CDM-MPC: An Integrated Dynamic Planning and Control Framework for Bipedal Robots Jumping

Zhicheng He, Jiayang Wu, Jingwen Zhang, Shibowen Zhang, Yapeng Shi,
Hangxin Liu, Lining Sun, Yao Su, *Member, IEEE*, and Xiaokun Leng

arXiv:2405.11773v1 [cs.RO] 20 May 2024

Abstract—Performing acrobatic maneuvers like dynamic jumping in bipedal robots presents significant challenges in terms of actuation, motion planning, and control. Traditional approaches to these tasks often simplify dynamics to enhance computational efficiency, potentially overlooking critical factors such as the control of centroidal angular momentum (CAM) and the variability of centroidal composite rigid body inertia (CCRBI). This paper introduces a novel integrated dynamic planning and control framework, termed centroidal dynamics model-based model predictive control (CDM-MPC), designed for robust jumping control that fully considers centroidal momentum and non-constant CCRBI. The framework comprises an optimization-based kinodynamic motion planner and an MPC controller for real-time trajectory tracking and replanning. Additionally, a centroidal momentum-based inverse kinematics (IK) solver and a landing heuristic controller are developed to ensure stability during high-impact landings. The efficacy of the CDM-MPC framework is validated through extensive testing on the full-sized humanoid robot KUAVO in both simulations and experiments.

Index Terms—Jumping control, model predictive control, bipedal robot, optimization, acrobatic motion planning

I. INTRODUCTION

ACHIEVING acrobatic motions, a significant challenge in bipedal robotics, requires not only powerful robot actuators [1, 2] but also sophisticated motion planning and control algorithms [3, 4]. Unlike the control of walking or running—where the **centroidal angular momentum (CAM)** is typically overlooked to simplify the highly nonlinear multibody dynamics using models like linear inverted pendu-

Zhicheng He, Jiayang Wu, Yapeng Shi and Xiaokun Leng are with department of Computer Science, Harbin Institute of Technology, Harbin 150001, China (e-mails: hezhicheng@hit.edu.cn; 2021113679@stu.hit.edu.cn; shi.yapeng@hit.edu.cn; lengxiaokun@hit.edu.cn).

Jingwen Zhang, Shibowen Zhang, Hangxin Liu and Yao Su are with State Key Laboratory of General Artificial Intelligence, Beijing Institute for General Artificial Intelligence (BIGAI), Beijing 100080, China (e-mails: zhangjingwen@bigai.ai; zhangshibowen@bigai.ai; liuhx@bigai.ai; suyao@bigai.ai).

Lining Sun is with School of Mechatronics Engineering, Harbin Institute of Technology, Harbin 150080, China, and also with Jiangsu Provincial Key Laboratory of Advanced Robotics, School of Mechanical and Electric Engineering, Soochow University, Suzhou 215000, China (e-mail: lnsun@hit.edu.cn).

This letter has supplementary downloadable material available at https://doi.org/10.1109/LRA.2024.xxxxxxx, provided by the authors.

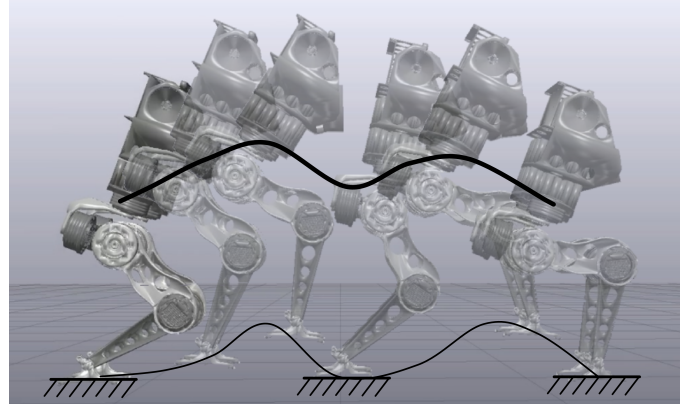Digital Object Identifier (DOI): see top of this page.

Fig. 1: **The proposed integrated dynamic planning and control framework endows bipedal robots capable of continuously forward jumping.** The trajectories of the foot and the torso links are plotted in thin and bold black lines, respectively.

lum model (LIPM), spring-loaded inverted pendulum model (SLIPM), or single rigid body model (SRBM) for computational efficiency, CAM plays a key role in the jumping control of bipedal robots. This introduces unique challenges and necessitates additional considerations in the control strategy to accurately manage CAM throughout the entire process [5, 6].

The dynamic jumping process of bipedal robots, depicted in Fig. 1, consists of three distinct phases: the **launching phase**, the **flight phase**, and the **landing phase**. Each phase poses unique challenges in dynamics planning and control. Specifically, (i) the launching phase requires an accurate kinodynamic motion planner to generate trajectories for the center of mass (CoM) and CAM within the hardware capabilities [7]; (ii) the flight phase necessitates a fast trajectory replanning scheme to address motor execution errors and unknown disturbances, alongside an advanced whole-body control (WBC)-based CAM controller for accurate trajectory tracking [6, 8]; (iii) the landing phase demands a robust landing controller to preserve platform stability under substantial impact forces [1, 6].

Researchers have developed various integrated motion planning and control frameworks [6, 8–11] to address this challenge. Specifically, Zhang *et al.* employed CAM for planning the entire jumping motion with a fixed foothold location and a heuristic landing planner to compensate for disturbances or tracking errors [6]. Qi *et al.* designed a CAM controller based on LIPM for real-time planning of whole-body dynamics with predefined CoP [8]. Li *et al.* utilized a reinforcement learning (RL)-based approach to establish a robot-specific jumping

control policy [9]. Saloutos *et al.* and Mesesan *et al.* generated jumping trajectories through trajectory optimization leveraging full-order dynamics or a 3-D divergent component of motion (3D-DCM) framework, respectively [10, 11]. Despite their theoretical contributions, these frameworks have not achieved the continuous jumping of a humanoid robot, which requires more robustness than a single vertical jumping.

In this study, we introduce a novel integrated dynamic planning and control framework—centroidal dynamics model-based model predictive control (CDM-MPC), tailored for bipedal robot jumping. This framework thoroughly accounts for the full centroidal momentum while considering non-constant centroidal composite rigid body inertia (CCRBI) to achieve precise body pose control. Initially, we formulate an optimization-based kinodynamic motion planner that concurrently optimizes the CAM trajectory, touchdown point, and contact force trajectories. Subsequently, we design an innovative model predictive control (MPC) controller for precise trajectory tracking and fast online replanning. Further, we develop a centroidal momentum-based Inverse Kinematics (IK) solver to achieve accurate centroidal momentum tracking and a landing heuristic controller to ensure robust stabilization during high-impact landings. The efficacy of the proposed framework is corroborated through rigorous testing on our full-scale humanoid robot platform, KUAVO, within both realistic simulation environments and real-world experimental settings.

A. Related Work

MPC-based motion planning/control frameworks have gained significant attention in recent years due to their capability to account for kinematics, dynamics, physical, and contact constraints through a unified optimization formulation. These frameworks have demonstrated success across various robotic configurations, including wheeled robots [12, 13], UAVs [14], bipedal robots [1, 15, 16], quadruped robots [17–20], and hexapod robots [21, 22]. In the realm of bipedal robots, Pratt *et al.* first proposed the capture point method to determine the ideal footstep locations post-disturbance within an MPC framework [23], albeit lacking consideration of kinematic and dynamic constraints. Scianca *et al.* enhanced this approach by incorporating Zero Moment Point (ZMP), kinematics, and stability constraints for improved gait generation capability [24],

while Brasseur *et al.* extends the method to a 3D walking scenario with online gait generation. Concurrently optimization of footstep locations and step time was achieved by a nonlinear MPC formulation [25]. These methodologies excelled in low-speed walking scenarios by utilizing simplified models, such as LIPM and SLIPM for computational efficiency, which is less suited for dynamic motions where the CAM of the robot can not be neglected.

