Requirement derivation method for a legged robot with series-elastic actuators.[*]

Daniela Vacarini de Faria[1,2][0000−0003−1027−1827], Marcos R. O. A. Máximo[1,3][0000−0003−2944−4476], and Luiz C. S. Góes[1,4][0000−0003−3770−0601]

[1] Aeronautics Institute of Technology, Praça Marechal Eduardo Gomes, 50, Vila das Acácias, 12228-900, São José dos Campos, SP, Brazil
[2] Aeronautical Mechanics Division, danivacarini@gmail.com
[3] Autonomous Computational Systems Lab (LAB-SCA), Computer Science Division, mmaximo@ita.br
[4] Aeronautical Mechanics Division, goes@ita.br

Abstract. In this work we propose a new methodology for requirement derivation of the dynamical requirements of a series elastic actuator applied to a legged robot. The leg model consists of a mechanism composed of three links – representing the thigh, the shin and the foot – and two Series Elastics Actuators (SEA) – representing the knee and ankle. The stance phase of a running gait is modeled according to the Spring Loaded Inverted Pendulum (SLIP) method. To make sure that sufficient extent of running patterns is covered, the SLIP parameters are sampled inside a predefined range using the Improved Distributed Hypercube Sampling method. The number of samples used in this study is selected through a convergence test. The leg performance is then studied through a comparison between the CoM trajectory obtained simulating the mechanism with ideal actuators on its joints and with SEAs. A closed loop Impedance Controller is used to calculate the torque required by each joint that allows the system to behave as a spring, thus mimicking the spring-like behavior of the leg during the SLIP movement. The SEAs are modeled by a parametric transfer function that is also presented in this work. To the best of our knowledge, this work is the first to propose a method that accounts for the performance of this task execution.
Student level: MSc. Date of conclusion (defense): December 2019. To be considered in CTDR.

Keywords: Legged robots · Robot dynamics · Impedance control · Series elastic actuators.

1 Introduction

Wheeled robots are simple to manufacture and their functioning is easy to comprehend, nevertheless they have some drawbacks, as they are highly dependent

---

of an relatively well structured ground. The use of robots in activities that are usually restricted only to humans, such as rescue or exploration of places of difficult access with irregular ground are now possible due to the recent advances in legged locomotion

The animal muscle skeletal system is a nonlinear spring-mass system, nevertheless it can be described as a spring-mass model consisting of a massless spring attached to a point mass [1]. This model is know as Spring Loaded Inverted Pendulum (SLIP) [13].

Heglund *et al.* [5] studied how speed and stride frequency changed with body size and mass. McMahon *et al.* [10] showed a correlation between leg stiffness and body speed. Other studies also show correlations between leg stiffness and other parameters such as surface type [4], extra load [16], and so on. As shown by Blickhan [1], animals use muscles, tendons and leg position to control leg stiffness and force applied to the ground. A contemporaneous work by Hogan [7], presented the possibility of controlling a manipulator end effector apparent stiffness, damping and inertia – i.e. impedance – with redundant degrees of freedom and force feedback control. This was called Impedance control.

The use of force control in robotics was made possible by work through the 1990's and 2000's on force actuators. Pratt *et al.* [11] showed force source actuators were achievable adding compliant elements between the power output and load. They called them Series Elastic Actuators (SEA). Another important work in this field can be found on [12], which also shows the design of SEAs to be used in the robot Corndog, a planar running robot.

The contributions of this work are a novel framework for requirement derivation for a running legged robot, as well as parametric equation for SEA that aids this task. Some similar works can be found on literature, [14] developed a quadruped robot with hydraulic SEAs, the HyQ. Although, the robot requirements ability to walk and run in different conditions, the task chosen to derive the actuator requirements is jumping in place higher than 0.15 m, which is not guaranteed to be sufficient for running. Likewise, [9] designed a quadruped robot with series elastic actuators, he focused in the design and control strategy for the robot instead of requirement derivation.

This work is outlined as follows: Section 2 shows aspects of a legged robot locomotion during running, Section 3 discuss the impedance control, SEA modeling is shown on Section 4. A framework for requirement derivation and new parametric equation for SEA are introduced on Section 5. The results and conclusion are shown on Sections 6 and 7.

