

Robust Foot Placement Control for Dynamic Walking using Online Parameter Estimation

Qingbiao Li, Iordanis Chatzinikolaïdis, Yiming Yang, Sethu Vijayakumar and Zhibin Li

Abstract—This paper presents an estimation scheme to control foot placement for achieving a desired dynamic walking velocity in presence of sensor and model errors. Inevitable discrepancies, such as sensors’ noise, delay, and modelling errors, degrade the performance of model-based control methods or even cause instabilities. To resolve these issues, an on-line parameter estimation approach based on Tikhonov regularisation is formulated using measurement data, which is particularly robust for more accurately approximating the dynamics. The proposed scheme initially uses the foot placement predicted by the linear inverted pendulum model, while the control parameters are being optimised using adequate measurements to represent the real dynamics within and in-between steps; and then, the estimation based control is used to predict the future foot placement accurately in the presence of discrepancies.

I. INTRODUCTION

Humanoid robots, designed with a human morphology, offer advantages of traversing environments that are easily accessible by humans, such as stairs, passageways, rugged terrains, etc. [1] as well as using human-oriented tools [2]. A humanoid robot is a floating-base system of two-legs [3] with morphological adaptation to various surfaces, providing adaptability and maneuverability [4]. They have potentials to be indispensable in emergency and disaster responses, where wheeled robots are limited by the terrain irregularities. In turn, the mechanical complexity of humanoids imposes control challenges compared to wheeled robots.

Many model-based approaches have been studied to address the problem of bipedal locomotion. Kajita et al. [5] proposed the Linear Inverted Pendulum (LIP) model, which regards the robot as a point mass, to generate horizontal motions and keep the Centre of Mass (COM) height constant. Given a target COM motion, the corresponding Zero Moment Point (ZMP) or Centre of Pressure (COP) for achieving it can be analytically computed. LIP model and its extensions have been widely applied in bipedal walking, and its simplified modelling is illustrated in Fig. 1.

However, model-based approaches, e.g. LIP model-based foot placement control, generally have fixed coefficients and parameters manually tuned off-line for controlling legged locomotion [6]. For example, Raibert’s control of a one-leg hopping robot has decoupled regulation of hoping height by delivering a fixed vertical thrust during stance, forward speed by foot placement, and an upright posture by exerting a torque around the hip [7]. Therefore, proper tuning of all variables was very crucial and usually relied on experience, which could only be done by experimental trial and error. However, this manual tuning has limitations because parameters might be time-varying or state-dependent. The same

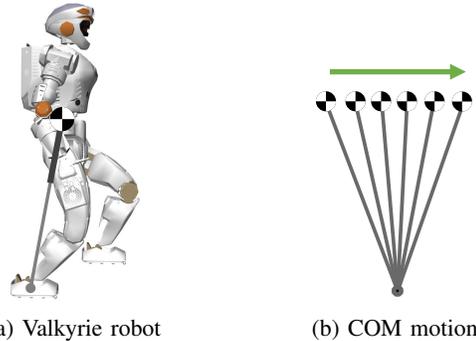


Fig. 1: Bipedal walking control of the Valkyrie robot using the Linear Inverted Pendulum model (sagittal scenario).

problems exist in other model-based approaches, especially when unexpected changes occur [8].

To resolve this, auto-tuning of parameters has been explored given a known control structure [9]. Nakanishi et al. [10] developed a framework to learn bipedal locomotion through movement primitives by locally weighted regression while the frequency of the learnt trajectories is adjusted automatically. You et al. [11] used linear regression based on past measurements for updating the coefficients of an extended formulation based on Raibert’s model to achieve accurate velocity tracking. This method improved the system’s flexibility to unknown changes, such as a mass offset, and was later extended to bipedal walking and running [12]. However, the convergence rate in You’s method is significantly limited, because its formulation has two coefficients coupled with the measured velocity: one is directly for the velocity, and the other is for the velocity error where measured velocity appear as well. Hence, the coupling of these two coefficients resulted in the fluctuation of estimated values.

We propose an online estimation approach derived from the analytic insights of the LIP model, which has a major advantage in comparison with Raibert’s linear model, i.e. the decoupling between the current forward velocity and the desired one. As a consequence, we expect a much faster and stable convergence of the coefficients needed for foot placement, thus better adaptation to unexpected changes, e.g. an unknown mass offset. We elaborated on a more robust calculation of the coefficients as well as on the effects of downgraded sensory information.

This paper contributes in the following aspects:

- A rigorous analysis of the propagation of sensor errors in walking control, and the inevitable uncertainties of model-based methods;
- Identification of control parameters derived from the

model, and solution of the parameter estimation problem with nominal values using optimization [13];

- Online estimation for both the continuous dynamics within a step and the discrete dynamics between steps.

This paper is organised as follows. The limitations and inevitable uncertainties of LIP model are discussed in Section II. The proposed methodology is elaborated in Section III. Benchmark results studied in simulation are presented in Section IV, followed by conclusions in Section V.

