Don't Break the Egg: Branch-Rejoining Trajectory Optimization under Contact Timing Uncertainties

Zhuocheng Zhang[1], Haizhou Zhao[1,2], Xudong Sun[1], Aaron M. Johnson[3,4], and Majid Khadiv[1]

***Abstract*— Robotic tasks involving contact interactions pose significant challenges for Trajectory Optimization (TO) due to discontinuous dynamics. Conventional formulations typically assume deterministic contact events, limiting robustness and adaptability in real-world settings. In this work, we propose SURE, a robust TO framework that explicitly accounts for uncertainty in contact timing. By allowing multiple trajectories to branch from possible pre-impact states and subsequently merge into a shared trajectory, SURE achieves both robustness and computational efficiency within a unified optimization framework. We evaluate SURE on two representative tasks with unknown impact times. In a cart-pole impact task with uncertain wall location, SURE achieves an average improvement of 21.6% in success rate when branch switching is enabled during control. In an egg-catching experiment using a robotic manipulator, SURE improves the success rate by 40%. These results demonstrate that SURE substantially enhances robustness compared to conventional nominal formulations. This paper is a short version of our full paper [1].**

I. INTRODUCTION

Most recent trajectory optimization approaches for contact-rich problems adopt hierarchical formulations [2], in which contact planning and whole-body motion generation are treated separately. These methods typically assume deterministic contact timings [3], [4]. In practice, however, such assumptions are overly restrictive: environmental perception is inherently uncertain, and both modeling errors and end-effector tracking inaccuracies during contact establishment can cause discrepancies between planned and actual contact times. To mitigate these effects, existing frameworks often rely on the inherent robustness of model predictive control (MPC) through frequent replanning [5], [6], [7], [8]. Nevertheless, contact switches introduce very fast dynamics that cannot be captured by the re-planning frequency of the MPC controllers, underscoring the need for a more systematic framework that explicitly addresses contact-timing uncertainty.

The primary contribution of this work is the development of a robust trajectory optimization framework, termed SURE(Safe Uncertainty-Aware Robot-Environment Interaction), for hybrid systems with uncertain contact timing. The framework introduces a branching phase, in which nodes can branch and subsequently rejoin a common final trajectory, thereby achieving both robustness and computational efficiency. Using two case studies, we demonstrate the improved robustness of the proposed approach compared to nominal trajectory optimization and analyze different implementations.

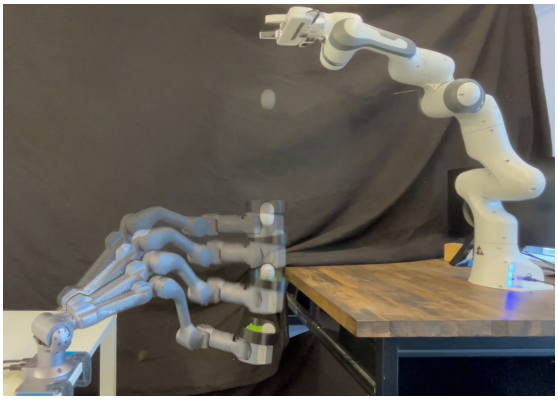


Fig. 1. Illustration of the egg-catching task. The Unitree Z1 robot arm attempts to catch a falling egg while minimizing impact. The egg serves both as the test object and as a passive sensor for detecting contact forces. In this task, timing deviations on the order of milliseconds can significantly affect performance, highlighting the importance of robustness to contact-timing uncertainties.

II. NOMINAL TRAJECTORY OPTIMIZATION

In this section, we present the formulation of the nominal trajectory optimization using multiple shooting [9], which assumes that contact occurs at a deterministic time. The trajectory starts from the initial state $\mathbf{x}_0$, under control $\mathbf{u}$, and is discretized into N shooting nodes, Fig. 2a. We assume that the contact occurs at the node indexed by c. The trajectory optimization problem can then be formulated as follows:

$$\min_{\mathbf{x},\mathbf{u},\Delta t} \quad \sum_{i=0}^{N-1} L_i(\mathbf{x}_i, \mathbf{u}_i, \Delta t_i) + L_N(\mathbf{x}_N) \tag{1a}$$