To enhance the generation of dynamic motions for legged robots across various speeds and magnitudes, the **centroidal dynamics model (CDM)** was integrated into trajectory optimization formulations [26–28]. Specifically, Budhiraja *et al.* introduced a formulation to reconcile the CDM with the complete dynamics model [26]; Ponton *et al.* employed the CDM for efficient walking pattern generation [27]; Kwon *et al.* utilized the CDM for footstep planning, integrating a momentum-mapped IK solver to design whole-body motion [28]. Moreover, the CDM-based real-time MPC frameworks have been successfully implemented on agile maneuver control of quadruped robots, considering the constant CCRBI of the robot [29, 30]. In this study, we concentrate on the jumping control of biped robots while accounting for non-constant CCRBI. We first derive the relationship between CCRBI and robot leg length as a constraint, then incorporate it into a real-time MPC framework to accurately regulate the body posture of a biped robot during the jumping process.

B. Overview

We organize the remainder of the paper as follows. Sec. II introduces the kinematics and dynamics model of the system. Sec. III describes the dynamical planning and control framework. Sec. IV presents the simulation and experiment results with comprehensive evaluations. Finally, we conclude the paper in Sec. V.

II. PLATFORM MODEL

A. Hardware Introduction

The hardware platform depicted in Fig. 2 is developed to evaluate the performance of the proposed CDM-MPC framework. The KUAVO bipedal robot platform stands $1.2\ m$ in height and weighs $34.5\ kg$. It incorporates 18 motors: each
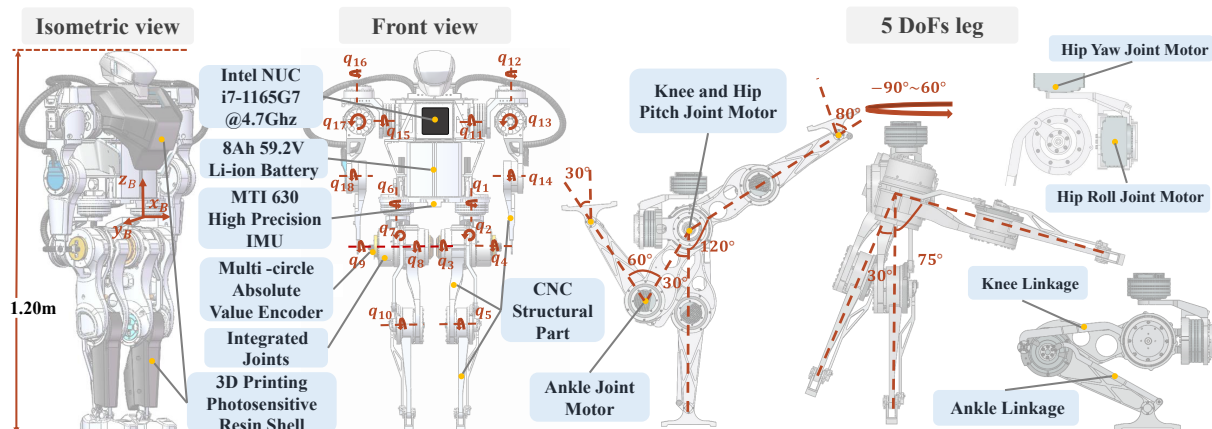

Fig. 2: **Hardware design and configuration of the bipedal humanoid robot KUAVO.** Each leg contains 5 Degree of Freedoms (DoFs): 3 DoFs for the hip joint, 1 DoF for the knee joint and 1 DoF for the ankle joint.

TABLE I: **Main Physical Parameters of KUAVO Robot**

Dimension Parameters				
Total mass	Pelvis width	Thigh length	Calf length	Foot length
34.5 [kg]	0.22 [m]	0.23 [m]	0.26 [m]	0.15 [m]
Motion Range & Joint Peak Torque				
Hip Yaw	Hip Roll	Hip Pitch	Knee Pitch	Ankle Pitch
$-90° \sim 60°$	$-30° \sim 75°$	$-30° \sim 120°$	$-120° \sim 10°$	$-30° \sim 80°$
48 [Nm]	110 [Nm]	110 [Nm]	110 [Nm]	48 [Nm]

leg is equipped with 5 DoFs and each arm with 4 DoFs. A key design innovation implemented to optimize the robot's performance involves the strategic placement of the ankle joint motor at the knee and the knee joint motor at the hip. These modifications, coupled with linkage mechanisms, are intended to reduce leg inertia and improve the robot's maneuverability. The robot operates in a Linux system with an augmented real-time kernel patch to prompt the reading of sensor inputs, the whole system runs on an onboard Intel NUC computer (Intel Core i7-1165G7@4.7GHz). Notably, all joint actuators are configured to operate in the torque control mode, ensuring precise and efficient control. For a comprehensive overview of the hardware platform's specifications, please refer to Tab. I.

B. Dynamics

1) Floating Base Dynamics: The comprehensive dynamics of a biped robot can be described as:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \boldsymbol{\Lambda}\boldsymbol{\tau} + \boldsymbol{J}^\top(\boldsymbol{q})\boldsymbol{f}_{\text{ext}}, \quad (1)$$

where $\boldsymbol{M}(\boldsymbol{q})$ is the inertia matrix, $\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})$ is the vector of Coriolis, centrifugal and gravitational forces, $\boldsymbol{\tau}$ is the joint torque, $\boldsymbol{\Lambda}$ is the input mapping matrix, $\boldsymbol{J}(\boldsymbol{q})$ is the Jacobian matrix and $\boldsymbol{f}_{\text{ext}}$ is the external force vector [31]. However, when utilizing this highly nonlinear multi-body dynamics model for controller design, computational efficiency becomes a significant concern. Even with the advanced nonlinear optimizers [32], real-time control applications remain problematic.

2) Centroidal Dynamics: Consequently, the CDM proposed by Orin *et al.* [33] is employed in this work, offering a holistic approach to capturing whole body momentum while maintaining simplicity. It can be formally written as:

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{h} \\ m\dot{\boldsymbol{r}} \end{bmatrix} = \boldsymbol{A}(\boldsymbol{q})\dot{\boldsymbol{q}}, \quad (2)$$

where the centroidal momentum vector $\boldsymbol{H} \in \mathbb{R}^{6\times1}$ consists of the angular centroidal momentum $\boldsymbol{h} \in \mathbb{R}^{3\times1}$ and the linear centroidal momentum $m\dot{\boldsymbol{r}}$ of the robot. Here, m denotes the total mass and $\boldsymbol{r} \in \mathbb{R}^{3\times1}$ represents the CoM position, $\boldsymbol{A} \in \mathbb{R}^{6\times16}$ is the centroidal momentum matrix (CMM), $\boldsymbol{q} \in \mathbb{R}^{16\times1}$ is the generalized position vector. Notably, for model simplicity the dynamics and control of the arms are not considered in work, the whole upper body of the robot is treated as one single body. Therefore, $\boldsymbol{q}$ includes the position and attitude of the base (6-DoFs), and the 10-DoFs of the legs.

