Data-based Optimisation for Adaptive Foot Placement in Dynamic Legged Locomotion

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

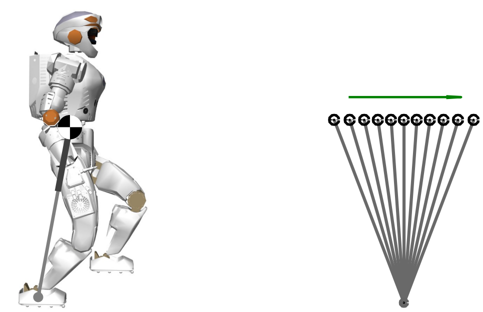
***Abstract*— This paper presents an online optimisation approach to control foot placement based on measured data for achieving desired walking velocity with minimum steady state error in legged robots. Model-based control approaches are widely used by the state-of-the-art methods for legged locomotion. However, discrepancies are inevitable due to sensors noise, delays, model mismatch, etc. which degrade the performance or even cause instabilities. To resolve these issues, particularly during dynamic walking, a data-based optimisation is proposed for better approximating the real robot dynamics. This method increases the accuracy of the obtained walking velocity and is particularly suited for situations where unexpected changes take place. Initially, a model-based approach uses the Linear Inverted Pendulum model for the foot placement control; After collecting adequate measurements, parameters are optimized to better represent the system dynamics within and in-between steps; Finally, the data-based optimisation takes over and is used to predict the next foot placement accurately in the presence of discrepancies.**

I. INTRODUCTION

Humanoid robots are designed with a morphology similar to that of humans offering a twofold advantage: ability to utilize tools designed based on the human anatomy [1], and ability to traverse environments easily accessible by humans such as stairs, passageways, rugged terrains etc [2]. They can be indispensable in emergency and disaster response situations, where other kinds of robots face severe limitations; for example after the Fukushima nuclear disaster [3].

In terms of mobility, a humanoid robot can be regarded as a floating-base system having two legs [4]. As a result, they are able to operate without considerations on the quality of the ground, exhibiting traits like adaptability and manoeuvrability [5]. In contrast, they require much more power and their mechanical complexity is usually much higher than that of other mobile robots.

As a result, bipedal locomotion is a field that draws more and more attention over the years. Kajita et al. [6] proposed the Linear Inverted Pendulum (LIP) model in order to generate motions under real-time constraints. The LIP model regards the robot as a point mass, with the total mass concentrated in the Centre of Mass (COM). The equations of the Inverted Pendulum become linear when the COM is kept at a constant height, which can be achieved by extending massless, telescopic legs. Given a target COM motion, a corresponding foot trajectory – in terms of the Zero Moment Point (ZMP) – can be generated to achieve it. LIP model and its extensions have been widely applied in bipedal walking as illustrated in Fig. 1.



(a) Valkyrie robot (b) COM motion

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

However, in classical model-based approaches, the initial values of the coefficients and parameters used to control the walking behaviour are manually tuned off-line and are fixed during walking. For example, Raibert [7] proposed a control approach based on the decoupled regulation of the hoping height, forward speed, and posture of an one-legged hopping robot. Based on experiments and observation, he proposed a simple – yet very efficient – state machine, which regulated hoping height by delivering a fixed vertical thrust during the support period, forward speed by moving the foot in an appropriate forward position, and posture by exerting a torque about the hip in order to keep the body upright. As a consequence, proper tuning of all variables was very crucial and usually relied on experience, which could be done only with a series of physical experiments. Also, the parameters might needed change under different situations in order to achieve an acceptable response. The same problems are present in other model-based approaches, especially when unexpected changes occur [8].

To overcome such limitations, a number of researchers explored approaches that allowed to auto-tune the parameters of fixed controllers [9] [10]. For example, You et al. [11] proposed a method that utilised linear regression for updating the model's coefficients in order to track a desired forward velocity. A law was devised that updated the coefficients based on previous measurements, where Raibert's model was used only within a certain period of steps, and then the updated coefficients were used to estimate the next foot placement. This method improved the system's flexibility to unexpected changes, such as adding an unknown mass offset. However, the convergence rate in You's method is significantly limited, because there are two coefficients that multiply the same measured velocity: one used directly for the velocity, and the other used for the term of the velocity

Preprint submitted to 2017 IEEE-RAS International Conference on Humanoid Robots.
Received July 25, 2017.

error. Hence, these coefficients were coupled, which resulted in fluctuation of their estimated values.

To overcome the limitations of the previous method and extend its applicability, we opt for an online estimation approach derived from the insights of the LIP model. The LIP model 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. Also, we expect better adaptation in situations where unexpected changes take place, such as an unknown mass offset. Finally, a more principled approach for handling the real-time sensory information for even more robust calculation of the coefficients is studied.

This paper's contribution is threefold:

- A rigorous analysis of how the errors propagate in walking control, and thus inevitable uncertainties of model-based methods which are manually solved by tuning.
- The problem formulation, taking a form derived by the model-based approach, and then applying regression to solve the optimisation problem analytically using measured data [12] [13].
- Different general expressions of the LIP model are used in order to avoid the coupling issue exhibited in Railbert's model. Moreover, in the regularisation part, the model coefficients of the LIP model serve as an nominal value and overly large deviations are penalised. Hence, the convergence of the model coefficients is guaranteed, while the rate of the convergence is increased.

