

Design of galloping robots with elastic spine: tracking relations between dynamic model parameters based on motion analysis of a real cheetah*

Olga Borisova¹, Ivan Borisov¹, Sergey Kolyubin¹, Stefano Stramigioli^{1,2}

Abstract—One way to create a quadruped galloping robot from scratch is to design a brick-shaped body and utilize relatively simple open-chain leg mechanisms controlled with relatively complex control algorithms. Alternatively, we can look at how nature solved the same task designing fast mammals such as cheetah, and by means of morphological computation, we can design a complex mechanical system that has much of the desired behavior within inherent dynamics and only a little control effort is needed to stabilize or augment the motion.

In this paper, we have analyzed a real cheetah motion using video tracking and looked for a way to match the dynamic model parameters of the real cheetah with a galloping robot with an elastic spine. We believe the elastic spine is the essential feature for a fast-running energy-efficient galloping robot. Within this paper, we are focused on the flying stage when the elastic spine affects the motion of the robot's front and rear bodies. We have found how to optimize mass distribution and elasticity in the spine in order to get the cheetah-like galloping motion of a quadruped robot.

I. INTRODUCTION

Legged robots, i.e., running or galloping robots, are designed to operate under an unstructured environment where robots with wheels show poor performance. Galloping as highly dynamic locomotion is influenced by the dynamics of the robot's mechanical structure and its control algorithms' dynamics since they both shape the robot's motion.

The current generation of legged robots such as MIT Cheetah [1], ANYmal robot [2], Unitree [3], and HyQ [4] utilize the similar mechanical morphology, e.g., a brick-shaped body with an open-chain mechanism for legs. This kind of morphology is considered as a successful design since it allows to get various behaviors: standing, walking, crouching, lying, running, galloping, jumping, and etc.

At the same time, mammals can perform rotary gallop to achieve fast and energy-efficient motion. The essential feature is a *flexible spine* that allows to store potential energy during the impact and transfer it back to the kinetic energy during the takeoff to perform energy recuperation [5]. It is possible to use inspiration from biology to design a quadruped robot with a flexible spine such that it can gallop with little energy injection into the system.

*This work was supported by the Russian Science Foundation grant (project No17-79-20341)

¹Olga Borisova, Ivan Borisov, Sergey Kolyubin, Stefano Stramigioli are with the Biomechatronics and Energy-Efficient Robotics Lab, ITMO University, Saint Petersburg, Russia e-mail: {ovborisova, borisovii, s.kolyubin}@itmo.ru

²Stefano Stramigioli is also with the Department of Electrical Engineering, Mathematics and Computer Science, University of Twente, The Netherlands

There are contributions such as [6], [7], [8], and [9] that shows explicitly the benefits of utilizing the elastic spine for a quadruped robot to achieve fast, stable, and energy-efficient motion. For example, in [9] quadrupedal robots with stiff and flexible spine have been compared. To ensure that the comparison is fair the optimal control strategy has been used to find the most energy efficient joint trajectories, actuator inputs, and footfall timing for each model across a broad range of gaits and locomotion velocities. The results showed that the model with the flexible spine allows to achieve much higher speeds than with the rigid spine.

Proper allocation of passive elastic elements and mass distribution in the robot's mechanical structure potentially allow achieving faster dynamic locomotion with a more straightforward control system and less energy consumption. Such an approach is called *morphological computation*, when *computation* of control signal is carried out not only via algorithms but via a mechanical structure, e.g., its morphology. It allows to minimize control effort and use it only to stabilize, excite or augment the inherent dynamics [10].

In this paper, we are looking for a way to match the motion of a real cheetah with the motion of a galloping robot equipped with a spatial elastic spine. Our task is not to copy the cheetah's appearance but to copy its dynamics. This paper gives insights into how we can perform a top-down design approach to design a galloping robot with an elastic spine.

Section II focuses on analyzing the real cheetah motion using video tracking and developing a model of a galloping robot following the same motion. Knowing trajectories, velocities, and preferable mass of the robot, it is possible to solve inverse dynamics, which is given in Section III-D. The robot must obey the physics of cheetah galloping. Due to this, the optimization of mass distribution is given in Section IV. Section V is focused on search of spatial spring attachment and its coefficients. Knowing the proper mass distribution, position for a spatial spring, and coefficients of stiffness, we can repeat the motion of the real cheetah. Verification of the founded results is given in Section VI.

