Impact-Aware Online Motion Planning for Fully-Actuated Bipedal Robot Walking

Yuan Gao, Xingye Da, and Yan Gu

***Abstract*— The ability to track a general walking path with specific timing is crucial to the operational safety and reliability of bipedal robots for avoiding dynamic obstacles, such as pedestrians, in complex environments. This paper introduces a motion planner that generates the desired full-body motion in real-time for fully-actuated bipedal robotic walking. The key novelty of the proposed planner lies in its capability of online generation of impact-aware non-periodic motions along general walking paths that respect the discrete impact behaviors. The planner is derived based on full-order modeling of hybrid walking dynamics and the proposed construction of keyframe posture library. Simulation results of a three-dimensional bipedal robot are presented to confirm the effectiveness of the proposed planner in generating full-body motions for each walking step within 0.6 second.**

I. INTRODUCTION

Motion planning of legged robotic locomotion demands a high computational load due to the complexity of the associated robot dynamics. Thus, previous motion planning methods [1] utilized offline trajectory optimization [2], [3] to generate periodic walking patterns. Later on, the methods were extended to generate non-periodic walking motions to enable robot navigation in complex, static environments [4], [5]. One major limitation of offline planning is that it is not suitable for navigation in dynamic environments (e.g., crowded hallways with moving pedestrians). To address this issue, online planning is required.

To enable online planning, reduced-order models of legged locomotion [6], [7] have been used as the basis to reduce the computational load of motion planning [8]–[10]. Despite their advantages in helping to alleviate the planning burden, these reduced-order models fail to capture an essential behavior of legged locomotion, which is the swing-foot landing impact. The impact occurs when a robot's swing foot strikes the ground and causes a sudden jump in joint velocities. Without considering the impact in planning, the feasibility of the generated motions may not be guaranteed due to the discrepancy between the modeled and the actual dynamic behaviors of a walking robot [11], [12].

Previously, we have developed an impact-aware motion planner as part of a global-position tracking control framework to produce walking motions that respect landing impact dynamics. The overall control framework can achieve provably accurate tracking of non-periodic desired motions for planar fully-actuated walking [13], [14], 3-D fully-actuated walking [15], [16], and planar multi-domain walking [17]. Still, our previous planner has two main limitations: a) the planner is not suitable for online planning as one walking steps takes approximately 8 minutes to generate [16] and b) the planner is only capable of generating walking motions along straight paths instead of general paths.

Motivated by the current research needs, this study introduces an online, impact-aware motion planning method that respects the impact behaviors involved in legged locomotion (e.g., the impact dynamics and the desired impact timing). The input to the proposed planner is the desired footstep sequence consisting of a) the desired position and orientation of each footstep and b) the desired timestamps of swing-foot landings. The output of the planner is non-periodic full-body motions that agree with the desired footstep sequence and the impact dynamics. The key to the high efficiency of the proposed planner is a novel method, keyframe posture library, that helps to greatly reduce the computational load of motion planning.

The paper is organized as follows. Section II presents the full-order model of hybrid walking dynamics, which serves as the basis of the proposed motion planner, along with a brief explanation of our global-position tracking controller design [13], [14]. Section III proposes the construction of keyframe posture library together with the formulation of a set of optimization problems for creating the proposed motion planner. Section IV reports the simulation results.

II. MODELING AND CONTROL OF BIPEDAL ROBOTIC WALKING

This section presents a full-order model of hybrid bipedal walking dynamics and a global-position tracking control strategy.

As walking inherently involves both continuous dynamics and discrete behaviors, it is natural to model bipedal walking as a hybrid dynamical system. The following assumptions are considered in this study [11]:

- The walking surface is flat and horizontal.
- During continuous phases, the support foot remains a static, full contact with the walking surface.
- The impact is modeled as a rigid-body contact, which occurs within an infinitesimal period of time.

Based on these assumptions, the robot is fully actuated during continuous phases.

The generalized coordinates of the floating-base bipedal robot can be expressed as

$$[\mathbf{p}_b^T, \boldsymbol{\gamma}_b^T, q_1, ..., q_n]^T \in \mathcal{Q}, \tag{1}$$

Yuan Gao and Yan Gu are with the Department of Mechanical Engineering, University of Massachusetts Lowell, Lowell, MA 01854, U.S.A. Emails: yuan_gao@student.uml.edu, yan_gu@uml.edu. Xingye Da is a AI and Robotics Research and Development Engineer in Nvidia Coporation. Email: xda@nvidia.com.

where $\mathcal{Q} \subset \mathbb{R}^{n+6}$ is the configuration space, $\mathbf{p}_b := [x_b, y_b, z_b]^T \subset \mathbb{R}^3$ represents the position of the floating-base with respect to (w.r.t) the world coordinate frame, $\boldsymbol{\gamma}_b := [\phi_b, \theta_b, \psi_b]^T$ represents the pitch, roll, and yaw angles of the floating base w.r.t. the world coordinate frame, and $q_1, ..., q_n$ represent the robot's joint angles. The robot model used for simulation validation is ROBOTIS-OP3 [18] (Fig. 1), which has 20 (i.e., $n = 20$) independent joints.

