Learning-based Leg Contact Detection using Position Feedback Only

Jiří Kubík[1], Jan Faigl[1]

[1]Czech Technical University, Faculty of Electrical Engineering, Technická 2, 166 27, Prague, Czechia

Abstract

The multilegged walking robots benefit from their complex morphology when traversing rough terrains. However, reliable foot contact detection is required to exploit their locomotion capabilities fully. Based on a dynamic model of the leg movements, foot-contact detection is possible using position feedback only. Since dynamic model determination can be demanding and laborious, we propose employing spare identification of nonlinear dynamics to construct the closed-form contact-free leg dynamics model without explicit manual identification. Model predictions and current measurements are then used to detect deviations from contact-free leg dynamics and thus determine the leg contact with an obstacle or terrain. The feasibility of the proposed approach is validated and compared with a precise high-fidelity analytical model.

Keywords

walking robot, machine learning, position feedback, physics-informed machine learning

1. Introduction

In rough terrain locomotion with multi-legged robots, the crucial part of locomotion control is a timely and reliable sense of the leg contact with the terrain or obstacles. It is specifically essential for position-based leg control, where the internal model of leg dynamics can be utilized to estimate the foot-contact [1]. However, in adverse environments or long-term deployments, the robot leg dynamics can change for various reasons, such as increased leg weight caused by mud deposits, increased friction caused by sludge in the servomotors, or by usage. Therefore, the dynamics model needs to be adjusted to such changes to support reliable contact sensing.

The model-based contact detection method can be based on an inverse dynamics model to estimate contact force [2, 3, 4, 5, 6]. The model accuracy relies on identifying the robot's kinematic and dynamic parameters. Hence, it might be cumbersome [7], and parameters become outdated as the robot properties change over time. On the other hand, machine learning-based approaches estimate input-output relation directly from the training data, including phenomena omitted by the analytical models. However, a black-box-based machine learning approach might result in a physically infeasible model. Therefore, using physics-informed machine learning can be advantageous to overcome the laborious parameter identification yet have a "gray-box" model that can fit the properties of the handcrafted models. Furthermore, we can generalize the approach toward an online learnable system that adapts to non-stationarities and changes in the system caused by external factors.

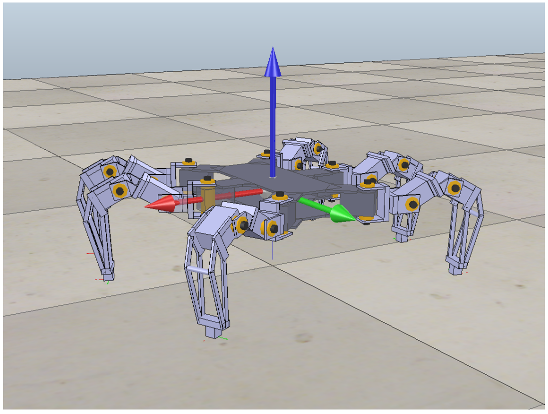

Figure 1: Six-legged walking platform in the CoppeliaSim simulation environment used to evaluate the proposed model learning. The x, y, and z-axis are depicted in red, green, and blue.

We propose to develop a lightweight learning-based physics-informed contact detection method using only position feedback from the servomotors. In [8], the general inverse leg dynamics black-box models were benchmarked and deployed on a single leg, which was initially shown as a promising approach. However, the detector constructed upon the best-benchmarked model did not yield reliable results supporting locomotion over rough terrains [9]. Therefore, we limited the regressor operation range to a specific context of the leg swing phase to increase the robustness and reliability of the detection [10]. In this paper, we hypothesize that in-

ITAT'23: Information technologies – Applications and Theory, Computational Intelligence and Data Mining - 11th international workshop, September 22–26, 2023, Tatranské Matliare, SK
kubikji2@fel.cvut.cz (J. Kubík); faiglj2@fel.cvut.cz (J. Faigl)
0000-0002-2219-5764 (J. Kubík); 0000-0002-6193-0792 (J. Faigl)
© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)
CEUR-WS.org/Vol-3498/paper9.pdf

corporating the physics information allows generalizing over a range of the leg trajectories contexts to enable reliable and robust contact detection. The developed solution has been studied using the CoppeliaSim (formerly VREP) [11] simulation environment using the high-fidelity model [12] of real hexapod walking robot depicted in Fig. 1. The robot is actuated by 18 Robotis Dynamixel AX-12A servomotors with position feedback only. Thus, it represents a relatively complex robotics system with 18 controllable degrees of freedom (DoF) and 24 total DoF.