According to Newton's laws of motion, the rate of change of angular and linear momentum at the CoM, $\dot{\boldsymbol{h}}$ and $m\ddot{\boldsymbol{r}}$, is equivalent to the resultant effect of all external forces. As illustrated in Fig. 3, we can have:

$$\begin{aligned} \dot{\boldsymbol{h}} &= (\boldsymbol{\rho} - \boldsymbol{r}) \times \boldsymbol{f}_\rho, \\ m\ddot{\boldsymbol{r}} &= m\boldsymbol{g} + \boldsymbol{f}_\rho, \end{aligned} \quad (3)$$
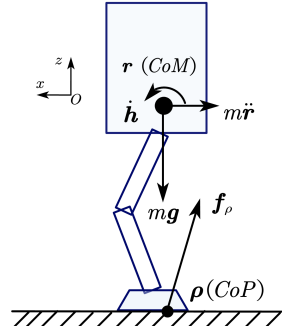

Fig. 3: **External forces analysis.** In the xOz plane, the robot is subjected to gravitational force $m\boldsymbol{g}$ and ground reaction force $\boldsymbol{f}_\rho$.

where $m\boldsymbol{g}$ and $\boldsymbol{f}_\rho \in \mathbb{R}^{3\times1}$ are the gravitational force and ground reaction force of the robot, respectively. $\boldsymbol{\rho} \in \mathbb{R}^{3\times1}$ is the Center of Pressure (CoP) location.

III. DYNAMIC PLANNING & CONTROL FRAMEWORK

Building upon the CDM, we develop the CDM-MPC framework to address the complexities encountered during bipedal robot jumping in the launching, flight, and landing phases, as introduced in Sec. I. Our framework consists of four primary components: (i) an optimization-based kinodynamic motion planner to produce the CDM trajectory, contact force, and contact position; (ii) a real-time MPC controller with CDM for trajectory tracking and fast replanning; (iii) an IK solver based on centroidal momentum that calculates whole-body trajectories; (iv) a landing heuristic controller for robust stabilization. The overall framework is summarized in Fig. 4.

A. Kinodynamic Motion Planning

Assuming each foot has two contact points—the heel and the toe, with the contact force at each point constrained by a linearized friction cone comprising four edges, we formulate the kinodynamic motion planning as a trajectory optimization problem and define the set of decision variables as:

$$\begin{aligned} \boldsymbol{\Gamma}^{\text{kmp}} = \{&\boldsymbol{q}_{[k]}, \dot{\boldsymbol{q}}_{[k]}, \boldsymbol{r}_{[k]}, \dot{\boldsymbol{r}}_{[k]}, \ddot{\boldsymbol{r}}_{[k]}, \boldsymbol{h}_{[k]}, \dot{\boldsymbol{h}}_{[k]}, \boldsymbol{f}_{\rho[k]}^i, \boldsymbol{\rho}_{[k]}^i, \beta_{[k]}^{ij}, \\ &\forall i = 1, \cdots, N_c, j = 1, \cdots, N_d, k = 1, \cdots, N\} \in \mathbb{R}^{87\times N}, \end{aligned} \quad (4)$$

where the subscript $[k]$ represents the kth time interval, N is the total number of timesteps, $\boldsymbol{\rho}^i$ and $\boldsymbol{f}_\rho^i$ denote the ith contact location and corresponding contact force, $N_c = 4$ is the number of contact points, β^{ij} represents the portion of the friction along each edge of the linearized friction pyramid, $N_d = 16$ is the total number of these approximated edges.

The kinematic constraints include [3]

$$\boldsymbol{H}_{[k]} = \boldsymbol{A}(\boldsymbol{q}_{[k]})\dot{\boldsymbol{q}}_{[k]}, \quad (5a)$$

$$\boldsymbol{r}_{[k]} = \boldsymbol{\mathcal{F}}_r(\boldsymbol{q}_{[k]}), \quad (5b)$$

$$\boldsymbol{\rho}_{[k]}^i = \boldsymbol{\mathcal{F}}_\rho^i(\boldsymbol{q}_{[k]}), \qquad i = 1, \cdots, N_c, \quad (5c)$$

$$\left\|\boldsymbol{\rho}_{[k]}^i - \boldsymbol{\rho}_{[k-1]}^i\right\| \leqslant \boldsymbol{D}\left(1 - \boldsymbol{S}_{[k]}^i\right), \quad i = 1, \cdots, N_c, \quad (5d)$$

$$\left\|\boldsymbol{r}_{[k]} - \boldsymbol{\rho}_{[k]}^i\right\| \geqslant \xi_{\min}, \qquad i = 1, \cdots, N_c, \quad (5e)$$

where $\boldsymbol{\mathcal{F}}_r(\cdot)$ and $\boldsymbol{\mathcal{F}}_\rho^i(\cdot)$ are forward kinematics functions to compute $\boldsymbol{r}$ and $\boldsymbol{\rho}^i$, $\boldsymbol{S}^i$ is a binary variable representing the predefined contact sequence, $\boldsymbol{D}$ is the maximum allowable
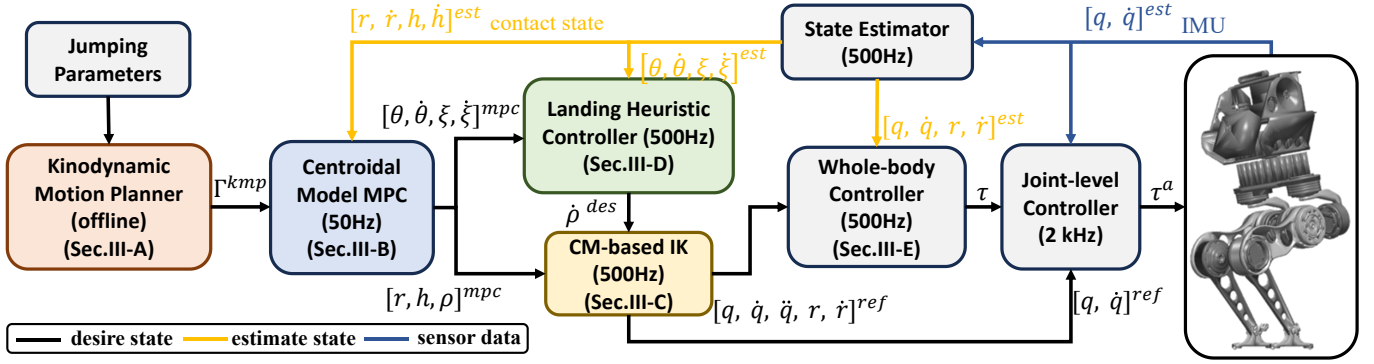
Fig. 4: **The CDM-MPC dynamic planning and control framework.** (i) The CDM-based kinodynamic motion planner produces the centroidal momentum reference trajectory. (ii) The real-time MPC controller provides accurate trajectory tracking and fast replanning under disturbances. (iii) The centroidal moment-based IK solves whole-body trajectory without simplifying leg dynamics. (iv) The landing heuristic controller guarantees robust landing stabilization.

distance between consecutive timesteps, $\xi_{\min}$ is the minimal distance between r and ρ^i. Of note, Eq. (5d) limits the stepsize of each contact point while avoiding the movement of the contact point which is in contact. Eq. (5e) is an approximate collision avoidance constraint to keep the minimal safety distance without adding computational complexity.