2 Legged robot locomotion

2.1 Dynamic Model

The position of the robot in any given time can be described by its center of mass (CoM) position $\mathbf{x}_{CoM} \in \mathbb{R}^3$, orientation of a reference body part in relation to a reference Coordinates System (CS) $\boldsymbol{\theta}_{CoM} \in \mathbb{R}^3$, and the angular position $\boldsymbol{\theta} \in \mathbb{R}^N$ of its N joints. The state vector is $\mathbf{q} \in \mathbb{R}^{N+6}$:

$$\mathbf{q} = \begin{bmatrix} \boldsymbol{\theta} \\ \mathbf{x}_{CoM} \\ \boldsymbol{\theta}_{CoM} \end{bmatrix}. \tag{1}$$

The robot dynamics can be written as [3]:

$$\mathbf{I}\left(\mathbf{q}\right)\ddot{\mathbf{q}} + \mathbf{C}\left(\mathbf{q},\dot{\mathbf{q}}\right) + \mathbf{V}\left(\dot{\mathbf{q}}\right) + \mathbf{S}\left(\mathbf{q}\right) = \begin{bmatrix} \boldsymbol{\tau} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + \sum_i \mathbf{J}_i^t(\mathbf{q})\mathbf{f}_i, \tag{2}$$

where $\mathbf{I}\left(\mathbf{q}\right) \in \mathbb{R}^{(N+6)\times(N+6)}$ is the inertia tensor, $\mathbf{C}\left(\mathbf{q},\dot{\mathbf{q}}\right) \in \mathbb{R}^{N+6}$ are the terms due to the Coriolis effect, $\mathbf{V}\left(\dot{\mathbf{q}}\right) \in \mathbb{R}^{N+6}$ are the velocity-dependent torques, and $\mathbf{S}\left(\mathbf{q}\right) \in \mathbb{R}^{N+6}$ represent position-dependent torques. On the right hand side, $\boldsymbol{\tau} \in \mathbb{R}^N$ is the joint torque vector, $\mathbf{f}_i = [f_{i,x}, f_{i,y}, f_{i,z}]^t \in \mathbb{R}^3$ with $i \in \mathbb{N}$ is a external force applied to the robot. The $\mathbf{J}_i(\mathbf{q}) \in \mathbb{R}^{3\times(N+6)}$ is the associated Jacobian matrix, i.e. the matrix that maps the joints velocities to the end effector velocities [3].

2.2 Spring Loaded Inverted Pendulum

The Spring Loaded Inverted Pendulum (SLIP) model can be described as a spring-mass model consisting of a massless spring attached to a point mass. In this case, the leg is seen as a line linking the body CoM to the foot that touches the ground, K_{leg} is the leg stiffness, and m is the body mass (see Fig. 1).

Figure 1 shows the parameters used to describe the stance phase movement. l_0 is the uncompressed leg length, $l(t)$ is the leg length in a given time t. θ_0 is the angle formed between the leg and the vertical in the instant that the foot touches the ground, $\theta(t)$ the angle between the leg and the vertical in a given time t. $u = \dot{x}$ is the horizontal velocity of the CoM, and is assumed to have the same value during the touch down and the take off. $v = \dot{y}$ is the vertical velocity of the CoM, and is assumed to have the same absolute value during the touch down and the take off, but opposite direction. The positive direction is oriented upwards as seen in Fig. 1.
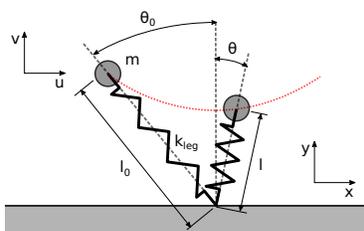
Fig. 1. SLIP parameters, based on [10].

3   Impedance Control

The Impedance control approach consists in modulating the way the environment "sees" the physical properties of the robot. Although it is not possible for a controller to make the hardware appear as anything other than a physical system [6], one can manipulate how they are perceived externally. There are two types of physical systems: admittances and impedances. Admittances transform an effort input (torque, force) into a flow output (displacement, velocity). On the other hand, impedances convert a flow input into an effort output. In other words, this type of control could give the robot a behavior similar to a mass-spring system, which can be useful to perform the movements needed in SLIP.

