Cascaded Newton-based Augmented Lagrangian Method for Robotic Multi-Contact Simulation

Jeongmin Lee[1], Minji Lee[2], Sunkyung Park[2], Jinhee Yun[2], and Dongjun Lee[2]

***Abstract*— The multi-contact nonlinear complementarity problem (NCP) is a naturally arising challenge in robotic simulations. Achieving high performance in terms of both accuracy and efficiency remains a significant challenge, particularly in scenarios involving intensive contacts and stiff interactions. In this paper, we introduce a new multi-contact NCP solvers based on the theory of the Augmented Lagrangian (AL). We detail how the standard derivation of AL in convex optimization can be adapted to handle multi-contact NCP through the iteration of surrogate problem solutions and the subsequent update of primal-dual variables. Specifically, we present tailored variation of AL for robotic simulations: the Cascaded Newton-based Augmented Lagrangian (CANAL). We demonstrate how CANAL can manage multi-contact NCP in an accurate and robust manner, through robotic manipulation scenarios with intensive contact.**

I. Introduction

Contact simulation is a fundamental tool for the development of manipulation intelligence, as it enables scalable data acquisition, training, and safe testing of various algorithms and designs. This significance has led to the development of diverse open-source platforms [1]–[6], which are increasingly being utilized in various research endeavors. An essential focus in contact simulation for robotics revolves around achieving results that are both accurate and efficient in terms of memory and computation time. This presents a comprehensive problem, encompassing geometry, contact modeling, and numerical algorithms.

Modeling multi-contact interactions in simulation typically induces a nonlinear complementarity problem (NCP) [7]. In practice, contact solvers must balance three crucial factors: efficiency, accuracy, and robustness. However, finding a universal solution remains challenging. Methods developed for graphics and game engines tend to prioritize efficiency and robustness, aiming to deliver visually plausible results, even if early termination occurs. However, they are known to converge slowly and may struggle with achieving highly accurate solutions. They frequently encounter difficulties in handling intensive contact interactions (i.e., where constraints are dense and numerous relative to the system degrees of freedom), which is common in robotic manipulation. Conversely, achieving a highly accurate solution for NCP often involves complex matrix operations and numerically sensitive processes, which generally lack efficiency and robustness for practical robotic applications. Moreover, some approaches aim to enhance efficiency and robustness by relaxing the contact constraints and exploiting them during the solving stage. However, such relaxations can be challenging to physically interpret, and the solutions they produce may exhibit undesirable physical behaviors.

This research was supported by the National Research Foundation (NRF) grant funded by the Ministry of Trade, Industry & Energy (MOTIE) of Korea (RS-2024-00419641), and the Ministry of Science and ICT (MSIT) of Korea (RS-2022-00144468).

[1]Holiday Robotics, [2]Department of Mechanical Engineering, IAMD and IOER, Seoul National University, Seoul, Republic of Korea. {ljmlgh,mingg8,sunk1136,yjhs0932,djlee}@snu.ac.kr.

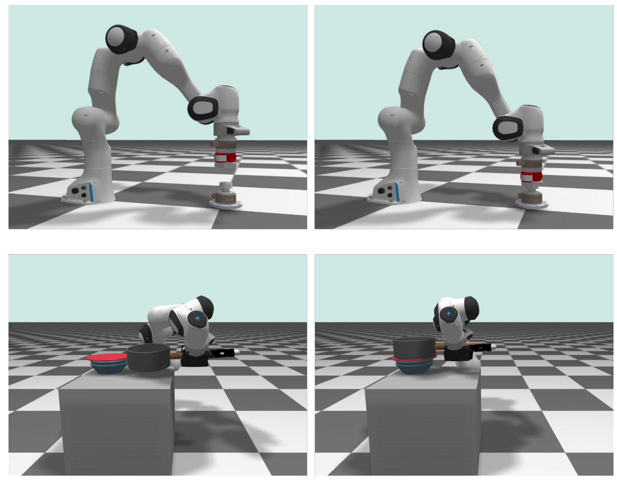

Fig. 1: Snapshots of a robotic simulation using our multi-contact solver. Top: bolt-nut assembly. Bottom: dish piling. Although intensive contact formation and stiff interactions make these scenarios challenging to simulate, our solvers successfully complete the simulations less than a ms of time budget per step.

