2024 European Control Conference (ECC)
June 25-28, 2024. Stockholm, Sweden

Model-based unified state and phase estimation for torque actuated dissipative spring-mass runner using limited sensory information

Osman Kaan Karagoz[1,2], Aysegul Kilic[2] and Mustafa Mert Ankarali[1].

***Abstract*— In the field of autonomous legged robotics, accurate state estimation is crucial for control and planning. While traditional methods suffice for fully-actuated platforms, under-actuated systems face challenges due to sensory limitations and uncertainties. This paper presents a novel methodology for state estimation and phase prediction, integrating a torque-actuated spring-mass model with limited sensors using a multiple-hypotheses extended Kalman filter. Within this estimation framework, the optimal estimate is determined at each iteration by evaluating the likelihood functions associated with two distinct phase hypotheses, either stance or flight. We evaluate different sensor and motion model combinations, showing that our method achieves precise state and phase estimation even without advanced sensors for compliant and under-actuated platforms.**

I. INTRODUCTION

In terrestrial environments, legged robots offer advantages over wheeled or tracked ones, particularly in unstructured surroundings [1]. The effectiveness of locomotion in such robots critically depends on accurate phase and state estimation.

While numerous studies have addressed state estimation in the context of legged robotics, the primary focus of these investigations has typically centered on fully actuated and comprehensively sensed systems. In such setups, where rich kinematic models and abundant measurement data are at their disposal, successful implementations of state estimation have been achieved [2]–[4].

However, for legged platforms that incorporate elastic and soft materials in their linkages to achieve energy efficiency and robustness [5]–[7], such approaches are not always viable. The use of elastic materials introduces additional uncertainties and complexities in control and estimation tasks, rendering the system not entirely observable through kinematic information alone. In such cases, traditional kinematic models often fail to effectively capture the system's dynamics.

One potential solution involves enhancing sensory subsystems with exteroceptive sensors like depth, vision, GPS, inertial measurement units (IMUs), force sensors, or range sensors [8]–[10]. These sensors can indirectly mitigate the kinematic unobservability problem. However, this approach comes with trade-offs, including increased cost and hardware complexity. Moreover, these sensors are often susceptible to environmental conditions that can hamper their effectiveness, such as light intensity variations, reflective surfaces, or temperature fluctuations.

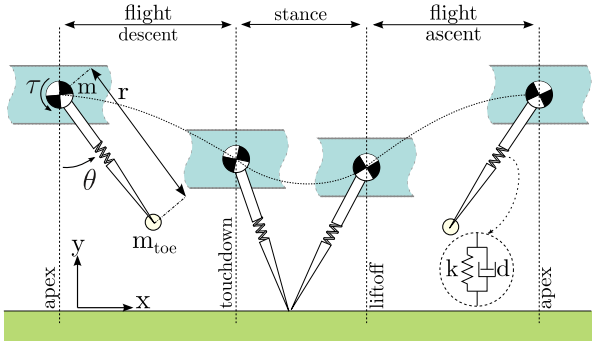

Fig. 1. Enhanced Torque-Actuated Dissipative Spring-Loaded Inverted Pendulum (TD-SLIP) model [11] with added toe mass and DC motor at the hip joint. Also, this figure illustrates the progression of a single step in the TD-SLIP model, highlighting the different locomotion phases and key transition events.

Furthermore, the conventional approach to the estimation problem in legged robots typically involves segregating ground contact estimation from state estimation, leading to the development of multi-step strategies. These strategies employ separate estimation blocks to tackle these two problems individually [3], [12], [13]. Attempting to decouple these problems and merging the individual blocks often introduces compatibility issues, including initialization and calibration challenges.

This paper aims to tackle the challenge of accurately estimating the phase (ground contact) and system states for under-actuated and compliant legged robots, even when equipped with minimal sensory information.

Our approach relies on mathematical modeling of the system, allowing us to develop a state estimation framework that uses minimal proprioceptive sensors, even when the robot incorporates compliant structures. Proprioceptive sensor readings include primary DC motor encoders and current measurements inherent to the system. This strategy circumvents the challenges of integrating exteroceptive sensors or other proprioceptive sensors. Similar to the approach in [12], this work emphasizes a model-based state estimation framework without requiring an extensive array of sensors.

Additionally, we propose a methodology that enables us to estimate ground contact detection and state within a single framework. We design and implement dynamic observers

All authors are with Department of Electrical and Electronics Engineering, Middle East Technical University, 06800 Ankara, Turkey: {karagoz.osman@metu.edu.tr, aysegul.kilic@metu.edu.tr, mertan@metu.edu.tr}

[1]Authors are also with Robotics and Artificial Intelligence Technologies Application and Research Center (ROMER), Middle East Technical University, 06800 Ankara, Turkey.