II. REAL CHEETAH MOTION

A. Video tracking

To analyze the motion of a real cheetah, the famous video "Cheetahs on the Edge" by National Geographic magazine [11] has been used. The video shows the rotational galloping of a real cheetah chasing a lure. Thanks to its flexible spine and long, fluid legs, a cheetah can do zero to 100 km/h in under three seconds. The detailed analysis of the cheetah rotary gallop is presented in [5].



Fig. 1: The motion sequence of a real cheetah from National Geographic [11]

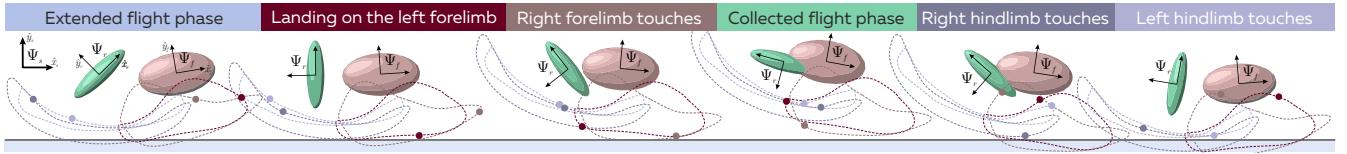


Fig. 2: Kinematic model of a set of rigid bodies representing a front and a rear body parts and feet

From the sequence of motion depicted in Fig. 1 it can clearly be seen that the front part of the body nearly keeps the orientation, while the rear part makes a significant rotational motion. Thus, we have decided to split the cheetah's body into two components: a relatively big and heavy front body part that almost keeps its orientation and a smaller and lighter rear body part that inject momenta into the whole body to move forward.

To model cheetah motion, we use ellipsoids of inertia to represent front and rear bodies. The trajectories of the running cheetah were retrieved from video footage. We have used MATLAB Computer Vision Toolbox and Image Processing Toolbox to perform object tracking using color markers. We put three color markers for the cheetah's rib cage, three more for the muscular hips, four more for feet, and one more for a branch on the background to measure motion in the coordinate system fixed in space.

The video analysis gave us the position of geometrical centers of the bodies x_i^c , y_i^c , and the angle of rotation φ_i^c in pixel units expressed in camera frame Ψ_c . Taking into account the video resolution and the length of an average cheetah, it is possible to restore the motion in meters. However, it is possible to scale the data to get a robot with the desired dimensions. Within this study, we consider a robot with specific dimensions and mass for the front body part, while the rear body's mass and dimensions have to be optimized to obey the physics.

Within this study, we considered that 358 video frames give 1 second of running. This velocity is less than the actual one in the video. We have decided to simplify the behavior of our model and decrease the velocity down to 10.0 km/h.

B. Calculation of trajectories

The received trajectories are relevant for a camera frame Ψ_c , which moves with respect to a frame fixed in space Ψ_s . Knowing x_i^c , y_i^c , and φ_i^c for each of the body i we can collect all these data into homogeneous matrix of configuration's transformation $H_i^c \in SE(3)$ from Ψ_i to Ψ_c

$$H_i^c = \begin{pmatrix} R_i^c & p_i^c \\ 0 & 1 \end{pmatrix},$$

where $R_i^c \in SO(3)$ is the rotation matrix and p_i^c is position.

To express motion in the frame Ψ_s we use the chain rule

$$H_r^s = H_c^s H_r^c = (H_s^c)^{-1} H_r^c,$$

where H_r^s represents motion of the rear body with frame Ψ_r with respect to Ψ_s , H_s^c is the motion of space frame Ψ_s with respect to Ψ_c . The same is relevant for spheres representing feet. Thus, we get trajectories that have to be followed by the bodies to get a cheetah-like dynamic behavior.

III. DYNAMIC MODEL FOR GALLOPING MOTION DESCRIPTION

A. Model overview

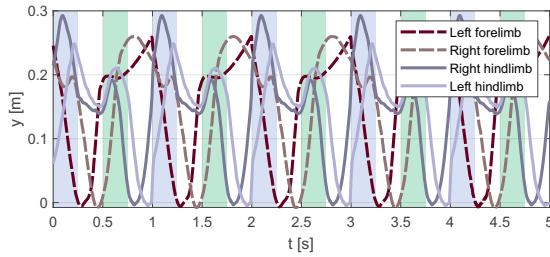
Since from the video analysis, we know the trajectories of the front and the rear robot's components along with the feet expressed in the frame fixed in space Ψ_s , it is possible to animate the desired motion of a robot to be designed. The model discussed within this section is a set of rigid bodies that mimics the cheetah motion.