The remainder of this paper is organized as follows: The limitations and inevitable uncertainties of current model-based methods are discussed in Section II. The proposed methodology is elaborated in Section III. Results obtained in simulation are presented in Section IV, followed by some final remarks and future directions in Section V.

II. Problem Statement

Model-based approaches constitute a powerful 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 in turn gives rise to a specific instantiation of the model. For example, according to the LIP model, given a certain transition time t, the COM motion can be computed based on the current COM state and by hyperbolic functions as follows [8]

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

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

where $\tau = t/T_c$ is the normalized transition time. The time constant $T_c = \sqrt{z_c/g}$ is defined by the constant COM height z_c assumed in the LIP model. The transition time t is the duration from the current state to the moment of interest. In the following, x_0 and $\dot{x}_0$ are the COM position and velocity at the current time instant, x_f and $\dot{x}_f$ are the COM position and velocity after the transition time t, while p^* is the position of the stance foot; all variables are expressed in a global coordinate system.

Tuning of the model's parameters requires substantial effort in model-based control approaches of bipedal walking [14]. Furthermore, a fixed set of model parameters is not adequate to capture the real-time response of actuation under all circumstances. Consequently, degradation of performance often occur due to errors and uncertainties introduced by the following sources:

- *Sensory errors*: All sensors have a number of limitations due to accuracy, resolution, bandwidth, etc. which might result in residuals or drifting in the measurements [15].
- *Delays*: Delays introduced by the communication architecture, signal processing etc. can degrade the performance of the control loop [16].
- *Model mismatch*: Models are always approximations of the real physical phenomena. For example, kinematic models assume that there is no bending and deformation of the mechanical structure. Also, some common forms of non-linearities (e.g. backlash) are very difficult to model.

To mitigate the performance degradation and extend the applicability of the models in more situations, we will exploit an underlying model that governs the general walking behaviour, but its specific instantiation will depend on the measurements. For a better understanding of the problem, we will analyse how errors propagate through the LIP model.

Specifically, we focus on how the current COM state error affects the prediction of future COM state and the resulting foot placement. The current COM state, i.e. COM position and velocity, as given by the sensor measurements $\tilde{x}_0$ and $\tilde{\dot{x}}_0$ can be decomposed into two parts: the true COM state x_0^{real} and $\dot{x}_0^{\text{real}}$ and the measurement errors e_{x_0} and $e_{\dot{x}_0}$,

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

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

In the following, [˜] and [ˆ] indicate measured and predicted variables, respectively.

To keep intuition and simplicity, we place the global frame during 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),

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

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

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

Preprint submitted to 2017 IEEE-RAS International Conference on Humanoid Robots.
Received July 25, 2017.

Clearly, (6) shows that the error propagates through the dynamics as time goes by.

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

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

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

Since the swing foot cannot be placed instantaneously, we need some look-ahead time τ_{la}. As a result, a predicted future velocity $\hat{\dot{x}}_f^n$ is needed. Substituting $\dot{x}_0^{n+1}$ in (7) by $\hat{\dot{x}}_f^n$ in (5), yields

$$p^* = \tilde{x}_0^{n+1} + T_c(\dot{x}_f^{\text{real}} + e_{\dot{x}_f}^n) \coth(\tau_s) - T_c \dot{x}_d^{n+1} \operatorname{csch}(\tau_s). \quad (8)$$

Note that p^* in (8) is defined in a global coordinate frame during the n-th step. However, a relative foot placement w.r.t. the body, defined as p without [*], is of more interest in terms of control

$$\begin{aligned} p^{n+1} &= p^{*,n+1} - \tilde{x}_0^{n+1} \\ &= \underbrace{T_c \dot{x}_f^{\text{real},n} \coth(\tau_s) - T_c \dot{x}_d^{n+1} \operatorname{csch}(\tau_s)}_{p^{\text{real},n+1}} + \underbrace{T_c \coth(\tau_s) \hat{e}_{\dot{x}_f}^n}_{\hat{e}_p^{n+1}} . \end{aligned} \quad (9)$$

Based on $\hat{e}_{\dot{x}_f}^n$ in (6), the uncertain error term $\hat{e}_p$ in (9) is

$$\begin{aligned} \hat{e}_p^{n+1} &= T_c \coth(\tau_s) \left[e_{x_0}^n \frac{\sinh(\tau_{la})}{T_c} + e_{\dot{x}_0}^n \cosh(\tau_{la}) \right] \\ &= \coth(\tau_s) \sinh(\tau_{la}) e_{x_0}^n + T_c \coth(\tau_s) \cosh(\tau_{la}) e_{\dot{x}_0}^n, \end{aligned} \quad (10)$$

which provides further insights on how the errors of the current COM state $(e_{x_0}^n,\ e_{\dot{x}_0}^n)$ propagate, and consequently downgrade the accuracy of the foothold prediction. The uncertainty is largely determined in an exponential manner by the look-ahead time τ_{la} and the allowable step time τ_s for achieving a desired walking velocity.