[2]Authors are contributed equally in this work.

tailored to different phases of the system and combine this information using probabilistic techniques to determine the system's current mode and acquire state information.

Our work is centered around a locomotion model (see Fig. 1) based on the torque-actuated spring-loaded inverted pendulum (TD-SLIP) model with damping [11]. This model represents an elaborate and physically realistic extension of the spring-loaded inverted pendulum (SLIP) model [14]–[16], addressing the complexities of real legged robotic platforms. Using this proposed model, we compare various sensor inputs and motion models and evaluate estimation performance.

In the subsequent sections, we detail the model dynamics and the proposed estimation framework in Section II. We present the results obtained from our simulations in Section III. Finally, we conclude our paper in Section IV.

II. STATE ESTIMATION WITH THE TD-SLIP MODEL

A. TD-SLIP Model and Dynamics

The Spring-Loaded Inverted Pendulum (SLIP) model serves as a fundamental bridge connecting the fields of biology and engineering. It is a widely studied dynamic model used to explain the running behavior of animals and also serves as a foundational design reference and control objective for robotic systems [14], [17], [18]. The appeal of the SLIP model lies in its simplicity, but its highly abstract structure can create challenges when translated into model-based controllers for physical robotic platforms. These inaccuracies can pose significant challenges for model-based controllers and estimators [13].

To overcome these limitations, researchers have proposed various extensions to the SLIP model, enhancing its applicability to robotic platforms. These extensions address body dynamics [19], [20] and dissipative and actuator components [21], [22], enabling its adaptation to complex models with multiple legs and bodies. In this study, we extend a variant of the SLIP model known as the TD-SLIP model, originally introduced in [11]. This model strikes a balance between simplicity and extensibility, making it well-suited for fundamental analyses and more complex representations involving multiple legs and bodies.

Unlike the conventional SLIP model, the TD-SLIP model comprises a planar rigid body with a fixed orientation (two degrees of freedom) linked to a toe by an entirely passive leg. This leg incorporates a linear spring k and linear viscous damping d, with a rest length (r_0), as shown in Fig. 1. This model's only actuated degree of freedom is the revolute joint located at the hip position.

In this work, we introduce two minor extensions to the TD-SLIP model proposed in [11]. First, we incorporate a toe mass m_t positioned at the ground-tip of the TD-SLIP, which is relatively smaller than the body mass. Additionally, unlike the prevalent assumption in the legged robotics community, we do not presume the existence of an infinitely fast actuator capable of reactively regulating the hip-torque. Instead, we integrate an armature voltage-controlled DC motor that generates the hip-torque based on voltage commands and internal electrical dynamics.

TABLE I
NOTATION USED THROUGHOUT THE PAPER

System Parameters	
$r, \theta, \dot{r}, \dot{\theta}$	Leg length, angle and respective velocities
$x, y, \dot{x}, \dot{y}$	Body COM cartesian positions and velocities
k, d, r_0	Leg stiffness, leg damping and rest leg length
m, m_t, g	Body mass, toe mass, gravitational acceleration
Motor Parameters	
V_m, i	Armature voltage and current
K_t, K_e	Torque and back electromotive force constant
R_a, L_a	Armature resistance and inductance
b	Motor damping
Estimation Procedure Parameters	
T_s	Sampling period
$\mathbf{A}_k^{s,f}, \mathbf{C}^{o,m}, \mathbf{G}_k^{s,f}$	System, output and process noise matrices
$\mathbf{w}_k^{s,f}, \mathbf{v}_k^{o,m}$	Process and measurement noise
$\mathbf{Q}^{s,f}, \mathbf{R}^{o,m}$	Process and measurement noise covariances
$\hat{\mathbf{q}}_{\alpha\|\beta}, \mathbf{P}_{\alpha\|\beta}$	Estimate and covariance of the states at time α given observations up to β

Table I provides an overview of the system, motor, and estimation procedure parameters, with the parameter values selected based on the platform developed by Uyanik et al. [23], including values such as $m = 3.21\ kg$, $m_t = 0.19\ kg$, $k = 3570\ N/m$, $r_0 = 0.22\ m$, and $d = 9.9\ kg/s$. Gravitational acceleration g is set at $9.8\ m/s^2$.

