Dexterous Contact-Rich Manipulation via the Contact Trust Region

H.J. Terry Suh*,[1], Tao Pang*,[2], Tong Zhao[2] and Russ Tedrake[1]

***Abstract*—What is a good local description of contact dynamics for contact-rich manipulation, and where can we trust this local description? While many approaches often rely on the Taylor approximation of dynamics with an ellipsoidal trust region, we argue that such approaches are fundamentally inconsistent with the unilateral nature of contact. As a remedy, we present the Contact Trust Region (CTR), which captures the unilateral nature of contact while remaining efficient for computation. With CTR, we first develop a Model-Predictive Control (MPC) algorithm capable of synthesizing local contact-rich plans. Then, we extend this capability to plan globally by stitching together local MPC plans, enabling efficient and dexterous contact-rich manipulation. To verify the performance of our method, we perform comprehensive evaluations, both in high-fidelity simulation and on hardware, on two contact-rich systems: a planar** IiwaBimanual **system and a 3D** AllegroHand **system. On both systems, our method offers a significantly lower-compute alternative to existing RL-based approaches to contact-rich manipulation. In particular, our Allegro in-hand manipulation policy, in the form of a roadmap, takes fewer than 10 minutes to build offline on a standard laptop *using just its CPU*, with online inference taking just a few seconds. Experiment data, video and *the full version of this paper* are available at** ctr.theaiinstitute.com**.**

I. Introduction

ROBOTS today rarely leverage their embodiment to the fullest due to the limitations of our computational algorithms: robot arms only establish contact with the end-effectors and only perform collision-free motion planning, and robot hands often only establish contact with the fingertips instead of leveraging the entire surface of the hand. This stands in stark contrast to humans, as we are able to utilize every part of our body to strategically establish contact with the environment. In order to address this gap, dexterous contact-rich manipulation, where a robot must autonomously decide where to establish contact without restricting possible contacts, remains an important problem for us to solve.

At the heart of many iterative algorithms for manipulation lies the question: what is a good local description of contact mechanics for contact-rich manipulation that (i) faithfully captures local behavior, and (ii) is sufficiently simple for efficient and scalable computation? Classical treatments express how friction-cone–bounded contact forces perturb the object via the contact Jacobian [1]–[3]. This Jacobian view has powered much of the subsequent planning and control literature; for example, standard grasp-synthesis tools rely on local wrench-space arguments [4], [5]. Yet these analyses assume sticking contacts at fixed points—formalized as *contact modes*—and therefore struggle with dexterous tasks that involve rapid mode switching. In order to handle rapid mode switching, contact-implicit trajectory optimization (CITO) encodes contact modes with complementarity constraints that couple normal forces and separations [6]–[9]. Linearizing these constraints inside an iterative optimization loop exposes the familiar contact Jacobians of classical mechanics.

*Equal contributions. [1]Computer Science and Artificial Intelligence Laboratory (CSAIL), Massachusetts Institute of Technology. [2]RAI Institute (formerly Boston Dynamics AI Institute). Correspondence to pangtao@csail.mit.edu.

This work is funded by the Defense Science & Technology Agency Award DST00OECI20300823, Amazon Award PO#2D-15694085, and the Boston Dynamics AI Institute Award Agmd Dtd 8/1/2023.

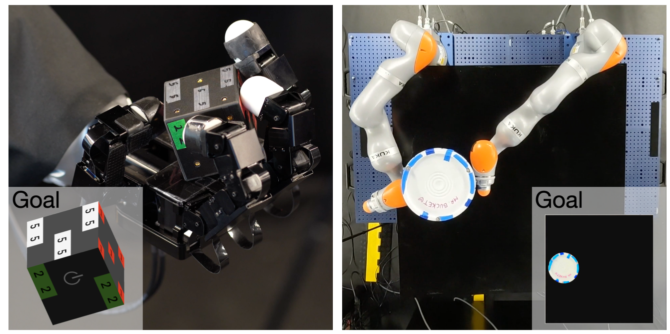

Fig. 1: Hardware experiments illustrating the utility of our proposed method in contact-rich manipulation. Left: Dexterous in-hand manipulation with the Allegro hand moving a cube. Right: Whole-body manipulation with bimanual iiwas moving a bucket.

