is prone to penetrate.

Multi-body. We evaluate multi-body contact for different numbers of bodies K with the MULTICUBES dataset. Renderings of final-frame pose predictions and ground truth can be found in Figure 6. Figure 5 shows a weak linear increase in positional and rotational errors as the number of bodies increases; as expected, ground penetration for each model is agnostic to the number of bodies. As can be seen later in 5.2, this error is correlated with inter-body penetration, which is a major source of the error. These results, showcase the importance to evaluate how well each method scales with increasing number of colliding bodies.

5.2. Analysis

In this section we probe the results of our experiments in order to understand why certain methods outperform others under specific conditions. Understanding when and why a particular method succeeds is key to developing future contact modeling algorithms.

Inter-body penetration. In Figure 7a, we report the average inter-penetration distance. We find that GNN has significantly more inter-penetration between freely moving bodies compared to floor

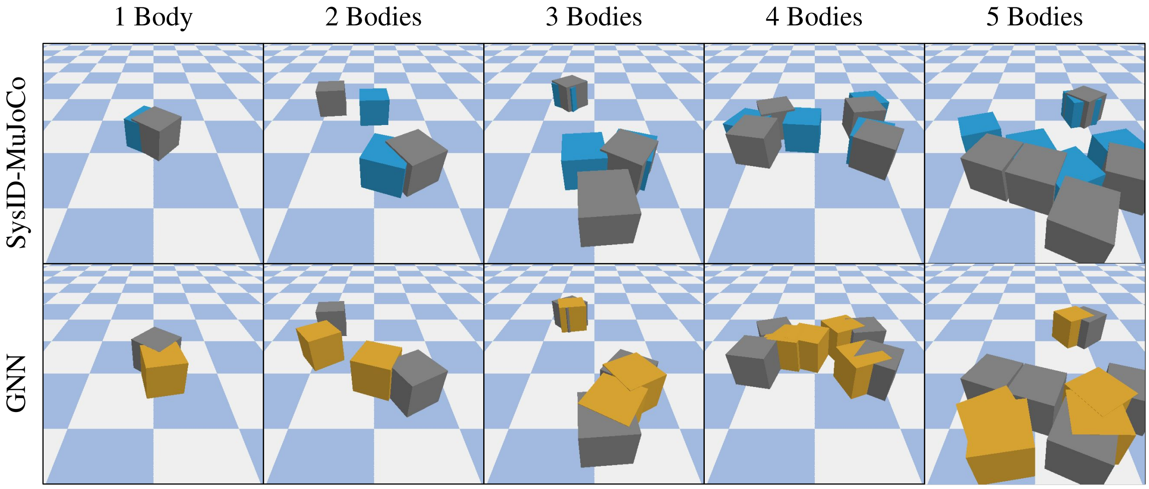
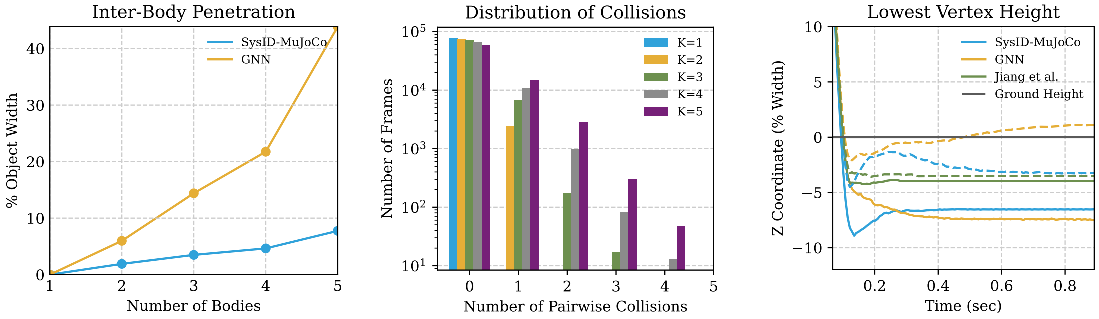


Figure 6: **Rendering of different number of bodies.** We render the ground truth (gray) and predicted (blue for SysID-MuJoCo and gold for GNN) final-frame poses for different baseline methods across different numbers of bodies from the MULTICUBES dataset.

penetration. This is likely due to the sparsity of the contact events: while ground-contact is consistent, occurring at similar spatial coordinates and directional velocities, inter-body collisions can happen at any location and in any direction. This provides less signal to supervise on. This is supported by the fact that error increases with the number of bodies, where extreme collision events are sparse. By contrast, while analytical methods have error, it is bounded (sometimes with guarantees) since rigid body simulators are typically tuned so as to ensure penetrations are not catastrophic. For learning-based methods, data augmentation schemes specialized to contact events will likely improve learned model performance.

Pairwise collisions. While the number of pairwise interactions can theoretically increase quadratically with K, in practice not all possible collisions occur. In our experiments, the maximum number of pairwise collisions for a given K is K rather than $K(K-1)/2$. See Figure 7b for the distribution of pairwise collisions in the MULTICUBES dataset. The fact that contact events are relatively sparse (and are in fact thin-tailed) explains the slow degradation of performance in Figure 5.