We approach the TD-SLIP model by examining two distinct phases: stance and flight (Fig. 1). Following a similar methodology to locomotion models used for the analysis of compliant legged robotic systems [18], [24], during the flight phase, we assume that compliant leg forces do not influence body dynamics but instead govern the dynamics of the toe mass, capitalizing on the substantial difference in mass between the body and leg. Consequently, in this context, the body exhibits ballistic flight dynamics, and we can formulate the equations of motion as follows:

$$\begin{aligned} \ddot{r} &= r\dot{\theta}^2 + \frac{k(r - r_o)}{m_t} - \frac{d\dot{r}}{m_t}, \\ \ddot{\theta} &= -\frac{2\dot{r}\dot{\theta}}{r} + \frac{K_t i}{m_t r^2} - \frac{b\dot{\theta}}{m_t r^2}, \\ \ddot{x} &= 0,\ \ddot{y} = -g. \end{aligned} \tag{1}$$

In this context, the flight phase states pertaining to the main body, represented by $[x,\ \dot{x},\ y,\ \dot{y}]$, become both uncontrollable and unobservable.

In the stance phase, we assume that the toe mass dynamics are inactive and the touchdown point behaves as a pin joint without slippage. Therefore, we solely consider the body dynamics during the stance phase, as articulated in the following stance dynamics:

$$\begin{aligned} \ddot{r} &= r\dot{\theta}^2 + \frac{k(r_o - r)}{m} - g\cos\theta - \frac{d\dot{r}}{m} \\ \ddot{\theta} &= \frac{g\sin\theta}{r} - \frac{2\dot{r}\dot{\theta}}{r} + \frac{K_t i}{mr^2} - \frac{b\dot{\theta}}{mr^2}. \end{aligned} \tag{2}$$

Furthermore, we incorporate the governing equation for the motor as

$$\dot{i} = -\frac{K_e \dot{\theta}}{L_a} - \frac{R_a i}{L_a} + \frac{V_m}{L_a}, \tag{3}$$

where V_m serves as the control input to our system, along with the touchdown angle, regulated by a proportional-derivative (PD) controller. This controller aims to track a time-based reference trajectory defined for the toe mass as in [25], ensuring the system's running behavior is sustained through open-loop clock-driven voltage commands.

With these dynamic equations in place, we can define the observable states regardless of the phase as $\mathbf{q} := \left[r, \theta, i, \dot{r}, \dot{\theta}\right]^T$. To describe the complete system trajectory, we introduce $\mathbf{q_d} := \left[\mathbf{q}^T, x, y, \dot{x}, \dot{y}\right]^T$ and define the nonlinear continuous-time differential equation of motion as follows:

$$\dot{\mathbf{q}}_\mathbf{d}(t) = h^{s,f}(\mathbf{q_d}(t), u(t), \mathbf{w}^{s,f}(t)), \tag{4}$$

This equation incorporates the dynamics presented in (1), (2), and (3). The superscripts s and f denote stance and flight phases, while $u(t)$ represents the controller input.

B. Estimation Framework

We employ an Extended Kalman Filter (EKF) to fuse available sensory readings and perform hybrid state estimation. To create a more realistic experimental environment, we assume that filters and sensors operate at the sampling period T_s, ensuring that all readings are *synchronous.*

Between two sampling instants, we can define the discrete motion model and observation equations as follows:

$$\mathbf{q}_{k+1} = g_k^{s,f}(\boldsymbol{q}_k, u_k, \boldsymbol{w}_k^{s,f}) \tag{5}$$

$$\boldsymbol{z}_k = \boldsymbol{C}^{o,m}\boldsymbol{q}_k + \boldsymbol{v}_k^{o,m}. \tag{6}$$

Here, the superscript o, m denotes different selections of sensor models, which will be further explained in Section II-D. $g_k^{s,f}$ represents the nonlinear discrete-time state transition function, which depends on the motion model. Noises $\mathbf{w}_k^{s,f}$ and $\mathbf{v}_k^{o,m}$ are assumed to be zero-mean multivariate Gaussian noises with covariances $\mathbf{Q}^{s,f}$ and $\mathbf{R}^{o,m}$, respectively. While the vector $\mathbf{w}_k^{s,f}$ influences the leg force and leg torque, $\mathbf{v}_k^{o,m}$ represents additive noise present in the measurement.

The estimation procedure begins with both filters having the same foreknown initial condition. As shown in Fig. 2, both filters adopt the same noisy sensor model $\boldsymbol{C}^{o,m}$ and utilize the same motion model with different phases, which are denoted as $\boldsymbol{A}_k^{s,f}$, $[\boldsymbol{G}_k^{s,f}$, and $g_k^{s,f}$, to estimate the same states $\hat{\boldsymbol{q}}_{k|k}^{s,f} := [r, \theta, i, \dot{r}, \dot{\theta}]^T$. For each time step, there are two hypotheses regarding the best state estimate, either $\hat{\boldsymbol{q}}_{k|k}^{s}$ or $\hat{\boldsymbol{q}}_{k|k}^{f}$, which the likelihood-based phase identification block will later determine.