The derivation can be found in [8]. The actuator torque $\boldsymbol{\tau}_{act}$ can be calculated by (3) :

$$
\begin{aligned}
\boldsymbol{\tau}_{act} = {} & \mathbf{I}\left(\theta\right)\mathbf{J}^{-1}\left(\theta\right)\mathbf{M}^{-1}K\left[\mathbf{x}_0 - \mathbf{L}\left(\theta\right)\right] + \mathbf{S}\left(\theta\right) \\
& + \mathbf{I}\left(\theta\right)\mathbf{J}^{-1}\left(\theta\right)\mathbf{M}^{-1}B\left[\dot{\mathbf{x}}_0 - \mathbf{J}\left(\theta\right)\boldsymbol{\omega}\right] + \mathbf{V}\left(\boldsymbol{\omega}\right) \\
& + \mathbf{I}\left(\theta\right)\mathbf{J}^{-1}\left(\theta\right)\mathbf{M}^{-1}\mathbf{f}_{int} - \mathbf{J}^{t}\mathbf{f}_{int} \\
& - \mathbf{I}\left(\theta\right)\mathbf{J}^{-1}\left(\theta\right)\mathbf{G}\left(\theta,\boldsymbol{\omega}\right) + \mathbf{C}\left(\theta,\boldsymbol{\omega}\right),
\end{aligned}
\tag{3}
$$

where $\mathbf{I}\left(\theta\right)$ is the inertia tensor, $\mathbf{C}\left(\theta,\boldsymbol{\omega}\right)$ are the terms due to the Coriolis effect, $\mathbf{V}\left(\boldsymbol{\omega}\right)$ are the velocity-dependent torques, $\mathbf{S}\left(\theta\right)$ are position-dependent torques.

4   Series Elastic Actuators

According to Robinson [12], the transfer function of an SEA can be represented by (4).

$$
T_s(s) = \frac{\left(K_a K_d s + K_a K_p\right)T_d(s) - \left(J_m s^2 + b_m s\right)\Theta_l(s)}{\frac{J_m}{K_s}s^2 + \frac{b_m + K_s K_d K_a}{K_s}s + K_p K_a + 1}.
\tag{4}
$$

Where $T_s(s)$ and $T_d(s)$ are the Laplace transform of the measured spring torque and desired torque. $\Theta_l(s)$ is the Laplace transform of the load position. K_a and K_d are the proportional and derivative gains, K_s is the elastic constant, J_m is the motor inertia, and b_m is the motor damping coefficient.

5   Methodology

5.1   Parametric equation for a SEA

Since (4) contains physical parameters, using it to derive requirements could over-restrict the designers' choices. Instead, its parameterization allows us to study the impact of groups of parameters instead of choosing the value of each one, which is a better practice.

Multiplying both the numerator and denominator by $\frac{K_s}{J_m}$:

$$T_s(s) = \frac{\frac{K_s(K_aK_ds+K_aK_p)}{J_m}T_d(s) - \left(K_ss^2 + \frac{K_sb}{J_m}s\right)\Theta_l(s)}{s^2 + \frac{b_m+K_sK_dK_a}{J_m}s + \frac{K_s(K_pK_a+1)}{J_m}}. \tag{5}$$

Using the approximation: $K_pK_a + 1 \simeq K_pK_a$ [12], we arrive at

$$\omega_n^2 = \frac{K_s(K_pK_a)}{J_m}, \tag{6}$$

and

$$2\xi\omega_n = \frac{b_m + K_sK_dK_a}{J_m}. \tag{7}$$

Since $b_m << K_sK_dK_a$ [12], we have

$$\xi = \frac{K_sK_dK_a}{2\omega_nJ_m}. \tag{8}$$

At last we define:

$$\rho = \frac{b_m}{J_m}. \tag{9}$$

Substituting Equations (6), (8) and (9) into Equation (5) and rearranging the terms:

$$F_s(s) = \frac{\left(2\xi\omega_ns + \omega_n^2\right)F_d(s) - \left(K_ss^2 + \rho K_ss\right)X_l(s)}{s^2 + 2\xi\omega_ns + \omega_n^2}. \tag{10}$$

5.2 Robotic Leg Outline

Figure 2 below shows a schematic drawing of the robotic limb used for the actuator requirement derivation. The layout is intended to be similar to a cat rear legs, with degrees of freedom on the hip, the knee and the ankle. This is a design already present in the literature, as can be seen on [15]. The center of mass is assumed to be placed on the robot shoulder, as done in [18].
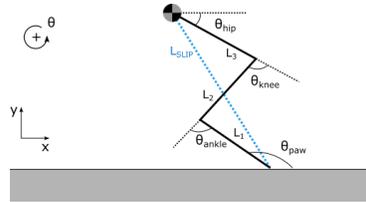
Fig. 2. Robot leg.