Fig. 2 shows an animation sequence of a set of rigid bodies which follows the calculated trajectories. It can be interpreted as there are three actuators per body which move them along \hat{x}_s and \hat{y}_s axes and rotate around \hat{z}_s axis, which is orthogonal to \hat{x}_s and \hat{y}_s . The pink ellipsoid with the frame Ψ_f indicates the massive front body's component, the green ellipsoid with the frame Ψ_r indicates the rear light component, and the spheres represent the feet which move along the trajectories expressed in the frame Ψ_f . The model has been implemented in MATLAB using Simscape Multibody.

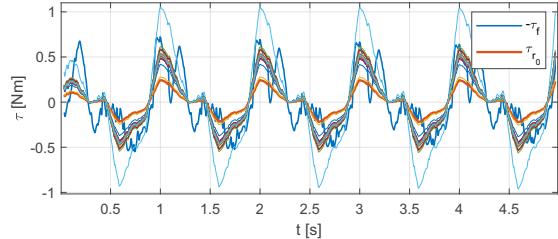
B. Rotary gallop phases

The model follows the rotary gallop of the real cheetah, but it is scaled down in terms of trajectories and velocities. However, the model follows the same type of periodic motion that can be splitted on several phases depicted in Fig. 2 [5].

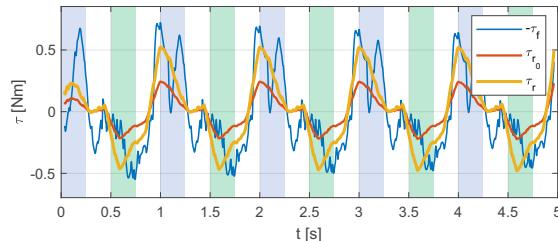
Following the major flight phase of the gait, which is called *the extended phase*, the robot lands on its left forelimb. While the spine is starting to load, the front robot's body lands on the right forelimb. The brief intervening *collected flight phase* follows. The spine gets maximum deflection to inject the stored energy to help the hindlimbs lift off after the following landing on the robot's hindlimbs. After the second hindlimb contact, the robot vaulting into the extended flight phase of the cycle.



(a) Feet's y coordinates expressed in Ψ_s : $y > 0$ means the flight phase



(b) Progress for τ_r optimization: $-\tau_f$ indicates the front body torque with the opposite sign, τ_{r0} shows the initial rear body torque, and all other lines stand for possible rear body torques



(c) Optimization results for τ_r : $-\tau_f$ is the front body torque with the opposite sign, τ_{r0} and τ_r indicate the initial and optimized rear body torque respectively

Fig. 3: Inputs and outputs for the rear body torque optimization. The blue and green stripes indicate the extended and collected flight phases respectively

C. Spatial forces applied to the bodies

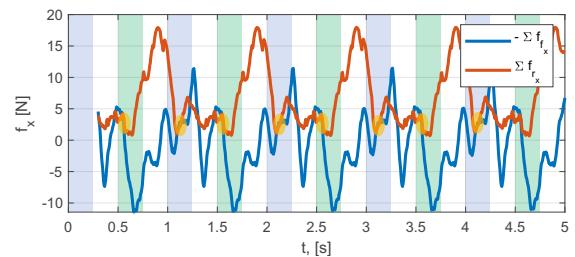
Since we extracted the trajectories from the video footage and know preferable mass of the running cheetah-robot, it is possible to use the *inverse dynamics* to determine the inter-body and external forces and torques that can be used later for the design optimization. The similar approach has been introduced in [12] and [13].

We can merge the torques τ and forces f applied to a body i into a single six-dimensional spatial wrench $W \in se^*(3)$ expressed in Ψ^j

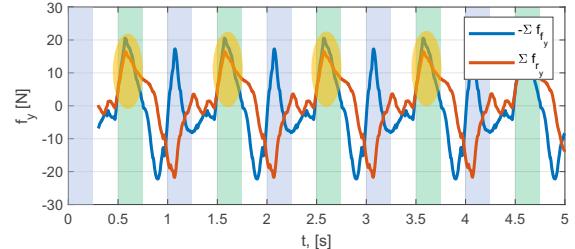
$$(W_i^j)^\top = \begin{pmatrix} \tau_{i_x}^j & \tau_{i_y}^j & \tau_{i_z}^j & f_{i_x}^j & f_{i_y}^j & f_{i_z}^j \end{pmatrix}.$$

Although the planar motion is discussed, we consider bodies as three-dimensional inertia ellipsoids, such as the rotational inertia matrix $J \in \mathbb{R}^{3 \times 3}$. For the planar motion, considered within the paper, we have that $\tau_{i_x}^j = \tau_{i_y}^j = f_{i_z}^j = 0$.