Using the measurement innovation and innovation covariances, defined as:

$$\tilde{\boldsymbol{z}}_k^{s,f} := \boldsymbol{z}_k^{s,f} - \boldsymbol{C}^{o,m}\hat{\boldsymbol{q}}_{k|k-1}^{s,f},$$

$$\boldsymbol{S}_k^{s,f} := \boldsymbol{C}^{o,m}\boldsymbol{P}_{k|k-1}\boldsymbol{C}^{o,m^T} + \boldsymbol{R}^{o,m}$$

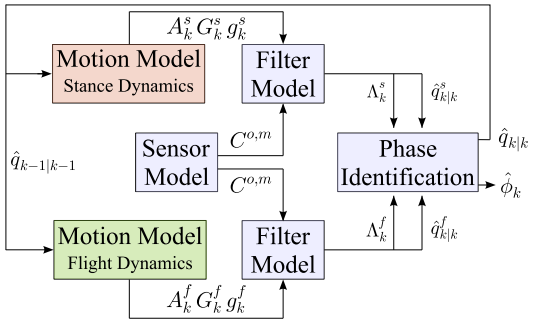

Fig. 2. Flowchart of the estimation routine. Two Extended Kalman Filters are utilized simultaneously, employing a common motion model featuring one stance and one flight phase dynamics. Both filters share the same sensor model. The phase identification block determines the most appropriate phase and estimation outcome by integrating the results based on the likelihood presented in Section II-B.

the likelihood of each estimate $\Lambda_k^{s,f}$ is computed as:

$$\Lambda_k^{s,f} = \frac{1}{\sqrt{|2\pi \mathbf{S}_k^{s,f}|}} e^{(\tilde{\mathbf{z}}_k^{s,f})^T \mathbf{S}_k^{s,f} \tilde{\mathbf{z}}_k^{s,f}}. \tag{7}$$

These likelihood values are then used to calculate the likelihood ratio (LR_k) for each time step:

$$LR_k = \frac{\Lambda_k^f}{\Lambda_k^s}. \tag{8}$$

The phase identification block in Figure 2 decides the best-suited phase and estimation result by fusing the influences on a likelihood basis. The identification block initially evaluates LR_k. If $LR_k > 1$, it implies that, based on the current measurement and EKF predictions, the phase is more likely to be in the flight phase. Conversely, if $LR_k < 1$, it indicates that the model is likely to be in the stance mode. Depending on the value of LR_k, the estimation routine determines the phase $\hat{\phi}_k$, and uses the final state estimation value of the most likely phase as $\hat{\mathbf{q}}_{k|k}$ for the next step.

However, a challenge arises when LR_k becomes almost equal to one in the close proximity of the liftoff event due to smooth transitions between the stance and flight phases, making it harder to identify the phase solely based on LR_k. To better capture the liftoff instant, the estimation routine also computes the ground reaction force (GRF) simultaneously using the selected $\hat{\mathbf{q}}_{k|k}$ and fuses this information with likelihood while deciding on the phase. Naturally, GRF cannot be negative in the stance phase; thus, a negative GRF estimate suggests that it is an unlikely prediction based on the hypothesis that the current mode is stance. In this context, we simply set:

$$\Lambda_k^s = 0 \quad \rightarrow \quad LR_k \rightarrow \infty. \tag{9}$$

which automatically triggers the liftoff event.

C. Motion Models for TD-SLIP

In this paper, we employ two fundamental motion models for state estimation. Firstly, we utilize the constant acceleration model (CAM) as a simple baseline, a model frequently

used in estimation problems due to its simplicity [26], [27]. Additionally, we compare the results with the model-based linearized model (LM) derived from first principles. This section provides a detailed explanation of the motion models used throughout our study.

1) Constant Acceleration Motion Model (CAM): The CAM is a widely used motion model based on the assumption that the system's acceleration remains approximately constant until the next sampling instant. We compute the corresponding acceleration values using (1), (2), and (3) with the most recent state estimation. The resulting state evaluation equation is as follows:

$$\mathbf{q}_{k+1} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & T & 0 \\ 0 & 1 & 0 & 0 & T \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{A}_k^{s,f}} \mathbf{q}_k + \begin{bmatrix} (\ddot{r}_k T^2)/2 \\ (\ddot{\theta}_k T^2)/2 \\ \dot{i}_k T \\ \ddot{r}_k T \\ \ddot{\theta}_k T \end{bmatrix}, \quad (10)$$