In this paper, we introduce Cascaded Newton-based Augmented Lagrangian (CANAL), a new multi-contact solver for robotic simulation. We explain how CANAL is advantageous in scenarios requiring precise management of high-density intensive contact, by leveraging a cascaded Newton scheme within the theory of augmented Lagrangian (AL). Several robotic simulations, particularly those involving challenging multi-contact scenarios, are demonstrated to validate our framework.

II. Multi-Contact Simulation via Augmented Lagrangian

A. Problem Formulation

We consider following discrete-time equations of motion:

$$\begin{aligned} \text{Solve } & A\hat{v} = b + J^T\lambda \\ \text{s.t. } & (J\hat{v}, \lambda) \in \mathcal{S}_c \end{aligned} \tag{1}$$

where $A \in \mathbb{R}^{n\times n}, b \in \mathbb{R}^n$ are the dynamics matrix/vector, and $\mathcal{S}_c$ denotes the set of all pairs $(J\hat{v}, \lambda)$ satisfying the contact condition.

Contact condition typically includes nonlinear complementarity relation between primal (i.e., velocity) and dual (i.e., impulse) variables. For each contact point, the corresponding 3-DOF relation is

$$\begin{aligned} &0 \leq \lambda_{i,n} \perp J_{i,n}\hat{v} + e_{i,n} \geq 0 \\ &0 \leq \delta_i \perp \mu_i\lambda_{i,n} - \|\lambda_{i,t}\| \geq 0 \\ &\delta_i\lambda_{i,t} + \mu_i\lambda_{i,n}J_{i,t}\hat{v} = 0 \end{aligned} \tag{2}$$

where $\perp$ denotes complementarity, $e_{i,n} \in \mathbb{R}$ and $J_{i,n} \in \mathbb{R}^{1\times n}$ denote the error and Jacobian for contact normal, $J_{i,t} \in \mathbb{R}^{2\times n}$ is the Jacobian for contact tangential, and μ_i is the friction coefficient and δ_i is the auxiliary variable. The first condition, known as the velocity-level Signorini condition, captures the complementarity nature of the contact occurrence and gap. The remaining conditions involve the complementarity between slipping velocity and the friction cone boundary, with the maximal dissipation law indicating that slip opposes the direction of impulse.

B. Augmented Lagrangian for Multi-Contact NCP

Although the problem (1) shares commonalities with optimization, it diverges due to the introduction of complementarity relations between primal and dual variables. Our aim is to establish a foundation for deriving AL techniques specifically tailored to multi-contact. We start by equivalently expressing (1) as follows:

$$\begin{aligned} \text{Solve } & A\hat{v} = b + J^T\lambda \\ \text{s.t. } & (z,\lambda) \in \mathcal{S}_c,\ J\hat{v} = z \end{aligned} \tag{3}$$

where $z \in \mathbb{R}^{n_c}$ serves as the slack variable for the constraint interface. The expression in (3) bears resemblance to the optimality condition of the following optimization problem:

$$\min_{\hat{v},z} \ \frac{1}{2}\hat{v}^T A\hat{v} - b^T\hat{v} + g(z) \quad \text{subject to} \quad J\hat{v} = z \tag{4}$$

as the matrix A is always symmetric positive definite. In this context, g serves to enforce the constraint in dynamics, although $(z,\lambda) \in \mathcal{S}_c$ is not integrable into the function if the multi-contact condition included. Recalling the structure of augmented Lagrangian [8] applicable to the optimization problem (4), we can similarly solve (3) as follows:

$$\begin{aligned} \text{Solve } & \begin{bmatrix} A + \beta J^T J & -\beta J^T \\ -\beta J & \beta I \end{bmatrix} \begin{bmatrix} \hat{v} \\ z \end{bmatrix} = \begin{bmatrix} b - J^T u \\ u + \lambda \end{bmatrix} \\ \text{s.t. } & (z,\lambda) \in \mathcal{S}_c \end{aligned} \tag{5}$$

$$u \leftarrow u + \beta(J\hat{v} - z). \tag{6}$$

The rationale of the above structure is that, at the fixed-point of the iteration (therefore, $J\hat{v} = z$), the result satisfies both dynamics equation and constraint relation. Similar to the original augmented Lagrangian, the process can be interpreted as iterating between solving the problem relaxed via a penalty term and updating the Lagrange multipliers. We refer this relaxed problem (5) as the *surrogate* problem.