Ground penetration. All methods display at least some penetration, with a slight downward trend for SysID-MuJoCo and GNN. We visualize the average penetration over time in Figure 7c. While penetrations stabilize, many analytical solvers converge to nonzero penetration. Solvers that employ penalty methods or use explicit soft contact models are especially prone to ground penetration, but penetration can also persist for solvers that solve an LCP. That penetration occurs because solving LCPs neutralizes velocities of penetrating bodies but does not necessarily correct positional errors accrued. Purely learned models have the flexibility to discover dynamics that minimize contact penetration ($\mu = 0$ in Figure 7c); however, as dynamics become more complex, model capacity may be exhausted on accurately representing other phenomena and penetration may worsen ($\mu = 1$ in Figure 7c). Augmenting analytical models with strong contact priors with learned models that explicitly penalize contact penetration may provide for the best of both worlds.



(a) Inter-body penetration (b) Pairwise collisions (c) Lowest vertex height

Figure 7: **Analysis and ablations.** (a) Maximum inter-body penetration by GNN, per-frame, averaged over the test set. (b) For each number of bodies K, we plot the number of unique pairwise collisions across frames. (c) Lowest vertex height averaged over predictions with $\mu = 0.0$ (solid lines) and $\mu = 1.0$ (dashed lines).

Methods	Friction	Restitution	Number of Faces	Number of Bodies
Analytical (SysID-MuJoCo)	No	Yes	Yes	Yes
GNN (Sanchez-Gonzalez et al., 2020)	No	Yes	Yes	Yes
Hybrid (Jiang et al., 2022)	Yes	Yes	N/A	N/A

Table 3: Importance of physical attributes for evaluating rigid body contact models. We report "Yes" or "No" if the dimension (column) is important for assessing the method (row).

6. Discussion and Conclusion

In this work, we presented a diverse benchmark containing multiple collision datasets as a first step to systematically evaluate different data-driven contact modeling approaches. From our evaluation results on recent methods, we found that while learning-based methods have their advantages in some scenarios, they are not yet one-size-fit-all solutions, and nevertheless still suffer from accuracy drops when geometry, material, or the scene becomes more complex (Table 3). Part of the reason for such performance drop, is exactly the fact that they cannot satisfy hard physics constraints (e.g., non-interpenetration) precisely, and such constraint violation tends to accumulate when the task becomes more complex and challenging to model.

We view this work as a prerequisite for proposing key attributes to consider with designing future datasets in the real world. There are many more attributes we would like include, in a future synthesized or real-world dataset, such as rigid articulation (i.e., multiple links), irregular and non-convex geometry, and non-flat terrains. Method-wise, we would like to evaluate more recent approaches such as other types of hybrid simulators, and differentiable simulations. A study to evaluate the sensitivity of methods with respect to noise and temporal resolution would also be useful. We hope such an open-sourced benchmark with baseline implementations will help improve the code availability of future work, which currently poses a limiting constraint on this study.

Acknowledgments

We thank Keenon Werling for insightful discussions and advice on working with Nimble Physics. Our research was supported by a Meta Research PhD Fellowship, National Science Foundation Graduate Research Fellowship Program, and a DARPA I2O Fellowship.

References

Brian Acosta, William Yang, and Michael Posa. Validating robotics simulators on real-world impacts. In *Robotics and Automation Letters (RA-L)*. IEEE, 2022.

Anurag Ajay, Jiajun Wu, Nima Fazeli, Maria Bauza, Leslie P Kaelbling, Joshua B Tenenbaum, and Alberto Rodriguez. Augmenting physical simulators with stochastic neural networks: Case study of planar pushing and bouncing. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 3066–3073. IEEE, 2018.

Kelsey R Allen, Tatiana Lopez Guevara, Yulia Rubanova, Kim Stachenfeld, Alvaro Sanchez-Gonzalez, Peter Battaglia, and Tobias Pfaff. Graph network simulators can learn discontinuous, rigid contact dynamics. In *Conference on Robot Learning (CoRL)*, 2022.

Mihai Anitescu. Optimization-based simulation of nonsmooth rigid multibody dynamics. *Mathematical Programming*, 105(1):113–143, 2006.

Mihai Anitescu and Florian A Potra. A time-stepping method for stiff multibody dynamics with contact and friction. *International Journal for Numerical Methods in Engineering*, 55(7):753–784, 2002.

David Baraff. Analytical methods for dynamic simulation of non-penetrating rigid bodies. In *Conference on Computer Graphics and Interactive Techniques*, pages 223–232, 1989.

Bibit Bianchini, Mathew Halm, Nikolai Matni, and Michael Posa. Generalization bounded implicit learning of nearly discontinuous functions. In *Learning for Dynamics and Control Conference (L4DC)*, pages 1112–1124. PMLR, 2022.

Yevgen Chebotar, Ankur Handa, Viktor Makoviychuk, Miles Macklin, Jan Issac, Nathan Ratliff, and Dieter Fox. Closing the sim-to-real loop: Adapting simulation randomization with real world experience. In *International Conference on Robotics and Automation (ICRA)*, pages 8973–8979. IEEE, 2019.