The remainder of the paper is organized as follows. The addressed problem of foot contact detection is stated in Section 2. The proposed detection method is described in Section 3. Experimental evaluation results are reported in Section 4. Concluding remarks are summarized in Section 5.

2. Problem

The multi-legged robot locomotion can be based on the coordinated repetitive motion pattern called *gait*. Within each *gait cycle*, legs follow a prescribed trajectory and alternate between the *stance phase* supporting the body, and the *swing phase*, where the legs move to new footholds. An inverse dynamics model can be integrated into an adaptive force threshold-based locomotion controller to detect the leg contact with the surface using the position feedback only [1]. For position feedback only, it is possible to detect deviations from the collision-free dynamics model [13]. The idea is to predict a nominal (collision-free leg dynamics) feedback in the swing phase of the leg and compare the predicted and measured values. A difference between the values can be interpreted as a motion anomaly; hence, a leg contact with an obstacle or terrain.

For the used robotic platform depicted in Fig. 1, two data types are available to model the robot dynamics at any given discrete time-step k and each i-th joint. The data are the measured joint positions $\theta^i_{\mathrm{real}}[k]$ and desired joint positions $\theta^i_{\mathrm{des}}[k]$ set by the locomotion controller. The measured and desired joint positions can be recorded for N and N' past time steps, respectively. Furthermore, each time step, M future time steps of the desired joint positions are available.

Hence, we formulate the foot-contact detection as a time-series prediction of the i-th joint position $\theta^i_{\mathrm{pred}}[k+1]$ from the current and historical data

$$\begin{aligned}\theta^i_{\mathrm{pred}}[k+1] = \tilde{f}^i(&\boldsymbol{\theta}_{\mathrm{real}}[k-N], \ldots, \boldsymbol{\theta}_{\mathrm{real}}[k], \\ &\boldsymbol{\theta}_{\mathrm{des}}[k-N'], \ldots, \boldsymbol{\theta}_{\mathrm{des}}[k], \ldots, \boldsymbol{\theta}_{\mathrm{des}}[k+M]),\end{aligned} \tag{1}$$

where $\boldsymbol{\theta}_{\mathrm{real}}[k]$ denotes the measured joint positions vector of all joints coupled with the i-th joint. Similarly, we define $\boldsymbol{\theta}_{\mathrm{des}}[k]$. The function $\tilde{f}^i$ approximates θ^i dynamics at the next step $k+1$ out of the N measured positions, N' past, and M future desired positions.

3. Method

In multi-legged locomotion, the robot legs are moved in repetitive patterns defined by the utilized gait. Therefore, the particular leg trajectory can be divided into *gait segments* based on the trajectory shape. For foot contact detection, the most relevant segments are those in which we expect the contact of the leg with the terrain. Such segments are referred to as *contact (gait) segments*. In the presented work, we consider a 3-DoF leg consisting of three servomotors references as *coxa*, *femur*, and *tibia* and the whole class of trajectories when the foot-tip descends from the upper sub-space of the workspace to the lower one.

We aim at learning-based physics-informed contact detection; hence, we propose to utilize the *Sparse Identification of Nonlinear Dynamical systems* (SINDy) [14] by J. Brunton *et al.* that yielded closed-form equation describing system dynamics. The central concept of SINDy is briefly presented in the following paragraphs to make the paper self-contained, as the proposed method is its direct instantiation.

The SINDy algorithm combines sparsity-promoting techniques with machine learning to discover governing equation of a dynamical system

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t)), \tag{2}$$

where $\boldsymbol{x}(t) \in \mathbb{R}^n$ is the system state at the time instant t, and the function $\boldsymbol{f}(\boldsymbol{x}(t))$ describes the system dynamics constraints. The time history of the system state $\boldsymbol{x}(t)$ and either measure of the system state derivatives $\dot{\boldsymbol{x}}(t)$ history or its numerical approximation from $\boldsymbol{x}(t)$ are needed to determine the function $\boldsymbol{f}(\boldsymbol{x}(t))$. These time histories are organized into respective rows of the $\boldsymbol{X}$ and $\dot{\boldsymbol{X}}$ matrices.