C. Closed-Form Formulation of Slack Variables

Compared to the original problem (1), the surrogate problem (5) should be easier to solve in order to maintain the rationality of the framework. A crucial difference between (1) and (5) is that the constraint condition is defined on the slack variable z as shown below:

$$\beta z = \beta J\hat{v} + u + \lambda, \quad \text{s.t.} \quad (z,\lambda) \in \mathcal{S}_c. \tag{7}$$

This implies that the relationship between z and λ is matrix-free and involves only a simple scalar weight β. Based on this feature, we can derive the *closed-form* representation for λ (therefore, also for z) with respect to $\hat{v}$ by substituting (7) into the contact condition (2) as

$$\lambda_i = \Pi_{\mathcal{C}}^{\text{strict}}\left(-\beta J_i\hat{v} - u_i - \beta e_i\right) \tag{8}$$

where $\Pi_{\mathcal{C}}$ denotes the projection onto the friction cone $\mathcal{C}$. Specifically, the projection $\lambda_i = \Pi_{\mathcal{C}}^{\text{strict}}(\lambda_i^*)$ is carried out by the following steps:

$$\begin{aligned} \lambda_{i,n} &= \max(\lambda_{i,n}^*, 0) \\ \lambda_{i,t} &= \Pi_{\mathcal{C}(\lambda_{i,n})}(\lambda_{i,t}^*). \end{aligned} \tag{9}$$

Here, $\mathcal{C}(\lambda_{i,n})$ represents the cross-section of $\mathcal{C}$ where the plane at height $\lambda_{i,n}$ intersects the cone. This nested projection is distinct from the closest distance projection, commonly known as the proximal operator when applied to the indicator function of the friction cone [9]. As in [10], we refer to (9) as the strict operator as the resulting (z_i, λ_i) strictly satisfies the contact condition (2).

The resulting (8) derived above allows us to write it as:

$$\lambda_i = T(\lambda_i^*) \quad \text{where} \quad \lambda_i^* = -\beta J_i\hat{v} - u_i - \beta e_i \tag{10}$$

where T is a closed-form operator which is continuous yet may nonsmooth depending on the constraint type. Accordingly, by the linear relation (7), the slack variable z is also expressed in closed-form with respect to $\hat{v}$. Based on the closed-form operation (10), solving (5) can be now expressed as solving following nonlinear equation:

$$\begin{aligned} r(\hat{v}) &= A\hat{v} - b - \sum_i J_i^T\lambda_i \\ &= A\hat{v} - b - \sum_i J_i^T T(-\beta J_i\hat{v} - u_i - \beta e_i) \end{aligned} \tag{11}$$

then computing $z = J\hat{v} + \frac{1}{\beta}(u + \lambda)$ accordingly. Due to the projection operator (9), $r : \mathbb{R}^n \to \mathbb{R}^n$ is a continuous, yet semismooth equation. Therefore, one can handle the surrogate problem by solving this nonlinear equation (11) using the Newton method, whose theories developed under semismooth case [11] by employing the generalized derivatives.

However, typical (semismooth) Newton methods are known to exhibit superlinear convergence near the solution but lack robustness. Furthermore, the derivative of the closed-form operator (10) might become non-symmetric in contact cases, and cannot guarantee that $\frac{dr}{d\hat{v}}$ will always be non-singular. This issue makes the computation both expensive and unreliable.

III. Cascaded Newton-Based Augmented Lagrangian

A. Cascaded Structure

A crucial issue of the Newton-based solution of (11) is that the landscape of the merit function $\frac{1}{2}\|r(\hat{v})\|^2$ is non-convex. Our core strategy to address this issue employs a cascaded method that relaxes each surrogate problem into a convex form, facilitating fast and stable solutions, while updating terms at each AL step to compensate for discrepancies between the convex problem and the original NCP. For the convex relation, we utilize the equivalence of $(z, \lambda) \in \mathcal{S}_c$ and (2) with the following condition:

$$\mathcal{C} \ni \lambda_i \perp z_i + \underbrace{\begin{bmatrix} 0 \\ 0 \\ \mu_i\|z_{i,t}\| \end{bmatrix}}_{p_i} \in \mathcal{C}^* \tag{12}$$

where $\mathcal{C}^*$ denotes the dual cone of $\mathcal{C}$. This equivalence can be easily verified, as we refer [12] for details. The reformulated relation in (12) essentially constitutes a cone complementarity condition, if the perturbation term p_i is excluded.