5.3   Framework

The requirements derivation method proposed in this work will adopt the following iterative strategy:

1. A high level task will be set, with some performance requirements attached to it;
2. This task will be simulated varying a set of parameters through a parameters space;
3. The performance of the system will be assessed and the range of the parameters will be chosen;
4. The performance and parameters range from this set of simulations will be used for a more detailed and complex model of the same task, returning to step 1.

Following this framework, SLIP model simulations will be performed in order to obtain a set of compatible SLIP parameters. Then, a robotic leg model with ideal actuators will be used to refine the SLIP parameter set and obtain CoM trajectories for different SLIP initial conditions. At last, a set of SEA parameters will be obtained through simulations performed with a robotic leg model with SEAs modeled by (5).

6   Results

6.1   SLIP Model Simulation

For the SLIP model simulation, the Drake toolbox was used. It is an open source collection of tools for analysing robot dynamics, designed by the Robot Locomotion Group of MIT Computer Science and Artificial Intelligence Lab (CSAIL) [17].

The set of parameters to be studied were sampled using Improved Distributed Hypercube Sampling (IHS) method. Convergence of the metrics was reached with a set of 300 samples. The boundaries of the parameters space are shown in Table 1.

Table 1. IHS limits for the SLIP parameters.

Parameter	Minimum value	Maximum value
l_0 [m]	0.35	0.40
$\dot{x}$ [m/s]	1.5	3.0
$\dot{y}$ [m/s]	-1.75	0.5
θ_0 [°]	35	45

For each sample point, an elastic constant that leads to a stable running was obtained through a method called "Shooting K" [10]. In this case, stability means that the final conditions for each step are symmetrical to the initial conditions.

6.2   Robotic leg with ideal actuators

The robotic leg with ideal actuators model was simulated using Matlab Simulink (Figure 3). The simulation consisted on the following steps:
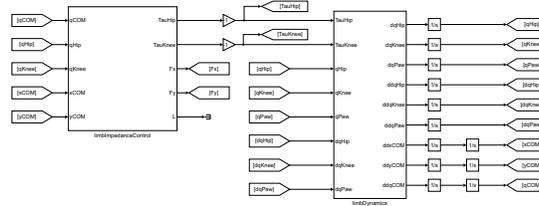
Fig. 3. Simulink model for a robotic leg with ideal actuators.

1. **Force feedback control block**: the center of mass (CoM) position and velocity, along with desired leg virtual elastic coefficient provided by K_{opt}, are used to calculate the force that should be provided by the leg mechanism on its end effector (in this case, the shoulder). The CoM position and velocity are also used as inputs for determining the joints position, velocities and accelerations through inverse kinematics. Then, the joints positions and kinematics were used to obtain the Jacobian of the mechanism, latter used to calculate the desired torque that should be provided by each joint ideal actuator;

2. **Limb Dynamics block**: the torques calculated by the Force Feedback Control block, along with integrated states from last iteration, are used as an input for direct dynamics of the leg mechanism. New values for the states were obtained.

The following hypothesis and simplifications were made:

– The simulation will be performed using rear leg configuration;
– The weight is evenly distributed on both rear legs;
– The paw of the leg touches the ground in one point;
– There is enough friction between the paw and the ground to avoid slipping or any loss of contact;
– The paw joint is not actuated and is free to rotate;
– The center of mass is ideally placed on the shoulders, this assumption was already used on other works such as [18];
– The actuators are able to provide instantaneously any torque required by the control block.

The robot mass and inertia properties are shown in Table 2. To better understand the kinematic chain, see Figure 2.

Table 2. Mass and Inertia properties of the leg parts.