At first glance this explicit, complementarity-based treatment might seem unrelated to the growing family of planners that build first-order Taylor models from differentiable simulators [10]–[16]. In those methods, the linearization process absorbs friction cones, non-penetration, and complementarity, so the planner sees only linear state-space equations. Several recent works have exploited this locally linear view for trajectory optimization and control [12], [17]–[21].

The apparent discrepancy between these two approaches raises the question of how they are related. In fact, analytically deriving the gradients from differentiable simulators shows that the Taylor expansions they produce and the complementarity constraints ubiquitous in contact-implicit trajectory optimization are fundamentally connected [12, Example 5]. However, the form of each approximation hides a deeper question: *where can we trust this local model*? In optimization, this region is known as the *trust region* [22]. Intuitively, the trust region describes where the local model closely approximates the original function, making it reliable for local improvements. As the quality of Taylor approximations typically degrades as we move further from the nominal point, previous works have often employed an *ellipsoidal trust region*

(ETR) [22]–[24].

Our **first contribution** (Section II-A) is to elucidate the inconsistency between the ETR and the *unilateral* nature of contact. We then address this issue by examining the structure of modern differentiable simulators and proposing the *contact trust region* (CTR). By augmenting Taylor expansions from these simulators explicitly with feasibility constraints from contact-implicit trajectory optimization, the CTR correctly characterizes the local region in which the Taylor expansions approximate the true contact dynamics well. Not only does the proposed CTR consist entirely of convex constraints, it can also be readily derived by combining standard components of existing differentiable simulators.

As our **second contribution**, we present a highly efficient, contact-implicit Model Predictive Control (MPC) method [25], [26] in Section II-B. Specialized for contact-rich manipulation, it can be viewed as a natural extension of iterative LQR [27]: we leverage Taylor expansions from differentiable contact dynamics as linear dynamics constraints, and we include the convex CTR constraints to capture local contact dynamics. Convexity of the CTR ensures that each iteration of the proposed MPC remains a convex optimization problem. We show the efficacy of the proposed CTR and MPC method on two representative contact-rich manipulation problems in Figure 1: (i) whole-body manipulation on the bimanual iiwa and cylinder system, and (ii) in-hand reorientation on the Allegro hand and cube system. Through 2000 trajectory runs in simulation and 100 runs on hardware, our results delve into the efficacy and the limitations of our approach, as well as the benefits of CTR over the ellipsoidal one in the context of planning and control.

As our **final contribution** (Section II-C), we address global planning by chaining local trajectories discovered by MPC. To do this, we follow a roadmap [28] approach where we seed each node with a stable object configuration, and then connect the nodes using MPC. Compared to single-query algorithms [12], [29]–[31], most of the computation time for our algorithm happens offline in the roadmap-building stage, enabling fast online inference. Moreover, the offline roadmap construction only takes minutes on a standard laptop, which is orders-of-magnitude less than approaches based on deep RL [32]–[34].

II. Method

In this work, we develop CTR based on the quasidynamic differentiable simulator proposed in [12], called the Convex Quasidynamic Differentiable Contact (CQDC) contact model. The approach could be extended to other contact dynamics models.

Due to the quasi-dynamic assumption, we assume the state consists of configurations, which we denote by $q \in \mathbb{R}^{n_q}$, and omit velocities. These configurations are divided into actuated configurations $q^{\mathrm{a}} \in \mathbb{R}^{n_{q_\mathrm{a}}}$ that belong to the robot, and object configurations $q^{\mathrm{o}} \in \mathbb{R}^{n_{q_\mathrm{o}}}$ which are unactuated. Furthermore, the actions are represented as a position command $u \in \mathbb{R}^{n_{q_\mathrm{a}}}$ to a stiffness-controlled robot. We denote the contact dynamics and its smoothed variant by $q_+ = f(q,u)$ and $q_+ = f_\kappa(q,u)$, respectively, where q_+ is the next state and κ the dynamics smoothing parameter.

The CQDC model is very representative of approaches taken by modern differentiable simulators through contact [12], [13], [35], [36]. It treats simulation of contact as a convex optimization problem whose primal solution becomes the next state, dual solution becomes contact forces, and gives gradients of both primal and dual solutions by performing sensitivity analysis in convex optimization.