It can be inferred from (10) 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 can be a variable due to different robot configurations, e.g. squatting, standing, walking, or carrying a payload.

If we could obtain each term in (3) and (4) this situation would not constitute a problem: simple subtraction of the error terms e_{x_0} and $e_{\dot{x}_0}$ could correct the calculations. But, these terms are influenced by all the factors contributing to the performance degradation, so their calculation is practically impossible. Mitigating their influence requires some form of parameter adaptation using real-time information.

Thus, this paper suggests an online estimation strategy that considers the walking model as a grey box, rather than an exactly known function. Our proposed approach tunes the unknown terms that affect the foot placement estimation, with their value calculated from the measurements during walking. This way the uncertainties caused by errors, delays, and unmodelled quantities are taken into consideration.

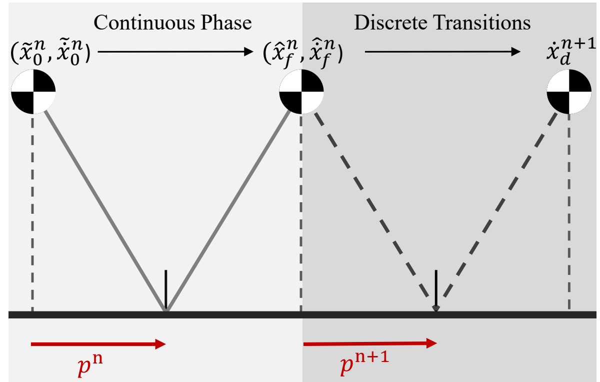


Fig. 2: Foot placement control for the $n+1$ step based on the COM state $(\tilde{x}_0^n,\ \dot{x}_0^n)$ at the current n step and the target velocity $\dot{x}_d^{n+1}$ at the $n+1$ step (sagittal scenario).

III. Foot Placement Control Based on Regularised Least Squares

Legged locomotion is a problem which is characterised by hybrid dynamics; that is, there are both *continuous* and *discrete* phases. In specific, legged locomotion can be viewed as the evolution of a continuous dynamical system which – during leg touch down or take off – undergoes discrete transitions.

As a result, our proposed optimisation approach (Fig. 2) is initially applied for estimating the state transition of the COM during the continuous phase (Section III-A). Afterwards, based on the predicted final velocity of the step, a similar optimisation problem is formulated which accounts for the discrete transitions. The result is the computation of an accurate foot placement based on the real-time sensory information, which achieves the desired walking velocity with minimum steady state error (Section III-B).

A. Optimisation of Velocity Estimation During a Step

While predicting the future COM state, uncertainty is inevitable; this can be recognised in (5). From the previous step we can create a dataset X_s which holds all the corresponding measurements $\tilde{x}_0$, $\tilde{\dot{x}}_0$, $\tilde{x}_f$, $\tilde{\dot{x}}_f$, and the final foot placement $\tilde{p}$.

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

$$\begin{aligned} \tilde{\dot{x}}_f^n &= \dot{x}_f^{\text{real},n} + e_{\dot{x}_f}^n \\ &= x_0^{\text{real},n} \frac{\sinh(\tau_{la})}{T_c} + \dot{x}_0^{\text{real},n} \cosh(\tau_{la}) + e_{\dot{x}_f}^n \\ &= (\tilde{x}_0^n - e_{x_0}^n) \frac{\sinh(\tau_{la})}{T_c} + (\tilde{\dot{x}}_0^n - e_{\dot{x}_0}^n) \cosh(\tau_{la}) + e_{\dot{x}_f}^n \\ &= \tilde{x}_0^n \frac{\sinh(\tau_{la})}{T_c} + \tilde{\dot{x}}_0^n \cosh(\tau_{la}) - e_{x_0}^n \frac{\sinh(\tau_{la})}{T_c} \\ &\quad - e_{\dot{x}_0}^n \cosh(\tau_{la}) + e_{\dot{x}_f}^n . \end{aligned} \quad (11)$$

Preprint submitted to 2017 IEEE-RAS International Conference on Humanoid Robots.
Received July 25, 2017.

Assuming that the measured foot placement is given w.r.t. the COM, i.e. $\tilde{p} = -\tilde{x}_0$, we can express the initial and final velocity as

$$\tilde{\dot{x}}_f^n = -\tilde{p}^n \frac{\sinh(\tau_s)}{T_c} + \tilde{\dot{x}}_0^n \cosh(\tau_s) - e_{x_0}^n \frac{\sinh(\tau_s)}{T_c} - e_{\dot{x}_0}^n \cosh(\tau_s) + e_{\dot{x}_f}^n. \quad (12)$$

Thus, (12) can be expressed in a more general form by defining a vector of coefficients $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \alpha_3]^T$ as

$$\tilde{\dot{x}}_f = \begin{bmatrix} -\tilde{p}^n & \tilde{\dot{x}}_0^n & 1 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} = -\tilde{p}^n \alpha_1 + \tilde{\dot{x}}_0^n \alpha_2 + \alpha_3, \quad (13)$$