Erwin Coumans. Bullet physics simulation. In *ACM SIGGRAPH Courses*, page 1, 2015.

Nima Fazeli, Elliott Donlon, Evan Drumwright, and Alberto Rodriguez. Empirical evaluation of common contact models for planar impact. In *International Conference on Robotics and Automation (ICRA)*, pages 3418–3425. IEEE, 2017.

Nima Fazeli, Anurag Ajay, and Alberto Rodriguez. Long-horizon prediction and uncertainty propagation with residual point contact learners. In *International Conference on Robotics and Automation (ICRA)*, pages 7898–7904. IEEE, 2020.

C. Daniel Freeman, Erik Frey, Anton Raichuk, Sertan Girgin, Igor Mordatch, and Olivier Bachem. Brax - a differentiable physics engine for large scale rigid body simulation, 2021. URL http://github.com/google/brax.

Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, Thomas Kipf, Abhijit Kundu, Dmitry Lagun, Issam Laradji, Hsueh-Ti (Derek) Liu, Henning Meyer, Yishu Miao, Derek Nowrouzezahrai, Cengiz Oztireli, Etienne Pot, Noha Radwan, Daniel Rebain, Sara Sabour, Mehdi S. M. Sajjadi, Matan Sela, Vincent Sitzmann, Austin Stone, Deqing Sun, Suhani Vora, Ziyu Wang, Tianhao Wu, Kwang Moo Yi, Fangcheng Zhong, and Andrea Tagliasacchi. Kubric: a scalable dataset generator. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Eric Heiden, David Millard, Erwin Coumans, Yizhou Sheng, and Gaurav S Sukhatme. Neuralsim: Augmenting differentiable simulators with neural networks. In *International Conference on Robotics and Automation (ICRA)*, pages 9474–9481. IEEE, 2021.

Krishna Murthy Jatavallabhula, Miles Macklin, Florian Golemo, Vikram Voleti, Linda Petrini, Martin Weiss, Breandan Considine, Jerome Parent-Levesque, Kevin Xie, Kenny Erleben, Liam Paull, Florian Shkurti, Derek Nowrouzezahrai, and Sanja Fidler. gradsim: Differentiable simulation for system identification and visuomotor control. *International Conference on Learning Representations (ICLR)*, 2021.

Yifeng Jiang, Jiazheng Sun, and C Karen Liu. Data-augmented contact model for rigid body simulation. In *Learning for Dynamics and Control Conference (L4DC)*, pages 378–390. PMLR, 2022.

Quentin Le Lidec, Igor Kalevatykh, Ivan Laptev, Cordelia Schmid, and Justin Carpentier. Differentiable simulation for physical system identification. In *Robotics and Automation Letters (RA-L)*, volume 6(2), pages 3413–3420. IEEE, 2021.

Laurent Meunier, Herilalaina Rakotoarison, Pak Kan Wong, Baptiste Roziere, Jeremy Rapin, Olivier Teytaud, Antoine Moreau, and Carola Doerr. Black-box optimization revisited: Improving algorithm selection wizards through massive benchmarking. *IEEE Transactions on Evolutionary Computation*, 26(3):490–500, 2021.

Matthias Müller, Bruno Heidelberger, Matthias Teschner, and Markus Gross. Meshless deformations based on shape matching. *ACM Transactions on Graphics (TOG)*, 24(3):471–478, 2005.

Mihir Parmar, Mathew Halm, and Michael Posa. Fundamental challenges in deep learning for stiff contact dynamics. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 5181–5188. IEEE, 2021.

Samuel Pfrommer, Mathew Halm, and Michael Posa. Contactnets: Learning discontinuous contact dynamics with smooth, implicit representations. In *Conference on Robot Learning (CoRL)*, pages 2279–2291. PMLR, 2021.

Gabriella Pizzuto and Michael Mistry. Physics-penalised regularisation for learning dynamics models with contact. In *Learning for Dynamics and Control Conference (L4DC)*, pages 611–622. PMLR, 2021.

Jeremy Rapin and Olivier Teytaud. Nevergrad-a gradient-free optimization platform, 2018.

Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter Battaglia. Learning to simulate complex physics with graph networks. In *International Conference on Machine Learning (ICML)*, pages 8459–8468. PMLR, 2020.

David E Stewart and Jeffrey C Trinkle. An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and coulomb friction. *International Journal for Numerical Methods in Engineering*, 39(15):2673–2691, 1996.

Emanuel Todorov. Convex and analytically-invertible dynamics with contacts and constraints: Theory and implementation in mujoco. In *International Conference on Robotics and Automation (ICRA)*, pages 6054–6061. IEEE, 2014.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 5026–5033. IEEE, 2012.

Keenon Werling, Dalton Omens, Jeongseok Lee, Ioannis Exarchos, and C Karen Liu. Fast and feature-complete differentiable physics for articulated rigid bodies with contact. *Robotics: Science and Systems (RSS)*, 2021.