A. Contact Trust Region

Definition 1 (Contact Trust Region). We define the Contact Trust Region (CTR) at the nominal configuration $(\bar{q}, \bar{u})$ as the set of all allowable perturbations that do not result in violation of the primal (1d) and dual (1e) feasibility constraints under a linear model,

$$\begin{aligned}
\mathcal{S}_{\mathbf{\Sigma},\kappa}(\bar{q},\bar{u}) \coloneqq \{(\delta q, \delta u) | \delta z^\top \mathbf{\Sigma} \delta z \leq 1, \delta z = (\delta q, \delta u), \quad &\text{(1a)}\\
\hat{q}_+ = \mathbf{A}_\kappa \delta q + \mathbf{B}_\kappa \delta u + f_\kappa(\bar{q},\bar{u}), \quad &\text{(1b)}\\
\hat{\lambda}_{+,i} = \mathbf{C}_{\kappa,i}\delta q + \mathbf{D}_{\kappa,i}\delta u + \lambda_{\kappa,i}(\bar{q},\bar{u}), \quad &\text{(1c)}\\
\hat{\mathbf{J}}_i(\hat{q}_+ - q) + [\phi_i, 0, 0]^\top \in \mathcal{K}_i, \quad &\text{(1d)}\\
\hat{\lambda}_{+,i} \in \mathcal{K}_i^\star \}. \quad &\text{(1e)}
\end{aligned}$$

Here (1a) is the ETR with $\mathbf{\Sigma} \in \mathbb{R}^{n_z \times n_z}$ being the ellipsoid radius. (1b) is the linearized state dynamics, which describes how the next state $\hat{q}_+$ changes due to perturbations to the nominal state δq and action δu. Similarly, (1c) describes how the contact force at contact i, denoted by $\hat{\lambda}_+$, respond to perturbations under a linear model. The gradients $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$ are obtained from the smoothed differentiable CQDC dynamics. Constraint (1e) requires the linearized contact force to stay inside the friction cone $\mathcal{K}_i^\star$. Lastly, constraint (1d) is a generalization of the non-penetration constraint at contact i, where $\mathcal{K}_i$ is the dual cone of $\mathcal{K}_i^\star$, $\hat{\mathbf{J}}_i$ is the contact Jacobian and ϕ_i the signed distance at contact i.

The constraints (1b)-(1e) are convex, yet they remain locally equivalent to the full complementarity conditions commonly used in CITO formulations. As the complementarity constraints violate standard constraint qualification violations and often destabilize nonlinear solvers [23], the convex structure of the CTR sidesteps those pathologies and is markedly easier to optimize.

B. CTR-based MPC

We first recap the standard iterative-LQR (iLQR) trajectory optimization scheme [27]. In iLQR, we start with an initial guess of the nominal input $\bar{u}_{0:T-1}$, then roll it out under the dynamics $q_+ = f(q,u)$ to get the nominal configuration trajectory $\bar{q}_{0:T}$. Utilizing gradients from a differentiable simulator (e.g. (1b)), we obtain a local linearization around this nominal trajectory $(\bar{q}_{0:T}, \bar{u}_{0:T-1})$, under which we search for optimal perturbations $(\delta q_{0:T}, \delta u_{0:T-1})$ by solving a convex optimization program with quadratic cost and linear constraints. After updating the nominal trajectory using the optimal perturbations, we repeat these steps until convergence or reaching iteration limit.

We argue that the state linearization alone (1b) is insufficient for capturing, even locally, the unilateral nature of contact dynamics. Therefore, we augment the linearized state dynamics used in standard iLQR with the full set of CTR constraints (1). As (1) is a convex set, each inner iteration of the modified "iLQR" algorithm remains a convex program, preserving iLQR's efficiency while enforcing non-penetration and friction-cone feasibility. Embedding this CTR-augmented iLQR in a receding-horizon loop naturally yields a contact-rich MPC controller.

C. Global Planning

We present a simple recipe for global search, in which we chain local plans together to efficiently reach *global* goals which are challenging for local MPC. Our method, inspired by the Probabilistic Roadmap (PRM) [37], consists of an offline phase in which the roadmap is constructed, and an online phase where we reach arbitrary goals using the roadmap.

In the offline phase, we build a roadmap in which the vertices are grasping configurations and the edges are local plans connecting those configurations. The edges are generated using the MPC controller in Section II-B.

Online, we can synthesize plans connecting any starting configuration q_0 to any goal object configuration $q^{\mathrm{o}}_{\mathrm{goal}}$. To do this, we first connect q_0 and $q^{\mathrm{o}}_{\mathrm{goal}}$ to their respective nearest vertices in the roadmap, which can be done using the same MPC inSection II-B. Then, the problem reduces to finding the shortest path between two vertices on a graph, which can be solved with standard methods.