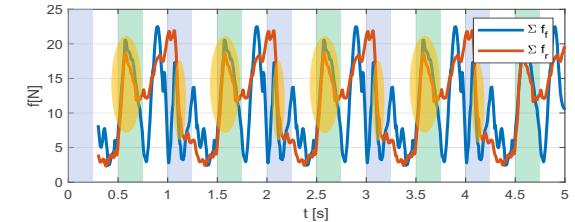
Front and rear bodies experience two external wrenches. The first wrench W_{l_i} is generated by the limbs because of



(a) f_x components for the front $-\sum f_{fx}$ and the rear f_{rx} bodies



(b) f_y components for the front $-\sum f_{fy}$ and the rear f_{ry} bodies



(c) Force magnitudes for the front $\sum f_f$ and the rear $\sum f_r$ bodies

Fig. 4: Comparison of the forces components. The blue and green stripes indicate the extended and collected flight phases respectively. Orange ellipsoids indicate equal forces for the flight phases

the swing motion in the flight phase and the impact during the stance phase. The second wrench W_{s_i} is generated by the spatial spine that holds the robot's components together. For each component we know that the sum of all applied Wrenches equal to the total wrench $\sum W_i$

$$\sum W_r = W_{sr} + W_{lr}, \quad (1)$$

$$\sum W_f = W_{sf} + W_{lf}, \quad (2)$$

where r stands for the rear body and f for the front body.

In the flight phase, the wrenches have to match

$$\dot{P}_r = -\dot{P}_f, \quad (3)$$

where P_i is the generalized momentum of a body i .

We use inertia matrix I^i to describe mass distribution for a body i with the mass m_i and the rotational inertia matrix $J_i \in \mathbb{R}^{3 \times 3}$

$$I^i = \begin{pmatrix} J_i & 0 \\ 0 & m_i I \end{pmatrix}, \quad J_i = \begin{pmatrix} j_{x_i} & 0 & 0 \\ 0 & j_{y_i} & 0 \\ 0 & 0 & j_{z_i} \end{pmatrix},$$

where I is identity matrix, principle inertia's $j_{x_i} = \frac{1}{3}m_i(b_i^2 + c_i^2)$, $j_{y_i} = \frac{1}{3}m_i(a_i^2 + c_i^2)$, $j_{z_i} = \frac{1}{3}m_i(a_i^2 + b_i^2)$, and a_i , b_i , and c_i are principal semi-axes.

The mass m_f and dimensions of the front body, e.g., principal semi-axes a_f , b_f , and c_f of the inertia ellipsoid, have to be defined by a designer based on actuators, batteries, controllers, materials, etc. Thus, the inertia I^f is fully defined and the rear's body properties must be found to satisfy eq.(3).

D. Inverse dynamics

For the given homogeneous transformation matrices $H \in SE(3)$ we can calculate the Twists $T \in se(3)$

$$\dot{T}_i^{s,s} = \dot{H}_r^s H_s^r, \quad \dot{T}_f^{s,s} = \dot{H}_f^s H_s^f,$$

where Twist $\dot{T}_a^{b,c}$ indicates velocity of a body a with respect to a frame Ψ_c expressed in a frame Ψ_b . For the motion expressed in the principal inertia frame Ψ_i of a body i according to the Newton's second law we have

$$(\dot{P}^i)^\top = I^i \dot{T}_i^{i,s} = \sum W_i^{i,\top} + ad_{T_i^{i,s}}^\top P^i \top,$$

where $ad_{T_i^{i,s}}$ is the adjoint of the Lie algebra $se(3)$.

Thus, we can calculate a total wrench $\sum W_i^i \in se^*(3)$ exerted on the body i , i.e., solve the *inverse dynamics task*

$$\sum W_i^{i,\top} = I^i \dot{T}_i^{i,s} - ad_{T_i^{i,s}}^\top P^i \top. \quad (4)$$

The wrenches calculated according to eq. (4) are noisy because of the video tracking measurements and the Twists' time derivative. Because of that, the calculated wrenches has been smoothed using *movmean* function in MATLAB.