$$\text{s.t.} \quad \forall i \in [0, N]: \quad \mathbf{w}_i(\mathbf{x}_i, \mathbf{u}_i) = 0, \tag{1b}$$

$$\mathbf{h}_i(\mathbf{x}_i, \mathbf{u}_i) \leq 0, \tag{1c}$$

$$\forall i \notin \{c, N\}: \quad \mathbf{f}_i(\mathbf{x}_i, \mathbf{u}_i, \mathbf{x}_{i+1}, \Delta t_i) = 0, \tag{1d}$$

$$g(\mathbf{x}_i) > 0, \tag{1e}$$

$$i = c: \quad \mathbf{x}_{i+1} = R(\mathbf{x}_i), \tag{1f}$$

$$g(\mathbf{x}_i) = 0, \tag{1g}$$

$$\Delta t \in [\Delta t_{\min}, \Delta t_{\max}]. \tag{1h}$$

Here, Δt_i denotes the time interval between nodes, varying within the bounds specified in (1h); L_N is the terminal cost;

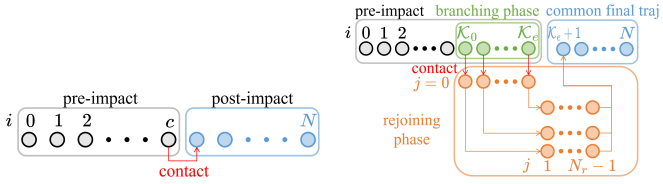
* This work was partially supported by the Huawei-TUM joint laboratory, SIEMENS AG and the Technical University of Munich - Institute for Advanced Study, Germany.

[1] Munich Institute of Robotics and Machine Intelligence (MIRMI), Technical University of Munich(TUM), Germany.

[2] Machines in Motion laboratory, New York University, New York, NY, USA.

[3] Institute for Advanced Study, Technical University of Munich, Garching, Germany.

[4] Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA, USA.



(a) Nominal Approach (b) SURE (ours)

Fig. 2. (a) The nominal trajectory optimization problem. The contact occurs at the node indexed by c, separating the pre- and post-impact phases. (b) SURE trajectory optimization problem. The gray nodes represent the pre-contact trajectory, the green nodes correspond to the branching phase (serving the same role as the robust phase in (b)), and the blue nodes denote the common final trajectory. Together, they form the common trajectory. The orange nodes represent the branch trajectories, each diverging from a branching node and rejoining the common final trajectory, composing the rejoining phase.

L_i is the running cost; $\mathbf{w}$ and $\mathbf{h}$ represent the state-input equality and inequality constraints, respectively; $\mathbf{f}$ specifies the discrete dynamics; g is the guard function, whose value determines the triggering condition of the contact event (contact occurs at $g = 0$, while $g > 0$ holds at all other times); and R is the impact or reset function.

III. SURE TRAJECTORY OPTIMIZATION

The formulation in (1) assumes a deterministic contact event. However, this assumption is often unrealistic due to modeling or perception errors. In such cases, the trajectory obtained from (1) cannot accommodate the timing uncertainty: if the contact occurs earlier or later than expected, it may cause a large impact or even lead to instability.

A. SURE Formulation

Similar to the robust phase in [10], we construct a branching phase $\mathcal{K}$, which includes all the possible pre-impact nodes. To address the drawback of [10], we force these branches to eventually rejoin a common final trajectory. From each node in the branching phase, we generate a rejoining phase trajectory, each of which ends at the start of this common final trajectory, as shown in Fig. 2b. For trajectories with multiple contact transitions, this process is repeated for each.