	Mass [kg]	I_{zz}	Length [m]
CoM	3.0	−	−
Thigh (link 3)	0.5	21.88×10^{-2}*	0.25
Shin (link 2)	0.5	3.13×10^{-2}	0.25
Foot (link 1)	0.5	3.13×10^{-2}	0.25

* The thigh inertia includes the CoM mass and position.

6.3 Robotic leg with SEA actuators

The robotic leg with series elastic actuators model was simulated using Matlab Simulink (Figure 4). The simulation consisted of steps similar to the ones shown in Section 6.2.
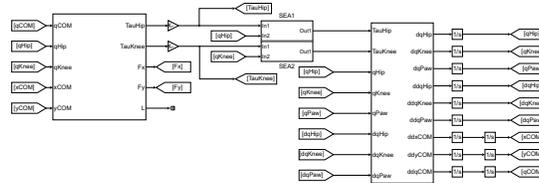
Fig. 4. Simulink model for a robotic leg with SEA actuators.

1. **Force feedback control blocks**: same procedure as in previous section;
2. **SEA block**: the required torques and joints positions will serve as inputs to the SEA transfer function shown on Equation (10) and the torques provided by the SEAs will be calculated;
3. **Limb Dynamics block**: same procedure as in previous section.

Also, the same hypothesis and simplifications described on previous section were used, except that in this case we are using SEAs instead of ideal actuators.

Now that the trajectories have been selected, a range of SEA parameters were tested and the results were compared to the trajectories obtained by the ideal actuators simulated in Section 6.2. As a metric of performance, the root mean square error (RMSE) between the shoulder position of the robot with ideal actuators and the robot with SEA actuators was calculated for all 300 trajectories. Then, the RMSE between all 300 trajectories for a given set of SEA parameters were calculated as shown in (11).

$$RMSE = \sqrt{\frac{1}{N_{samples}} \sum_{j=1}^{N_{samples}} \left(\sum_{i=1}^{N_{pts}} \frac{(y_{CoM,SEA,i,j} - y_{CoM,ideal,i,j})^2}{N_{pts}} \right)}. \quad (11)$$

Where N_{pts} is the number of points used for error calculation on a given trajectory, $N_{samples}$ is the number of IHS trajectory samples, in this case $N_{samples} = 300$, i is iterating the trajectory points, j is iterating over the samples, $y_{CoM,ideal,i,j}$ is the CoM vertical position i on the j-th ideal trajectory, $y_{CoM,SEA,i,j}$ is the CoM vertical position i on the j-th trajectory obtained using the simulation model with SEAs.

The RMSE value was then compared for all sets of SEA parameters. Instead of IHS sampling, now a more traditional grid sampling was used.

Since there are four SEA parameters being studied – namely: ω_n, ξ, K_s and ρ – they were divided into two groups, in order to allow for better graphical view of the problem. The two groups are: ω_n *versus* ξ, and K_s *versus* ρ.

6.4 Influence of SEA parameters: ω_n *versus* ξ and K_s *versus* ρ

For the first round of SEA requirements derivation, only ω_n and ξ were varied and K_s and ρ were kept constant. Their range was chosen as shown in Table 3:

Table 3. SEA parameters range I.

Parameter	Minimum value	Maximum value
ω_n [rad/s]	$2\pi 10$	$2\pi 60$
ξ	0.3	1.0

The range for ω_n was chosen based on values found in the literature or datasheets of commercial SEAs. Although ω_n is not exactly the natural frequency of this system, it can be used as a first rough guess when the real values are not known. The commercial actuator ANYdrive was chosen as a top *commercial off-the-shelf* SEA to set the top limit of ω_n range. The ANYdrive has a bandwidth of 60Hz and is the actuator used on the quadruped robot ANYmal, both developed by the ANYbotics AG.

Figure 5 shows the RMSE for each set of parameters, Figure 6 shows a comparison of a trajectory obtained with the ideal actuator simulation and the ones obtained with the SEA simulation.