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 (13) 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

$$\mathbf{x}_f = \begin{bmatrix} \tilde{\dot{x}}_f^{n-k} \\ \vdots \\ \tilde{\dot{x}}_f^{n-1} \end{bmatrix}_{k\times 1}, \quad \mathbf{X}_1 = \begin{bmatrix} -\tilde{p}^{n-k} & \tilde{\dot{x}}_0^{n-k} & 1 \\ \vdots & \vdots & \vdots \\ -\tilde{p}^{n-1} & \tilde{\dot{x}}_0^{n-1} & 1 \end{bmatrix}_{k\times 3}. \quad (14)$$

We would like to solve for $\boldsymbol{\alpha}$ in a way which reflects the dynamics in the collected data, while having minimum deviation from the values calculated by the LIP model. This can be achieved by introducing a penalised least-squares problem of the form

$$\min_{\boldsymbol{\alpha}} \left\|\mathbf{X}_1 \boldsymbol{\alpha} - \mathbf{x}_f\right\|_{\mathbf{P}_1}^2 + \|\boldsymbol{\alpha} - \boldsymbol{\alpha}_0\|_{\mathbf{Q}_1}^2, \quad (15)$$

where $\|\cdot\|_{\mathbf{M}}^2$ denotes a weighted euclidean norm.

It shall be noted that in our formulation the second term $\|\boldsymbol{\alpha} - \boldsymbol{\alpha}_0\|_{\mathbf{Q}_1}^2$ is important, because our study found that in the prior work [11], using only the least square term sometimes produced undesirable fluctuation of $\boldsymbol{\alpha}$. With (15), we guarantee that $\boldsymbol{\alpha}_0$ serves as an initial guess and a very large deviation is penalised.

The minimisation problem expressed in (15) is also known as Tikhonov regularisation [12] [13]. The closed-form solution can be readily computed as

$$\boldsymbol{\alpha} = \boldsymbol{\alpha}_0 + \left[\mathbf{X}_1^T \mathbf{P}_1 \mathbf{X}_1 + \mathbf{Q}_1\right]^{-1} \left[\mathbf{X}_1^T \mathbf{P}_1 (\mathbf{x}_f - \mathbf{X}_1 \boldsymbol{\alpha}_0)\right], \quad (16)$$

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

$$\mathbf{P}_1 = G_P \begin{bmatrix} w_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & w_k \end{bmatrix}, \mathbf{Q}_1 = G_Q \begin{bmatrix} w_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & w_k \end{bmatrix}. \quad (17)$$

For the regression term, G_P is used to weight the influence of the regression part and $w_i = i$ is a weight for the walking state in $\mathbf{X}_1$, where i is the number of samples as well as the index of the weight in the matrix, and k represents the latest data. For the regularisation term, G_Q is used to weight the influence of the regularisation part, and $w_i = i$ serves the same purpose as before.

By solving (16), we obtain an $\boldsymbol{\alpha}$ that best approximates the state transition expressed by the collected data until the $n-1$ step. Once the n step starts, we can measure $\mathbf{x}^n = \begin{bmatrix} -\tilde{p}^n & \tilde{\dot{x}}_0^n & 1 \end{bmatrix}$ and predict the future velocity $\hat{\dot{x}}_f^n$ at the end of the n step by

$$\hat{\dot{x}}_f^n = \mathbf{x}^n \boldsymbol{\alpha}. \quad (18)$$

B. Optimisation of the Foot Placement for the Next Step

Section III-A describes the optimisation 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 optimisation. This is essential because once $\hat{\dot{x}}_f^n$ is calculated by (18), 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 optimisation for estimating the unknown error terms, we mitigate the degradation effecting both sensing and control.

For each step that has happened, we can measure $\tilde{\dot{x}}_f^{n-1}$, $\tilde{\dot{x}}_f^n$, and $\tilde{p}^n$. From (9), the measured foot placement of the step expressed w.r.t. the stance foot can be described as

$$\begin{aligned} \tilde{p}^n &= p^{\text{real},n} + e_p^n \\ &= T_c \dot{x}_f^{\text{real},n-1} \coth(\tau_s) - T_c \dot{x}_f^{\text{real},n} \operatorname{csch}(\tau_s) + e_p^n \\ &= T_c \left(\tilde{\dot{x}}_f^{n-1} - e_{\dot{x}_f}^{n-1}\right) \coth(\tau_s) - T_c \left(\tilde{\dot{x}}_f^n - e_{\dot{x}_f}^n\right) \operatorname{csch}(\tau_s) + e_p^n \\ &= T_c \tilde{\dot{x}}_f^{n-1} \coth(\tau_s) - T_c \tilde{\dot{x}}_f^n \operatorname{csch}(\tau_s) \\ &\quad - T_c e_{\dot{x}_f}^{n-1} \coth(\tau_s) + T_c e_{\dot{x}_f}^n \operatorname{csch}(\tau_s) + e_p^n, \end{aligned} \quad (19)$$

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 estimation formula in (19) can be expressed in a general form using coefficients β_1, β_2, and β_3 as

$$p^n = \beta_1 \tilde{\dot{x}}_f^{n-1} + \beta_2 \tilde{\dot{x}}_f^n + \beta_3, \quad (20)$$

where β_1 and β_2 replace the model-based coefficients, and β_3 accounts for the error terms expressed in (6) and (10).