A key idea of our cascaded Newton approach is to substitute the perturbation term p_i by borrowing z_i from the previous AL iteration. In other words, we treat p_i as a constant in every surrogate problem, and temporarily consider the relationship between z_i and λ_i as a cone complementarity condition. Consequently, in the $(l+1)$-th AL iteration, we solve the following nonlinear equation that replaces the strict operator (8) with the proximal operator:

$$\begin{aligned} r(\hat{v}^{l+1}) &= A\hat{v}^{l+1} - b - \sum_i J_i^T \lambda_i^{l+1} \\ \lambda_i^{l+1} &= \Pi_{\mathcal{C}}^{\text{prox}}(\underbrace{-\beta J_i \hat{v}^{l+1} - u_i^l - \beta \tilde{e}_i^l}_{\tilde{\lambda}_i^*}) \end{aligned} \tag{13}$$

where $\tilde{e}_i^l = e_i + p_i^l = e_i + \left[0\ 0\ \mu_i\|z_{i,t}^l\|\right]^T$. Even after this replacement, the nonlinear equation in (13) remains semismooth. However, we can demonstrate that it is integrable, as detailed in the following proposition. Note that to streamline the explanation, we will focus exclusively on the contact constraints below, as the other types (i.e., hard and soft) follow straightforwardly.

Proposition 1: The function $r(\hat{v})$ from (13) is the derivative of the following strongly-convex function:

$$h(\hat{v}) = \frac{1}{2}\hat{v}^T A\hat{v} - b^T\hat{v} + \sum_i \frac{1}{2\beta}\|\lambda_i\|^2 \tag{14}$$

Proof: The derivative of $h(\hat{v})$ can be expressed as:

$$\begin{aligned} \frac{dh(\hat{v})}{d\hat{v}} &= A\hat{v} - b - \sum_i J_i^T \frac{d\lambda_i}{d\lambda_i^*}^T \lambda_i \\ &= A\hat{v} - b - \sum_i J_i^T \lambda_i \end{aligned}$$

The latter equality holds due to the identity $\lambda_i^T(\lambda_i - \lambda_i^*) = 0$ in the proximal operator. The symmetric positive-definite property of A ensures that the quadratic term is strongly convex. Furthermore, since the squared distance to a convex set is convex, $\|\lambda_i\|^2$ is convex with respect to λ_i^*, and thus also for $\hat{v}$. Therefore, $h(\hat{v})$ is a strongly-convex function. ■

This result is closely related to those presented in [13], [14], although the objective function is defined differently based on our AL-based formulation. Given this property, we can apply the exact Newton method to the strongly-convex function (14) by computing the derivative of $r(\hat{v})$ (i.e., the Hessian), which is proven to exhibit global convergence [8].

B. Newton Step

Computing the derivative of $r(\hat{v})$ in (13) with respect to $\hat{v}$ is straightforward, except for the part involving T. As the operator T is a proximal operator on a friction cone, it involves a continuous concatenation of three formulaic forms, yet the function is semismooth at the connection points. Below, we provide derivative of each form which can be obtained from a few algebraic calculation:

$$\frac{d\lambda_i}{d\tilde{\lambda}_i^*} = \begin{cases} 0_{3\times3}, & \text{open} \\ I_{3\times3}, & \text{stick} \\ \frac{1}{\mu^2+1}\begin{bmatrix} \mu_i^2 I_{2\times2} + \frac{\mu_i\tilde{\lambda}_{i,n}^*}{\|\tilde{\lambda}_{i,t}^*\|}P(\bar{\lambda}_{i,t}) & \mu_i\bar{\lambda}_{i,t}^T \\ \mu_i\bar{\lambda}_{i,t} & 1 \end{bmatrix}, & \text{slip} \end{cases} \tag{15}$$

where $\bar{\lambda}_{i,t}^*$ is the normalized vector of $\tilde{\lambda}_{i,t}^*$ and $P(\bar{\lambda}_{i,t}) = I - \bar{\lambda}_{i,t}\bar{\lambda}_{i,t}^T$ is the tangential projection matrix. Then the derivative can be written as

$$\frac{dr(\hat{v})}{d\hat{v}} = A + \sum_i \beta J_i^T \frac{d\lambda_i}{d\tilde{\lambda}_i^*} J_i. \tag{16}$$

Due to the structure given in (15), and consequently the matrix (16), is guaranteed to be symmetric positive definite, therefore always invertible. Followingly, the direction of the Newton step is computed as

$$d(\hat{v}) = -\left(\frac{dr(\hat{v})}{d\hat{v}}\right)^{-1} r(\hat{v}) \tag{17}$$

where the $d(\hat{v})$ denotes the direction of $\hat{v}$ update.