The dynamics constraints include

$$\dot{h}_{[k]} = \sum_{i=1}^{N_c} (\rho^i_{[k]} - r_{[k]}) \times f^i_{\rho[k]}, \tag{6a}$$

$$\ddot{r}_{[k]} = \frac{1}{m} \sum_{i=1}^{N_c} f^i_{\rho[k]}, \tag{6b}$$

$$f^i_{\rho[k]} = \sum_{j=1}^{N_d} \beta^{ij}_{[k]} v^{ij}_{[k]}, i = 1, \cdots, N_c, j = 1, \cdots, N_d, \tag{6c}$$

$$\beta^{ij}_{[k]} \geqslant 0, \qquad i = 1, \cdots, N_c, j = 1, \cdots, N_d, \tag{6d}$$

$$\left\| f^i_{\rho[k]} \right\| \leqslant f_{\max} S^i_{[k]}, \quad i = 1, \cdots, N_c, \tag{6e}$$

where the unit vector v^{ij} denotes the jth edge of the friction pyramid at the ith contact point, $f_{\max}$ is the maximum contact force for each contact point. The complete formulation is as follows:

$$\min_{\Gamma^{\text{kmp}}} \sum_{k=1}^{N} \left(\sum_{i=1}^{N_c} \left\| f^i_{\rho[k]} \right\|^2 + \left\| \ddot{r}_{[k]} \right\|^2 + \left\| \dot{h}_{[k]} \right\|^2 + \left\| \dot{q}_{[k]} \right\|^2 + \left\| q_{[k]} - q^* \right\|^2 \right) \tag{7}$$

$s.t.$ Eqs. (5) and (6)

where the object function includes (i) a smoothness cost $\left\| \ddot{r}_{[k]} \right\|^2 + \left\| \dot{h}_{[k]} \right\|^2$ which penalizes for the rate of change in centroidal momentum; (ii) a contact force cost $\sum_{i=1}^{N_c} \left\| f^i_{\rho[k]} \right\|^2$ that minimizes the contact force; and (iii) a tracking cost $\left\| q_{[k]} - q^* \right\|^2 + \left\| \dot{q}_{[k]} \right\|^2$ to regularize the solution around the nominal posture q^*.

B. Real-time MPC Controller

To achieve real-time replanning after the launching phase, a MPC controller is designed by simplifying the trajectory planning schema Eq. (7) for computing efficiency improvement. Diverging from other MPC formulations [11, 26–28], our MPC controller design accounts for non-constant leg

inertia throughout the flight phase and accurately manipulates the body pose through actuated CCRBI regulation [34].

1) Centroidal Inertia Decomposition: In centroidal dynamics, the average spatial velocity is defined as [33]:

$$h = {}^{\bar{\omega}}I_h(q)\bar{\omega}, \tag{8}$$

where $\bar{\omega}$ is the average angular velocity of CoM, ${}^{\bar{\omega}}I_h(q) \in \mathbb{R}^{3\times3}$ is the rotational part of the CCRBI, defined as the sum of all the link inertia projected to the CoM, and can be divided into the floating-base and actuated parts:

$$\begin{aligned}
{}^{\bar{\omega}}I_h(q) &= X_h(q)^\mathsf{T} I X_h(q) \\
&= X_h^\mathsf{T} \begin{bmatrix} I_1 & & & \\ & 0 & & \\ & & \ddots & \\ & & & 0 \end{bmatrix} X_h + X_h^\mathsf{T} \begin{bmatrix} 0 & & & \\ & I_2 & & \\ & & \ddots & \\ & & & I_{N_L} \end{bmatrix} X_h \\
&= {}^{\bar{\omega}}I_h^f + {}^{\bar{\omega}}I_h^a(q),
\end{aligned} \tag{9}$$

where X_h projects motion vectors from centroidal coordinate frame to link coordinate frames, I_i is the spatial inertia for link i with $i = 1$ refers to the body link, N_L is the number of links. Within CCRBI, ${}^{\bar{\omega}}I_h^f$ is unactuated and invariant whereas ${}^{\bar{\omega}}I_h^a(q)$ changes according to the leg configuration.

Combining Eq. (9) with Eq. (8), we obtain:

$$h = {}^{\bar{\omega}}I_h(q)\bar{\omega} = ({}^{\bar{\omega}}I_h^f + {}^{\bar{\omega}}I_h^a(q))\bar{\omega}, \tag{10}$$

which indicates that although h remains conserved in the flight phase, the average angular velocity $\bar{\omega}$ can be regulated by manipulating the actuated part ${}^{\bar{\omega}}I_h^a(q)$ of the CCRBI, thereby influencing the overall body posture.

2) MPC Formulation: ${}^{\bar{\omega}}I_h^a(q)$ has a complex nonlinear relationship with the robot configuration q. For simplification, we consider the actuated leg as a constant density cuboid with variable dimension, resulting in a diagonal inertia matrix that can be estimated through the leg length ξ, defined as

$$\xi = r - \rho. \tag{11}$$

Furthermore, we define the Euler angle vector between the CoM coordinate frame and the leg length vector ξ as θ, expressed in the X-Y-Z convention. Since the floating-base part maintains a constant inertia ${}^{\bar{\omega}}I_h^f$ during the motion, we have:

$${}^{\bar{\omega}}I_h = {}^{\bar{\omega}}I_h^f + {}^{\bar{\omega}}I_h^a \approx {}^{\bar{\omega}}I_h^f + R(\theta)^\mathsf{T} \bar{I}_\xi R(\theta), \tag{12}$$

where

$$\bar{\boldsymbol{I}}_\xi = \begin{bmatrix} k_{\xi 1} & 0 & 0 \\ 0 & k_{\xi 2} & 0 \\ 0 & 0 & k_{\xi 3} \end{bmatrix} \begin{bmatrix} \|\xi\|^2 & 0 & 0 \\ 0 & \|\xi\|^2 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (13)$$

and $\boldsymbol{R}(\cdot)$ represents the transformation from Euler angles to a standard rotation matrix, $\boldsymbol{k}_\xi = diag(k_{\xi 1}, k_{\xi 2}, k_{\xi 3})$ is a diagonal matrix that can be calibrated with the multi-body model of actuated leg.

Define the integral of $\bar{\boldsymbol{\omega}}$ with time as equimomental ellipsoid orientation $\boldsymbol{\mathcal{L}} \in \mathbb{R}^{3 \times 1}$:

$$\boldsymbol{\mathcal{L}}_{[t_1]} - \boldsymbol{\mathcal{L}}_{[t_2]} = \int_{t_1}^{t_2} \bar{\boldsymbol{\omega}}\, dt = \int_{t_1}^{t_2} ({}^{\bar{\omega}}\boldsymbol{I}_h^f + {}^{\bar{\omega}}\boldsymbol{I}_h^a)^{-1} \boldsymbol{h}\, dt. \quad (14)$$

Discretize Eq. (14) at k^{th} time interval and utilize approximated inertia Eq. (12), we can have the following equality constraint:

$$\boldsymbol{\mathcal{L}}_{[k]} = \sum_{i=1}^{k} \left({}^{\bar{\omega}}\boldsymbol{I}_h^f + {}^{\xi}\boldsymbol{R}(\boldsymbol{\xi})^\top \bar{\boldsymbol{I}}_\xi\, {}^{\xi}\boldsymbol{R}(\boldsymbol{\xi})\right)^{-1} \boldsymbol{h}_{[k]} + \boldsymbol{\mathcal{L}}_{[0]}. \quad (15)$$