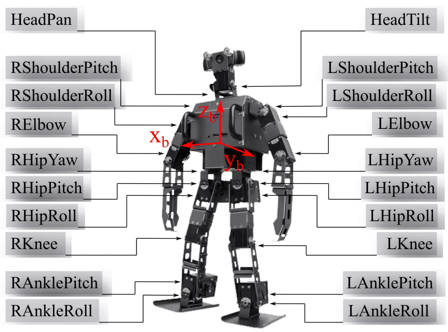

Fig. 1. An illustration of the revolute joints of a ROBOTIS-OP3 bipedal humanoid robot. The coordinate system of the robot's floating base is located at the center of the chest.

A. Continuous Dynamics

The continuous-phase equation of motion is obtained through Lagrange's method:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{B}\mathbf{u} + \mathbf{J}^T\mathbf{F}, \tag{2}$$

where $\mathbf{M}(\mathbf{q}) : \mathcal{Q} \to \mathbb{R}^{(n+6)\times(n+6)}$ is the inertia matrix, $\mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) : \mathcal{TQ} \to \mathbb{R}^{(n+6)}$ is the sum of Coriolis, centrifugal, and gravitational terms, $\mathbf{B} \subset \mathbb{R}^{(n+6)\times m}$ $(m = 20)$ is a joint-torque projection matrix. $\mathbf{u} \subset \mathbb{R}^m$ is the input vector, $\mathbf{F} \subset \mathbb{R}^6$ is the vector of the generalized external force caused by the contact between the support foot and the ground, and $\mathbf{J}(\mathbf{q}) : \mathcal{Q} \to \mathbb{R}^{6\times(n+6)}$ is the corresponding Jacobian matrix.

The holonomic constraints that the robot is subject to can be expressed as:

$$\mathbf{J}\ddot{\mathbf{q}} + \dot{\mathbf{J}}\dot{\mathbf{q}} = \mathbf{0}. \tag{3}$$

Combining Eqs. (2) and (3) yields the complete continuous dynamics, which can be expressed as:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \bar{\mathbf{c}}(\mathbf{q}, \dot{\mathbf{q}}) = \bar{\mathbf{B}}\mathbf{u}, \tag{4}$$

where $\bar{\mathbf{c}}(\mathbf{q}, \dot{\mathbf{q}}) := \mathbf{c} - \mathbf{J^T}(\mathbf{J}\mathbf{M^{-1}}\mathbf{J^T})^{-1}(\mathbf{J}\mathbf{M^{-1}}\mathbf{c} - \dot{\mathbf{J}}\dot{\mathbf{q}})$ and $\bar{\mathbf{B}}(\mathbf{q}) := \mathbf{B} - \mathbf{J^T}(\mathbf{J}\mathbf{M^{-1}}\mathbf{J^T})^{-1}\mathbf{J}\mathbf{M^{-1}}\mathbf{B}$. Details of the derivation can be found in [16].

B. Switching Surface

The switching surface that represents a foot-landing event can be defined as:

$$S_q(\mathbf{q}, \dot{\mathbf{q}}) := \{(\mathbf{q}, \dot{\mathbf{q}}) \in \mathcal{TQ} : z_{sw}(\mathbf{q}) = 0, \dot{z}_{sw}(\mathbf{q}, \dot{\mathbf{q}}) < 0\}, \tag{5}$$

where $z_{sw} : \mathcal{Q} \to \mathbb{R}$ represents the swing-foot height above the ground and $\dot{z}_{sw} < 0$ indicates that the swing foot is moving toward the ground.

C. Discrete Dynamics

Upon a swing-foot landing, an instantaneous rigid-body impact occurs. This impact does not cause discontinuities in joint positions, but joint velocities will experience a sudden jump. The joint velocities right after an impact can be described as:

$$\dot{\mathbf{q}}^+ = \mathbf{R}_{\dot{q}}(\mathbf{q}^-)\dot{\mathbf{q}}^-, \tag{6}$$

where $\dot{\mathbf{q}}^-$ and $\dot{\mathbf{q}}^+$ represent the the joint velocities right before and after the impact, respectively. The derivation of $\mathbf{R}_{\dot{q}} : \mathcal{Q} \to \mathbb{R}^{(n+6)\times(n+6)}$ can be found in [11].