As shown in Figure 5, the performance of the robot is visibly low for lower ω_n values: there is a raising ω_n from 10 to 35 Hz decreases the RMSE in 93.9%, but raising the same parameter from 10 to 47.5 Hz decreases the RMSE in 96.7%. This low performance is also visible in the trajectories shown in Figure 6, since the actuator with $\omega_n = 10$ Hz is not even able to perform the required trajectory.
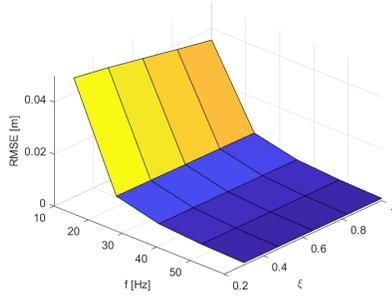
Fig. 5. RMSE between the shoulder trajectory of a simulation with SEA versus ideal actuators.
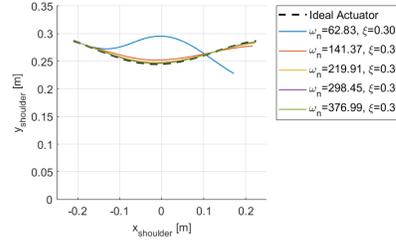
Fig. 6. Comparison of a trajectory obtained with the simulation using ideal actuators and the ones obtained with the SEA actuators.

Therefore, $\omega_n = 35$ Hz was chosen as a bottom threshold for this variable. The influence of ξ is also visible, but not as prominent as the influence of ω_n, $\xi = 0.65$ was chosen as a bottom limit for the actuator. All trajectories were simulated once more, with new ranges set as shown in Table 4. Figure 7 shows the RMSE of all sets of SEA parameters.

Table 4. SEA parameters range II.

Parameter	Minimum value	Maximum value
ω_n [rad/s]	$2\pi35$	$2\pi60$
ξ	0.65	1.0

For the last round of requirements derivation for the SEAs, the influence of both K_s and ρ was studied. It is important to note that the previous round already sets some boundaries for K_s, since it influences the values of ω_n and ξ, as shown in Equations (6) and (8). Although there are other parameters influencing ω_n and ξ, K_s cannot be as low as desired, since it would require raising too much the values of the gains K_p and K_a of the SEA controller, which is not a good practice [2].

Table 5 shows the range of the K_s and ρ parameters used in this study. The range of K_s is very broad, since there are few information about the SEA elastic constant in the literature. Commercial values are not available, since this is a sensible information, and similar performance studies were found in the literature.

The RMSE was obtained as described in Section 6.4. Figure 8 shows the results.

Table 5. SEA parameters range III.

Parameter	Minimum value	Maximum value
K_s [N/rad]	10	500
ρ	1	10
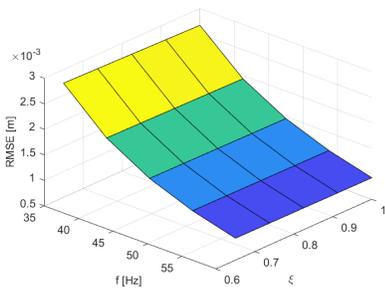
Fig. 7. RMSE between the shoulder trajectory of a simulation with SEA versus ideal actuators (ω_n *versus* ξ).
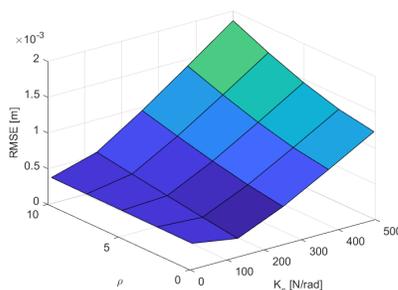
Fig. 8. RMSE between the shoulder trajectory of a simulation with SEA versus ideal actuators (K_s *versus* ρ).

7   Conclusion

In this work, we presented a method for requirement derivation of a SEA to be employed on a running legged robot. The CoM movement was modeled according to the SLIP method. Then the spring-like behavior of the leg was executed using Impedance Control, the each link position and actuator required forces were calculated through inverse kinematics and the Jacobian of the system, respectively. The local optimal virtual elastic constant of the leg was obtained through *Shooting K* method, then the CoM trajectories were sampled using IHS.

To avoid over-restricting the design engineer, a parametric transfer function for the SEA was proposed. All trajectories were simulated with a dynamical model considering both ideal actuators or SEAs. The SEA performance was then compared to the ideal actuators by calculation the RMSE between the trajectories obtained on both simulations, for all samples.