IV. MASS DISTRIBUTION ANALYSIS

The flexible spine that holds the bodies together generates the wrench applied to each component such that the magnitude is the same, but the direction is opposite

$$W_{s_r} = -W_{s_f}. \quad (5)$$

The legs' wrenches take the maximum values when impact with the ground takes place. During the flight phase, legs make the swing motion; however, we can suppose that the legs' wrenches are equal to zero in the apex flight point $W_{l_r} = W_{l_f} = 0$. Fig. 3a shows feet's y coordinates expressed in Ψ_s . When $y > 0$ it means the robot in the flight phase. Considering eq. (1), (2), and (5) for the flight phase we get the necessary condition for bodies' inertia to obey the physics

$$\sum W_f + \sum W_r = 0. \quad (6)$$

The mass m_r and the rotational inertia matrix J_r of the rear body must match the eq. (6). Since the front body's physical parameters are given, and the desired wrench values are known for *only the flight phases*, we can set the optimization task to find the principal semi-axes a_r , b_r , and c_r of the inertia ellipsoid and the mass m_r for the rear body. Thus, the next optimization cost functions have been chosen

$$F = \sum_{j=1}^n \sqrt{(\tau_{f_j} + \tau_{r_j})^2 + k_x(f_{fx_j} + f_{rx_j})^2 + k_y(f_{fy_j} + f_{ry_j})^2},$$

where τ_i , f_{ix} , and f_{iy} are the components of a wrench $\sum W_i$, $k_x = \frac{\max \tau_f^2}{\max f_{fx}^2}$ and $k_y = \frac{\max \tau_f^2}{\max f_{fy}^2}$ are the weight coefficients.

TABLE I: Bodies' physical parameters

| Parameter | Front | Rear (initial) | Rear (optimized) |
|--------------|-------|----------------|------------------|
| a_i , [m] | 0.22 | 0.10 | 0.1419 |
| b_i , [m] | 0.12 | 0.03 | 0.0708 |
| c_i , [m] | 0.04 | 0.03 | 0.0310 |
| m_i , [kg] | 4 | 3 | 2.8213 |

Inertia matrix for the front body I^f is given, the inertia matrix for the rear body I^r must be found. The rotational inertia matrix J_r depends on the principal semi-axes a_r , b_r , and c_r and m_r , thus we have chosen them as the parameters to search.

In addition to restrictions on the maximum and minimum sizes for the ellipsoid, we define that the rear body has to be smaller than the front body

$$a_r < k_a \cdot a_f, \quad b_r < k_b \cdot a_r, \quad c_r < k_c \cdot b_r,$$

where k_a , k_b , and k_c are the weight coefficients.

Fig. 3b shows the initial rear body torque τ_{r_0} , the front body torque with the opposite sign $-\tau_f$, which is considered as the desired one, and possible rear body torques with different values of principal semi-axes. The optimized torque τ_r for the rear body is shown in Fig. 3c. The optimization has been conducted in MATLAB using Global Optimization Toolbox and *patternsearch* has been used to find minimum of the cost function. The physical parameters for the front and the rear bodies (initial and optimized) are given in the Table I.

Fig. 4a and Fig. 4b compare the f_x and f_y force components of wrenches eq. (6) for both bodies. Fig. 4c compare force magnitudes for the front $\sum f_f = \sqrt{f_{fx}^2 + f_{fy}^2}$ and the rear $\sum f_r = \sqrt{f_{rx}^2 + f_{ry}^2}$ bodies. Orange ellipsoids indicate equal forces for the flight phases: f_x components coincide only in apex points for both flight phases, while the f_y components coincide during the whole collected phase and are completely opposite during the extended phase. During the take-off phase f_x components are directed on opposite sides, while f_y components have the similar profile but with time offset.

V. ELASTICITY ALLOCATION

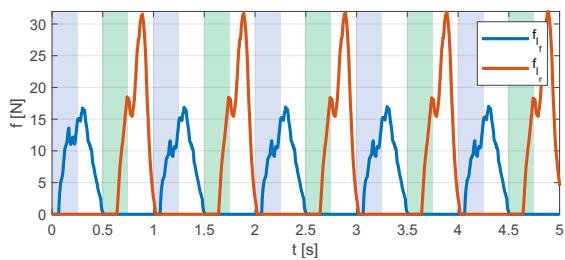
A. Wrenches generated by limbs

For the stance phases, taking into account eq. (1), (2), and (5) we have the total wrench produced by the limbs

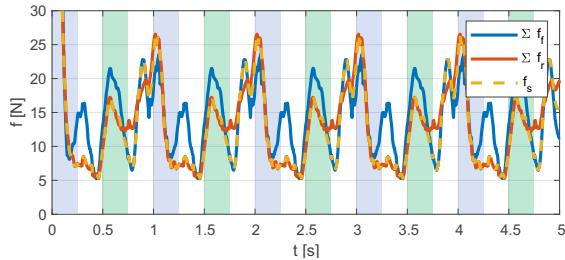
$$\sum W_l = W_{l_r} + W_{l_f} = \sum W_r + \sum W_f. \quad (7)$$

Since we know when each limb touches the ground (Fig. 3a), we can split the total legs wrench $\sum W_l$ on the front W_{l_f} and the rear W_{l_r} components.