Furthermore, a *candidate library* $\boldsymbol{\Theta}(\boldsymbol{X})$ is an essential part of the concept as it consists of candidate non-linearities applied to the system stated. For example, the relevant functions for our setup are the identity and goniometric functions $\boldsymbol{\Theta}(\boldsymbol{X}) = [\boldsymbol{X} \ \sin(\boldsymbol{X}) \ \cos(\boldsymbol{X})]$. Then, the sparse vector of coeffiecients $\boldsymbol{\Xi} = [\boldsymbol{\xi}_1 \ \boldsymbol{\xi}_2 \ \cdots \ \boldsymbol{\xi}_n]$ is to be determined using sparse-promoting algorithm

$$\dot{\boldsymbol{X}} = \boldsymbol{\Theta}(\boldsymbol{X})\boldsymbol{\Xi}. \tag{3}$$

The control input $\boldsymbol{u}(t)$ to the system dynamics

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)), \tag{4}$$

can be incorporated into the model using the approach by E. Kaiser *et al.* [15] to extend the candidate library

$\boldsymbol{\Theta}(\boldsymbol{X}, \boldsymbol{U})$ by the non-linearities applied to the vector of the aggregated inputs $\boldsymbol{U}$. For the discrete time domain (4), it can be rewritten as

$$\boldsymbol{x}[k+1] = \boldsymbol{f}(\boldsymbol{x}[k], \boldsymbol{u}[k]). \tag{5}$$

By defining the system state vector $\boldsymbol{x}[k]$ and the input vector $\boldsymbol{u}[k]$ as

$$\boldsymbol{x}[k] = [\boldsymbol{\theta}_{\text{real}}[k-N] \ \cdots \ \boldsymbol{\theta}_{\text{real}}[k]] \tag{6}$$

$$\boldsymbol{u}[k] = \left[\boldsymbol{\theta}_{\text{des}}[k-N'] \ \cdots \ \boldsymbol{\theta}_{\text{des}}[k] \ \cdots \ \boldsymbol{\theta}_{\text{des}}[k+M]\right], \tag{7}$$

the function $\tilde{f}^i$ from (1) can be found using the SINDy.

For the considered robotic platform, the coupling effect between each pair of legs is below the resolution of the utilized servomotors [9]. Therefore, each leg can be considered an independent model with unknown dynamics $\tilde{f}^i$ to be found by the SINDy.

Having the learned model (function $\tilde{f}^i$), the foot contact is based on the adaptive thresholding as in [1]. The interpolated trajectory consisting of a series of position values $\theta^i_{\text{des}}[k]$ is executed step-wise. At each step k, the real joint position $\theta^i_{\text{real}}[k]$ is read out from the servomotor and compared with the predicted value using (1).

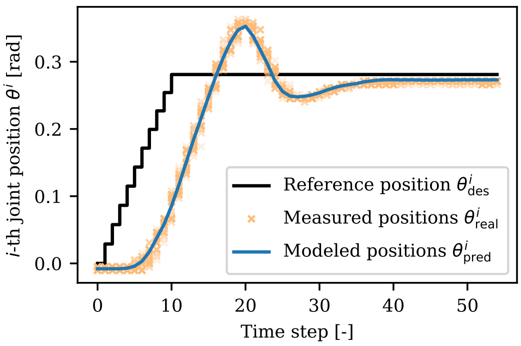

Figure 2: Example of several measured femur servomotor responses θ^i_{real} (yellow) to reference signal θ^i_{des} (black) and an example of the model predictions θ^i_{pred} (blue).

An example of the collision-free trajectory for the second joint (femur servomotor) is depicted in Fig. 2. It can be observed that the prediction accuracy might vary within the selected operational space. Hence, using a fixed threshold value might be challenging. Therefore, we opt for using the absolute prediction error difference between two consecutive predictions of selected joint subsets and experimentally found threshold values to detect a contact of the particular leg with the terrain.

4. Results

The proposed method has been evaluated using the high-fidelity model of a real hexapod robot [12]. The utilized CoppeliaSim [11] simulator has been operated in the stepped mode when each simulation step is triggered. The simulation step length has been set to $dt = 1\,\text{ms}$ as required by the high fidelity model [12]. The selected robot control loop $f_s = 50\,\text{Hz}$ reflects that (i) each read-/write operation to the servomotors takes about 1 ms; and (ii) up to three legs can be moved simultaneously.

The model has been learned using datasets collected within the simulation environment; see Section 4.1. Then, we study the influence of the model parameters on the learned model, and the results are reported in Section 4.2. Finally, the contact detection results are presented in Section 4.3.