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 past k steps and its consecutive. The vector $\mathbf{p}$ is the concatenation of the corresponding foot placement locations

$$\mathbf{p} = \begin{bmatrix} \tilde{p}^{n-k} \\ \vdots \\ \tilde{p}^{n-1} \end{bmatrix}_{k\times 1}, \mathbf{X}_2 = \begin{bmatrix} \tilde{\dot{x}}_f^{n-k} & \tilde{\dot{x}}_f^{n-k+1} & 1 \\ \vdots & \vdots & \vdots \\ \tilde{\dot{x}}_f^{n-1} & \tilde{\dot{x}}_f^n & 1 \end{bmatrix}_{k\times 3}. \quad (21)$$

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

$$\min_{\boldsymbol{\beta}} \|\mathbf{X}_2 \boldsymbol{\beta} - \mathbf{p}\|_{\mathbf{P}_2}^2 + \|\boldsymbol{\beta} - \boldsymbol{\beta}_0\|_{\mathbf{Q}_2}^2. \quad (22)$$

Preprint submitted to 2017 IEEE-RAS International Conference on Humanoid Robots.
Received July 25, 2017.

The solution and the weighting matrices $\mathbf{P}_2$ and $\mathbf{Q}_2$ are defined similarly with the ones in (16) and (17), respectively. After calculating $\boldsymbol{\beta}$, the next foot placement is given by

$$p^{n+1} = \mathbf{x}^{n+1}\boldsymbol{\beta}, \tag{23}$$

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

C. Implementation details of the 2-stage Optimisation

The process described below presents the implementation of the continuous transition "$\dot{x}_0^n \rightarrow \dot{x}_f^n$", and the step-to-step transition "$\dot{x}_f^n \rightarrow \dot{x}_f^{n+1}$" during the initial stage, i.e. the stage before a sufficient number of samples is acquired, and the optimisation once online estimation starts.

Step 1: Data generation using a fixed model: During the initial stage, a fixed model – the LIP model in this work – is used in order to predict the final velocity $\hat{\dot{x}}_f$ of the current step based on the COM state $\tilde{x}_0$ and $\tilde{\dot{x}}_0$ of each step, and then calculate the foot placement location p; that is we collect data performing a usual model-based approach with predefined parameters.

Step 2: Selection, storage, and update of the dataset X_s: In the beginning of every step the COM position $\tilde{x}_0^i$ and velocity $\tilde{\dot{x}}_0^i$, as measured by the sensors, are inserted in the dataset X_s. Consequently, the COM velocity at the beginning of the next step $i+1$ is stored as the final velocity of the current step $\tilde{\dot{x}}_0^{i+1} = \tilde{\dot{x}}_f^i$. Furthermore, the relative foot placement location that resulted in the measured initial and final velocity of the step is stored in the dataset too. It can be used to obtain matrices $\mathbf{X}_1$ and $\mathbf{X}_2$.

The dataset is updated with the measurements at the touch-down moment of the swing foot. The update law is currently implemented using a fixed size First In First Out (FIFO) data structure. The optimal selection of the dataset size k will be further elaborated in future work in order to achieve good prediction results.

Step 3: Estimation of the model coefficients: Based on the data available, we apply the Tikhonov regularisation method as already proposed in Section III-A and Section III-B.

To sum up the approach, once the online estimation starts, matrices $\mathbf{X}_1$ and $\mathbf{X}_2$ are formed using a fixed number of the most recent measurements. The matrix $\mathbf{X}_1$ contains the relative COM positions and velocities at the beginning of every step, while the vector $\mathbf{x}_f$ is the concatenation of the final velocities from the past k_{vel} steps. Then, $\mathbf{X}_1$ and $\mathbf{x}_f$ are used to estimate the optimal model coefficients $\boldsymbol{\alpha}$ in the continuous transition "$\dot{x}_0^n \rightarrow \dot{x}_f^n$" by (16).

Similarly, matrix $\mathbf{X}_2$ contains the COM velocities at the end of each of the k_{fp} steps and its consecutive, 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 (22).

Step 4: Prediction of the next foot placement: While updating the estimates of the model's coefficients $\boldsymbol{\alpha}$ 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 $\hat{\dot{x}}_f^n$ can be estimated using $\mathbf{x}^n$ and the model coefficients $\boldsymbol{\alpha}$ based on (18). Afterwards, $\hat{\dot{x}}_f^n$ will used in $\mathbf{x}^{n+1}$ to control the next foot placement p^{n+1} together with model coefficients $\boldsymbol{\beta}$ in order to achieve the target velocity $\dot{x}_d^{n+1}$ in (23).