Choosing the set of decision variables as:

$$\boldsymbol{\Gamma}^{\text{mpc}} = \{\boldsymbol{r}_{[k]}, \dot{\boldsymbol{r}}_{[k]}, \ddot{\boldsymbol{r}}_{[k]}, \boldsymbol{\mathcal{L}}_{[k]}, \boldsymbol{h}_{[k]}, \dot{\boldsymbol{h}}_{[k]}, \boldsymbol{f}_{\rho[k]}^i, \boldsymbol{\rho}_{[k]}^i, \beta_{[k]}^{ij},$$
$$\forall i = 1, \cdots, N_c, j = 1, \cdots, N_d, k = 1, \cdots, N\} \in \mathbb{R}^{58 \times N}, \quad (16)$$

we formulate the real-time MPC as follows:

$$\min_{\boldsymbol{\Gamma}^{\text{mpc}}} \quad \sum_{k=1}^{N} \left(\sum_{i=1}^{N_c} \left\|\boldsymbol{f}_{\rho[k]}^i\right\|^2 + \left\|\ddot{\boldsymbol{r}}_{[k]}\right\|^2 + \left\|\dot{\boldsymbol{h}}_{[k]}\right\|^2\right) + \left\|\boldsymbol{\mathcal{L}}_{[N]} - \boldsymbol{\mathcal{L}}_{[N]}^*\right\|^2 \quad (17)$$
$$\text{s.t.} \quad \text{Eqs. (5d), (5e), (6a) to (6e) and (15)}$$

This formulation utilizes the feedback signal from the state estimator as the initial value. While similar to the trajectory optimization problem Eq. (7), it eliminates $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$ in decision variables, along with the complex nonlinear constraints Eqs. (5a) to (5c). These modifications lead to a significantly faster convergence speed. $\boldsymbol{\mathcal{L}}_{[0]}$ and $\boldsymbol{\mathcal{L}}_{[N]}^*$ are the manually specified equimomental ellipsoid orientation at the first and last time intervals.

C. Centroidal Momentum-based IK

To accurately track the optimized centroidal momentum and equimomental ellipsoid orientation obtained from the MPC, we designed a centroidal momentum-based IK method. A primary challenge in mapping the centroidal trajectory is the angular velocity of the frame attached to the CoM can not be integrated due to the non-constant nature of CCRBI as introduced before [5].

1) Instantaneous Mapping Method: To address this issue, our initial approach first maps the centroidal momentum to general velocity $\dot{\boldsymbol{q}}$ based on the previous general position $\boldsymbol{q}^{\text{pre}}$, and then integrates $\dot{\boldsymbol{q}}$ to obtain new general position $\boldsymbol{q}$, resulting in an optimization problem in each control cycle:

$$\min_{\boldsymbol{q}, \dot{\boldsymbol{q}}} \boldsymbol{W}_{cm} \|\boldsymbol{H}^{\text{mpc}} - \boldsymbol{A}(\boldsymbol{q})\dot{\boldsymbol{q}}\| + \boldsymbol{W}_{\dot{q}} \|\dot{\boldsymbol{q}}\| + \boldsymbol{W}_q \|\boldsymbol{q} - \boldsymbol{q}^*\|$$
$$\text{s.t.} \quad \boldsymbol{q} = \boldsymbol{q}^{\text{pre}} + \dot{\boldsymbol{q}}\Delta t$$
$$\boldsymbol{r}^{\text{mpc}} = \boldsymbol{\mathcal{F}}_r(\boldsymbol{q}) \quad (18)$$
$$\boldsymbol{\rho}^{\text{mpc}} = \boldsymbol{\mathcal{F}}_\rho(\boldsymbol{q})$$

where the superscript $[\cdot]^{\text{mpc}}$ is the optimized value of a variable from the MPC, $\boldsymbol{H}$ is defined in Eq. (2), $\boldsymbol{q}^*$ is the nominal

posture of the robot. $\boldsymbol{W}_{cm}$, $\boldsymbol{W}_{\dot{q}}$, and $\boldsymbol{W}_q$ are weighting matrices.

2) Simplified Mapping Method: However, simultaneously solving for $\boldsymbol{q}$ and $\dot{\boldsymbol{q}}$ leads to slow computational speeds. Therefore, we simplify the problem by choosing $\dot{\boldsymbol{q}}$ as the only decision variable and solve $\boldsymbol{q}$ through integration.

$$\min_{\dot{\boldsymbol{q}}} \quad \boldsymbol{W}_{cm} \|\boldsymbol{H}^{\text{mpc}} + \boldsymbol{H}^{\text{aug}} - \boldsymbol{A}(\boldsymbol{q})\dot{\boldsymbol{q}}\| + \boldsymbol{W}_{\dot{q}} \|\dot{\boldsymbol{q}}\|$$
$$\text{s.t.} \quad \boldsymbol{J}\dot{\boldsymbol{q}} = \dot{\boldsymbol{\rho}}^{\text{des}} + \dot{\boldsymbol{\rho}}^{\text{aug}} \quad (19)$$

The cost function of this simplified optimization problem accounts for the centroidal momentum and generalized velocity while the constraint limits the foot endpoint velocity. $\dot{\boldsymbol{\rho}}^{\text{des}}$ is the desired foot endpoint speed acquired from the landing heuristic controller (to be introduced next),

$$\dot{\boldsymbol{\rho}}^{\text{aug}} = \boldsymbol{k}_\rho \left(\boldsymbol{\rho}^{\text{mpc}} - \boldsymbol{\rho}^{\text{fk}}\right) \quad (20)$$

is a feedback augment to improve the foot endpoint trajectory tracking where the superscript $[\cdot]^{\text{fk}}$ denotes a variable calculated through forward kinematics.

$$\boldsymbol{H}^{\text{aug}} = [\boldsymbol{k}_r(\boldsymbol{r}^{\text{mpc}\top} - \boldsymbol{r}^{\text{fk}\top}), \boldsymbol{k}_R(\boldsymbol{R}^{\text{fk}\top}\boldsymbol{R}^* - \boldsymbol{R}^{*\top}\boldsymbol{R}^{\text{fk}})_\vee]^\top \quad (21)$$

is a feedback augment to improve the CoM and torso orientation trajectory tracking, where $\boldsymbol{R}^*$ stands for the nominal torso orientation expressed as standard rotation matrix, and $(\cdot)_\vee$ is the mapping from SO(3) to $\mathbb{R}^3$, $\boldsymbol{k}_\rho$, $\boldsymbol{k}_r$, $\boldsymbol{k}_R$ are the feedback gains.

D. Landing Heuristic Controller

The landing angle $\boldsymbol{\theta}$ and leg length $\boldsymbol{\xi}$ are critical for landing stability, and thus are controlled by the PD feedback controllers. As the CoM is uncontrollable in the air, we control the foot endpoint position to achieve the desired landing angle and leg length, the control law is designed as follows:

$$\boldsymbol{\xi}^{\text{des}} = \boldsymbol{\xi}^{\text{mpc}} + \boldsymbol{k}_p^\xi \left(\boldsymbol{\xi}^{\text{mpc}} - \boldsymbol{\xi}^{\text{est}}\right) + \boldsymbol{k}_d^\xi \left(\dot{\boldsymbol{\xi}}^{\text{mpc}} - \dot{\boldsymbol{\xi}}^{\text{est}}\right)$$
$$\boldsymbol{\theta}^{\text{des}} = \boldsymbol{\theta}^{\text{mpc}} + \boldsymbol{k}_p^\theta \left(\boldsymbol{\theta}^{\text{mpc}} - \boldsymbol{\theta}^{\text{est}}\right) + \boldsymbol{k}_d^\theta \left(\dot{\boldsymbol{\theta}}^{\text{mpc}} - \dot{\boldsymbol{\theta}}^{\text{est}}\right) \quad (22)$$
$$\boldsymbol{\rho}^{\text{des}} = \boldsymbol{r}^{\text{est}} + \|\boldsymbol{\xi}^{\text{des}}\| / \|\boldsymbol{\theta}^{\text{des}}\| \boldsymbol{\theta}^{\text{des}}$$

where the superscript $[\cdot]^{\text{est}}$ is the estimated value of a variable from the state estimator, the optimized leg length $\boldsymbol{\xi}^{\text{mpc}}$ is calculated with MPC decision variable set $\boldsymbol{\Gamma}^{\text{mpc}}$ and Eq. (11), which is then utilized to calculate the optimized landing angle $\boldsymbol{\theta}^{\text{mpc}}$, $\boldsymbol{k}_p^\xi$, $\boldsymbol{k}_d^\xi$, $\boldsymbol{k}_p^\theta$, $\boldsymbol{k}_d^\theta$ are feedback gains.