As future works, one could also include studies about the vibration and transient dynamics that occur during the feet impact on the ground, and also the influence of sensor characteristics (noise, quantization, sample rate, among others things) on the robot performance.

References

1. Blickhan, R.: The spring-mass model for running and hopping. Journal of Biomechanics **22**(11-12), 1217–1227 (1989)

2. Calanca, A., Muradore, R., Fiorini, P.: Impedance control of series elastic actuators: Passivity and acceleration-based control. Mechatronics **47**, 37–48 (Nov 2017). https://doi.org/10.1016/j.mechatronics.2017.08.010
3. Craig, J.J.: Introduction to robotics: mechanics and control. Pearson/Prentice Hall, Upper Saddle River, N.J, 3rd ed edn. (2005)
4. Ferris, D.P., Louie, M., Farley, C.T.: Running in the real world: adjusting leg stiffness for different surfaces. Proceedings of the Royal Society B: Biological Sciences **265**(1400), 989–994 (Jun 1998). https://doi.org/10.1098/rspb.1998.0388
5. Heglund, N.C., Taylor, C.R.: Speed, stride frequency and energy cost per stride: how do they change with body size and gait? The Journal of Experimental Biology **138**, 301–318 (Sep 1988)
6. Hogan, N.: Impedance Control: An Approach to Manipulation. In: 1984 American Control Conference. pp. 304–313. IEEE, San Diego, CA, USA (Jul 1984). https://doi.org/10.23919/ACC.1984.4788393
7. Hogan, N.: Impedance Control: An Approach to Manipulation: Part I – Theory. Journal of Dynamic Systems, Measurement, and Control **107**(1), 1 (1985). https://doi.org/10.1115/1.3140702
8. Hogan, N.: Impedance Control: An Approach to Manipulation: Part II – Implementation. Journal of Dynamic Systems, Measurement, and Control **107**(1), 8 (1985). https://doi.org/10.1115/1.3140713
9. Hutter, M.: StarlETH & Co.: Design and control of legged robots with compliant actuation. PhD Thesis, ETH Zurich (2013). https://doi.org/10.3929/ethz-a-009915229
10. McMahon, T.A., Cheng, G.C.: The mechanics of running: How does stiffness couple with speed? Journal of Biomechanics **23**, 65–78 (Jan 1990). https://doi.org/10.1016/0021-9290(90)90042-2
11. Pratt, G., Williamson, M.: Series elastic actuators. In: Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots. vol. 1, pp. 399–406. IEEE Comput. Soc. Press, Pittsburgh, PA, USA (1995). https://doi.org/10.1109/IROS.1995.525827
12. Robinson, D.W.: Design and Analysis of Series Elasticity in Closed-loop Actuator Force Control by. PhD Thesis, Massachusetts Institute of Technology (2000)
13. Schwind, W., Koditschek, D.: Characterization of monoped equilibrium gaits. In: Proceedings of International Conference on Robotics and Automation. vol. 3, pp. 1986–1992. IEEE, Albuquerque, NM, USA (1997). https://doi.org/10.1109/ROBOT.1997.619161
14. Semini, C.: HyQ - Design and Development of a Hydraulically Actuated Quadruped Robot. Thesis (Ph.D.), Italian Institute of Technology, University of Genoa (2010)
15. Seok, S., Wang, A., Meng Yee Chuah, Otten, D., Lang, J., Kim, S.: Design principles for highly efficient quadrupeds and implementation on the MIT Cheetah robot. In: 2013 IEEE International Conference on Robotics and Automation. pp. 3307–3312. IEEE, Karlsruhe, Germany (May 2013). https://doi.org/10.1109/ICRA.2013.6631038
16. Silder, A., Besier, T., Delp, S.L.: Running with a load increases leg stiffness. Journal of Biomechanics **48**(6), 1003–1008 (Apr 2015). https://doi.org/10.1016/j.jbiomech.2015.01.051
17. Tedrake, R., the Drake Development Team: Drake: Model-based design and verification for robotics (2019), https://drake.mit.edu
18. Yu, H., Gao, H., Fan, Z., Deng, Z., Zhang, L.: Dual-slip model based galloping gait control for quadruped robot: A task-space formulation. In: 2017 American Control Conference (ACC). pp. 191–197. IEEE, Seattle, WA (May 2017)