II. PROBLEM STATEMENT

Model-based approaches constitute an analytic framework for controlling robots and therefore are extensively used. In a classical model-based approximation of bipedal walking—Raibert’s model, LIP, etc.—an analytical solution is defined to estimate the next foot placement, which is model specific. For example, given a certain transition time t , the COM motion of a LIP model can be computed based on the current COM state by the hyperbolic functions as [8]:

$$x_f = (x_0 - p^*) \cosh(\tau) + \dot{x}_0 T_c \sinh(\tau) + p^* \quad (1)$$

$$\dot{x}_f = (x_0 - p^*) \frac{\sinh(\tau)}{T_c} + \dot{x}_0 \cosh(\tau), \quad (2)$$

where $\tau = t/T_c$ is the normalised transition time. The time constant $T_c = \sqrt{z_c/g}$ is defined by the fixed COM height z_c in the LIP model. The transition time t is the duration from the current state to the moment of interest. x_0 and \dot{x}_0 are the initial COM state, x_f and \dot{x}_f are the final COM state after the transition time t , while p^* is the position of the point foot. All variables are expressed in a global coordinate.

First of all, tuning of the model’s parameters requires substantial effort in model-based approaches of bipedal walking [14]. Secondly, a set of model parameters are still not adequate to capture the real system dynamics under all circumstances. Thus, degradation of performance occurs due to errors and uncertainties in the following sources.

- *Sensory errors*: limited resolution, bandwidth, etc. result in noises, residuals or drifts in the measurements [15];
- *Delays*: latency and phase lag introduced by the communication, signal filtering, etc. can degrade the control performance and stability [16];
- *Model errors*: discrepancies between a simplified model and a real system, un-modeled non-linearities such as mechanical backlash and deformation.

To mitigate the aforementioned degradation and achieve robust walking, we exploit an underlying model that governs the general walking behaviour, and then estimate the model specific coefficients from the measurements. We first analyse how the propagation of errors of the current COM state affects the prediction of the future COM state as well as the resulting foot placement.

The current COM state \tilde{x}_0 and $\tilde{\dot{x}}_0$ from the sensor measurements is composed by the true COM state x_0^{real} and \dot{x}_0^{real} , and the measurement errors e_{x_0} and $e_{\dot{x}_0}$,

$$\tilde{x}_0 = x_0^{\text{real}} + e_{x_0}, \quad (3)$$

$$\tilde{\dot{x}}_0 = \dot{x}_0^{\text{real}} + e_{\dot{x}_0}. \quad (4)$$

In the following, $[\tilde{\cdot}]$ and $[\hat{\cdot}]$ indicate measured and predicted variables, respectively.

To keep intuition and simplicity, we place the coordinate frame at the n -th step with respect to (w.r.t.) the stance foot. The predicted future COM velocity can be calculated by the current COM state, using the analytic solution defined in (2),

$$\begin{aligned} \hat{x}_f^n &= \tilde{x}_0^n \frac{\sinh(\tau_{la})}{T_c} + \tilde{\dot{x}}_0^n \cosh(\tau_{la}) \\ &= x_0^{\text{real},n} \frac{\sinh(\tau_{la})}{T_c} + \dot{x}_0^{\text{real},n} \cosh(\tau_{la}) + e_{x_0}^n \frac{\sinh(\tau_{la})}{T_c} \\ &\quad + e_{\dot{x}_0}^n \cosh(\tau_{la}) \\ &= \hat{x}_f^{\text{real},n} + \hat{e}_{\hat{x}_f}^n, \end{aligned} \quad (5)$$

where $\tau_{la} = t_{la}/T_c$ is the normalised look-ahead time, and $\hat{e}_{\hat{x}_f}^n$ is the error of the predicted future COM velocity

$$\hat{e}_{\hat{x}_f}^n = e_{x_0}^n \frac{\sinh(\tau_{la})}{T_c} + e_{\dot{x}_0}^n \cosh(\tau_{la}). \quad (6)$$

Clearly, (6) shows that the error propagates through time and amplifies by the dynamics.

A similar uncertainty exists when errors propagate to the foot placement for the *next step* ($n+1$). In (2), let $\dot{x}_f = \dot{x}_d^{n+1}$, given an initial COM state and the next step time T_{step} , the foot placement for achieving a desired COM velocity is

$$p^* = \tilde{x}_0^{n+1} + T_c \dot{x}_0^{n+1} \coth(\tau_s) - T_c \dot{x}_d^{n+1} \text{csch}(\tau_s), \quad (7)$$

where $\tau_s = T_{\text{step}}/T_c$ is the normalised step time. Note that according to the LIP model, the final velocity of a step is equal to the initial velocity of the next step, i.e. $\dot{x}_0^{n+1} = \dot{x}_f^n$.