4.1. Dataset

In the collected dataset, the selected hexapod leg follows the prescribed trajectory while the other legs support the robot body in the elevated pose, ensuring a collision-free trajectory for the selected leg. The trajectory consists of the trajectory segments alternating between the leg ascension and descension to reflect the expected leg motion during the locomotion. The start-point and end-point of each segment are selected randomly from the lower (upper) subspace and upper (lower) subspace for ascension (descension) while each segment has a length of 10 interpolation steps. Both subspaces are depicted in Fig. 3 and they are 5 cm in height and separated by the additional 5 cm of free space. During the trajectory execution, the real joint positions θ^i_{real} and desired joint positions θ^i_{des} are collected. The training and testing datasets have been collected using 5000 and 1000 random configurations, respectively.

4.2. Model Learning and Influence of the Model Parameters

The Python programming language module PySindy [16] implementing the SINDy has been used to construct the model. The performance of the model is mainly influenced by the selection of non-linear functions in the candidate library $\boldsymbol{\Theta}(\boldsymbol{X}, \boldsymbol{U})$ for both state variable and the inputs, and the threshold for the sparsity promoting algorithm. Selecting proper candidate non-linearities is necessary for $\tilde{f}^i$ to approximate f^i accurately; namely, SINDy eliminates the additional non-linearities but cannot substitute missing elements. Hence, based on the analytical model of the morphologically similar robot [8], the following functions have been utilized in the candi-

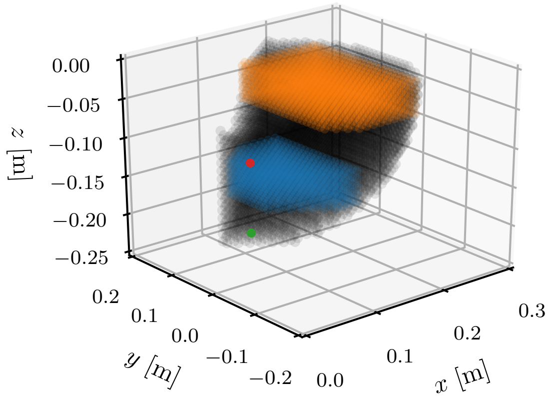

Figure 3: The grid with granularity 1 cm representing the leg workspace (black), with the lower subspace (blue) and the upper subspace (yellow) highlighted, is used for the selected leg trajectory generation. A small green disk represents the pose of the remaining legs during the collection, and the red disk represents the remaining leg pose in the collision scenarios.

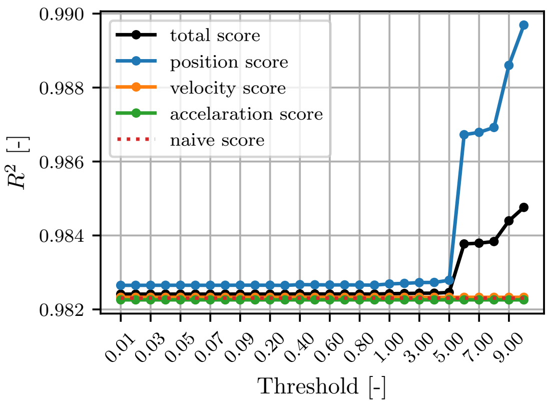

Figure 4: Prediction score R^2 using training dataset at the selected threshold values for all state variables (black), position (blue), velocity (yellow), and acceleration (green) in comparison to the naive prediction (red) when the predicted position remains unchanged.

date library

$$\mathbf{\Theta}(\boldsymbol{X},\boldsymbol{U}) = \left[\sin(\boldsymbol{X},\boldsymbol{U})\ \cos(\boldsymbol{X},\boldsymbol{U})\ \boldsymbol{P}^2(\boldsymbol{X},\boldsymbol{U})\right], \tag{8}$$

where $\boldsymbol{P}^2$ represents the polynomial features up to the second-degree polynomial, including the constant term.

We have selected a universal *Sequentially Thresholded Least Squares* (STLSQ) algorithm optimizer [14] that solves $\mathbf{\Xi}$ using *Least Squares* (LSQ) algorithm and then thresholds all candidate non-linearities that are smaller than the cut-off threshold λ. These steps are repeated with new $\mathbf{\Theta}(\boldsymbol{X},\boldsymbol{U})$ until the coefficients converge. The parameter search has been conducted using the log-like threshold values to train the SINDy model using the training dataset and identify the proper threshold λ. An evolution of the predicting using R^2 metrics is depicted in Fig. 4.