This equation is used in the prediction step of the EKF. It is important to note that the input V_m is implicitly included in the equation through $\dot{i}_k$. For the update steps, we compute the related $\mathbf{G}_k^{s,f}$ matrices as follows:

$$\mathbf{G}_k^f = \begin{bmatrix} 0 & 0 & 0 & m_t^{-1} & 0 \\ 0 & 0 & 0 & 0 & (m_t r^2)^{-1} \end{bmatrix}^T, \quad \mathbf{G}_k^s = \begin{bmatrix} 0 & 0 & 0 & m^{-1} & 0 \\ 0 & 0 & 0 & 0 & (mr^2)^{-1} \end{bmatrix}^T. \quad (11)$$

2) Linearized Motion Model (LM): (1) and (2) are nonlinear and nonintegrable, which means they do not admit analytical solutions. To perform the prediction step of the EKF algorithm, we solve (5) using a variable fourth-order Runge-Kutta differential equation solver using (4). For the update and correction steps of the EKF, we first linearize the continuous dynamics, i.e., (4), around the current estimated state, and then discretize the dynamics under zero-hold operation to be consistent with the digital control framework.

In the stance phase, the linearization of the dynamics with respect to $\mathbf{q}_k$ and $\mathbf{w}_k$ yields:

$$D_{\mathbf{q}}^s g = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{R_a}{L_a} & 0 & -\frac{K_e}{L_a} \\ \ddot{\theta}^2 - \frac{k}{m} & g\sin\theta & 0 & -\frac{d}{m} & 2r\dot{\theta} \\ M & \frac{g\cos\theta}{r} & \frac{K_t}{mr^2} & -\frac{2\dot{\theta}}{r} & N \end{bmatrix}, \quad (12)$$

$$D_{\mathbf{w}}^s g = \begin{bmatrix} 0 & 0 & 0 & m^{-1} & 0 \\ 0 & 0 & 0 & 0 & (mr^2)^{-1} \end{bmatrix}^T,$$

where

$$M = -\frac{g\sin\theta}{r^2} + \frac{2\dot{\theta}\dot{r}}{r^2} - \frac{2K_t i}{mr^3} + \frac{2b\dot{\theta}}{mr^3}, \quad N = -\frac{2\dot{r}}{r} - \frac{b}{mr^2}, \quad (13)$$

and D is the Jacobian operator.

In the flight phase, these matrices become more complex and take the form of:

$$D_{\mathbf{q}}^f g = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{-R_a}{L_a} & 0 & \frac{-K_e}{L_a} \\ P & 0 & 0 & \frac{-d}{m_t} & 2r\dot{\theta} \\ S & 0 & U & V & W \end{bmatrix}, \quad D_{\mathbf{w}}^f g = \begin{bmatrix} 0 & 0 & 0 & m_t^{-1} & 0 \\ 0 & 0 & 0 & 0 & (m_t r^2)^{-1} \end{bmatrix}^T, \quad (14)$$

where

$$P = \dot{\theta}^2 - \frac{k}{m_t}, \quad S = \frac{2\dot{r}\dot{\theta}}{r^2} - \frac{2(K_t i - b\dot{\theta})}{m_t r^3}, \quad U = \frac{K_t}{m_t r^2}, \quad V = -\frac{2\dot{\theta}}{r}, \quad W = -\frac{2\dot{r}}{r} - \frac{b}{m_t r^2}.$$

We then discretize (12) and (14) with a sampling period T_s and compute $\mathbf{A}_k^{s,f}$ and $\mathbf{G}_k^{s,f}$ to use in covariance estimate prediction and update steps of the EKF, as shown in Fig. 2.

D. Sensor Models

In this section, we introduce the alternative sensor models utilized in this research.

1) Optimistic Sensor Model (OSM): The OSM assumes that motor armature current, leg length, leg angle, and their respective derivatives in both stance and flight phases are all measured, essentially representing a full observable state space. We define the associated observation vector and observation model as follows:

$$\mathbf{z}^o := [r, \theta, i, \dot{r}, \dot{\theta}]^T, \ \mathbf{C}^o := \mathbf{I}^{5x5}. \quad (15)$$

This model is considered optimistic because it assumes that states associated with the compliant leg are directly measurable. However, obtaining such measurements poses significant challenges in practice, as mentioned earlier. Theoretically, we expect the best estimation performance from this sensor model. We will compare its prediction performance with a limited sensor model that captures state information obtained only through the hip joint.