Since the swing foot cannot be placed instantaneously, a look-ahead time τ_{la} is needed to predict a the future foot placement. We can substitute \dot{x}_0^{n+1} in (7) by a predicted future velocity $\hat{\dot{x}}_f^n$ in (5). Also, a relative foot placement w.r.t. the body, defined as p without $[\cdot]^*$, is of more interest

$$\begin{aligned} p^{n+1} &= p^{*,n+1} - \tilde{x}_0^{n+1} \\ &= T_c (\hat{\dot{x}}_f^{\text{real},n} + e_{\hat{\dot{x}}_f}^n) \coth(\tau_s) - T_c \dot{x}_d^{n+1} \text{csch}(\tau_s) \\ &= p^{\text{real},n+1} + \hat{e}_p^{n+1}. \end{aligned} \quad (8)$$

Based on $\hat{e}_{\hat{x}_f}^n$ in (6), the uncertain error term \hat{e}_p in (8) is

$$\hat{e}_p^{n+1} = T_c \coth(\tau_s) \left[e_{x_0}^n \frac{\sinh(\tau_{la})}{T_c} + e_{\dot{x}_0}^n \cosh(\tau_{la}) \right], \quad (9)$$

where the uncertainty and error of foothold prediction increases in an exponential manner by the look-ahead time τ_{la} and the next step time τ_s , which provides further insights on how the current COM errors, $e_{x_0}^n$ and $e_{\dot{x}_0}^n$, propagate.

It can be inferred from (9) that since $e_{x_0}^n$ and $e_{\dot{x}_0}^n$ vary from time to time and are perhaps phase-dependent, proper tuning of τ_{la} and τ_s is rather challenging; especially when T_c (z_c) can be a variable due to different robot configurations, e.g. squatting, standing, walking, or carrying a payload.

If each parameter in (3) and (4) can be precisely known, then simple subtraction of the error terms e_{x_0} and $e_{\dot{x}_0}$ could correct the resulted errors in prediction. However, since these terms are multiple and influenced by all factors contributing

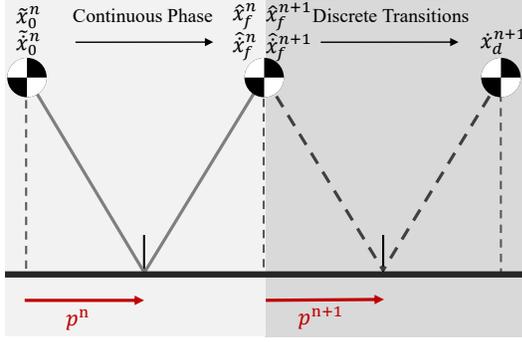


Fig. 2: Foot placement control for the $n+1$ step based on the COM state $(\tilde{x}_0^n, \dot{x}_0^n)$ at the current n step and the target velocity \dot{x}_d^{n+1} at the $n+1$ step (sagittal scenario).

to the performance degradation, their accurate estimates are very difficult to obtain. Hence, an online estimation approach is proposed here that considers the foot placement controller as a grey box, and estimates only the lumped-up terms that are the resulted coefficients including the uncertainties caused by errors, delays, and unmodelled quantities.

III. FOOT PLACEMENT CONTROL BASED ON REGULARISED LEAST SQUARES

Legged locomotion is characterised by hybrid dynamics including *continuous* and *discrete* phases, which can be viewed as a continuous dynamical system undergoing discrete transitions during the touch-down or take-off of support legs. Hence, our proposed optimization approach (Fig. 2) is firstly applied for estimating the state transition of the COM during the continuous phase (Section III-A). Then, based on the predicted final velocity of the current step, a similar optimization problem is formulated to account for the discrete transitions. The final result is the computation of an accurate foot placement which achieves the desired walking velocity with minimum steady state error (Section III-B) based on the imperfect real-time sensory feedback.

A. Optimization of Velocity Estimation During a Step

The inevitable uncertainty of predicting the future COM state can be understood from (5). From past steps, a dataset X_s can be created to hold the corresponding measurements \tilde{x}_0 , \tilde{x}_f , and the final foot placement \tilde{p} .

Furthermore, if we substitute equations (3) and (4) in (5), the measured final velocity of a step expressed in the local stance foot frame is

$$\begin{aligned} \tilde{x}_f^n &= \dot{x}_f^{\text{real},n} + e_{\dot{x}_f}^n \\ &= \dot{x}_0^{\text{real},n} \frac{\sinh(\tau_{la})}{T_c} + \dot{x}_0^{\text{real},n} \cosh(\tau_{la}) + e_{\dot{x}_f}^n \\ &= -\tilde{p}^n \frac{\sinh(\tau_s)}{T_c} + \tilde{x}_0^n \cosh(\tau_s) - e_{\dot{x}_0}^n \frac{\sinh(\tau_s)}{T_c} \\ &\quad - e_{\dot{x}_0}^n \cosh(\tau_s) + e_{\dot{x}_f}^n, \end{aligned} \quad (10)$$

where the foot placement \tilde{p}^n is w.r.t. the COM ($\tilde{p} = -\tilde{x}_0$).

Thus, (10) can be expressed in a more general form by defining a vector of coefficients $\alpha = [\alpha_1, \alpha_2, \alpha_3]^T$ as

$$\tilde{x}_f = -\tilde{p}^n \alpha_1 + \tilde{x}_0^n \alpha_2 + \alpha_3, \quad (11)$$

where α_1 and α_2 capture the uncertainties of the model-based coefficients, and α_3 accounts for the lumped terms of both propagated and current measurement errors.