E. Whole-body Controller

Our study employs a typical weighted WBC controller to regulate the movements of the entire robot [35]. Distinct weights are set for tracking the trunk and foot trajectories of the robot. The robot's reference positions and velocity trajectories are obtained through the centroidal momentum-based IK, while the reference accelerations are derived by differentiating velocities. Additionally, we integrate a constraint in line with the law of conservation of momentum [36], aiming to achieve improved tracking of the centroidal angular momentum. This integration enhances the robot's dynamic balance and stability during complex maneuvers.

IV. Verification

To rigorously evaluate the efficacy of the proposed CDM-MPC framework, simulations and experiments were executed using the KUAVO bipedal robot platform. The dynamic simulator was developed in the Drake simulation environment [37] and utilized the KUAVO's detailed multibody model. Besides incorporating the mass and inertia parameters reflecting the physical robot's specifications, the simulator also accounted for the torque saturation and motion range of each motor, control frequency, and measurement noise, as described in Sec. II-A. Physical validation trials were carried out on the physical KUAVO platform. Main control parameters are summarized in Tab. II.

Four test cases were designed as follows: (i) **Case 1:** we compared the disturbance rejection performance of the proposed CDM-MPC method with a baseline SRBM-MPC method during in-place jumping; (ii) **Case 2:** we studied the robustness of proposed landing controller with variable forward jumping velocities; (iii) **Case 3:** we validated the pro-
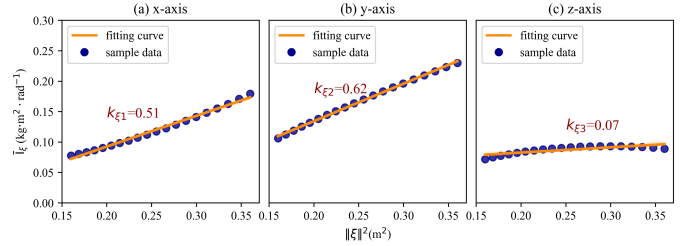

Fig. 6: **Inertia parameter calibration results.** The x and y axes components of $\bar{\boldsymbol{I}}_\xi$ has linear relationship with $\|\xi\|^2$, while the z axis component of $\bar{\boldsymbol{I}}_\xi$ is invariant.

posed framework on the physical KUAVO robot in a jumping experiment; and (iv) **Case 4:** we explored the versatility of the framework by applying it to walking locomotion.

A. Inertia Parameter Calibration

To calibrate the inertia parameter matrix $\boldsymbol{k}_\xi$ in Eq. (13), 20 robot configuration data points were collected in the simulator with different leg lengths from 0.4 to 0.6 m. Utilizing the calculation of actuated part $^{\bar{\omega}}\boldsymbol{I}_h^a(\boldsymbol{q})$ of the CCRBI in Eq. (9) as ground truth, Eqs. (12) and (13) are utilized to calculate the calibration matrix $\boldsymbol{k}_\xi$. The calibration results are illustrated in Fig. 6, where the component of $\bar{\boldsymbol{I}}_\xi$ in x and y axes have an obvious linear relationship with $\|\xi\|^2$, while the component in z axis is almost invariant to the changes of leg length. Subsequently, the parameter matrix is calculated from the sampled data by leveraging a linear regression method. The calibrated parameters are included in Tab. II.

B. In-Place Jumping Simulation

To verify the effectiveness of the CDM-MPC method in rejecting disturbances, we compared it with a baseline SRBM-MPC method. The simulation results are shown in Fig. 5. Both methods employed the same optimized in-place jumping trajectory from the kinodynamic motion planning. At the

TABLE II: **Planing & Control Parameters**

Group	Parameters	Value
KMP Eq. (7)	Step time	$50\ ms$
	Horizon N	$2\ s$
	f_{max}	$2000N$
	ξ_{min}	$0.4m$
	Solve time	$1 \sim 10\ s$
Real-time MPC Eq. (17)	Step time	$10\ ms$
	Horizon N	$1\ s$
	f_{max}	$2000N$
	ξ_{min}	$0.4m$
	$k_{\xi 1}, k_{\xi 2}, k_{\xi 3}$	$0.51, 0.62, 0.07$
	Solve time	$\approx 20\ ms$
CMIK Eq. (18)	Solve time	$\approx 2.5\ ms$
CMIK Eq. (19)	Solve time	$\approx 1\ ms$
	$\boldsymbol{k}_\rho, \boldsymbol{k}_r, \boldsymbol{k}_R$	$[10.0, 10.0, 1.0] * \mathbf{1}_3$
Landing Eq. (22)	$\boldsymbol{k}_p^\xi, \boldsymbol{k}_d^\xi$	$[0.1, 0.03] * \mathbf{1}_3$
	$\boldsymbol{k}_p^\theta, \boldsymbol{k}_d^\theta$	$[0.5, 0.2] * \mathbf{1}_3$
	Solve time	$\leqslant 0.1\ ms$