III. Local MPC Experiments

In this section, we evaluate the performance of the CTR-based MPC in Section II-B. We are particularly interested in (i) comparing the proposed CTR against the standard ETR, and (ii) finding out if the proposed method successfully reach a diverse set of goals on complex systems.

To answer these questions and demonstrate the scalability of our method, we conduct statistical analysis on two contact-rich robotic systems:

- the planar IiwaBimanual system, which comprises of 3 unactuated DOFs, 6 actuated DOFs and 29 collision geometries. The whole system is constrained to the xy plane, with gravity pointing along the negative z direction. The bucket measures $0.28\mathrm{m}$ in diameter. The task is to rotate the object, a cylindrical bucket, to target $SE(2)$ poses.
- the 3D AllegroHand system, which comprises 6 unactuated DOFs, 16 actuated DOFs and 39 collision geometries. The object, a $6\mathrm{cm}$ cube, is unconstrained. The hand's wrist is welded to the world frame. The task is to reorient the cube to target $SE(3)$ poses.

A. Experiment Setup

1) Goal Generation : When evaluating the proposed MPC statistically, the success rate depends on *which goals* are commanded from *which initial configurations*. For local optimization, selecting these pairs is nontrivial: goals that are too easy are uninformative, while those requiring highly non-local movements are beyond the scope of local stabilization.

For both the IiwaBimanual and AllegroHand systems, we generate about 1000 pairs of initial conditions and goals which are locally reachable yet far enough to be challenging for MPC. Details about the goal generation scheme can be found in the full paper.

2) Evaluation Metrics: We evaluate the performance of MPC by comparing the difference between $q^{\mathrm{o}}_{\mathrm{final}}$, the final object configuration reached by the by MPC, and $q^{\mathrm{o}}_{\mathrm{goal}}$, the goal object configuration.

We split the object configuration q^{o} into a quaternion Q and a position p: $q^{\mathrm{o}} := (Q, p)$, both expressed relative to the world frame. We divide the error in q^{o} into a *translation error* $\|p_{\mathrm{goal}} - p_{\mathrm{final}}\|$, and an *rotation error* $\Delta\theta(Q_{\mathrm{goal}}, Q_{\mathrm{final}})$, where $\Delta\theta(\cdot,\cdot)$ returns the angular difference between two unit quaternions.

3) Implementation Details: The numerical experiments are run on a M2 Max Macbook Pro with 64GB of RAM. We use the same open-sourced implementation of the CQDC dyanmics as in [12]. The convex subproblems in MPC is solved with [38].

B. Results on CQDC Dynamics

In this section, we evaluate the proposed MPC's ability to reach goals generated in Section III-A1, using the metrics in Section III-A2.

As shown in Table I, MPC with CTR achieves the lowest average error and variance on both IiwaBimanual and AllegroHand. The proposed MPC is able to reach the vast majority of generated goals with tight tolerance.

	IiwaBimanual		AllegroHand	
	Trans. [mm]	Rot. [mrad]	Trans. [mm]	Rot. [mrad]
CTR	2.0 (11.5)	2.1 (10.1)	2.2 (5.7)	9.8 (26.9)
ETR	8.6 (23.8)	9.6 (37.9)	4.5 (53.8)	21.4 (136.4)

TABLE I: Mean translation and rotation errors for local MPC experiments on CQDC dynamics. Each cell displays the mean (std).

On IiwaBimanual, CTR does significantly better than ETR, suggesting ETR can be overly relaxed and does not capture the contact dynamics constraints faithfully. In contrast, for AllegroHand, the advantage of CTR over ETR is much more nuanced, suggesting that AllegroHand may activate the feasibility constraints less frequently than IiwaBimanual.

C. Results on Second-Order Dynamics

In Section III-B, we presented results for running MPC on the CQDC dynamics. However, the CQDC dynamics is different from real contact physics: not only are second-order effects ignored, it also introduces a small gap between objects undergoing sliding friction [11, Section IV-A2], which is an artifact known as "hydroplaning" [39, Appendix B] shared by contact dynamics formulations that utilize Anitescu's convex relaxation for contact dynamics [40] (this includes Mujoco).