D. Global-Position Tracking Control

This subsection briefly introduces a controller that be utilized in simulation to help validate our proposed online motion planner. Let $\mathbf{h}_c(\mathbf{q}) : Q \to Q_c \subset \mathbb{R}^n$ denote the variables of interest. Let $\mathbf{h}_d(t) : \mathbb{R}^+ \to \mathbb{R}^n$ denote the desired position trajectories of $\mathbf{h}_c(\mathbf{q})$, which are generated by the proposed motion planner. By defining the trajectory tracking errors as $\mathbf{h}(t, \mathbf{q}) := \mathbf{h}_c(\mathbf{q}) - \mathbf{h}_d(t)$, the control objective becomes to drive $\mathbf{h}$ to zero exponentially. With the output function $\mathbf{y}$ designed as $\mathbf{h}$, an input-output linearizing control law [19] is derived as

$$\mathbf{u} = (\frac{\partial \mathbf{h}}{\partial \mathbf{q}}\mathbf{M^{-1}}\bar{\mathbf{B}})^{-1}[(\frac{\partial \mathbf{h}}{\partial \mathbf{q}})\mathbf{M^{-1}}\bar{\mathbf{c}} + \mathbf{v} + \ddot{\mathbf{h}}_d] \tag{7}$$

with

$$\mathbf{v} = -\mathbf{K}_p\mathbf{y} - \mathbf{K}_d\dot{\mathbf{y}},$$

where $\mathbf{K}_p \in \mathbb{R}^{n\times n}$ and $\mathbf{K}_d \in \mathbb{R}^{n\times n}$ are both positive definite diagonal matrices. Then, the continuous-phase closed-loop dynamics in Eq. (4) become $\ddot{\mathbf{y}} = -\mathbf{K}_d\dot{\mathbf{y}} - \mathbf{K}_p\mathbf{y}$. The closed-loop tracking error dynamics can be expressed as:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} := \begin{bmatrix} \mathbf{0}_{n\times n} & \mathbf{I}_{n\times n} \\ -\mathbf{K}_p & -\mathbf{K}_d \end{bmatrix}\mathbf{x} & \text{if } (t, \mathbf{x}^-) \notin S(t, \mathbf{x}); \\ \mathbf{x}^+ = \Delta(t, \mathbf{x}^-) & \text{if } (t, \mathbf{x}^-) \in S(t, \mathbf{x}), \end{cases} \tag{8}$$

where $\mathbf{x} := \left[\mathbf{y}^T, \ \dot{\mathbf{y}}^T\right]^T \in \mathcal{X} \subset \mathbb{R}^{2n}$ and the expressions of $S : \mathbb{R}^+ \times \mathcal{X} \to \mathbb{R}^{2n-1}$ and $\Delta : \mathbb{R}^+ \times \mathcal{X} \to \mathcal{X}$ can be obtained from S_q and $\mathbf{R}_{\dot{q}}$. The stability analysis is based on the multiple Lyapunov functions [20]. The detailed proof will not shown here due to the space limits.

III. ONLINE IMPACT-AWARE FULL-BODY MOTION PLANNING

This section introduces our proposed impact-aware online motion planner that produces desired full-body motions along the given general walking path.

In this study, it is assumed that a higher-level planner has provided the desired footstep sequence that the robot should follow in order to walk along the given path. The desired footstep sequence includes the position and orientation of each footstep as well as the desired timestamps of foot placement. The remaining task of planning is then to generate impact-ware, full-body motion profiles in real-time that respect the desired footstep sequence and the impact dynamics, which is the focus of this study.

To guarantee the feasibility of the planned motion, necessary constraints should be enforced in motion planning, including the impact behaviors and the continuous-phase physical constraints. However, respecting these conditions is a major cause of the high computational burden of walking motion planning, mainly because walking dynamics, including both continuous-phase and impact dynamics, are nonlinear and high-dimensional.

To alleviate the computational load for online planning, we first decompose the complete planning task into four subtasks (Fig. 2) such that the impact-aware and the continuous-phase feasibility constraints can be handled separately. These four subtasks include: a) posture interpolation, b) computing keyframe postures through inverse kinematics (IK), c) pre- and post-impact velocity assignment to keyframe postures, and d) continuous-phase motion generation.

To further enhance the efficiency of planning, we construct an offline keyframe posture library to mitigate the computational load of computing keyframe posture associated with impact events. Also, we utilize a reduced-order model and a full-order kinematic model to efficiently generate continuous-phase motions.

By decomposing the planning task into smaller elements and utilizing both pre-computed results and reduced-order models, our planner is able to generate the impact-aware full-body motion profile of one walking step within 0.6 second, which is typically less than the duration of one walking step. Thus, by planning the motion one walking step ahead, the proposed motion planner allows a robot to move continuously without pausing.