The SURE trajectory optimization problem under uncertain contact timing is formulated as follows:

$$\min_{\mathbf{x},\mathbf{u},\Delta t,d} \sum_{i=0}^{N-1} L_i(\mathbf{x}_i,\mathbf{u}_i,\Delta t_i) + \sum_{i\in\mathcal{K}}\sum_{j=0}^{N_r} L_{i,j}(\mathbf{x}_{i,j},\mathbf{u}_{i,j},\Delta t_{i,j}) + L_N(\mathbf{x}_N) \tag{2a}$$

$$\text{s.t.} \quad \forall i \in [0,N]: \quad \mathbf{w}_i(\mathbf{x}_i,\mathbf{u}_i) = 0, \tag{2b}$$

$$\mathbf{h}_i(\mathbf{x}_i,\mathbf{u}_i) \le 0, \tag{2c}$$

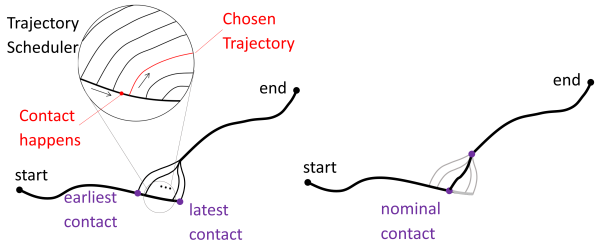
$$\forall i \notin \{\mathcal{K}_e, N\}: \mathbf{f}_i(\mathbf{x}_i,\mathbf{u}_i,\mathbf{x}_{i+1},\Delta t_i) = 0, \tag{2d}$$

$$\forall i \in \mathcal{K}: \quad \mathbf{x}_{i,0} = R(\mathbf{x}_i), \tag{2e}$$

$$\mathbf{x}_{i,N_r} = \mathbf{x}_{\mathcal{K}_e+1}, \tag{2f}$$

$$\forall j \in [0, N_r-1]: \quad \mathbf{f}_{i,j}(\mathbf{x}_{i,j},\mathbf{u}_{i,j},\mathbf{x}_{i,j+1},\Delta t_{i,j}) = 0, \tag{2g}$$

$$\mathbf{w}_{i,j}(\mathbf{x}_{i,j},\mathbf{u}_{i,j}) = 0, \tag{2h}$$



(a) Trajectory scheduling (b) Robust nominal trajectory

Fig. 3. Illustrations of (a) trajectory scheduling, and (b) the robust nominal trajectory. Trajectory scheduling utilizes all optimized trajectories obtained from the robust approach, whereas the robust nominal trajectory corresponds to the middle branch selected from these solutions.

$$\mathbf{h}_{i,j}(\mathbf{x}_{i,j},\mathbf{u}_{i,j}) \le 0, \tag{2i}$$

$$i = \mathcal{K}_0: \quad g(\mathbf{x}_i) = d, \tag{2j}$$

$$i = \mathcal{K}_e: \quad g(\mathbf{x}_i) = -d, \tag{2k}$$

$$\forall i < \mathcal{K}_0: \quad g(\mathbf{x}_i) > d, \tag{2l}$$

$$d \in [d_{\min}, d_{\max}], \tag{2m}$$

$$\Delta t \in [\Delta t_{\min}, \Delta t_{\max}]. \tag{2n}$$

All nodes can be classified into two categories: (1) nodes on the common trajectory (with a single subscript index, e.g., $\mathbf{x}_i$) and (2) nodes in the rejoining phase (with two subscript indices, e.g., $\mathbf{x}_{i,j}$).

On the common trajectory, $\mathcal{K}_0$ and $\mathcal{K}_e$ denote the indices of the first and last nodes in the branching phase, respectively. The variable d denotes half the width of the uncertainty range. At $i = \mathcal{K}_0$, the earliest possible contact, we require $g(\mathbf{x}_i) = d$, (2j), while at $i = \mathcal{K}_e$, the latest possible contact, we require $g(\mathbf{x}_i) = -d$, (2k). Each node in the branching phase represents a pre-contact state, and we apply a state transition to compute the corresponding post-impact state which is then the initial state on the corresponding branch,(2e).

In the rejoining phase, each branch contains N_r+1 nodes, indexed by $j = 0, 1, \ldots, N_r$. Node $\mathbf{x}_{i,j}$ denotes the j-th node on the branch originating from node $i \in \mathcal{K}$. The contact-related cost and constraints are incorporated into $L_{i,j}$ as well as (2h) and (2i). At $j = N_r$, the branch node rejoins the common final trajectory at the node with index $i = \mathcal{K}_e + 1$, (2f).