By indexing (11) in each step, we can extract k measurements from the dataset X_s that correlate the state transition from the beginning until the end of each step

$$\dot{\mathbf{x}}_f = \begin{bmatrix} \tilde{x}_f^{n-k} \\ \vdots \\ \tilde{x}_f^{n-1} \end{bmatrix}_{k \times 1}, \quad \mathbf{X}_1 = \begin{bmatrix} -\tilde{p}^{n-k} & \tilde{x}_0^{n-k} & 1 \\ \vdots & \vdots & \vdots \\ -\tilde{p}^{n-1} & \tilde{x}_0^{n-1} & 1 \end{bmatrix}_{k \times 3}. \quad (12)$$

To solve α in a way that reflects the dynamics in the collected data, while having minimum deviation from the values calculated by the LIP model, it can be achieved by introducing a penalised least-squares problem in the form of

$$\min_{\alpha} \|\mathbf{X}_1 \alpha - \dot{\mathbf{x}}_f\|_{\mathbf{P}_1}^2 + \|\alpha - \alpha_0\|_{\mathbf{Q}_1}^2, \quad (13)$$

where $\|\cdot\|_{\mathbf{M}}$ denotes a weighted euclidean norm.

It shall be noted that the second term $\|\alpha - \alpha_0\|_{\mathbf{Q}_1}^2$ in (13) is important, because our study found that the prior work [12] using only the least square term sometimes produced undesirable fluctuation of α . With (13), α_0 serves as an initial guess and a very large deviation is penalised.

The minimisation problem expressed in (13) is also known as Tikhonov regularisation [13]. The closed-form solution can be readily computed as

$$\alpha = \alpha_0 + [\mathbf{X}_1^T \mathbf{P}_1 \mathbf{X}_1 + \mathbf{Q}_1]^{-1} [\mathbf{X}_1^T \mathbf{P}_1 (\dot{\mathbf{x}}_f - \mathbf{X}_1 \alpha_0)], \quad (14)$$

where \mathbf{P}_1 is the diagonal weighting matrix for the regression term, and \mathbf{Q}_1 is the diagonal weighting matrix for the regularisation term

$$\mathbf{P}_1 = g_p \begin{bmatrix} w_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & w_k \end{bmatrix}, \quad \mathbf{Q}_1 = g_q \begin{bmatrix} w_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & w_k \end{bmatrix} \quad (15)$$

with w_i the weight for the i th index of sampled walking state in \mathbf{X}_1 , and k the number of stored data. Here we simply have a linear weight of $w_i = i$. The scalars g_p and g_q are used to weight the influence of the regression and the regularisation term, respectively.

By solving (14), we obtain an α that best approximates the state transition from the collected data until the $n-1$ step. Once the n step starts, we can measure $\mathbf{x}^n = [-\tilde{p}^n \quad \tilde{x}_0^n \quad 1]$ and predict the future velocity $\hat{\dot{x}}_f^n$ at the end of the n step by

$$\hat{\dot{x}}_f^n = \mathbf{x}^n \alpha. \quad (16)$$

B. Optimization of the Foot Placement for the Next Step

Section III-A describes the optimization approach concerning the continuous transition " $\dot{x}_0^n \rightarrow \dot{x}_f^n$ ". This section elaborates on how the dynamics of the step-to-step transition " $\dot{x}_f^n \rightarrow \dot{x}_f^{n+1}$ " can be better approximated using a similar optimization. This is essential because once $\hat{\dot{x}}_f^n$ is calculated by (16), we can predict an accurate foothold if the discrete transition " $\dot{x}_f^n \rightarrow \dot{x}_f^{n+1}$ " is known. Using prediction and our

optimization for estimating the unknown error terms, we mitigate the degradation effects in sensing and control.

For each step that has happened, we can measure \tilde{x}_f^{n-1} , \tilde{x}_f^n , and \tilde{p}^n . From (8), the measured foot placement of the step expressed w.r.t. the stance foot can be described as

$$\begin{aligned}\tilde{p}^n &= p^{\text{real},n} + e_p^n \\ &= T_c \tilde{x}_f^{\text{real},n-1} \coth(\tau_s) - T_c \tilde{x}_f^{\text{real},n} \text{csch}(\tau_s) + e_p^n \\ &= T_c \tilde{x}_f^{n-1} \coth(\tau_s) - T_c \tilde{x}_f^n \text{csch}(\tau_s) - T_c e_{\tilde{x}_f}^{n-1} \coth(\tau_s) \\ &\quad + T_c e_{\tilde{x}_f}^n \text{csch}(\tau_s) + e_p^n,\end{aligned}\quad (17)$$

where the target velocity \dot{x}_d^n can be regarded equal with the real velocity $\dot{x}_f^{\text{real},n}$ at the end of the step.