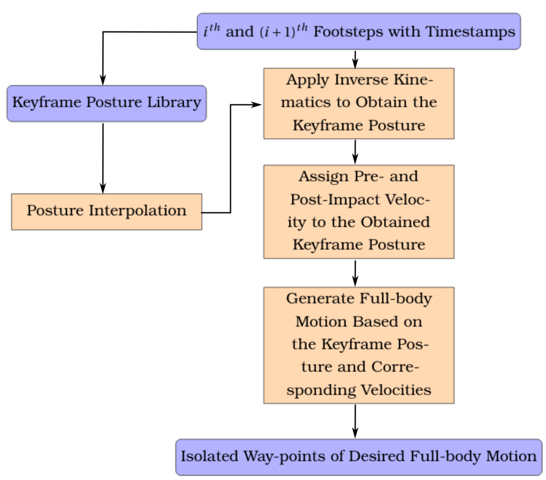

Fig. 2. Flowchart of the proposed planner. The orange boxes indicate the planning steps, and the blue boxes indicate the data involved in the online planning. The Keyframe Posture Library is highlighted with the blue box as it is pre-computed data.

A. Keyframe Posture and Keyframe Posture Library

In this study, we use the term, a keyframe posture, to define a pre-computed, kinematically feasible configuration of a walking robot at a swing-foot landing moment (Fig. 3). We denote the keyframe posture as $\mathbf{q}^k$, where the superscript "k" stands for "keyframe". With offline computing, we can construct a collection of keyframe postures that correspond to a set of relative displacements and orientations between two support feet. This collection is termed as *keyframe posture library* (*KPL*):

$$KPL = \{{}^i\mathbf{q}^k | i \in \mathbb{Z}^+, i \leq m\}, \tag{9}$$

where ${}^i\mathbf{q}^k \in \mathcal{Q}$ is the i^{th} keyframe posture within the library, $\mathbb{Z}^+$ is the set of positive integers, and the m is the total number of postures stored in this library.

The KPL provides a feasible initial guess for solving the IK associated with meeting the impact-aware constraint. The details are discussed next.

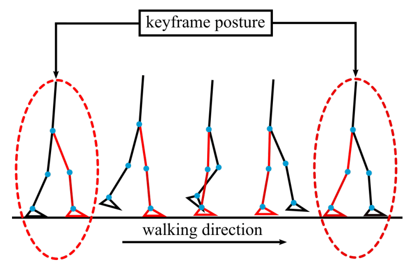

Fig. 3. An illustration of keyframe postures during walking.

B. Posture Interpolation

Let Γ denote the desired footstep sequence provided by a higher-level planner, which is mathematically expressed as

$$\Gamma = \{{}^i\gamma : i \in \mathbb{Z}^+, {}^i\gamma \subset \mathbb{R}^4\}, \tag{10}$$

where ${}^i\gamma := [{}^ix^\gamma, {}^iy^\gamma, {}^i\phi^\gamma, {}^i\tau^\gamma]^T$ represents the pose and timestamp of the i^{th} footstep in the given sequence. ${}^ix^\gamma$, ${}^iy^\gamma$, and ${}^i\phi^\gamma$ are the x, y and yaw angle of the i^{th} footstep with respect to the world coordinate frame. ${}^i\tau^\gamma$ is the timestamp of the footstep, indicating the desired moment for the robot to step onto that footstep. As this work addresses flat terrain and a full contact between the foot and the ground is assumed, the height, roll and pitch angle of the support foot are all 0. Thus, we only need to specify x, y and ϕ of the support foot.

Given two adjacent desired footsteps from Γ, the objective of this step is to obtain the desired posture ${}^i\mathbf{q}^* \in \mathbb{R}^{n+6}$ that are compatible with the two footsteps. One can perform IK to solve this problem. However, this IK problem is nonlinear, non-square and has infinitely many solutions, among which many can be infeasible. In order to get a feasible solution efficiently, we can exploit the pre-computed feasible postures in the proposed *KPL* as explained next:

- Given two adjacent footsteps, search for the two postures within the *KPL* that correspond to two footsteps closest to the given pair in terms of the relative displacement and orientation. Let these two postures be ${}^n\mathbf{q}^k$ and ${}^m\mathbf{q}^k$.
- Compute the initial guess of the IK by $\mathbf{q}_0 = ({}^n\mathbf{q}^k + {}^m\mathbf{q}^k)/2$.
- Perform the IK to obtain the feasible posture ${}^i\mathbf{q}^*$.

As the initial guess of the IK (i.e., $\mathbf{q}_0$) is obtained from the *KPL*, one can expect that the solution will be highly likely feasible, which helps to guarantee the reliability of the proposed planner.

C. Velocity Assignment to Keyframe Postures

The key novelty of the proposed online planner lies in its capability of generating full-body motions that respect the impact. To satisfy the impact-awareness condition, we assign the pre- and post-impact velocities to each keyframe posture.