B. Application of Solution Trajectories for Control

1) Trajectory Scheduling: When a sensor is available to detect contact, the reference trajectory can be switched according to the detected contact time, a strategy referred to as *trajectory scheduling* (Fig. 3a). Initially, the system follows the common pre-impact trajectory. Once it enters the branching phase, if contact does not occur, the system continues along the common trajectory. When contact occurs, the scheduler selects the nearest subsequent trajectory as the new reference and guides the system to the terminal state.

2) Robust Nominal Trajectory: If no sensor is available for contact detection, having multiple trajectories is still beneficial. By introducing additional decision variables and constraints, the desired trajectory before the contact event can also be reshaped to reduce the worst-case impact within

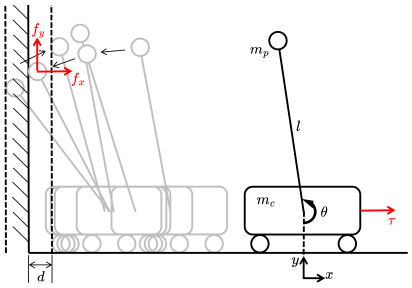


Fig. 4. Illustration of the cart-pole system with wall contact. The cart-pole starts from an initially disturbed state and, due to limited control input, moves toward the wall to induce an impact that reverses the pole's velocity. After the impact, the system seeks to regain balance and return to the desired position. The wall position is uncertain within a range of $\pm d$.

the uncertainty region. As a result, selecting a branch among these trajectories is expected to yield improved worst-case performance than the nominal trajectory, even though performance in some specific cases may degrade. It is reasonable to choose the middle branch, as it is located at almost equal distance to the extremes of the uncertainty set and thus represents the most representative trajectory. This trajectory branches from index $i = \lceil (\mathcal{K}_0 + \mathcal{K}_e)/2 \rceil$ and is referred to as the *robust nominal trajectory* (Fig. 3b).

IV. Case Study I: Cart-Pole System with Wall

In this case study, a cart-pole system stabilizes itself under disturbances using contact with an uncertain wall (Fig. 4). The system dynamics are given by:

$$M(\mathbf{q})\ddot{\mathbf{q}} + H(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tau + J_c^\top \mathbf{F} \tag{3}$$

where $\mathbf{q} = [x, \theta]^\top$ denotes the generalized coordinates, with x the cart position and θ the pole angle. The cart and pole masses are $m_c = 0.3\,\mathrm{kg}$ and $m_p = 1\,\mathrm{kg}$, respectively, and the pole has length $l = 0.4\,\mathrm{m}$ with mass concentrated at the tip. The system input is the external force τ acting on the cart. $M(\mathbf{q})$ denotes the mass matrix, and $H(\mathbf{q}, \dot{\mathbf{q}})$ captures Coriolis, centrifugal, and gravitational effects. The contact Jacobian at the pole tip is $J_c \in \mathbb{R}^{2\times 2}$, and the wall reaction force is $\mathbf{F} = [f_x, f_y]^\top \in \mathbb{R}^2$.

The trajectories generated by both nominal and SURE approaches are used as references to control the system in simulation, and their performance is compared to evaluate robustness. As the simulation platform, we developed a custom environment to reproduce the physical behavior of the cart-pole system and its interaction with the wall. For tracking the reference trajectory, a proportional-derivative (PD) controller with feedforward compensation is employed:

$$\tau(t) = \mathbf{k}_p\big(\mathbf{q}_{des}(t) - \mathbf{q}(t)\big) + \mathbf{k}_d\big(\dot{\mathbf{q}}_{des}(t) - \dot{\mathbf{q}}(t)\big) + \tau_{des}(t), \tag{4}$$

where $\mathbf{k}_p \in \mathbb{R}^2$ and $\mathbf{k}_d \in \mathbb{R}^2$ are the proportional and derivative gain vectors of the PD controller, respectively.