Hence, the foot placement formula in (17) can be expressed in a general form using coefficients β_1 , β_2 , and β_3 :

$$p^n = \beta_1 \tilde{x}_f^{n-1} + \beta_2 \tilde{x}_f^n + \beta_3, \quad (18)$$

where β_1 and β_2 replace the model-based coefficients, and β_3 accounts for the error terms expressed in (6) and (9).

Once the estimation starts, the dataset X_s can be used to form the matrix \mathbf{X}_2 that holds a fixed number of the most recent measurements. The matrix \mathbf{X}_2 contains the COM velocities at the end of every step for the last k steps. The vector \mathbf{p} is the concatenation of the corresponding foot placement locations:

$$\mathbf{p} = \begin{bmatrix} \tilde{p}^{n-k} \\ \vdots \\ \tilde{p}^{n-1} \end{bmatrix}_{k \times 1}, \quad \mathbf{X}_2 = \begin{bmatrix} \tilde{x}_f^{n-k} & \tilde{x}_f^{n-k+1} & 1 \\ \vdots & \vdots & \vdots \\ \tilde{x}_f^{n-1} & \tilde{x}_f^n & 1 \end{bmatrix}_{k \times 3}. \quad (19)$$

The Tikhonov regularisation method can be used again to calculate the model coefficients. Thus, the vector of coefficients $\boldsymbol{\beta} = [\beta_1 \ \beta_2 \ \beta_3]^T$ can be estimated by

$$\min_{\boldsymbol{\beta}} \|\mathbf{X}_2 \boldsymbol{\beta} - \mathbf{p}\|_{\mathbf{P}_2}^2 + \|\boldsymbol{\beta} - \boldsymbol{\beta}_0\|_{\mathbf{Q}_2}^2. \quad (20)$$

The solution and the weighting matrices \mathbf{P}_2 and \mathbf{Q}_2 are defined similarly to those in (14) and (15), respectively. After calculating $\boldsymbol{\beta}$, the next foot placement can be predicted by

$$p^{n+1} = \mathbf{x}^{n+1} \boldsymbol{\beta}, \quad (21)$$

where $\mathbf{x}^{n+1} = \begin{bmatrix} \hat{x}_f^n & \hat{x}_d^{n+1} & 1 \end{bmatrix}$.

C. Implementation Details of the 2-Stage Optimization

This section explains the implementation for estimating the continuous “ $\dot{x}_0^n \rightarrow \dot{x}_f^n$ ” and the discrete “ $\dot{x}_f^n \rightarrow \dot{x}_f^{n+1}$ ” transitions: the initial stage before sufficient samples are acquired; the optimization stage once online estimation starts.

Step 1: Data generation using LIP model: In the initial stage, the LIP model is used to predict the final velocity \hat{x}_f of the current step based on the COM state \tilde{x}_0 and \tilde{x}_0^i of each step, and then calculate the foot placement location p ; data are collected using predefined model-specific parameters.

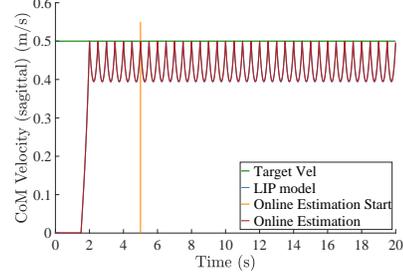


Fig. 3: Sagittal velocity profile generated by LIP model and online estimation to reach target velocity in *Simulation 1*

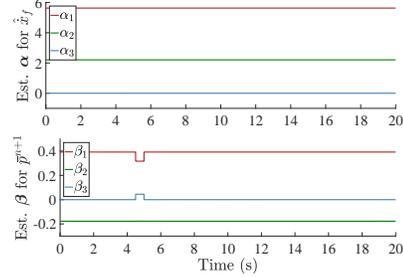


Fig. 4: Model coefficients α and β in *Simulation 1*

Step 2: Generation of the dataset X_s : In the beginning of every step, \tilde{x}_0^i and \tilde{x}_0^i measured by the sensors are inserted in the dataset X_s . The final velocity of the current step is stored as the initial COM velocity at the beginning of the next step $i + 1$, because $\tilde{x}_0^{i+1} = \tilde{x}_f^i$ in LIP model. For any other model where $\tilde{x}_0^{i+1} \neq \tilde{x}_f^i$, we can apply exactly the same approach by directly using measured \tilde{x}_0^{i+1} . The corresponding foot placement that resulted in the measured initial and final velocity of each step is stored in the dataset as well. These data are used to form matrices \mathbf{X}_1 and \mathbf{X}_2 .

The dataset is updated at each touch-down moment, using a fixed size First-In First-Out (FIFO) buffer. The dataset size k will be further studied for optimal selection.

Step 3: Estimation of parameters: Based on the data, we apply the Tikhonov regularisation method as proposed in Section III-A and Section III-B. Once the online estimation starts, matrices \mathbf{X}_1 and \mathbf{X}_2 are formed using a fixed number of the last k measurements. The matrix \mathbf{X}_1 contains the relative COM positions and velocities at the beginning of every step, while the vector $\dot{\mathbf{x}}_f$ is the concatenation of the final velocities from the past k_{vel} steps. Then, \mathbf{X}_1 and $\dot{\mathbf{x}}_f$ are used to estimate the optimal model coefficients α in the continuous transition “ $\dot{x}_0^n \rightarrow \dot{x}_f^n$ ” by (14).