(a) SRBM-MPC

(b) SRBM-MPC leg configuration

(c) SRBM-MPC jumping data

(d) CDM-MPC

(e) CDM-MPC leg configuration
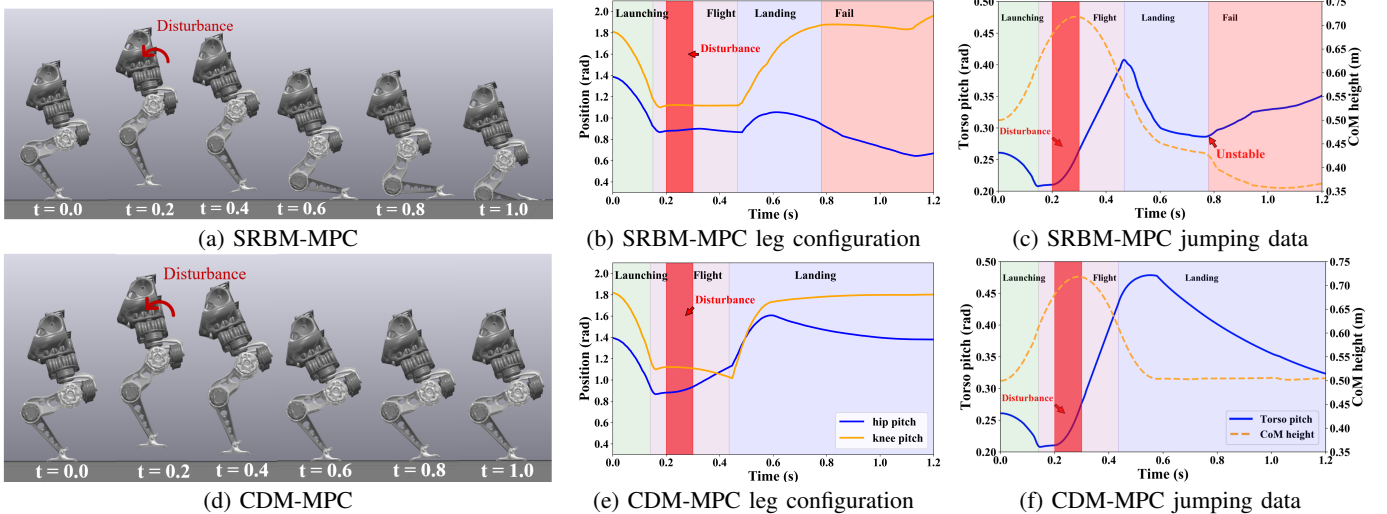

(f) CDM-MPC jumping data

Fig. 5: **Case1 (Simulation): Disturbance rejection performance study during in-place jumping.** (a) The SRBM-MPC method cannot maintain the robot's stability when subjected to disturbance torque, leading to its collapse. (c) Conversely, the CDM-MPC method preserves the robot's stability throughout the entire flight phase, culminating in a successful landing. (b)(e) depict the joint configuration of hip pitch and knee pitch joints while (c)(f) depict the pitch angle of the torso and the CoM height for both methods, respectively.
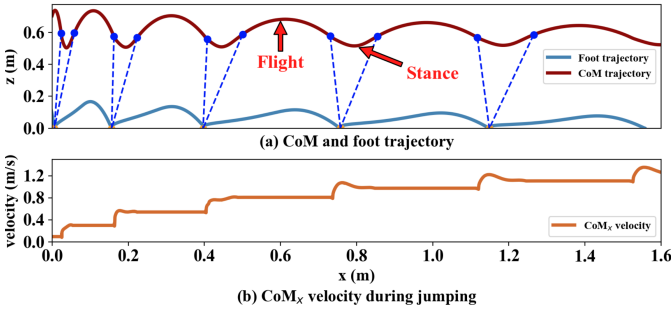
Fig. 7: **Case2 (Simulation): Continuous forward jumping with increasing jumping velocities.** (a) CoM and foot trajectory of the robot during the process. (b) Corresponding estimated velocity.

apex of the robot's flying phase, a perturbing torque with a magnitude of $20\ Nm$ with a duration of $0.1\ s$ was applied at the robot's torso as an unknown disturbance.

As shown in Fig. 5a, with the SRBM-MPC method, the body of the robot gradually tilts after the injected disturbance during the flight phase, resulting in an unstable landing. In contrast, as shown in Fig. 5d, the angular momentum-based control framework, which takes into account leg inertia changes, facilitates refined control over the robot's posture. During the flight phase, both body and leg postures are dynamically adjusted in accordance with a trajectory generated by the CDM-MPC and the momentum-based IK. This approach improves the in-flight disturbance-rejection capability of the robot, ensuring a stable landing.

C. Forward Jumping Simulation

In the forward jumping simulation, a range of initial jump trajectories corresponding to velocities of $0.5, 1.0, 1.5, 2.0\ m/s$ were provided to serve as starting points for optimization. The CDM-MPC algorithm is designed to automatically select the trajectory that aligns most closely with the commanded velocity as its initial guess, thereby improving the efficiency of the trajectory planning process. The robot's key performance metrics and jumping velocity commands throughout the jumping sequence are illustrated in Fig. 7. The corresponding visualization is shown in Fig. 1.

The results demonstrated (i) the robustness of the landing controller which makes subtle adjustments to regulate the landing angle as the robot descends (ii) the capability of the CDM-MPC framework to adaptively generate a range of jump trajectories from a pool of precomputed offline samples, facilitating smooth transitions between various jumping velocities.

D. Physical Jumping Experiment

In the experiment, we validate our proposed jumping framework using KUAVO, our full-sized humanoid robot platform, as detailed in Sec. II. As shown in Fig. 8b, the IMU-measured acceleration curve indicates significant impacts on the robot's torso during both the launching and landing phases. The contact forces are derived directly from joint current readings, which are significantly affected by joint current noise at the moment of landing, resulting in notable spikes. Despite these challenges, the robot's stable landing performance showcases
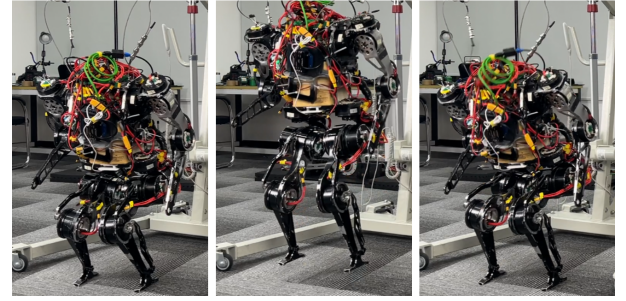
(a) Key frames in experiment.
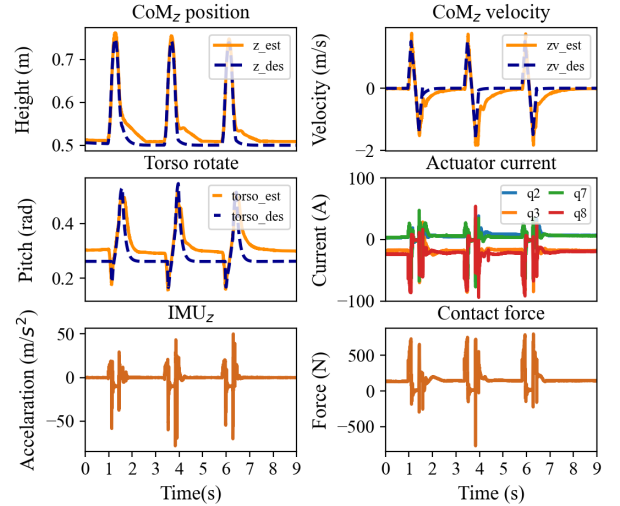
(b) Experiment results

Fig. 8: **Case3 (Experiment): Continuous in-place jumping on physical KUAVO robot.** The KUAVO robot executes three vertical jumps of approximately $0.28\ m$ and lands stably with the proposed jump control framework.
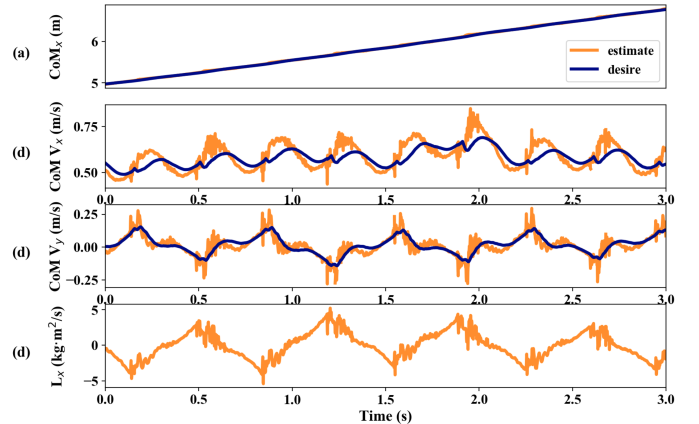

Fig. 9: **Case4 (Experiment): Apply the proposed framework for walking control.** The KUAVO robot achieves stable walking performance with a speed of $0.6\ m/s$.

our control method's robustness to impacts and noise, highlighting our framework's effectiveness in physical humanoid robot jumping dynamics. The keyframes are shown in Fig. 8a.

E. Walking Experiment

As shown in Fig. 9, to demonstrate the generalizability of our framework, we applied it to walking control. The robot walks stably at $1.2\ m/s$ in simulation and $0.6\ m/s$ on

hardware. This verifies that the framework can be extended to other locomotion modes.