In this sub-section, we would like to understand how the gap between the CQDC and real physics affect MPC performance

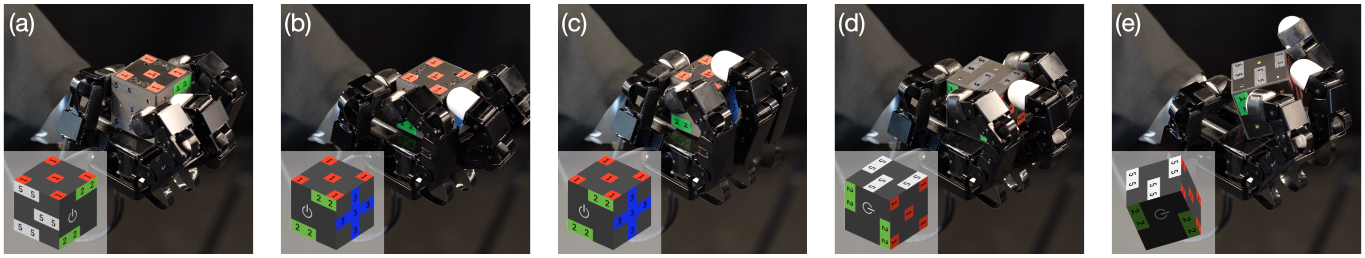

Fig. 2: A complete path generated by our roadmap-based global planner. The object configuration in each frame is highlighted at the lower-left corner. (**a**) shows the starting configuration, which in this example is a vertex already in the roadmap. (**b**) can be reached from (**a**) with a -90° yaw. (**c**) has the same object configuration as (**b**), but the hand has repositioned for a 90° pitch. (**d**) is system configuration after the pitch. (**e**) shows the system reaching the goal configuration from (**d**). The path (**a**)-(**d**) is part of the roadmap and generated offline. (**d**)-(**e**) is generated online using collision-free planning and local MPC.

	IiwaBimanual		AllegroHand	
	Trans. [m]	Rot. [rad]	Trans. [m]	Rot. [rad]
Simulation	0.020 (0.065)	0.039 (0.079)	0.014 (0.009)	0.290 (0.197)
Hardware	0.013 (0.016)	0.024 (0.038)	0.018 (0.011)	0.258 (0.215)

TABLE II: Translation and rotation errors for simulation and hardware experiments. Each cell displays the mean (standard deviation).

in more realistic settings. Specifically, we focus on answering the question: can the MPC controller using the CQDC dynamics perform closed-loop stabilization (i) on high-fidelity simulated second-order dynamics, and (ii) on hardware?

As shown in Table II, the performance on both simulated and hardware experiments deteriorates when compared with CQDC dynamics, despite our best effort to tune hyperparameters and apply reasonable heuristics.

Moreover, the average errors from MPC is much smaller on IiwaBimanual, which we attribute to the Allegro task's inherently difficulty. As reaching goal poses on the Allegro often requires lifting the cube, any slip can cause the cube to slide back to the palm, resetting progress and resulting in large errors. In contrast, on IiwaBimanual, the bucket remains on a tabletop, so slipping merely slows progress without eliminating it.

IV. Global Planning Experiments

To demonstrate the robustness of the roadmap constructed using the method in Section II-C, we perform on the Allegro hand system 150 consecutive edge traversals before the hardware shuts down due to overheating. An example path reaching an arbitrary goal generated using our method is shown in Figure 2. More roadmap planning examples can be found in the supplementary video.

V. Conclusion

Have we solved the problem of planning and control through contact dynamics? Locally—on CQDC dynamics—the proposed MPC with Contact Trust Region constraints can achieve small tracking errors for the vast majority of the goals we sampled based on local reachability criteria. However, we do not understand this problem nearly as well as we do the simple pendulum: not only do we lack solid explanations for the small but non-zero number of planner failures, but we also have yet to fully understand the different roles played by feasibility constraints on IiwaBimanual vs AllegroHand.

Much more remains to be understood for second-order dynamics, both in simulation and on hardware. In particular, keeping the robot in contact with the object, without exploiting the artifact of CQDC dynamics, remains one of the biggest unsolved challenges. Incorporating the CTR constraints in MPC greatly alleviates the problem of lost contact, and the initial guess heuristics empirically helps a bit more. However, we occasionally still observe loss of contact, and its root cause remains unclear.