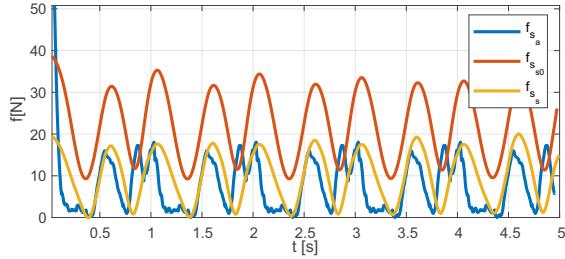
Fig. 5a shows the force f_{l_f} generated by the front legs and by the rear legs f_{l_r} . If a limb in contact the force for the plot is calculated as $f = \sqrt{f_x^2 + f_y^2}$, where f_x and f_y are components from the corresponding wrench (eq. 7), if the robot in the flight phase it is considered the wrench must be equal to zero in apex point $\sum W_l = 0$. The blue and green stripes indicate the extended and collected flight phases, respectively. It can be seen that the rear legs' force is twice bigger than the front body force.



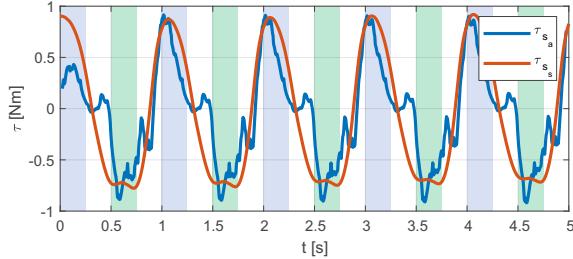
(a) Generated forces f_{l_f} and f_{l_r} by fore- and hindlimbs respectively



(b) Calculation of analytical estimation of the desired spring force: Σf_f and Σf_r are the total forces applied to the front and the rear body respectively, f_s is the analytical estimation of the spring force



(c) Optimization results for elasticity allocation: f_{s_a} is the analytical estimation of the desired spring force, $f_{s_{s0}}$ is the initial spatial spring force from the simulation, and f_{s_s} is the optimized value



(d) Optimization results for elasticity allocation: torque optimization

Fig. 5: Elasticity allocation optimization

B. The desired wrench generated by spine

If each leg's wrench component is known, we can get the pure wrench generated by the elastic spine

$$W_{s_r} = \sum W_r - W_{l_r} = -\sum W_f + W_{l_f} = -W_{s_f}.$$

Fig. 5b shows the analytical estimation of the desired spring force $f_s = \sqrt{f_{s_x}^2 + f_{s_y}^2}$, where f_{s_x} and f_{s_y} are the force components from W_s . It can be seen that the force is transferred between the front Σf_f and the rear Σf_r bodies.

C. Spring model

The idea is to use passive spatial spring for the elastic spine. Since the desired spine wrench W_s is found, we need to find where the spring should be attached and what are the stiffness coefficients.

The spine is modeled as a geometric spring parameterized by a center of compliance [14], where the spring locally behaves as a decoupled rotational stiffness k_z and translational stiffness $K_t = \begin{pmatrix} k_x & 0 \\ 0 & k_y \end{pmatrix}$. The spine wrench is applied in frame Ψ_{s_r} and Ψ_{s_f} as a function of $H_{s_r}^{s_f}$

$$W_s^{s,s_r}(H_{s_r}^{s_f}) = (\tau_z^{s_r} \quad f_x^{s_r} \quad f_y^{s_r})^\top,$$

where $\tau_z^{s_r}$ is a torque and $f_x^{s_r}$ and $f_y^{s_r}$ are linear forces generated by a spatial spring and applied to the frame Ψ_{s_r} . Implementation of a model of spatial spring is given in [15].