V. Conclusion

In this paper, we have presented an integrated dynamic planning and control framework (CDM-MPC) for the jumping motion of bipedal robots. This framework considers centroidal momentum in both dynamics planning of the launching phase and online tracking control of the flight phase, and it integrates a robust landing to ensure stability. Altogether, the proposed framework enables agile and continuous jumping motions on full-sized bipedal robots. We validated the effectiveness of this framework based on a novel full-sized bipedal robot KUAVO in both realistic simulations and real-world experiments and confirmed its applicability to other locomotion modes, such as walking. Future directions include generalizing the framework for unified walking, running, and jumping control with smooth transitional behavior and integrating reinforcement learning-based methods for performance and robustness improvement.

References

[1] M. Chignoli, D. Kim, E. Stanger-Jones, and S. Kim, "The mit humanoid robot: Design, motion planning, and control for acrobatic behaviors," in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2021.

[2] T. Zhu, J. Hooks, and D. Hong, "Design, modeling, and analysis of a liquid cooled proprioceptive actuator for legged robots," in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 2019.

[3] H. Dai, A. Valenzuela, and R. Tedrake, "Whole-body motion planning with centroidal dynamics and full kinematics," in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2014.

[4] A. Herzog, S. Schaal, and L. Righetti, "Structured contact force optimization for kino-dynamic motion generation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016.

[5] A. Saccon, S. Traversaro, F. Nori, and H. Nijmeijer, "On centroidal dynamics and integrability of average angular velocity," *IEEE Robotics and Automation Letters (RA-L)*, vol. 2, no. 2, pp. 943–950, 2017.

[6] J. Zhang, J. Shen, Y. Liu, and D. W. Hong, "Design of a jumping control framework with heuristic landing for bipedal robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023.

[7] 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 (T-RO)*, vol. 40, pp. 43–63, 2024.

[8] H. Qi, X. Chen, Z. Yu, G. Huang, Y. Liu, L. Meng, and Q. Huang, "Vertical jump of a humanoid robot with cop-guided angular momentum control and impact absorption," *IEEE Transactions on Robotics (T-RO)*, vol. 39, no. 4, pp. 3154–3166, 2023.

[9] Z. Li, X. B. Peng, P. Abbeel, S. Levine, G. Berseth, and K. Sreenath, "Robust and versatile bipedal jumping control through reinforcement learning," *arXiv preprint arXiv:2302.09450*, 2023.

[10] G. Mesesan, R. Schuller, J. Englsberger, C. Ott, and A. Albu-Schäffer, "Unified motion planner for walking, running, and jumping using the three-dimensional divergent component of motion," *IEEE Transactions on Robotics (T-RO)*, 2023.

[11] A. SaLoutos, E. Stanger-Joncs, Y. Ding, M. Chignoli, and S. Kim, "Design and development of the mit humanoid: A dynamic and robust research platform," in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2023.

[12] K. Zhou, P. Wu, Y. Su, H. Gao, J. Ma, H. Liu, and C. Liu, "Aspire: An informative trajectory planner with mutual information approximation for target search and tracking," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[13] H. Gao, P. Wu, Y. Su, K. Zhou, J. Ma, H. Liu, and C. Liu, "Probabilistic visibility-aware trajectory planning for target tracking in cluttered environments," in *American Control Conference (ACC)*, 2024.

[14] Y. Su, J. Zhang, Z. Jiao, H. Li, M. Wang, and H. Liu, "Real-time dynamic-consistent motion planning for over-actuated uavs," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[15] S. Kuindersma, R. Deits, M. Fallon, A. Valenzuela, H. Dai, F. Permenter, T. Koolen, P. Marion, and R. Tedrake, "Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot," *Autonomous robots*, vol. 40, no. 3, pp. 429–455, 2016.

[16] J. Luo, Y. Su, L. Ruan, Y. Zhao, D. Kim, L. Sentis, and C. Fu, "Robust bipedal locomotion based on a hierarchical control structure," *Robotica*, vol. 37, no. 10, pp. 1750–1767, 2019.

[17] G. Bledt, M. J. Powell, B. Katz, J. Di Carlo, P. M. Wensing, and S. Kim, "Mit cheetah 3: Design and control of a robust, dynamic quadruped robot," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[18] D. Kim, J. Di Carlo, B. Katz, G. Bledt, and S. Kim, "Highly dynamic quadruped locomotion via whole-body impulse control and model predictive control," *arXiv preprint arXiv:1909.06586*, 2019.

[19] 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.

[20] Z. Zhou, B. Wingo, N. Boyd, S. Hutchinson, and Y. Zhao, "Momentum-aware trajectory optimization and control for agile quadrupedal locomotion," *IEEE Robotics and Automation Letters (RA-L)*, vol. 7, no. 3, pp. 7755–7762, 2022.

[21] M. Wang, Y. Su, H. Liu, and Y. Xu, "Walkingbot: Modular interactive legged robot with automated structure sensing and motion planning," in *IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, 2020.

[22] X. Lin, J. Zhang, J. Shen, G. Fernandez, and D. W. Hong, "Optimization based motion planning for multi-limbed vertical climbing robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019.

[23] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture point: A step toward humanoid push recovery," in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2006.

[24] N. Scianca, D. De Simone, L. Lanari, and G. Oriolo, "Mpc for humanoid gait generation: Stability and feasibility," *IEEE Transactions on Robotics (T-RO)*, vol. 36, no. 4, pp. 1171–1188, 2020.

[25] P. Kryczka, P. Kormushev, N. G. Tsagarakis, and D. G. Caldwell, "Online regeneration of bipedal walking gait pattern optimizing footstep placement and timing," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015.

[26] R. Budhiraja, J. Carpentier, and N. Mansard, "Dynamics consensus between centroidal and whole-body models for locomotion of legged robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019.

[27] B. Ponton, M. Khadiv, A. Meduri, and L. Righetti, "Efficient multicontact pattern generation with sequential convex approximations of the centroidal dynamics," *IEEE Transactions on Robotics (T-RO)*, vol. 37, no. 5, pp. 1661–1679, 2021.

[28] T. Kwon, Y. Lee, and M. Van De Panne, "Fast and flexible multilegged locomotion using learned centroidal dynamics," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 4, pp. 46–1, 2020.

[29] R. Grandia, F. Jenelten, S. Yang, F. Farshidian, and M. Hutter, "Perceptive locomotion through nonlinear model-predictive control," *IEEE Transactions on Robotics (T-RO)*, 2023.

[30] 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 (T-RO)*, vol. 39, no. 2, pp. 905–922, 2023.

[31] J. Luo, Z. Gong, Y. Su, L. Ruan, Y. Zhao, H. H. Asada, and C. Fu, "Modeling and balance control of supernumerary robotic limb for overhead tasks," *IEEE Robotics and Automation Letters (RA-L)*, vol. 6, no. 2, pp. 4125–4132, 2021.

[32] P. E. Gill, W. Murray, and M. A. Saunders, "Snopt: An sqp algorithm for large-scale constrained optimization," *SIAM review*, vol. 47, no. 1, pp. 99–131, 2005.

[33] D. E. Orin, A. Goswami, and S.-H. Lee, "Centroidal dynamics of a humanoid robot," *Autonomous robots*, vol. 35, no. 2-3, pp. 161–176, 2013.

[34] S.-H. Lee and A. Goswami, "Reaction mass pendulum (rmp): An explicit model for centroidal angular momentum of humanoid robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2007.

[35] S. Feng, E. Whitman, X. Xinjilefu, and C. G. Atkeson, "Optimization based full body control for the atlas robot," in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2014.

[36] I. Vatavuk and Z. Kovačić, "Precise jump planning using centroidal dynamics based bilevel optimization," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[37] R. Tedrake and the Drake Development Team, "Drake: Model-based design and verification for robotics," 2019.