2) Minimal Proprioceptive Sensor Model (mPSM): This paper aims to demonstrate that we can still achieve accurate state estimation using minimal proprioceptive sensor readings obtained from existing primary DC motor encoder and current measurements without additional sensor suites. This model assumes that only the leg angle, the leg angle rate, and the armature current are measured. The associated observation vector and observation model are defined as:

$$\mathbf{z}^m := [\theta, i, \dot{\theta}]^T, \ \mathbf{C}^m := \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (16)$$

When we use this sensor model along with the CAM, the system becomes unobservable and undetectable for r

and $\dot{r}$ states. Since the states are decoupled, measurement information does not enter these modes, while the process noise is still directly present on $\dot{r}$ and indirectly on r. As a result, Kalman filter matrices updated estimate covariance $\mathbf{P}_{k|k}$ and predicted estimate covariance $\mathbf{P}_{k|k-1}$ grow indefinitely, leading to unstable and failed estimation. However, with the LM, where the states are coupled, the system remains observable, and we do not encounter this problem.

III. ESTIMATION PERFORMANCE

A. Simulation Environment

We evaluate the estimation performance of our proposed framework by analyzing various trajectories generated from the same initial conditions. We assess their performance using metrics defined in Section III-B. We create these trajectories by integrating the dynamics described in (4) using MATLAB's $ode45$ solver.

Our simulations implement a digital control system topology with a synchronous clock frequency of $500\ Hz$. We consider the input $u(t)$ and process noise $\mathbf{w}^{s,f}(t)$ as piece-wise constant over intervals of length T_s. Throughout the simulations, we set the process and measurement noise covariances as follows: $\mathbf{Q}^s = diag[5, 10]$, $\mathbf{Q}^f = diag[1, 1]$, $\mathbf{R}^o = diag[r_0/100, 5, 1, r_0/10, 20]$, and $\mathbf{R}^m = diag[5, 1, 20]$.

B. Performance Metrics and Comparision

In this section, we evaluate state estimation performance using different sensors and motion models. We employ several performance metrics to assess the accuracy of our estimations.

We start by using the mean absolute percentage error (MAPE) metric, which measures the prediction performance of state estimation. It is calculated as follows:

$$MAPE\ := \frac{100}{n} \sum_{k=1}^{n} |\frac{\mathbf{e}_k}{\mathbf{q}_k}|, \tag{17}$$

where n is the number of samples, $\mathbf{e_k}$ represents the state error defined as $\mathbf{e}_k = \mathbf{q}_k - \hat{\mathbf{q}}_k$, where $\hat{\mathbf{q}}$ is the estimated state. MAPE provides a percentage error rate for overall estimation performance.

To assess the phase estimation performance, we define the timing error for transition events and the overall phase estimation error, denoted as $\hat{\phi}_k$ Err. (%):

$$\hat{\phi}_k\ Err\ (\%) := \frac{\text{Erroneous Phase Est. Duration(s)}}{\text{Total Run Time (s)}} \times 100,$$
$$e_i(s) := \frac{|\hat{t}_i - t_i|}{\text{total number of i}},$$

where $e_i(s)$ represents the timing error for touchdown, td, and liftoff, lo, events.

In Table II, we present the performance metrics for different motion and sensor models. The simulations are conducted for an initial apex state with $\dot{x}_0 = 1.6m/s$ and $y_0 = 0.24m$. We perform 1000 simulations, each consisting of 5 successful strides, for each motion-sensor model pair. This approach allows us to qualitatively assess estimation performance, as the trajectories differ due to process noise.

TABLE II
PERFORMANCE FOR DIFFERENT MOTION AND SENSOR MODEL PAIRS

	LM		**CAM**	
	OSM	mPSM	OSM	mPSM
MAPE [%]	2.44 %	2.62 %	3.87 %	-
$\hat{\phi}_k$ Err [%]	3.1%	3.3 %	36.56 %	-
e_{td} [ms]	4.0	4.4	-	-
e_{lo} [ms]	5.2	5.6	-	-

Notably, the CAM model with OSM yields relatively high MAPE and phase estimation errors, indicating its limitations in accurately predicting the true phase information. On the other hand, LM outperforms CAM in all performance metrics, including when paired with mPSM. This result highlights LM's superiority in capturing the system's second-order dynamics, which CAM cannot fully represent. However, it is worth noting that LM increases computational costs by approximately fourfold.

Furthermore, LM demonstrates exceptional event detection performance, with timing errors for touchdown and liftoff events remaining low (around $4\ ms$ and $5.4\ ms$, respectively) for both sensor models. Consequently, the overall phase estimation error ($\hat{\phi}_k$ Err.) is only around 3%, demonstrating accurate phase estimation, particularly for liftoff detection. In contrast, the CAM model shows frequent faulty event detection, leading to poorly defined timing errors and a high phase estimation error ($\hat{\phi}_k$ Err. = 36.56%).