In planar case, the spring has only 3 degrees of freedom

$$\begin{pmatrix} 0 & -f_z^{s_r} \\ \tau_z^{s_r} & 0 \end{pmatrix} = -2\text{as}(G_o R_{s_r}^{s_f}) - \text{as}(G_t w w^\top),$$

$$f^{s_r} = -\frac{1}{2} R_{s_f}^{s_r} K_t p_{s_r}^{s_f} - \frac{1}{2} K_t R_{s_f}^{s_r} p_{s_r}^{s_f},$$

where $\text{as}()$ is an operator which takes the skew-symmetric part of a matrix, $R_{s_f}^{s_r}$ and $p_{s_f}^{s_r}$ are the subparts of the matrix $H_{s_f}^{s_r}$ and G_o , G_t , G_c are called respectively orientational, translational and coupling co-stiffnesses of the spring [16], $w = R_{s_f}^{s_r} \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} p_{s_r}^{s_f}$ is used to calculate the moment resulting from the translational forces of an anisotropic spring. Co-stiffness matrices follows

$$G_t = \frac{1}{2} \text{tr}(K_t) I - K_t, \quad G_o = \frac{1}{2} k_z I.$$

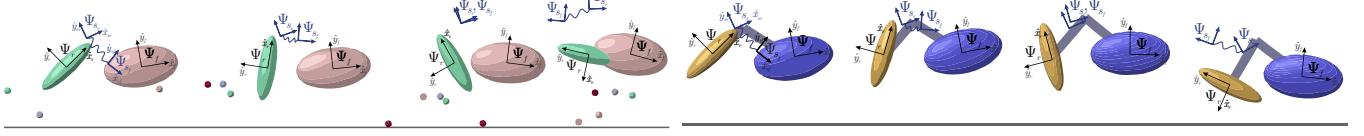
Since we know the spring deflection $H_{s_r}^{s_f}$ and the desired forces and torque that should be generated by the spine, we can set the optimization task. The next optimization cost functions has been chosen

$$F = \sum_{i=1}^n \sqrt{(\tau_{s_{a_i}} - \tau_{s_{s_i}})^2 + k(f_{s_{a_i}} - f_{s_{s_i}})^2},$$

where $\tau_{s_{a_i}}$ and $f_{s_{a_i}}$ are the desired torque and force values respectively from the analytical estimation of the spine Wrench W_{s_r} ; $\tau_{s_{s_i}}$ and $f_{s_{s_i}}$ are the springs torque and force values respectively from the spring Wrench $W_s^{s,s_r}(H_{s_r}^{s_f})$ have to be optimized, $k = \frac{\max \tau_{s_{a_i}}^2}{\max f_{s_{a_i}}^2}$ is the weight coefficients.

The minimal energy configuration, i.e., equilibrium position of the spring is when Ψ_{s_r} and Ψ_{s_f} coincide and $H_{s_r}^{s_f} = I$. Thus, to get rid of spring pretension we have adjusted the desired spring force f_{s_a} such that the minimum value during the touchdown is zero. The frames Ψ_{s_r} and Ψ_{s_f} are set as $H_{s_r}^r$ and $H_{s_r}^f$ respectively: x_{s_i} , y_{s_i} are coordinates, and ϕ_{s_i} is the angle of rotation around \hat{z} axis for a body i .

Stiffness coefficients k_x , k_y , and k_z are the parameters to be found. The optimization has been conducted in MATLAB using Global Optimization Toolbox and *patternsearch* has been used to find minimum of the cost function. The initial and optimized parameters are given in the table II.



(b) Dynamic model of floating bodies connected with spatial spring

Fig. 6: Verification of elasticity allocation optimization

TABLE II: Elasticity allocation parameters

| Parameter | ϕ_{sr} , [rad] | ϕ_{sf} , [rad] | x_{sr} , [m] | x_{sf} , [m] | y_{sr} , [m] | y_{sf} , [m] | k_x , [N/m] | k_y , [N/m] | k_z , [Nm/rad] |
|-----------|---------------------|---------------------|----------------|----------------|----------------|----------------|---------------|---------------|------------------|
| initial | -0.5 | -0.04 | -0.04 | -0.30 | -0.05 | 0.40 | 70 | 100 | 0 |
| optimized | -1.4312 | 0.2348 | 0.1858 | -0.2879 | -0.1580 | 0.2168 | 54.2833 | 83.4056 | 0.9 |

VI. VERIFICATION

Fig. 6a shows the initial kinematic model of a set of rigid bodies that move along the calculated trajectories. The spatial spring has been attached to the founded frames Ψ_{sr} and Ψ_{sf} . The total linear force generated by the spatial spring $f_{ss} = \sqrt{f_{ssx}^2 + f_{ssy}^2}$ is shown in Fig. 5c. The optimized torque τ_{ss} is presented in Fig. 5d. The force profile f_{ss} alike to the calculated estimation f_{sa} , and the torque coincide the τ_{sa} , τ_r , and $-\tau_f$.