Based on the displacement and the timestamp differences between two adjacent footsteps, the average velocity between i^{th} and $(i+1)^{th}$ footsteps ${}^{i}_{i+1}\mathbf{v}$ can be simply computed as

$$ {}^{i}_{i+1}\mathbf{v} = \begin{bmatrix} \frac{{}^{i+1}x^{\gamma} - {}^{i}x^{\gamma}}{{}^{i+1}\tau^{\gamma} - {}^{i}\tau^{\gamma}} & \frac{{}^{i+1}y^{\gamma} - {}^{i}y^{\gamma}}{{}^{i+1}\tau^{\gamma} - {}^{i}\tau^{\gamma}} & \frac{{}^{i+1}\phi^{\gamma} - {}^{i}\phi^{\gamma}}{{}^{i+1}\tau^{\gamma} - {}^{i}\tau^{\gamma}} \end{bmatrix}^T. \quad (11) $$

Then, an optimization problem is formulated to solve for the velocities assigned to the keyframe postures. It is important to note that any keyframe posture ${}^{i}\mathbf{q}^{*}$ are associated with two velocities, namely ${}^{i}\dot{\mathbf{q}}^{-}$ and ${}^{i+1}\dot{\mathbf{q}}^{+}$. The optimization problem of solving for the velocities assigned to the posture ${}^{i}\mathbf{q}^{*}$ can be formulated as follows:

$$
\begin{aligned}
\min_{{}^{i}\dot{\mathbf{q}}^{-},{}^{i+1}\dot{\mathbf{q}}^{+}} \quad & \mathbf{V}^{-T}\mathbf{Q}\mathbf{V}^{-} + \mathbf{V}^{+T}\mathbf{P}\mathbf{V}^{+} \\
\text{s.t.} \quad & \mathbf{J}_1 \; {}^{i+1}\dot{\mathbf{q}}^{+} = \mathbf{0_{6\times 1}} && \text{(C11-1)} \\
& \mathbf{J}_2 \; {}^{i}\dot{\mathbf{q}}^{-} = \mathbf{0_{6\times 1}} && \text{(C11-2)} \\
& \dot{z}_{sw}(\mathbf{q}, {}^{\mathbf{i}}\dot{\mathbf{q}}^{-}) < 0 && \text{(C11-3)} \\
& [{}^{i+1}\dot{q}_1^{+}, {}^{i+1}\dot{q}_2^{+}][cos^{i}\phi^{\gamma}, sin^{i}\phi^{\gamma}]^T > 0 && \text{(C11-4)} \\
& {}^{i+1}\dot{\mathbf{q}}^{+} = \mathbf{R}_{\dot{\mathbf{q}}}(\mathbf{q}^{-}) \; {}^{i}\dot{\mathbf{q}}^{-} && \text{(C11-5)}
\end{aligned}
\quad (12)
$$

where

$$ \mathbf{V}^{-} = [\dot{x}_b^{-}, \dot{y}_b^{-}, \dot{\psi}_b^{-}]^T - {}^{i}_{i+1}\mathbf{v} $$

and

$$ \mathbf{V}^{+} = [\dot{x}_b^{+}, \dot{y}_b^{+}, \dot{\psi}_6^{+}]^T - {}^{i}_{i+1}\mathbf{v}, $$

$\dot{x}_b$ and $\dot{y}_b$ are the velocities of the robot's base in x- and y-directions w.r.t. the world coordinate frame, and $\dot{\psi}_b$ is the yaw rate of the base w.r.t. the world coordinate frame. $\mathbf{Q} \in \mathbb{R}^{3\times3}$ and $\mathbf{P} \in \mathbb{R}^{3\times3}$ are any positive definite matrices. $\mathbf{J}_1$ and $\mathbf{J}_2$ are the contact Jacobian matrices, which are used to enforce the holonomic constraint at the contact points. The cost function ensures no dramatic changes in the desired velocity during one walking step.

The constraints are explained as follows:

- The constraint (C11-1) requires that right after the impact, the leading foot should become static on the ground.
- The constraint (C11-2) requires that right before the impact, the trailing foot should be static on the ground.
- The constraint (C11-3) requires that right before the impact, the leading foot should move toward the ground.
- The constraint (C11-4) requires that right after the impact, the velocity of the robot's base should not move backward.
- The constraint (C11-5) is the full-order dynamic relationship between the pre-impact and post-impact velocities.

It is important to note that the assigned pre- and post-impact velocities automatically satisfy the impact-awareness condition.

D. Full-Body Motion Generation

This subsection presents the last step of our proposed online planning method, which is continuous-phase motion generation. To enable online planning, it is reasonable to use reduced-order dynamic model for continuous phases because it significantly reduces the computational cost. Researchers have previously used Center of Mass (CoM) dynamics to successfully generate continuous-phase walking motions [1]. The CoM dynamics can be expressed as:

$$ m\ddot{\mathbf{r}} = \sum_{i=1}^{j} \mathbf{F}_i + m\mathbf{g}, \quad (13) $$

where m is the robot's total mass, $r \in \mathbb{R}^3$ is the CoM position w.r.t. the world coordinate frame, i is the i^{th} contact point, j is the total number of contact points, and $\mathbf{F}_i \in \mathbb{R}^3$ is the ground-reaction force applied at the i^{th} contact point.