The search is stopped when the threshold causes the elimination of all coefficients for any state variable. Note that the achieved scores are comparable regardless of the particular threshold. Therefore, the inner model structure has been examined to consider the sensitivity to the control input $\theta^i_{\text{des}}[k]$ since the accuracy of the naive approach where the prediction is the current position $\theta^i_{\text{pred}}[k+1] = \theta^i_{\text{real}}[k]$ is significantly close to the accuracy of the models regardless of the threshold. Based on the presented results, we select $\lambda = 5$ parametrization for the future evaluation of the model in contact detection.

4.3. Contact Detection

Ten collisions have been collected for a selected leg within the subspace of the workspace utilized to collect training data to evaluate foot contact detection of the proposed method. During the trajectory execution, real joint positions θ^i_{real} and desired joint positions θ^i_{des} have been collected at $f_s = 50$ Hz. Besides, the robot body orientation and selected leg servomotor torques are collected at the highest possible sampling rate of $f_{dt} = 1000$ Hz to provide ground truth measurements for the foot contact.

The plots of the robot orientation, torques of the moving leg's joints, and real and predicted joints' position errors are depicted in Fig. 5. The results support the hypothesis that the leg contact with the terrain can be detected using absolute prediction error difference between two consecutive predictions.

5. Conclusion

A learning-based approach is presented to model the leg dynamics in foot contact with the terrain. The proposed physics-informed lightweight learning approach has been evaluated to detect the leg contact with the terrain in locomotion control of a small hexapod walking robot with only position feedback. Based on the high-fidelity simulation, the proposed approach is vital, and for a given workspace subspace, the proposed method can detect contact. However, we plan to employ model ensembling to extend the method to the whole leg's workspace.

Acknowledgments

The work was supported by the Czech Science Foundation (GAČR) under research project No. 21-33041J.

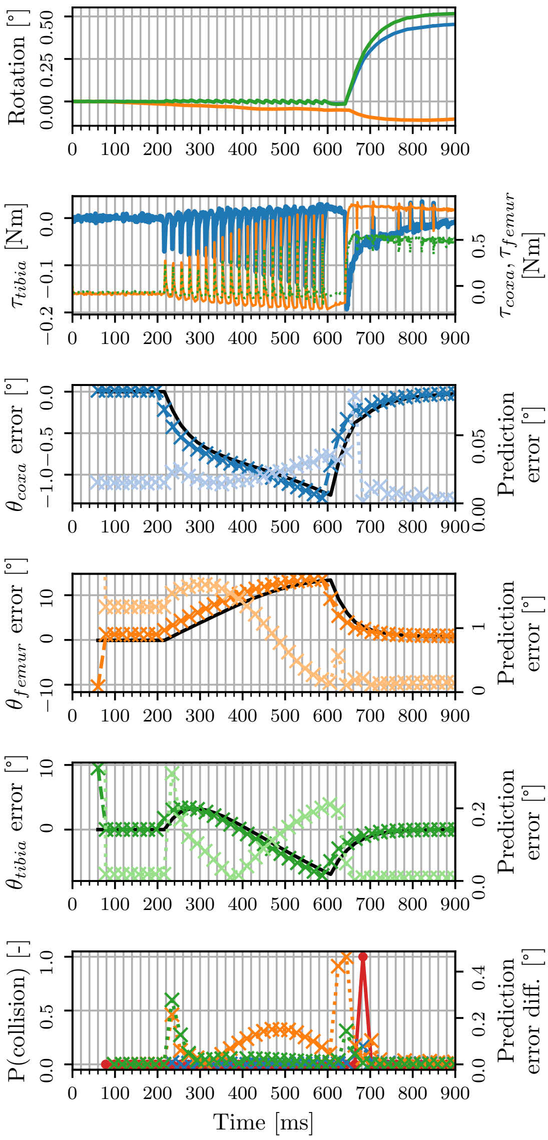

Figure 5: The leg starts moving at $t = 200$ ms, the collision occurs at $t = 640$ ms, and it is detected at $t = 680$ ms. The robot orientation is depicted as the robot body orientation in the x, y, and z-axis in blue, orange, and green. The torques and error values are color-coded as coxa joint in blue, femur joint in orange, and tibia joint in green. The real joint position error is in black, the predicted joint error is in color corresponding to the joint, and the predicted error is in a light shade. The bottom plot is an absolute prediction error difference between two consecutive predictions for each respective joint and contact detection (red).