Similarly, matrix \mathbf{X}_2 contains the COM velocities at the end of each of the k_{fp} steps, while \mathbf{p} is the concatenation of the foot placement locations. Afterwards, they are used to estimate the optimal coefficients $\boldsymbol{\beta}$ for the step-to-step transition “ $\dot{x}_f^n \rightarrow \dot{x}_f^{n+1}$ ” by (20).

Step 4: Prediction of the next foot placement: While updating the estimates of α and $\boldsymbol{\beta}$, the intermediate values can be used in order to obtain more accurate predictions of the final velocity and the next foot placement location for the current step. The final velocity of the current step \hat{x}_f^n can be estimated using \mathbf{x}^n and the model coefficients α based on

TABLE I: Simulation setup and results.

Cases	Noise & delay	COM offset	Methods	Steady state error (m/s)	α	β
1	None	None	LIP model	0	[5.63, 2.21, 0]	[0.392, -1.78, 0]
			Online Estimation	0	[5.63, 2.21, -2.42×10^{-4}]	[0.392, -0.178, -4.37×10^{-5}]
2	110dB noise filtering delay	None	LIP model	8.17×10^{-2}	[5.63, 2.21, 0]	[0.392, -0.178, 0]
			Online Estimation	2.00×10^{-4}	[5.63, 2.21, 2.00×10^{-2}]	[0.392, -0.178, 3.66×10^{-3}]
3	110dB noise filtering delay	10% vertical	LIP model	5.56×10^{-2}	[4.98, 2.08, 0]	[0.418, -0.201, 0]
			Online Estimation	2.80×10^{-3}	[4.98, 2.08, 1.84×10^{-2}]	[0.418, -0.201, 2.88×10^{-3}]
4	110dB noise filtering delay	$-0.01m$ horizontal	LIP model	0.280	[5.63, 2.21, 0]	[0.392, -0.176, 0]
			Online Estimation	1.30×10^{-3}	[5.63, 2.21, 7.51×10^{-2}]	[0.392, -0.176, 1.24×10^{-2}]

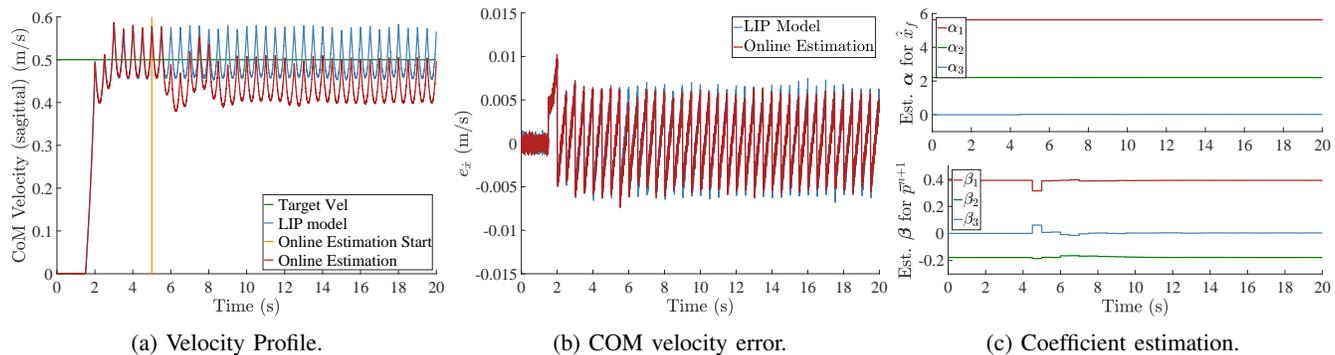


Fig. 5: Simulation 2 with 110dB noise and filtering delay.