To compute the desired continuous-phase motion, Eq. (13) is converted into difference equations to formulate the nonlinear optimization problem. Also, the full-order kinematics is considered in the optimization. In this case, we sample K points during each step. The optimization problem is solved for each step in real-time to obtain the desired continuous-phase motion that are dynamically feasible. Without loss of generality, we use the following cost function for our nonlinear optimization during the i^{th} step:

$$ \min_{\substack{{}^{i}\mathbf{q}[k], {}^{i}\dot{\mathbf{q}}[k], {}^{i}dt[k], \\ {}^{i}\mathbf{r}[k], {}^{i}\dot{\mathbf{r}}[k], {}^{i}\ddot{\mathbf{r}}[k], \\ {}^{i}\mathbf{F_j}[k]}} \begin{matrix} \sum_{i=1}^{K} (\|{}^{i}\mathbf{q}[k] - \mathbf{q}_{norm}[k]\| + \|{}^{i}\dot{\mathbf{q}}[k]\| + \|{}^{i}\ddot{\mathbf{r}}[k]\| \\ + \sum_{i=1}^{j} \|{}^{i}\mathbf{F}_j[k]\|) \end{matrix}, \quad (14) $$

where ${}^{i}\star[k]$ indicates the value of $\star$ at k^{th} point during the i^{th} step. $\mathbf{q}_{norm}$ is a single pre-computed nominal walking trajectory, which is to help ensure that the generated continuous-phase motion will not have drastically varying joint positions [5]. The same single $\mathbf{q}_{norm}$ is used in the cost function of any i^{th} step.

The constraints for this optimization include:

- Dynamic constraint:

$$ m^{i}\ddot{\mathbf{r}}[k] = \sum_{i=1}^{j} {}^{i}\mathbf{F}_j[k] + m\mathbf{g} \quad \text{(C12-1)} $$

- Kinematic constraint:

$$ {}^{i}\mathbf{r}[k] = {}^{i}\mathbf{r}({}^{i}\mathbf{q}[k]) \quad \text{(C12-2)} $$

- Step duration constraint:

$$\sum_{i=1}^{K} {}^{i}dt[k] = {}^{i+1}\tau^{\gamma} - {}^{i}\tau^{\gamma} \tag{C12-3}$$

- Keyframe posture constraint:

$$\begin{aligned} {}^{i}\mathbf{q}[1] &= {}^{i}\mathbf{q}^{*} \quad {}^{i}\mathbf{q}[K] = {}^{i+1}\mathbf{q}^{*} \\ {}^{i}\dot{\mathbf{q}}[1] &= {}^{i}\dot{\mathbf{q}}^{+} \quad {}^{i}\dot{\mathbf{q}}[K] = {}^{i}\dot{\mathbf{q}}^{-} \end{aligned} \tag{C12-4}$$

- Holonomic constraint:

$$\mathbf{J}({}^{i}\mathbf{q}[k])\,{}^{i}\dot{\mathbf{q}}[k] = \mathbf{0}_{6\times 1} \tag{C12-5}$$

- Keyframe posture constraint:
- Derivative approximation constraint:

$$\begin{aligned} \dot{\mathbf{r}}[k] &= \frac{\mathbf{r}[k+1]-\mathbf{r}[k]}{dt[k]} \\ \ddot{\mathbf{r}}[k] &= \frac{\dot{\mathbf{r}}[k+1]-\dot{\mathbf{r}}[k]}{dt[k]} \\ \dot{\mathbf{q}}[k] &= \frac{\mathbf{q}[k+1]-\mathbf{q}[k]}{dt[k]} \end{aligned} \tag{C12-6}$$

The constraints are explained next:

- The dynamic constraint (C12-1) requires that the planned motion satisfies Newton's law.
- The kinematic constraint (C12-2) indicates the kinematic relationship between the CoM and the configuration of the robot.
- The keyframe posture constraint (C12-4) ensures that the planed motion at the first and the last points (K^{th}) equals to the corresponding keyframe postures and velocities.
- The holonomic constraint (C12-5) ensures that the support foot is static on the ground during the step.
- The derivative approximation constraint (C12-6) is the finite difference method to compute the derivative in numerical computation.

IV. SIMULATIONS

This section presents the MATLAB and Webots [21] simulation results to demonstrate the effectiveness of our proposed online planning strategy.

MATLAB simulations were performed for initial validations, whereas Webots simulations were intended to provide more realistic validations.

Overall, our planner takes within 0.6 second to generate one walking step of impact-aware full-body motion, which is shorter than a typical bipedal walking step. Specifically, posture interpolation and keyframe-posture computation take approximately 0.05 second, pre- and post-impact velocity assignment takes approximately 0.13 second, and the continuous-phase motion generation, which is solved by the CasADi [22] optimization framework, takes approximately 0.4 second.