Computation of the step (17) requires the linear solving of (16), therefore assemble and factorization of the matrix is necessary. For better efficiency, we can exploit sparsity pattern of the inertia matrix and the constraint Jacobian during the process.

C. Exact Line-Search

Drawing from well-known convex optimization theory [8], we can guarantee that (17) provides a descent direction. However, we still need to integrate a suitable line-search scheme to ensure global convergence. Here, the line-search problem can be described as following one-dimensional, strictly convex optimization problem:

$$\min_{\alpha>0} f(\hat{v} + \alpha d(\hat{v})). \tag{18}$$

Similar to [14], we can find a globally optimal solution of the problem (18) using the `rtsafe` algorithm, which effectively

Algorithm 1: Multi-Contact Simulation via CANAL

while *simulation* **do**
  initialize $l = 0, \hat{v}^0, z^0, \beta > 0, \kappa > 1, 0 < \eta < 1$
  while *CANAL loop* **do**
    initialize $\hat{v}^{l+1} \leftarrow \hat{v}^l$
    compute $\tilde{e}$ based on z^l
    while *Newton loop* **do**
      compute $r(\hat{v}^{l+1})$ (13)
      if $\|r(\hat{v}^{l+1})\| < \theta_{th}^{N}$ **then**
        break
      end
      compute Newton step $d(\hat{v}^{l+1})$ (17)
      compute α via exact line-search (18)
      $\hat{v}^{l+1} \leftarrow \hat{v}^{l+1} + \alpha d(\hat{v}^{l+1})$
    end
    update z^{l+1} and multiplier u^{l+1} (6)
    if $\|J\hat{v}^{l+1} - z^{l+1}\| < \theta_{th}^{AL}$ **then**
      break
    else
      if $\|J\hat{v}^{l+1} - z^{l+1}\| > \zeta\|J\hat{v}^{l} - z^{l}\|$ **then**
        $\beta \leftarrow \min(\kappa\beta, \beta^{\max})$
      end
    end
    $l \leftarrow l + 1$
  end
  update system state using $\hat{v}^{l+1}$
end

combines the one-dimensional Newton-Raphson method and a bisection scheme. In practice, we find that the Newton step, when combined with the aforementioned exact line-search, performs robustly even with large values of β. The overall CANAL algorithm is summarized in Alg. 1.

IV. Examples

For examples, we consider two scenarios: bolt-nut assembly and dish piling. Both scenarios are characterized by the intensive formation of contacts and stiff interactions due to the complexity of the geometry. Moreover, we model light bowls and plates (0.1 kg) beneath a heavy pot (5 kg), resulting in a challenging mass ratio for the stable simulation.

A. Single Step Test

To precisely evaluate the quantitative performance, we measure the results of running different solvers: projected Gauss-Seidel (PGS [15], [16]) and subsystem-based ADMM (SubADMM [17]) single step at the same state and inputs. For test case generation, we sample various configurations of nut and dishes, then apply random external wrenches to the objects. The performance of the solvers in these cases is depicted in Fig. 2.

As shown, CANAL achieves significantly better accuracy in less time compared to PGS. CANAL achieves the highest accuracy, with residuals under 10^{-8}, and exhibits over linear

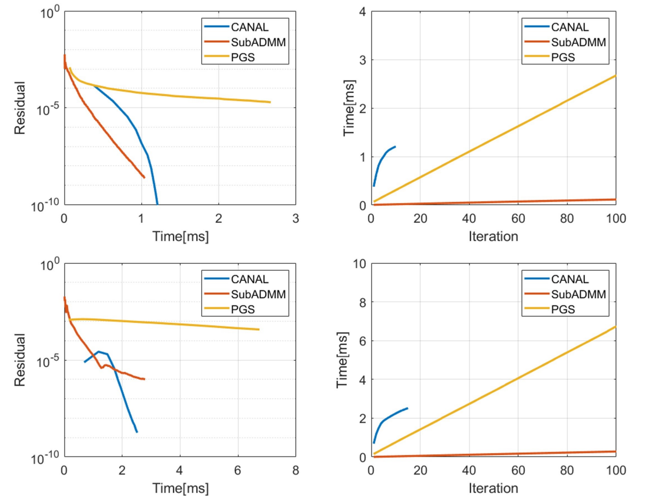

Fig. 2: Comparison of CANAL, SubADMM, and PGS for the bolt-nut assembly (top) and dish piling (bottom) simulation. Left: Residual decrease over computation time. Right: Computation time over iteration.