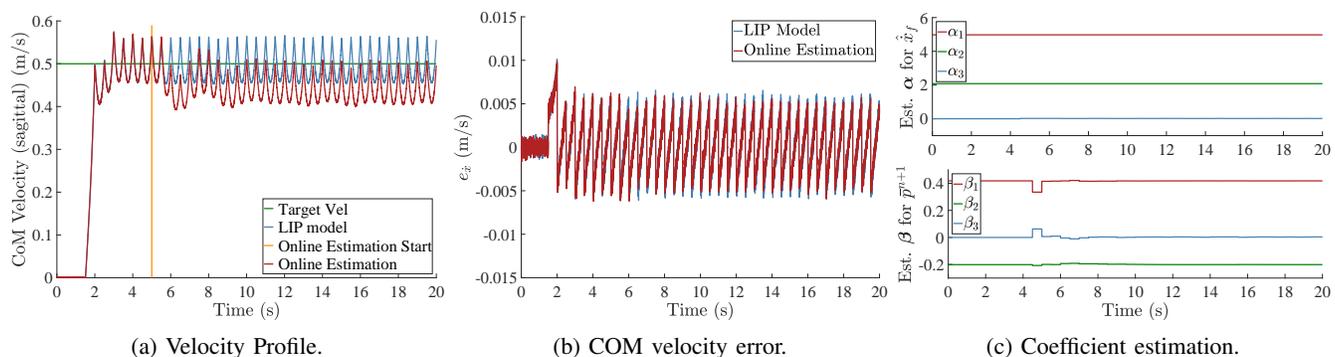
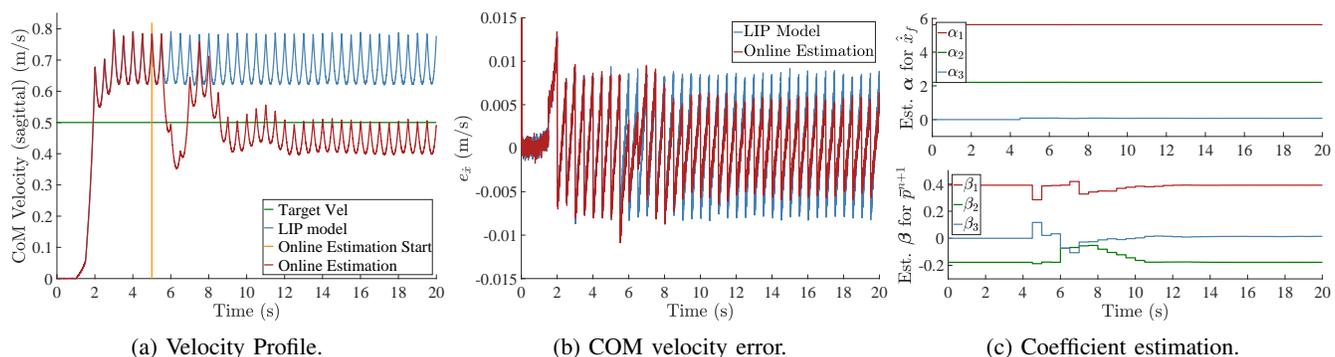


Fig. 6: Simulation 3 with 110dB noise, filtering delay and 10% vertical COM offset.


 Fig. 7: Simulation 4 with 110dB noise, filtering delay and COM has drifted reading of $-0.01m$ offset horizontally.

(16). Then, \hat{x}_f^n can be used in \mathbf{x}^{n+1} to control the next foot placement p^{n+1} together with model coefficients β in order to achieve the target velocity \dot{x}_d^{n+1} in (21).

The dataset and the model coefficients in Steps 2, 3, and 4 are recursively updated so as to obtain the optimal model

coefficients α and β , which are used to reach the target velocity with minimum steady state error.

IV. SIMULATION

The proposed approach was validated in terms of the accuracy of the next foot placement prediction, the convergence

of coefficients α and β , and the robustness of walking subject to delay and unknown mass offset. The performance was also compared to that of the LIP model with fixed parameters by the tracking of walking velocity.

In our simulation, the constant height of the LIP model was $1.2m$, matching the COM height of the humanoid Valkyrie, step time was $0.5s$, i.e. $1s$ stride time similar to humans. The target end velocity of each step was $0.5m/s$. The initial values of α and β are calculated by the LIP model.

A dataset size $k_{vel} = 2$ is used during the continuous transition " $\dot{x}_0^n \rightarrow \dot{x}_f^n$ ", while a size of $k_{fp} = 6$ is used during the step-to-step transition " $\dot{x}_f^n \rightarrow \dot{x}_f^{n+1}$ " similar to work in [12]. Also, we chose weighing matrices $\mathbf{P}_1 = \text{diag}(0.1, 0.2)$, $\mathbf{P}_2 = \text{diag}(0.1, 0.2, 0.3, 0.4, 0.5, 0.6)$, and $\mathbf{Q}_1 = \mathbf{Q}_2 = \text{diag}(0.1, 0.1, 0.001)$. In order to examine the robustness of the proposed method, we introduced noise with a $110dB$ signal-to-noise ratio, error and delay in filtering, constant vertical COM offset, and horizontal COM offset.

Four simulations have been carried out for evaluation, as summarised in Table I. According to the velocity profile, the robot started with an initial velocity \dot{x}_0 , decreased to minimum when the COM position was above the stance foot, and then increased to a new maximum velocity \dot{x}_f at end of the step. We are interested in reaching the target velocity \dot{x}^d at the end of each step instead of the average velocity.

Figs. 3 and 4 show the ideal case of Simulation 1 as a baseline, showing that both methods can achieve the desired velocity. The estimated coefficients agree with the analytical solution calculated by the LIP model.

In Simulation 2, the noise of $110dB$ and its filtering delay caused an average steady state error of $8.17 \times 10^{-2} m/s$ for the LIP model (see blue line in Fig. 5a). After a small fluctuation between $5s$ to $10s$, the online estimation method reached the desired velocity with a negligible error of $0.0002m/s$. Fig. 5b illustrates the effect of these errors as the subtraction of the measured by the ground truth value, ($e_{\dot{x}} = \tilde{\dot{x}} - \dot{x}$).