A. Trajectory Interpolation

As the output of the planner comprises isolated waypoints, trajectory interpolation is needed to generate the desired continuous trajectory.

Piecewise cubic Hermite functions are commonly used for trajectory interpolation, including: Piecewise Cubic Hermite Interpolating Polynomial (PCHIP), Cubic Spline Data Interpolation (SPLINE), and Modified Akima Piecewise Cubic Hermite Interpolation (MAKIMA).

We choose to use the PCHIP in our case because it preserves the shape of interpolated trajectories the best as compared with SPLINE and MAKIMA.

B. MATLAB Simulation

In MATLAB simulation, the global-position tracking control strategy is applied to drive the robot to follow the planned motion. The dynamic matrices, such as $\mathbf{M}(\mathbf{q})$ and $\mathbf{c}(\mathbf{q},\dot{\mathbf{q}})$, can be computed efficiently using FROST [23]. From the simulation results (Fig. 4), it is clear that our planning and control strategies result in satisfactory tracking of the desired footsteps with specific timing. The results demonstrate the effectiveness of our proposed planning strategy in generating impact-aware, dynamically feasible trajectory in real-time.

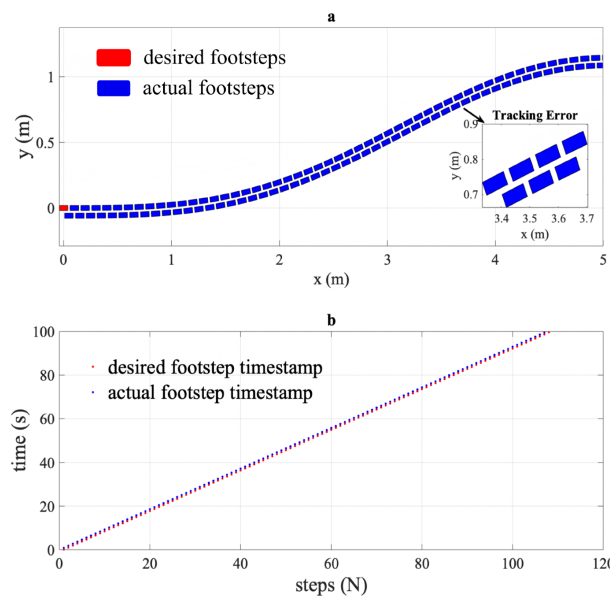

Fig. 4. MATLAB simulation results of a) satisfactory footstep tracking and b) satisfactory convergence of foot-landing timing.

C. Webots Simulation

In Webots simulation, we use the same optimization framework as implemented in MATLAB to generate the desired motion online. The setup of the Webots is illustrated in Fig. 5. For simplicity and without generality, decoupled joint-position control adapted from the global-position tracking control [16] is utilized in the Webots simulation. Fig. 6 shows the footstep tracking results in Webots. As compared with the MATLAB results, the tracking performance in Webots is less accurate because the decoupled joint controller ignores the nonlinear coupling among joints. However, as the steady-state tracking error is small and bounded, we can still

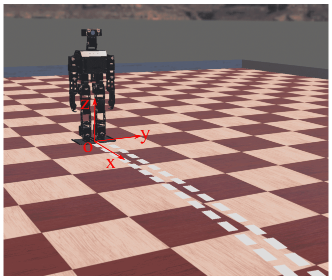

Fig. 5. Simulation setup in Webots. The shaded rectangles indicate the desired footstep sequence.

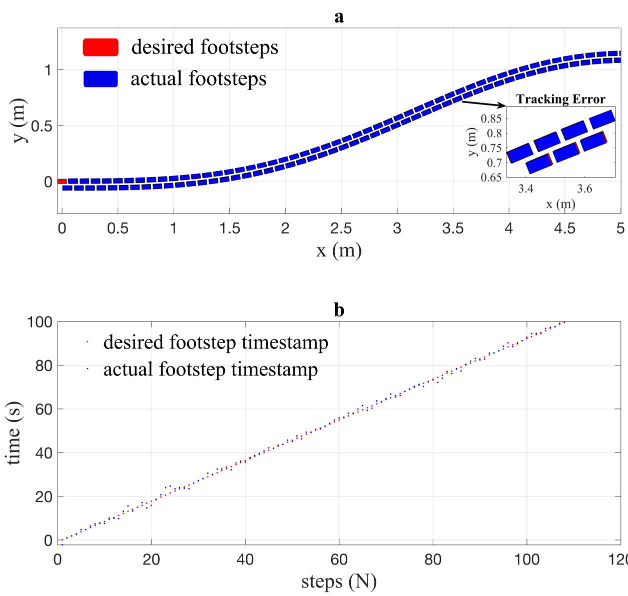

Fig. 6. Webots simulation results of a) satisfactory footstep tracking and b) satisfactory convergence of foot-landing timing.

consider that Webots simulation demonstrated a satisfactory trajectory tracking performance.