convergence. SubADMM, demonstrating first-order convergence, struggles to achieve very high accuracy. Due to the odd mass ratio present in the environment, the differences in achievable residuals between the solvers are larger in dish piling compared to those in the bolt-nut assembly. This suggests that CANAL may be the more preferable option in this case, although SubADMM remains a viable choice for achieving moderate results in a very short time. The trend in computation time per iteration is similar in both scenarios; per-iteration cost ranks as follows: CANAL $>$ PGS $>$ SubADMM, and the cost for each iteration in CANAL tend to decreases as the iterations proceed.

B. Task Simulation

We perform a full task simulation using a Franka Panda arm equipped with a Hebi X5 gripper or Allegro hand, as depicted in Fig. 1. For performance validation, we limit the computation budget for the solver to $1\ \mathrm{ms}$ for each time step. As a result, simulations using CANAL successfully complete the both task. In contrast, PGS solver fails, as significant penetrations or jittery movements are generated due to its lack of convergence.

V. Conclusion

In this paper, we introduce a new multi-contact solver algorithm CANAL, based on the theory of augmented Lagrangian to handle multi-contact NCP. We variate original AL-based structure into a cascaded form of convex optimization, which can be solved by exact Newton steps, thereby ensuring accurate and robust simulation results. We believe that the development of contact modeling and solver is a valuable step toward narrowing the sim-to-real gap. For future work, we plan to integrate more diverse physical contact models with our framework, further improving simulation fidelity in complex contact scenarios.

REFERENCES

[1] Bullet physics engine. https://pybullet.org/.

[2] Mujoco physics engine. http://www.mujoco.org/.

[3] Physx physics engine. https://developer.nvidia.com/physx-sdk.

[4] Brax - a differentiable physics engine for large scale rigid body simulation. http://github.com/google/brax.

[5] Raisim physics engine. https://raisim.com/.

[6] Sofa physics engine. https://www.sofa-framework.org/.

[7] D. E. Stewart and J. 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.

[8] J. Nocedal and S. J Wright. *Numerical optimization*. Springer, 1999.

[9] N. Parikh, S. Boyd, et al. Proximal algorithms. *Foundations and trends® in Optimization*, 1(3):127–239, 2014.

[10] J. Lee, M. Lee, and D. Lee. Large-dimensional multibody dynamics simulation using contact nodalization and diagonalization. *IEEE Transactions on Robotics*, 39(2):1419–1438, 2022.

[11] M. Hintermüller. Semismooth newton methods and applications. *Department of Mathematics, Humboldt-University of Berlin*, 2010.

[12] V. Acary, F. Cadoux, C. Lemaréchal, and J. Malick. A formulation of the linear discrete coulomb friction problem via convex optimization. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, 91(2):155–175, 2011.

[13] E. Todorov. Convex and analytically-invertible dynamics with contacts and constraints: Theory and implementation in mujoco. In *IEEE International Conference on Robotics and Automation*, pages 6054–6061, 2014.

[14] A. M. Castro, F. N Permenter, and X. Han. An unconstrained convex formulation of compliant contact. *IEEE Transactions on Robotics*, 39(2):1301–1320, 2022.

[15] M. Macklin, M. Müller, N. Chentanez, and T. Kim. Unified particle physics for real-time applications. *ACM Transactions on Graphics*, 33(4), 2014.

[16] P. C. Horak and J. C. Trinkle. On the similarities and differences among contact models in robot simulation. *IEEE Robotics and Automation Letters*, 4(2):493–499, 2019.

[17] J. Lee, M. Lee, and D. Lee. Modular and parallelizable multibody physics simulation via subsystem-based admm. In *IEEE International Conference on Robotics and Automation*, pages 10132–10138, 2023.