Regarding sensor models, Table II shows that OSM generally yields better performance metrics, as expected. However, the most significant observation is that when using LM, the performance difference between OSM and mPSM is not substantial. Our estimation framework exhibits close estimation performance for fully sensed and under-sensed sensor models, indicating that over-instrumentation is unnecessary.

Fig. 3 illustrates an example from the simulation results for the LM-mPSM pair. The estimation remains accurate for all states, regardless of whether the measurement is available. Notably, even though the toe mass dynamics resemble an under-damped system in the flight phase, the estimation framework adequately tracks the true state of the system.

IV. CONCLUSIONS AND FUTURE WORK

This study addresses the state and phase estimation challenges of a torque-actuated spring-mass runner. Our proposed estimation framework leverages a multiple hypotheses EKF approach. We have demonstrated each model's effectiveness and concluded that a combination of model-based motion modeling and minimal proprioceptive sensor readings provides satisfactory state and phase estimation results in compliant, under-actuated, and under-sensed-legged platforms.

One of our primary motivations for future work is to validate our results on a two-dimensional monopod platform and

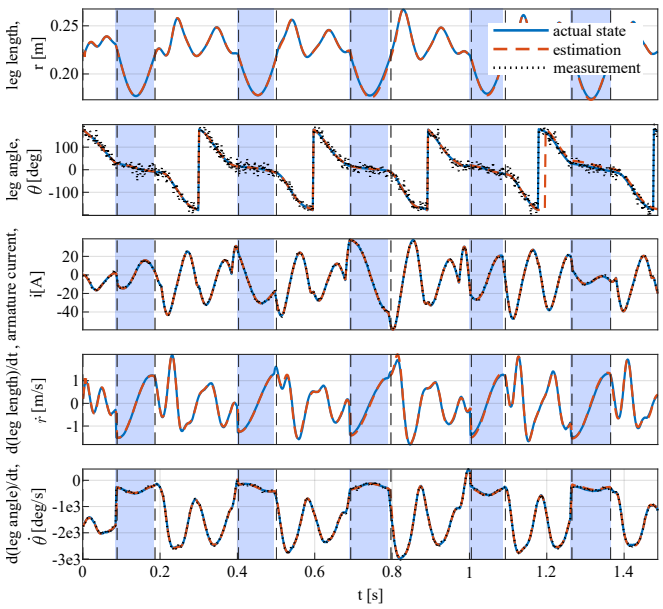

Fig. 3. An example from the simulation batch involving the LM-mPSM pair is illustrated. In the figure, blue and white regions represent the true stance and flight phases, respectively. Dashed lines indicate estimated transition events corresponding to touchdown or liftoff. The blue solid line represents the actual state, while the dashed orange line represents the estimation, and the dotted black line represents the measurement.

expand to three-dimensional models. This framework can be adapted for multi-legged systems for planar gaits without requiring substantial modifications. Additionally, enhancing the likelihood approach by incorporating the Interacting Multiple Model filter can improve performance.

ACKNOWLEDGMENT

This work was supported in part by Aselsan Inc. and The Scientific and Technological Research Council of Turkiye (TUBITAK) through AK's scholarship.

REFERENCES

[1] L. Bruzzone and G. Quaglia, "Locomotion systems for ground mobile robots in unstructured environments," *Mechanical sciences*, vol. 3, no. 2, pp. 49–62, 2012.

[2] T. Flayols, A. Del Prete, P. Wensing, A. Mifsud, M. Benallegue, and O. Stasse, "Experimental evaluation of simple estimators for humanoid robots," in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*. IEEE, 2017, pp. 889–895.

[3] G. Bledt, M. J. Powell, B. Katz, J. Di Carlo, P. M. Wensing, and S. Kim, "Mit cheetah 3: Design and control of a robust, dynamic quadruped robot," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 2245–2252.

[4] G. Fink and C. Semini, "Proprioceptive sensor fusion for quadruped robot state estimation," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 10 914–10 920.

[5] G. Secer and U. Saranli, "Control of planar spring–mass running through virtual tuning of radial leg damping," *IEEE Transactions on Robotics*, vol. 34, no. 5, pp. 1370–1383, 2018.

[6] A. Mazumdar, S. J. Spencer, C. Hobart, J. Salton, M. Quigley, T. Wu, S. Bertrand, J. Pratt, and S. P. Buerger, "Parallel elastic elements improve energy efficiency on the steppr bipedal walking robot," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 2, pp. 898–908, 2016.