V. CONCLUSIONS

In this paper, we have introduced an online planning method that generates impact-aware, dynamically feasible, full-body trajectories for fully actuated bipedal walking robots. The proposed planner consists of four main components, including posture interpolation, keyframe posture computation, keyframe posture velocity assignment, and full-body motion generation based on reduced-order dynamics and full-order kinematics. To validate the proposed planner through simulations on a fully actuated bipedal walking robot, a trajectory tracking control law was simulated to track the generated motions. Results of both MATLAB and 3-D realistic simulations demonstrated the effectiveness of the proposed online planning strategy in generating dynamically feasible, full-body motions that respect the impact behaviors associated with bipedal walking.

REFERENCES

[1] P.-B. Wieber, R. Tedrake, and S. Kuindersma, "Modeling and control of legged robots," in *Springer handbook of robotics*, pp. 1203–1234, Springer, 2016.

[2] M. Rostami and G. Bessonnet, "Impactless sagittal gait of a biped robot during the single support phase," in *Proc. of IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, vol. 2, pp. 1385–1391 vol.2, May 1998.

[3] G. Cabodevila, N. Chaillet, and G. Abba, "Energy-minimized gait for a biped robot," in *Autonome Mobile Systeme*, pp. 90–99, Springer, 1995.

[4] 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.

[5] H. Dai, A. Valenzuela, and R. Tedrake, "Whole-body motion planning with centroidal dynamics and full kinematics," in *Proc. of IEEE-RAS International Conference on Humanoid Robots*, pp. 295–302, 2014.

[6] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, "The 3D linear inverted pendulum mode: A simple modeling for a biped walking pattern generation," in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems.*, vol. 1, pp. 239–246, 2001.

[7] 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.

[8] K. Nishiwaki and S. Kagami, "Simultaneous planning of com and zmp based on the preview control method for online walking control," in *Proc. of 11th IEEE-RAS International Conference on Humanoid Robots*, pp. 745–751, 2011.

[9] M. Ciocca, P.-B. Wieber, and T. Fraichard, "Effect of planning period on MPC-based navigation for a biped robot in a crowd," in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1–8, 2019.

[10] N. Bohórquez, A. Sherikov, D. Dimitrov, and P.-B. Wieber, "Safe navigation strategies for a biped robot walking in a crowd," in *Proc. of IEEE 16th International Conference on Humanoid Robots*, pp. 379–386, 2016.

[11] J. W. Grizzle, G. Abba, and F. Plestan, "Asymptotically stable walking for biped robots: Analysis via systems with impulse effects," *IEEE Transactions on Automatic Control*, vol. 46, no. 1, pp. 51–64, 2001.

[12] E. R. Westervelt, J. W. Grizzle, and D. E. Koditschek, "Hybrid zero dynamics of planar biped walkers," *IEEE Transactions on Automatic Control*, vol. 48, no. 1, pp. 42–56, 2003.

[13] Y. Gu, B. Yao, and C. G. Lee, "Bipedal gait recharacterization and walking encoding generalization for stable dynamic walking," in *Proc. of IEEE International Conference on Robotics and Automation*, pp. 1788–1793, 2016.

[14] Y. Gu, B. Yao, and C. G. Lee, "Exponential stabilization of fully actuated planar bipedal robotic walking with global position tracking capabilities," *Journal of Dynamic Systems, Measurement, and Control*, vol. 140, no. 5, p. 051008, 2018.

[15] Y. Gu, B. Yao, and C. G. Lee, "Straight-line contouring control of fully actuated 3-D bipedal robotic walking," in *Proc. of American Control Conference*, pp. 2108–2113, 2018.

[16] Y. Gao and Y. Gu, "Global-position tracking control of a fully actuated nao bipedal walking robot," in *Proc. of American Control Conference*, pp. 4596–4601, 2019.

[17] Y. Gao and Y. Gu, "Global-position tracking control of multi-domain planar bipedal robotic walking," in *Proc. of ASME Dynamic Systems and Control Conference*, 2019.

[18] "Robotis." http://www.robotis.us/robotis-OP2-OP3/. Accessed: 2019-09-20.

[19] H. K. Khalil, *Noninear systems*. No. 5, Prentice Hall, 1996.

[20] M. S. Branicky, "Multiple lyapunov functions and other analysis tools for switched and hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 475–482, 1998.

[21] "Cyberbotics ltd." https://cyberbotics.com/. Accessed: 2018-09-20.

[22] J. A. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "Casadi: a software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019.

[23] A. Hereid and A. D. Ames, "FROST: Fast robot optimization and simulation toolkit," in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 719–726, 2017.