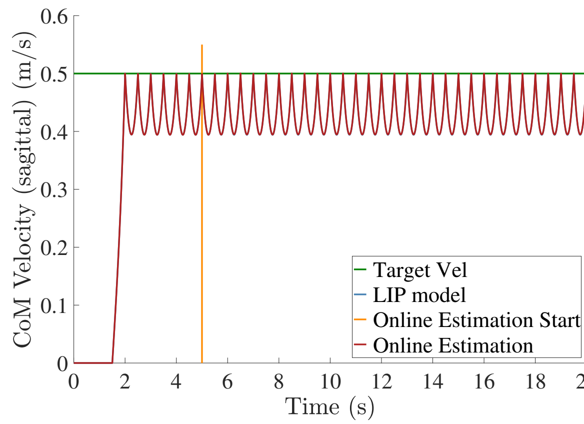


Fig. 3: Sagittal velocity profile generated by LIP model and online estimation to reach target velocity in *Case 1*

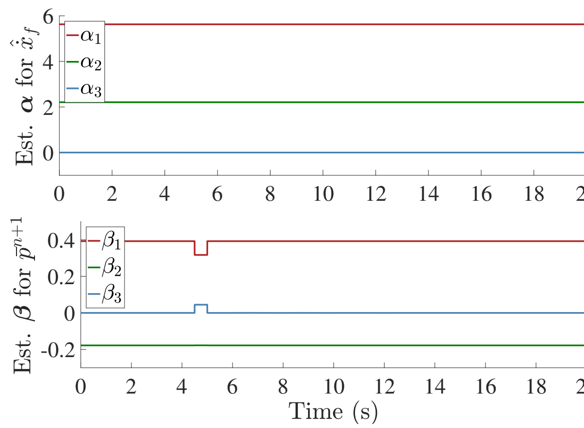


Fig. 4: Model coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ of online estimation in *Case 1*

The dataset and the model coefficients in *Step 2*, *Step 3*, and *Step 4* are recursively updated so as to obtain the optimal model coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, which are used to reach the target velocity with minimum steady state error.

IV. Simulation

The proposed approach can be validated by comparing against traditional models with the following criteria:

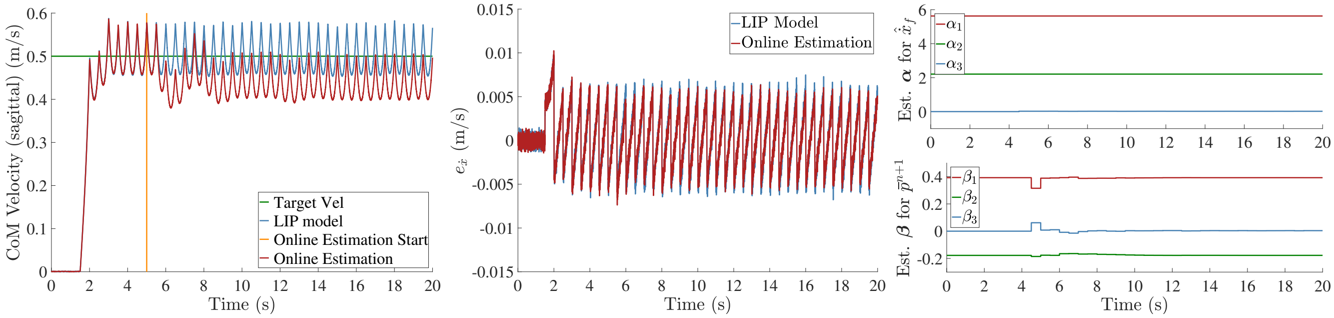
- Accuracy of the next foot placement prediction.
- The convergence rate of the model coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$.
- The robustness of the robot during walking subject to filtering delay and unknown mass offset.

In our simulation environment, the constant height of the LIP model is set at $1.2m$, matching the COM height of the humanoid robot Valkyrie, while the step time is set at $0.5s$, with a $0.5m/s$ target velocity at the end of each step. The results of the 2-stage approach for online estimation introduced in Section III is compared to the results by the LIP model. Note that, a size $k_{vel} = 2$ is used during the

Preprint submitted to 2017 IEEE-RAS International Conference on Humanoid Robots.
Received July 25, 2017.

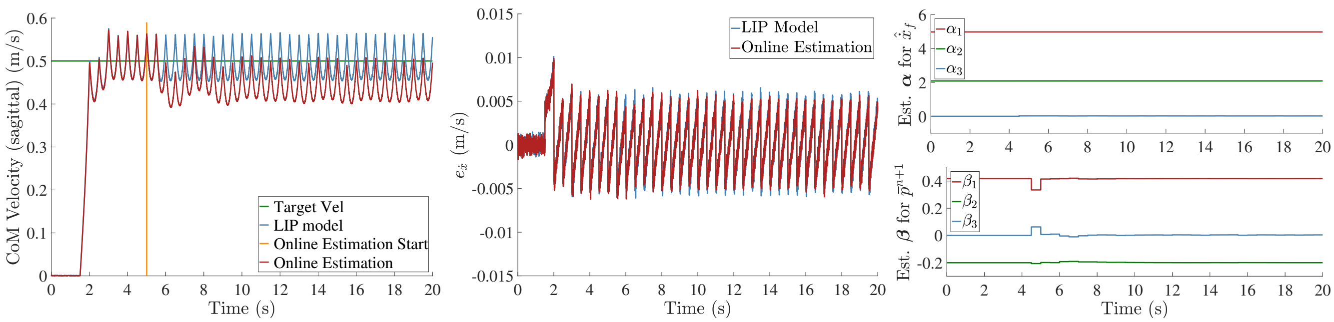
TABLE I: Simulation setup and results.