[7] S. Ş. Candan, O. K. Karagöz, Y. Yazıcıoğlu, and U. Saranlı, "Design of a parallel elastic hopper with a wrapping cam mechanism and template based virtually tunable damping control," in *Dynamic Systems and Control Conference*, vol. 84270. American Society of Mechanical Engineers, 2020, p. V001T05A009.

[8] S. Nobili, M. Camurri, V. Barasuol, M. Focchi, D. Caldwell, C. Semini, and M. Fallon, "Heterogeneous sensor fusion for accurate state estimation of dynamic legged robots," in *Proceedings of Robotics: Science and Systems*. Robotics: Science and Systems Foundation, 2017.

[9] M. F. Fallon, M. Antone, N. Roy, and S. Teller, "Drift-free humanoid state estimation fusing kinematic, inertial and lidar sensing," in *2014 IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2014, pp. 112–119.

[10] J. A. Cobano, J. Estremera, and P. G. De Santos, "Location of legged robots in outdoor environments," *Robotics and Autonomous Systems*, vol. 56, no. 9, pp. 751–761, 2008.

[11] M. M. Ankarali and U. Saranli, "Stride-to-stride energy regulation for robust self-stability of a torque-actuated dissipative spring-mass hopper," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 20, no. 3, p. 033121, 2010.

[12] O. Gur and U. Saranli, "Model-based proprioceptive state estimation for springmass running," *Robotics: Science and Systems VII*, pp. 105–112, 2012.

[13] W. C. Martin, A. Wu, and H. Geyer, "Experimental evaluation of deadbeat running on the atrias biped," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1085–1092, 2017.

[14] R. Blickhan, "The spring-mass model for running and hopping," *Journal of biomechanics*, vol. 22, no. 11-12, pp. 1217–1227, 1989.

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

[16] W. J. Schwind, *Spring loaded inverted pendulum running: A plant model*. University of Michigan, 1998.

[17] O. K. Karagoz, G. Secer, M. M. Ankarali, and U. Saranli, "Analysis and control of a running spring-mass model with a trunk based on virtual pendulum concept," *Bioinspiration & Biomimetics*, vol. 17, no. 4, p. 046009, 2022.

[18] U. Saranli and D. E. Koditschek, "Template based control of hexapedal running," in *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, vol. 1. IEEE, 2003, pp. 1374–1379.

[19] I. Poulakakis and J. W. Grizzle, "The spring loaded inverted pendulum as the hybrid zero dynamics of an asymmetric hopper," *IEEE Transactions on Automatic Control*, vol. 54, no. 8, pp. 1779–1793, 2009.

[20] M. A. Sharbafi, C. Maufroy, M. N. Ahmadabadi, M. J. Yazdanpanah, and A. Seyfarth, "Robust hopping based on virtual pendulum posture control," *Bioinspiration & biomimetics*, vol. 8, no. 3, p. 036002, 2013.

[21] U. Saranlı, Ö. Arslan, M. M. Ankaralı, and Ö. Morgül, "Approximate analytic solutions to non-symmetric stance trajectories of the passive spring-loaded inverted pendulum with damping," *Nonlinear Dynamics*, vol. 62, no. 4, pp. 729–742, 2010.

[22] C.-J. Hu, T.-K. Wang, C.-K. Huang, and P.-C. Lin, "A torque-actuated dissipative spring loaded inverted pendulum model with rolling contact and its application to hexapod running," *Bioinspiration & biomimetics*, vol. 14, no. 2, p. 026005, 2019.

[23] İ. Uyanık, Ö. Morgül, and U. Saranli, "Experimental validation of a feed-forward predictor for the spring-loaded inverted pendulum template," *IEEE Transactions on robotics*, vol. 31, no. 1, pp. 208–216, 2015.

[24] M. M. Ankaralı and U. Saranlı, "Control of underactuated planar pronking through an embedded spring-mass hopper template," *Autonomous Robots*, vol. 30, no. 2, pp. 217–231, 2011.

[25] U. Saranli, M. Buehler, and D. E. Koditschek, "Rhex: A simple and highly mobile hexapod robot," *The International Journal of Robotics Research*, vol. 20, no. 7, pp. 616–631, 2001.

[26] P.-C. Lin, H. Komsuoglu, and D. E. Koditschek, "Sensor data fusion for body state estimation in a hexapod robot with dynamical gaits," *IEEE Transactions on Robotics*, vol. 22, no. 5, pp. 932–943, 2006.

[27] M. Khan, N. Salman, A. Ali, A. Khan, and A. Kemp, "A comparative study of target tracking with kalman filter, extended kalman filter and particle filter using received signal strength measurements," in *2015 International Conference on Emerging Technologies (ICET)*. IEEE, 2015, pp. 1–6.