Nevertheless, the tools presented in this paper already enable capabilities that were previously out of reach for model-based methods. By accounting for the unilateral nature of contact, our contact trust region makes it possible to apply a broad range of robotics algorithms to contact-rich manipulation problems. We hope the MPC, grasp synthesis, and roadmap-based global planning methods introduced here are only a small sample of the many contact-rich manipulation algorithms yet to come.

REFERENCES

[1] J. Craig, *Introduction to robotics : mechanics & control*. Reading, Mass.: Addison-Wesley Pub. Co., 1986.

[2] R. M. Murray, S. S. Sastry, and L. Zexiang, *A Mathematical Introduction to Robotic Manipulation*, 1st ed. USA: CRC Press, Inc., 1994.

[3] M. T. Mason and J. K. Salisbury, *Robot hands and the mechanics of manipulation*. Cambridge, MA, USA: MIT Press, 1985.

[4] D. Prattichizzo and J. C. Trinkle, *Grasping*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 671–700. [Online]. Available: https://doi.org/10.1007/978-3-540-30301-5_29

[5] L. Han, J. Trinkle, and Z. Li, "Grasp analysis as linear matrix inequality problems," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 6, pp. 663–674, 2000.

[6] Y. Ding, A. Pandala, and H.-W. Park, "Real-time model predictive control for versatile dynamic motions in quadrupedal robots," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 8484–8490.

[7] M. Posa, C. Cantu, and R. Tedrake, "A direct method for trajectory optimization of rigid bodies through contact," *The International Journal of Robotics Research*, vol. 33, no. 1, pp. 69–81, 2014.

[8] P. M. Wensing, M. Posa, Y. Hu, A. Escande, N. Mansard, and A. D. Prete, "Optimization-based control for dynamic legged robots," *IEEE Transactions on Robotics*, vol. 40, pp. 43–63, 2024.

[9] Y. Li, H. Han, S. Kang, J. Ma, and H. Yang, "On the surprising robustness of sequential convex optimization for contact-implicit motion planning," *arXiv preprint arXiv:2502.01055*, 2025.

[10] H. J. Suh, M. Simchowitz, K. Zhang, and R. Tedrake, "Do differentiable simulators give better policy gradients?" in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, Eds., vol. 162. PMLR, 17–23 Jul 2022, pp. 20 668–20 696. [Online]. Available: https://proceedings.mlr.press/v162/suh22b.html

[11] T. Pang and R. Tedrake, "A convex quasistatic time-stepping scheme for rigid multibody systems with contact and friction," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 6614–6620.

[12] T. Pang, H. J. T. Suh, L. Yang, and R. Tedrake, "Global planning for contact-rich manipulation via local smoothing of quasi-dynamic contact models," *IEEE Transactions on Robotics*, vol. 39, no. 6, pp. 4691–4711, 2023.

[13] T. A. Howell, S. L. Cleac'h, J. Brüdigam, J. Z. Kolter, M. Schwager, and Z. Manchester, "Dojo: A differentiable physics engine for robotics," 2023. [Online]. Available: https://arxiv.org/abs/2203.00806

[14] C. D. Freeman, E. Frey, A. Raichuk, S. Girgin, I. Mordatch, and O. Bachem, "Brax - a differentiable physics engine for large scale rigid body simulation," 2021. [Online]. Available: http://github.com/google/brax

[15] M. Macklin, "Warp: A high-performance python framework for gpu simulation and graphics," https://github.com/nvidia/warp, March 2022, nVIDIA GPU Technology Conference (GTC).

[16] R. Tedrake and the Drake Development Team, "Drake: Model-based design and verification for robotics," 2019. [Online]. Available: https://drake.mit.edu

[17] Y. Tassa, T. Erez, and E. Todorov, "Synthesis and stabilization of complex behaviors through online trajectory optimization," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 4906–4913.

[18] T. A. Howell, S. L. Cleac'h, S. Singh, P. Florence, Z. Manchester, and V. Sindhwani, "Trajectory optimization with optimization-based dynamics," 2023. [Online]. Available: https://arxiv.org/abs/2109.04928

[19] H. J. T. Suh, T. Pang, and R. Tedrake, "Bundled gradients through contact via randomized smoothing," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4000–4007, 2022.

[20] V. Kurtz and H. Lin, "Contact-implicit trajectory optimization with hydroelastic contact and ilqr," 2022. [Online]. Available: https://arxiv.org/abs/2202.13986

[21] N. J. Kong, C. Li, G. Council, and A. M. Johnson, "Hybrid ilqr model predictive control for contact implicit stabilization on legged robots," *IEEE Transactions on Robotics*, vol. 39, no. 6, pp. 4712–4727, 2023.