Cases	Noise and delay	COM offset	Methods	Steady state error (m/s)	$\boldsymbol{\alpha}$	$\boldsymbol{\beta}$
Case 1	None	None	LIP model	0	$[5.63, 2.21, 0]$	$[0.392, -1.78, 0]$
			Online Estimation	0	$[5.63, 2.21, -2.42 \times 10^{-4}]$	$[0.392, -0.178, -4.37 \times 10^{-5}]$
Case 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}	$[5.63, 2.21, 2.00 \times 10^{-2}]$	$[0.392, -0.178, 3.66 \times 10^{-3}]$
Case 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}	$[4.98, 2.08, 1.84 \times 10^{-2}]$	$[0.418, -0.201, 2.88 \times 10^{-3}]$
Case 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}	$[5.63, 2.21, 7.51 \times 10^{-2}]$	$[0.392, -0.176, 1.24 \times 10^{-2}]$



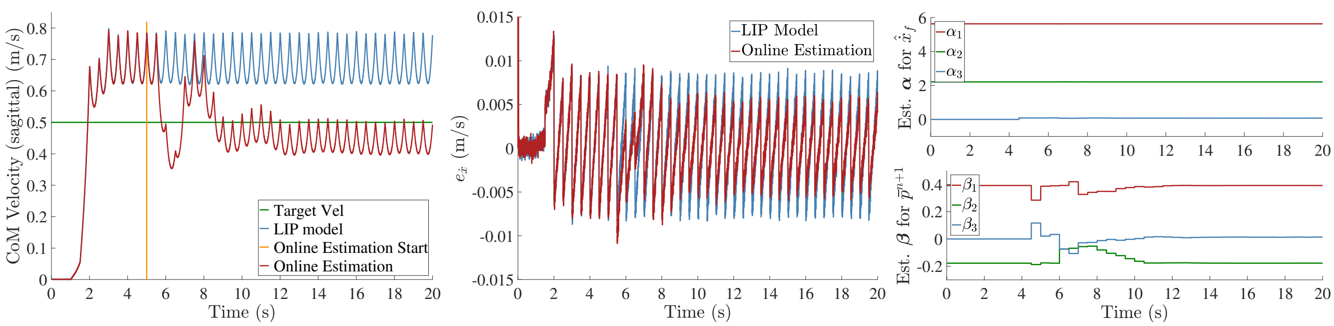
(a) Velocity Profile. (b) COM velocity error. (c) Coefficient estimation.

Fig. 5: Result of Case 2 with 110dB noise and filtering delay.



(a) Velocity Profile. (b) COM velocity error. (c) Coefficient estimation.

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



(a) Velocity Profile. (b) COM velocity error. (c) Coefficient estimation.

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

continuous transition "$\dot{x}_0^n \rightarrow \dot{x}_f^n$", while a dataset size of $k_{fp} = 6$ is used during the step-to-step transition "$\dot{x}_f^n \rightarrow \dot{x}_f^{n+1}$". Various types of errors are introduced in order to examine the robustness of the proposed method, including noise with a $110dB$ signal-to-noise ratio (SNR), error and delay in filtering, constant vertical COM offset and horizontal COM offset. A detailed comparison between the LIP method and the online estimation method has been carried out in

Preprint submitted to 2017 IEEE-RAS International Conference on Humanoid Robots.
Received July 25, 2017.

cases with different levels of noise.

A set of four simulations under different scenarios has been carried out to evaluate the performance of the proposed method, as highlighted in Table I. Note that the velocity profile within a step is a curve where the robot starts with a initial velocity $\dot{x}_0$ at the beginning of the step, decreases to minimum velocity when the COM position of the robot is vertically above the position of the stance foot, and increases up to a new maximum velocity $\dot{x}_f$ at end of the step. We are interested in whether the robot can reach the target velocity $\dot{x}^d$ at the end of each step, instead of the average velocity within the step. The optimal value of the coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ that affect the continuous transition "$\dot{x}_0^n \rightarrow \dot{x}_f^n$", and the step-to-step transition "$\dot{x}_f^n \rightarrow \dot{x}_f^{n+1}$" is also summarised.