Fig. 6b shows a dynamics model of two floating bodies representing a front and the rear robots component connected with the calculated spatial spring. The initial momenta are applied to the bodies, while bodies experience free falling. The whole extended flight phase is shown in Fig. 6b. The bodies move alike the original kinematic model. The swing motion of the orange rear body is perform by a spatial spring only. To restrict the possible motion a simple linkage with two revolute joints has been synthesised, that allows the robot to perform the motion alike the original kinematic model with a set of rigid bodies.

VII. CONCLUSION

This paper has solved the dynamics matching problem for the highly dynamic locomotion robot, which mimics the cheetah's rotary gallop. Here our attempt was to identify nature principles and transfer them to a robot design. The current generation of impressive galloping robots consists of a single rigid body with four legs attached. However, similarly to the fact that cheetah's legs are responsible mostly for ground clearance and adaptivity and spine does all the work, it is considered that the performance of galloping robots can be increased because of a flexible spine. We believe the morphological computation is the way to achieve a highly dynamic energy-efficient locomotion when the 'main' actuator is an elastic element that oscillates because of proper mass distribution and motion reconfiguration.

In this paper, we have found how we can distribute the mass among the robot's body components and the allocation of elasticity such that the spine does most of the work. Exploiting the natural dynamics can lead to energy efficiency and little control efforts.

REFERENCES

- [1] B. Katz, J. D. Carlo, and S. Kim, "Mini cheetah: A platform for pushing the limits of dynamic quadruped control," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 6295–6301.
- [2] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Science robotics*, vol. 5, no. 47, 2020.
- [3] X. Da, Z. Xie, D. Hoeller, B. Boots, A. Anandkumar, Y. Zhu, B. Babich, and A. Garg, "Learning a contact-adaptive controller for robust, efficient legged locomotion," *arXiv preprint arXiv:2009.10019*, 2020.
- [4] S. Fahmi, G. Fink, and C. Semini, "On state estimation for legged locomotion over soft terrain," *IEEE Sensors Letters (L-SENS)*, vol. 5, no. 1, pp. 1–4, Jan. 2021.
- [5] J. E. Bertram and A. Gutmann, "Motions of the running horse and cheetah revisited: fundamental mechanics of the transverse and rotary gallop," *Journal of the Royal Society Interface*, vol. 6, no. 35, pp. 549–559, 2009.
- [6] M. H. H. Kani, M. Derafshian, H. J. Bidgoly, and M. N. Ahmadabadi, "Effect of flexible spine on stability of a passive quadruped robot: Experimental results," in *2011 IEEE International Conference on Robotics and Biomimetics*, 2011, pp. 2793–2798.
- [7] C. D. V. Remy, "Optimal exploitation of natural dynamics in legged locomotion," Zürich, 2011.
- [8] P. Eckert, A. Spröwitz, H. Witte, and A. J. Ijspeert, "Comparing the effect of different spine and leg designs for a small bounding quadruped robot," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 3128–3133.
- [9] Y. Yesilevskiy, W. Yang, and C. D. Remy, "Spine morphology and energetics: how principles from nature apply to robotics," *Bioinspiration and biomimetics*, vol. 13 3, 2018.
- [10] G. A. Folkertsma, A. J. van der Schaft, and S. Stramigioli, "Morphological computation in a fast-running quadruped with elastic spine," *IFAC-PapersOnLine*, vol. 48, no. 13, pp. 170–175, 2015.
- [11] R. Smith, "Cheetahs on the edge," *National Geographic Magazine*, 2012. [Online]. Available: <https://www.nationalgeographic.com/magazine/article/cheetahs>
- [12] M. A. Brubaker, L. Sigal, and D. J. Fleet, "Estimating contact dynamics," in *2009 IEEE 12th International Conference on Computer Vision*. IEEE, 2009, pp. 2389–2396.
- [13] C. K. Liu, A. Hertzmann, and Z. Popović, "Learning physics-based motion style with nonlinear inverse optimization," *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 1071–1081, 2005.
- [14] G. Folkertsma, "Energy-based and biomimetic robotics," Ph.D. dissertation, University of Twente, Netherlands, Apr. 2017.
- [15] S. Stramigioli and V. Duindam, "Variable spatial springs for robot control applications," in *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No. 01CH37180)*, vol. 4. IEEE, 2001, pp. 1906–1911.
- [16] E. D. Fasse and J. F. Broenink, "A spatial impedance controller for robotic manipulation," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 4, pp. 546–556, 1997.