[22] D. C. Sorensen, "Newton's method with a model trust region modification," *SIAM Journal on Numerical Analysis*, vol. 19, no. 2, pp. 409–426, 1982. [Online]. Available: https://doi.org/10.1137/0719026

[23] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer, 2006.

[24] J. J. MORE, "Generalizations of the trust region problem," *Optimization Methods and Software*, vol. 2, no. 3-4, pp. 189–209, 1993.

[25] S. Le Cleac'h, T. A. Howell, S. Yang, C.-Y. Lee, J. Zhang, A. Bishop, M. Schwager, and Z. Manchester, "Fast contact-implicit model predictive control," *IEEE Transactions on Robotics*, vol. 40, pp. 1617–1629, 2024.

[26] A. Aydinoglu and M. Posa, "Real-time multi-contact model predictive control via admm," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 3414–3421.

[27] W. Li and E. Todorov, "Iterative linear quadratic regulator design for nonlinear biological movement systems," in *First International Conference on Informatics in Control, Automation and Robotics*, vol. 2. SciTePress, 2004, pp. 222–229.

[28] L. Kavraki, P. Svestka, J.-C. Latombe, and M. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.

[29] X. Cheng, E. Huang, Y. Hou, and M. T. Mason, "Contact mode guided motion planning for quasidynamic dexterous manipulation in 3d," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 2730–2736.

[30] X. Cheng, S. Patil, Z. Temel, O. Kroemer, and M. T. Mason, "Enhancing dexterity in robotic manipulation via hierarchical contact exploration," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 390–397, 2024.

[31] G. Khandate, T. L. Saidi, S. Shang, E. T. Chang, Y. Liu, S. Dennis, J. Adams, and M. Ciocarlie, "R times r: Rapid exploration for reinforcement learning via sampling-based reset distributions and imitation pre-training," *Autonomous Robots*, vol. 48, no. 7, p. 17, Aug 2024. [Online]. Available: https://doi.org/10.1007/s10514-024-10170-8

[32] OpenAI, M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, J. Schneider, S. Sidor, J. Tobin, P. Welinder, L. Weng, and W. Zaremba, "Learning dexterous in-hand manipulation," 2019. [Online]. Available: https://arxiv.org/abs/1808.00177

[33] T. Chen, J. Xu, and P. Agrawal, "A system for general in-hand object re-orientation," in *Proceedings of the 5th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, A. Faust, D. Hsu, and G. Neumann, Eds., vol. 164. PMLR, 08–11 Nov 2022, pp. 297–307. [Online]. Available: https://proceedings.mlr.press/v164/chen22a.html

[34] A. Handa, A. Allshire, V. Makoviychuk, A. Petrenko, R. Singh, J. Liu, D. Makoviichuk, K. Van Wyk, A. Zhurkevich, B. Sundaralingam, Y. Narang, J.-F. Lafleche, D. Fox, and G. State, "Dextreme: Transfer of agile in-hand manipulation from simulation to reality," *arXiv*, 2022.

[35] Y. D. Zhong, J. Han, and G. O. Brikis, "Differentiable physics simulations with contacts: Do they have correct gradients w.r.t. position, velocity and control?" 2022. [Online]. Available: https://arxiv.org/abs/2207.05060

[36] K. Werling, D. Omens, J. Lee, I. Exarchos, and C. K. Liu, "Fast and feature-complete differentiable physics for articulated rigid bodies with contact," *arXiv preprint arXiv:2103.16021*, 2021.

[37] H. Choset, K. M. Lynch, S. Hutchinson, G. A. Kantor, and W. Burgard, *Principles of robot motion: theory, algorithms, and implementations*. MIT press, 2005.

[38] M. ApS, *MOSEK Optimizer API for Python 10.2.1*, 2024. [Online]. Available: https://docs.mosek.com/latest/pythonapi/index.html

[39] R. Tedrake, *Underactuated Robotics*, 2023. [Online]. Available: https://underactuated.csail.mit.edu

[40] M. Anitescu, "Optimization-based simulation of nonsmooth rigid multibody dynamics," *Mathematical Programming*, vol. 105, no. 1, pp. 113–143, Jan 2006. [Online]. Available: https://doi.org/10.1007/s10107-005-0590-7