First of all, as shown in Fig. 3 and Fig. 4, simulations on an ideal case (Case 1) with no noise, delay and COM offset are performed as a baseline, showing that both the LIP model and the Online Estimation method can achieve the desired velocity. The estimated coefficients agree with the analytical solution calculated using LIP model. In the second case, we introduce noise of $110dB$ and filtering delay into the system, which causes an immediate $8.17\times10^{-2}m/s$ steady state error to LIP model, as shown in Fig. 5a. Meanwhile, after a small fluctuation between $5s$ to $10s$the online estimation method managed to reach the desired velocity with a negligible error, $0.0002m/s$. In the third case, we further introduce a 10% vertical COM offset, as highlighted in Fig. 6. The result shows that the proposed method is able to compensate not only for noise and delay, but also for the COM offset. Besides, as the constant height increased by the vertical COM offset, the time constant T_c in (2) increases. Hence, the initial value of the model coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ decreases as shown in (11) and (19), and the effect of those sources of error diminishes. That is why the steady error in the LIP model deceases compared with Case 2. Finally, a fourth simulation is performed by applying a $-0.01m$ horizontal COM offset. As shown in Fig. 7, although there exists a fluctuation in the estimated coefficients, online estimation method can still achieve the desired COM velocity with low steady state error after $12s$, whereas such an COM offset has caused a $0.28m/s$ steady state error to the velocity profile generated by the LIP model.

V. Conclusions

In this work, foot placement based on regularised least squares is introduced in order to tackle some common forms of errors that plague model-based approaches. Our method assumes unknown terms that affect the actual foot placement, while the exact value of those terms is on-line estimated from the real-time measurements during walking. The robustness of the proposed approach under various types of errors is validated through simulations in four different scenarios. Compared to traditional model-based methods, we were able to reach the commanded velocities with minimum steady state error, as illustrated in Section IV.

This work proved the feasibility of the proposed method. Currently, FIFO is used in order to update the dataset as already mentioned. In the future, we plan to extend the current approach into a 3D point mass simulation, which will include both sagittal and lateral motion. Finally, we are planning to validate it in a real humanoid robot.

Besides, we also plan to consider the centroidal angular momentum. For example, the Angular Momentum Pendulum Model (AMPM) proposed by Komura et.al [17], or the LIP plus flywheel (LIPF) model developed by J.Pratt [18] will be investigated.

References

[1] 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, p. 178, 2012.

[2] J. Denny, M. Elyas, S. A. Dcosta, and R. D. DSouza, "Humanoid robots–past, present and the future," *European Journal of Advances in Engineering and Technology*, vol. 3, no. 5, pp. 8–15, 2016.

[3] K. Kaneko, M. Morisawa, S. Kajita, S. Nakaoka, T. Sakaguchi, R. Cisneros, and F. Kanehiro, "Humanoid robot hrp-2kaiimprovement of hrp-2 towards disaster response tasks," in *Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on*. IEEE, 2015, pp. 132–139.

[4] N. B. Ignell, N. Rasmusson, and J. Matsson, "An overview of legged and wheeled robotic locomotion," *Available from: Mälardalen University, Web site: http://www. idt. mdh. se/kurser/ct3340/ht12/MINICONFERENCE/FinalPapers/i rcse12 sub mission*, vol. 21, 2012.

[5] 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, p. 35, 2016.

[6] S. Kajita, O. Matsumoto, and M. Saigo, "Real-time 3d walking pattern generation for a biped robot with telescopic legs," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, vol. 3, 2001, pp. 2299–2306 vol.3.

[7] M. H. Raibert, *Legged robots that balance*. MIT press, 1986.

[8] S. Kajita, H. Hirukawa, K. Harada, and K. Yokoi, *Introduction to humanoid robotics*. Springer, 2014, vol. 101.

[9] C.-M. Chew, "Dynamic bipedal walking assisted by learning," Ph.D. dissertation, Massachusetts Institute of Technology, 2000.

[10] L. T. Russell, "Applied optimal control for dynamically stable legged locomotion," Ph.D. dissertation, PhD thesis, Massachusetts Institute of Technology, 2004.

[11] Y. You, Z. Li, D. G. Caldwell, and N. G. Tsagarakis, "From one-legged hopping to bipedal running and walking: A unified foot placement control based on regression analysis," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2015-Decem, no. 1, pp. 4492–4497, 2015.

[12] A. N. Tikhonov, V. I. Arsenin, and F. John, *Solutions of ill-posed problems*. Winston Washington, DC, 1977, vol. 14.

[13] A. N. Tikhonov, A. Goncharsky, V. Stepanov, and A. G. Yagola, *Numerical methods for the solution of ill-posed problems*. Springer Science & Business Media, 2013, vol. 328.

[14] J. Pratt and G. Pratt, "Intuitive control of a planar bipedal walking robot," in *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, vol. 3. IEEE, 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, p. 26, 2016.

[16] A. S. Matveev and A. V. Savkin, "The problem of state estimation via asynchronous communication channels with irregular transmission times," *IEEE Transactions on Automatic Control*, vol. 48, no. 4, pp. 670–676, 2003.

[17] T. Komura, A. Nagano, H. Leung, and Y. Shinagawa, "Simulating pathological gait using the enhanced linear inverted pendulum model," *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 9, pp. 1502–1513, Sept 2005.

[18] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture point: A step toward humanoid push recovery," in *2006 6th IEEE-RAS International Conference on Humanoid Robots*, Dec 2006, pp. 200–207.

Preprint submitted to 2017 IEEE-RAS International Conference on Humanoid Robots.
Received July 25, 2017.