In Simulation 3, 10% of vertical COM offset was introduced (see Fig. 6). Note that a higher z_c increases the time constant T_c , hence, the initial value of the model coefficients α , β decreases (see (10) and (17)) and the effect of the errors reduces. That is why the steady error in the LIP model decreases compared to that in Simulation 2. Regardless, the proposed method was able to compensate for the COM offset.

In Simulation 4, a $-0.01m$ horizontal COM offset was applied. As shown in Fig. 7, although there exists a fluctuation in the estimated coefficients and a slightly longer converge time ($12s$), online estimation can still achieve the desired COM velocity with low steady state error, whereas such a COM offset has caused a $0.28m/s$ steady state error to the velocity profile generated by the LIP model. The first and second optimal value of the model coefficients (α and β) in Table I are close to the values based on the LIP model. However, the third coefficients (α_3 and β_3) reflect the overall effect of sensory noise and delay, and hence are different from those in the LIP model. These are also proved in (10) and (17).

V. CONCLUSIONS

In this paper, we first analyse how the unknown terms propagate and affect the accuracy of foot placement, and further we propose the regularised least squares for eliminating these common discrepancies in the LIP model for bipedal walking. The robustness of the proposed approach under various types of sensor and model errors is validated through simulations in four different scenarios. Compared to traditional model-based methods, the proposed control achieves smaller steady state error, as shown in Section IV.

In the future, we will investigate a better rule of exploiting the past dataset, similar to the long-term and short-term memories, instead of storing only a fixed number of past samples. Also, we will further extend this method from the planar case to the 3-dimensional space.

REFERENCES

- [1] J. Denny, M. Elyas, S. D'costa, and R. D'Souza, "Humanoid robots—past, present and the future," *European Journal of Advances in Engineering and Technology*, vol. 3, no. 5, pp. 8 – 15, 2016.
- [2] R. Xiong, Y. Sun, Q. Zhu, J. Wu, and J. Chu, "Impedance control and its effects on a humanoid robot playing table tennis," *International Journal of Advanced Robotic Systems*, vol. 9, no. 5, 2012.
- [3] N. B. Ignell, N. Rasmusson, and J. Matsson, "An overview of legged and wheeled robotic locomotion," in *Mini-conference on Interesting Results in Computer Science and Engineering*, vol. 21, 2012.
- [4] J. Yi, Q. Zhu, R. Xiong, and J. Wu, "Walking algorithm of humanoid robot on uneven terrain with terrain estimation," *International Journal of Advanced Robotic Systems*, vol. 13, no. 1, 2016.
- [5] S. Kajita, O. Matsumoto, and M. Saigo, "Real-time 3d walking pattern generation for a biped robot with telescopic legs," in *IEEE International Conference on Robotics and Automation*, vol. 3, 2001, pp. 2299 – 2306.
- [6] J. A. Castano, Z. Li, C. Zhou, N. Tsagarakis, and D. Caldwell, "Dynamic and reactive walking for humanoid robots based on foot placement control," *International Journal of Humanoid Robotics*, vol. 13, no. 02, p. 1550041, 2016.
- [7] M. Raibert, *Legged robots that balance*. The MIT Press, 1986.
- [8] S. Kajita, H. Hirukawa, K. Harada, and K. Yokoi, *Introduction to humanoid robotics*. Springer-Verlag Berlin Heidelberg, 2014.
- [9] R. Tedrake, "Applied optimal control for dynamically stable legged locomotion," Ph.D. dissertation, Massachusetts Institute of Technology, 2004.
- [10] J. Nakanishi, J. Morimoto, G. Endo, G. Cheng, S. Schaal, and M. Kawato, "Learning from demonstration and adaptation of biped locomotion," *Robotics and Autonomous Systems*, vol. 47, no. 2, pp. 79 – 91, 2004.
- [11] Y. You, Z. Li, N. Tsagarakis, and D. Caldwell, "Foot placement control for bipedal walking on uneven terrain: An online linear regression analysis approach," in *International Conference on Climbing and Walking Robots and Support Technologies for Mobile Machines*, 2015, pp. 478 – 485.
- [12] Y. You, Z. Li, D. Caldwell, and N. Tsagarakis, "From one-legged hopping to bipedal running and walking: A unified foot placement control based on regression analysis," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2015, pp. 4492 – 4497.
- [13] A. Tikhonov, A. Goncharky, V. Stepanov, and A. Yagola, *Numerical methods for the solution of ill-posed problems*. Springer Netherlands, 1995.
- [14] J. Pratt and G. Pratt, "Intuitive control of a planar bipedal walking robot," in *IEEE International Conference on Robotics and Automation*, vol. 3, 1998, pp. 2014 – 2021.
- [15] Q. Zhu, Y. Mao, R. Xiong, and J. Wu, "Adaptive torque and position control for a legged robot based on a series elastic actuator," *International Journal of Advanced Robotic Systems*, vol. 13, no. 1, 2016.
- [16] A. Matveev and A. Savkin, "The problem of state estimation via asynchronous communication channels with irregular transmission times," vol. 48, no. 4, 2003, pp. 670 – 676.