We evaluate whether stabilization can still be achieved under two sources of uncertainty: the wall position and the restitution coefficient of the wall. In each trial, the wall position and restitution coefficient are randomly sampled and held fixed. The system is released from an (often aggressive) initial condition, and the controller attempts to track the reference trajectory while exploiting wall contact for stabilization. Four sets of initial conditions are evaluated. Each case starts from a different initial state but shares the same terminal state, $\mathbf{x}_{\text{end}} = [0, \pi, 0, 0]^\top$. The results are summarized in Table I, and representative successful and failed samples for Condition 4 are shown in Fig. 5.

TABLE I

Experimental Results for the Cart–Pole-Wall Simulation

	State				Success Rate		
Initial Condition	x $[m]$	$\dot{x}$ $[m/s]$	θ $[rad]$	$\dot{\theta}$ $[rad/s]$	Nominal	Robust Nominal	Trajectory Scheduling
1	0	0	π	5.5	58.5%	60.0%	75.0%
2	0	0	π	6.5	37.0%	43.0%	48.0%
3	0	-1.0	3.53	3.5	37.0%	57.0%	70.0%
4	0	-0.5	3.45	4.5	46.5%	61.0%	72.5%
Total					44.8%	55.3%	66.4%

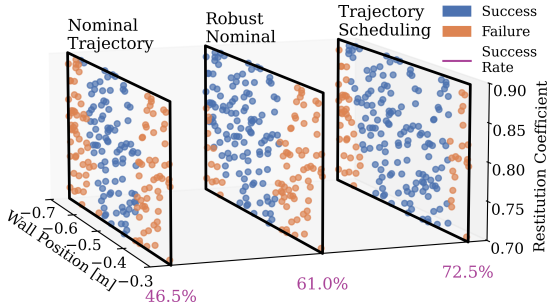


Fig. 5. Robustness comparison under Initial Condition 4 for three reference trajectories. The nominal wall position is $x_{\text{wall}} = -0.5\,\mathrm{m}$ and the nominal restitution coefficient is 0.8. SURE trajectory optimization is performed using 5 branches, with the half-width of the uncertainty range fixed at $d = 0.05\,\mathrm{m}$. The wall position varies within $x_{\text{wall}} \in [-0.7\,\mathrm{m}, -0.3\,\mathrm{m}]$, and the restitution coefficient ranges from 0.7 to 0.9. A total of 200 points are randomly sampled within this two-dimensional uncertainty space, and simulations are conducted for all three approaches to evaluate and compare their success rates.

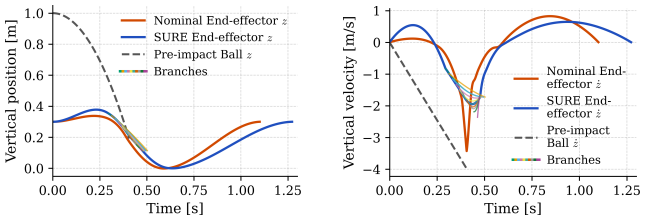
First, we observe that the nominal trajectories already exhibit a certain degree of robustness to variations in wall position and restitution coefficient with a success rate of 44.8%, even though these uncertainties are not explicitly considered during optimization. In comparison, the SURE solution with trajectory scheduling demonstrates a substantially higher success rate of 66.4%—even when evaluated over a wider uncertainty range than that specified during optimization (the intended range being $e = 0.8$, $x_{\text{wall}} = -0.5 \pm 0.05\,\mathrm{m}$).

V. Case Study II: Ball Catching

As shown in Fig. 1, we consider a Unitree Z1 robotic manipulator with a lightweight container as its end effector. The manipulator dynamics are given by:

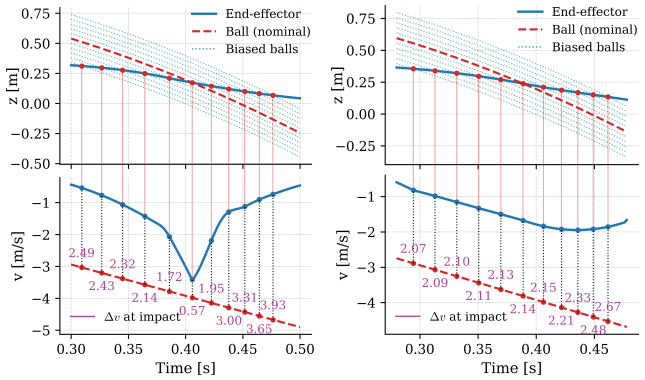
$$M(\mathbf{q})\ddot{\mathbf{q}} + H(\mathbf{q}, \dot{\mathbf{q}}) = \boldsymbol{\tau} \tag{5}$$

where $\mathbf{q} \in \mathbb{R}^6$ denotes the joint angles, $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{6\times 6}$ is the mass matrix, and $\mathbf{H}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^6$ collects the Coriolis, centrifugal, and gravitational terms. $\boldsymbol{\tau} \in \mathbb{R}^6$ is the vector of joint torques. A ball falls freely from an initial height



(a) Position (b) Velocity

Fig. 6. Nominal trajectory: The end effector moves upward only slightly and then descends rapidly, anticipating contact at a deterministic time. SURE robust nominal trajectory: The end effector initially moves higher, enabling a longer co-travel distance with the egg. During descent, it follows a smoother, gradually accelerating motion, thereby accommodating the possibility of earlier or later contact.



(a) Nominal (b) Robust Nominal

Fig. 7. Relative velocity between the ball and the end effector under varying initial ball heights. The ball is released from 11 uniformly spaced initial heights in the range $[p_{\text{ball},0,z}-0.2\,\text{m}, p_{\text{ball},0,z}+0.2\,\text{m}]$. The top row shows the vertical position profiles of the end effector and balls, while the bottom row shows the corresponding velocity profiles. Vertical dashed lines indicate the contact times for each trial, and the purple annotations report the relative velocity at impact. (a) Nominal trajectory, with a maximum impact relative velocity of 3.93m/s. (b) Robust nominal trajectory, with a reduced maximum impact relative velocity of 2.67m/s.

h_0 with uncertainty $\pm d$ and no horizontal velocity. Given a prescribed release time, the manipulator aims to catch the ball while minimizing impact and move it to a final position. Figure 6 compares the solution trajectories obtained from the nominal and SURE optimization formulations, including the corresponding position and velocity profiles.

To further analyze impact robustness, we evaluate the relative velocity between the ball and the end effector at contact for different initial ball heights, as shown in Fig. 7. The results indicate that the robust nominal trajectory provides substantially improved robustness. The nominal trajectory performs well only when the ball is released from the nominal height; deviations in the initial height lead to higher relative velocities at contact for all but the nominal and two closest heights. In contrast, the robust nominal trajectory keeps the relative velocity v_{lim} bounded below 2.67 m/s across all tested initial heights, effectively limiting impact despite uncertainty in contact timing. Meanwhile, the nominal trajectory reaches a maximum relative velocity of 3.93 m/s (47% higher).

We conduct physical experiments in which the robot attempts to catch an egg using either the nominal or the SURE robust nominal trajectory shown in Fig. 6, under identical initial conditions. The trajectory scheduling was not implemented as it requires accurate contact event sensing, which was not available at the time.

TABLE II

RESULT OF THE EGG DROP EXPERIMENT

t_{wait}	Trajectory	1	2	3	4	5	6	7	8	9	10	Success Rate
0.125s	nominal	✗	✓	✓	○	○	✗	○	✗	✗	✗	35%
	SURE	✓	✓	✓	✓	✓	✓	✓	○	✓	✓	95%
0.118s	nominal	✗	✗	✓	✓	✗	✓	✓	○	✓	✗	55%
	SURE	✓	○	○	✗	○	✓	✓	✓	✓	✓	75%
Total	nominal											45%
	SURE											85%

Opening the gripper and releasing the ball require a non-negligible amount of time. Consequently, the Z1 must wait for a fixed delay $t_{\text{wait}} \approx 0.12\text{s}$ to achieve approximate synchronization. For each trajectory, we tune t_{wait} such that the average contact time is close to the nominal contact time. This is achieved by iteratively adjusting t_{wait} and performing ten drop-catch trials until the number of early contacts approximately equals the number of late contacts. After calibration, we obtain $t_{\text{wait}} = 0.118\text{s}$ for the nominal trajectory and $t_{\text{wait}} = 0.125\text{s}$ for the SURE trajectory. Using both of these delays, we conduct ten additional trials for each trajectory. We classify the outcomes of each trial into three categories:

(i) successful catch without egg damage (score $+1$, denoted as ✓);
(ii) egg contacts the end effector, bounces upward by more than 10 cm, then falls back into the end effector (by luck) without breaking (score $+0.5$, denoted as ○);
(iii) egg breaks or drops (score 0, denoted as ✗).

The experimental results are summarized in Table II. The results show that, under both timing settings, the SURE trajectory consistently outperforms the nominal trajectory by achieving a higher success rate. Overall, SURE achieves an 85% success rate compared to 45% for the nominal approach, corresponding to a 40% improvement in egg-catching performance under timing uncertainty.

VI. CONCLUSION

In this paper, we proposed an efficient formulation for robust trajectory optimization that can handle uncertainties in contact timing. The main novelty of our formulation lies in branching possible solutions at different contact timings and rejoining all solutions at a common point outside the uncertain region. Through extensive simulation and experimental demonstrations, we showed that our framework adapts its strategy in the presence of uncertainty, resulting in more robust solutions than a nominal formulation.

In future work, we plan to extend our formulation to account for imperfections in low-level control. Furthermore, we are interested to learn uncertainty-conditioned policies from our robust formulation under various uncertainty conditions. We are also interested in extending the framework to floating-base systems, with the goal of enabling robust loco-manipulation behaviors.

REFERENCES

[1] Z. Zhang, H. Zhao, X. Sun, A. M. Johnson, and M. Khadiv, "Sure: Safe uncertainty-aware robot-environment interaction using trajectory optimization," *arXiv preprint arXiv:2602.06864*, 2026.

[2] P. M. Wensing, M. Posa, Y. Hu, A. Escande, N. Mansard, and A. Del Prete, "Optimization-based control for dynamic legged robots," *IEEE Transactions on Robotics*, vol. 40, pp. 43–63, 2023.

[3] R. Grandia, F. Jenelten, S. Yang, F. Farshidian, and M. Hutter, "Perceptive locomotion through nonlinear model-predictive control," *IEEE Transactions on Robotics*, vol. 39, no. 5, pp. 3402–3421, 2023.

[4] V. Dhédin, H. Zhao, and M. Khadiv, "Simultaneous contact sequence and patch planning for dynamic locomotion," in *IEEE-RAS International Conference on Humanoid Robots*, 2025, pp. 245–252.

[5] M. Toussaint, J. Harris, J.-S. Ha, D. Driess, and W. Hönig, "Sequence-of-constraints MPC: Reactive timing-optimal control of sequential manipulation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2022, pp. 13 753–13 760.

[6] C. Mastalli, W. Merkt, G. Xin, J. Shim, M. Mistry, I. Havoutis, and S. Vijayakumar, "Agile maneuvers in legged robots: a predictive control approach," *arXiv preprint arXiv:2203.07554*, 2022.

[7] A. Meduri, P. Shah, J. Viereck, M. Khadiv, I. Havoutis, and L. Righetti, "BiConMP: A nonlinear model predictive control framework for whole body motion planning," *IEEE Transactions on Robotics*, vol. 39, no. 2, pp. 905–922, 2023.

[8] L. Yan, T. Stouraitis, J. Moura, W. Xu, M. Gienger, and S. Vijayakumar, "Impact-aware bimanual catching of large-momentum objects," *IEEE Transactions on Robotics*, vol. 40, pp. 2543–2563, 2024.

[9] D. D. Morrison, J. D. Riley, and J. F. Zancanaro, "Multiple shooting method for two-point boundary value problems," *Communications of the ACM*, vol. 5, no. 12, pp. 613–614, 1962.

[10] H. Zhao and M. Khadiv, "Trajectory optimization under contact timing uncertainties," in *IEEE-RAS International Conference on Humanoid Robots*, 2024, pp. 1064–1071.