References

[1] J. Faigl, P. Čížek, Adaptive locomotion control of hexapod walking robot for traversing rough terrains with position feedback only, Robotics and Autonomous Systems 116 (2019) 136–147. doi:DOI: 10.1016/j.robot.2019.03.008.

[2] G. Bledt, P. M. Wensing, S. Ingersoll, S. Kim, Contact model fusion for event-based locomotion in unstructured terrains, in: IEEE International Conference on Robotics and Automation (ICRA), 2018, pp. 4399–4406. doi:DOI: 10.1109/ICRA.2018.8460904.

[3] M. Camurri, M. Fallon, S. Bazeille, A. Radulescu, V. Barasuol, D. G. Caldwell, C. Semini, Probabilistic contact estimation and impact detection for state estimation of quadruped robots, IEEE Robotics and Automation Letters 2 (2017) 1023–1030. doi:DOI: 10.1109/LRA.2017.2652491.

[4] J. Hwangbo, C. D. Bellicoso, P. Fankhauser, M. Hutter, Probabilistic foot contact estimation by fusing information from dynamics and differential/forward kinematics, in: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2016, pp. 3872–3878. doi:DOI: 10.1109/IROS.2016.7759570.

[5] S. Yang, H. Kumar, Z. Gu, X. Zhang, M. Travers, H. Choset, State estimation for legged robots using contact-centric leg odometry, arXiv (2019). `arXiv:1911.05176`.

[6] J. Yang, H. Sun, D. Wu, X. Chen, C. Wang, SLIP model-based foot-to-ground contact sensation via kalman filter for miniaturized quadruped robots, in: Intelligent Robotics and Applications (ICIRA), 2019, pp. 3–14.

[7] M. Forouhar, P. Čížek, J. Faigl, SCARAB II: A Small Versatile Six-legged Walking Robot, in: IEEE International Conference on Robotics and Automation (ICRA): 5th Full-Day Workshop on Legged Robots, 2021.

[8] J. Kubík, P. Čížek, R. Szadkowski, J. Faigl, Experimental leg inverse dynamics learning of multi-legged walking robot, in: 2020 Modelling and Simulation for Autonomous Systems (MESAS), 2021, pp. 154–168. doi:DOI: 10.1007/978-3-030-70740-8_10.

[9] J. Kubík, Learnable State Estimator for Multi-legged Robot, Master's thesis, Czech Technical University in Prague, Faculty of Electrical Engineering, 2021.

[10] J. Kubík, R. Szadkowski, J. Faigl, Learning-based detection of leg-surface contact using position feedback only, in: IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), 2022, pp. 1–4. doi:DOI: 10.1109/ETFA52439.2022.9921720.

[11] E. Rohmer, S. P. N. Singh, M. Freese, V-rep: A versatile and scalable robot simulation framework, in: IEEE/RSJ International Conference

on Intelligent Robots and Systems (IROS), 2013, pp. 1321–1326. doi:DOI: 10.1109/IROS.2013.6696520, www.coppeliarobotics.com.

[12] M. T. Nguyenová, P. Čížek, J. Faigl, Modeling proprioceptive sensing for locomotion control of hexapod walking robot in robotic simulator, in: Modelling and Simulation for Autonomous Systems, 2019, pp. 215–225.

[13] S. Haddadin, A. De Luca, A. Albu-Schaffer, Robot collisions: A survey on detection,isolation, and identification, IEEE Transactions on Robotics 33 (2017) 1292–1312. doi:DOI: 10.1109/TRO.2017.2723903.

[14] S. L. Brunton, J. L. Proctor, J. N. Kutz, Discovering governing equations from data by sparse identification of nonlinear dynamical systems, Proceedings of the National Academy of Sciences 113 (2016) 3932–3937. doi:DOI: 10.1073/pnas.1517384113.

[15] E. Kaiser, J. N. Kutz, S. L. Brunton, Sparse identification of nonlinear dynamics for model predictive control in the low-data limit, Royal Society A: Mathematical, Physical and Engineering Sciences 474 (2018) 20180335. doi:DOI: 10.1098/rspa.2018.0335.

[16] B. M. de Silva, K. Champion, M. Quade, J.-C. Loiseau, J. N. Kutz, S. L. Brunton, Pysindy: A python package for the sparse identification of nonlinear dynamical systems from data, Journal of Open Source Software 5 (2020) 2104. doi:DOI: 10.21105/joss.02104.