Optimal spine morphology for high-speed bounding in quadruped robots

Shalom Abiodun[1*], *Stacey* Shield[1], *Amir* Patel[2], *Reuben* Govender[3], and *Heather* Wimberley[1]

[1]Electrical Engineering Department, University of Cape Town, South Africa
[2]Computer Science Department, University College London, United Kingdom
[3]Mechanical Engineering Department, University of Cape Town, South Africa

Abstract. Quadrupedal animals leverage flexible spines to achieve agile locomotion, yet most robotic counterparts, like the Kemba robot, retain rigid or revolute spinal joints, limiting dynamic performance. While prior studies suggest that prismatic spines enhance acceleration in quadruped robots, these often neglect physical actuator constraints inherent in real-world systems. This study investigates spine morphology—rigid, revolute, and prismatic—on the steady-state bounding of Kemba using a dynamics-driven framework. Planar models for each spine configuration integrate actuator torque, velocity, and piston force constraints, alongside complementarity conditions for ground and valve contact interactions. A second-order Radau contact-implicit direct collocation method optimises trajectories using a bio-inspired contact sequence derived from cheetah and greyhound gaits. Results demonstrate the prismatic spine's superiority in stride length and velocity over rigid and revolute designs, attributed to enhanced leg extension and ground reaction force alignment. The prismatic spine also exhibits energy storage potential through linear actuation, suggesting efficiency gains. These findings underscore the importance of co-optimising spine morphology and actuator constraints in bio-inspired robots. This work bridges simulation-based biomechanics with practical robotic design, advancing Kemba's capabilities. Future efforts will experimentally validate these models through the physical integration of a prismatic spine.

1 Introduction

Quadrupedal animals achieve remarkable locomotion through specialised anatomical structures. A flexible spine, in particular, enhances agility, stability, and acceleration. Inspired by these biological systems, robotic quadrupeds have been developed to replicate their dynamic movement patterns. One such example is Kemba [1], a planar quadruped that leverages pneumatic actuation to mimic the explosive acceleration of a cheetah. However, unlike its biological counterpart, Kemba lacks a flexible spine—a key feature contributing to both agility and speed in nature.

* Corresponding author: abdsha033@myuct.ac.za

Fig. 1. Kemba, a hybrid pneumatic and electric actuated legged robot. [1]

Although speed is often the primary benchmark, a compliant spine also provides manoeuvrability and adaptability—critical for navigating unstructured and unpredictable environments. In nature, spinal flexibility extends beyond stride length: it enables agile turning, rapid reorientation, and dynamic stability during complex manoeuvres such as dodging obstacles or traversing uneven terrain. For robots, these capabilities could expand operational versatility in domains such as search-and-rescue, exploration, and military reconnaissance. While these broader benefits are outside the scope of this study, they underscore the potential value of exploring spine morphologies that enhance both speed and adaptability.

Studies of the cheetah's spine demonstrate a strong correlation between flexion–extension motion and running speed. Hildebrand [2] suggests that changes in spine length enable greater limb swing, thereby increasing stride length. As shown in Figure 2, spinal flexion allows the hind legs to reach further forward after the forelegs contact the ground [2]. This motion helps the cheetah generate greater acceleration from its hind legs, propelling it forward more effectively.

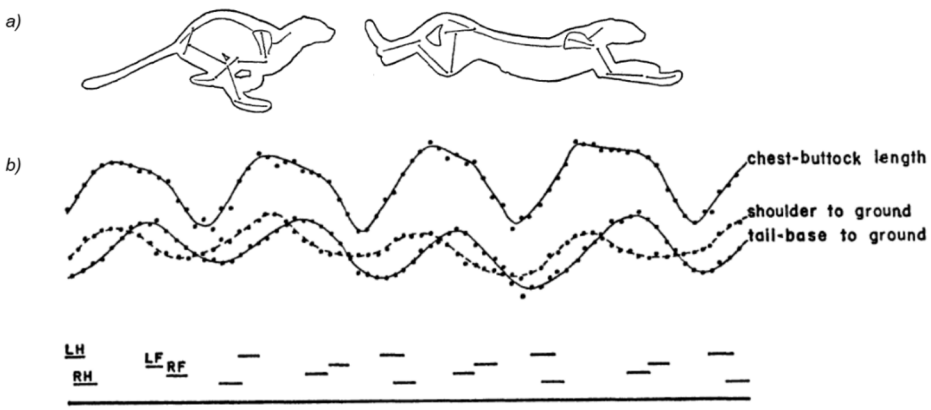

Fig. 2. (a) The bounding cheetah, showing maximum flexion and extension of the spine. [2]
(b) Cheetah's relation of body movement to action of the feet during four strides. Motion is from left to right. Lower broken lines show periods of contact of the feet with the ground; letters R, L, and F mean right, left, hind and front, respectively. Upper curve indicates variation in chest-buttock length, while middle curves depict the height of shoulders and tail base above the ground.[2]

Previous research suggests that a prismatic spine design is most effective for rapid acceleration in quadruped robots [3]. However, most existing quadrupeds incorporate either rigid or revolute spine designs [4], limiting their ability to reproduce the combined speed and

agility benefits observed in biological systems. Fisher et al. [3], for example, conducted a large-scale optimisation study on spine morphology from a bio-inspired perspective. Yet their approach did not account for the physical constraints of a specific robotic platform, instead applying biologically inspired rather than mechanically derived actuator limits.

This paper addresses these limitations by investigating the effect of spine morphology on Kemba's maximum speed. We develop dynamic models for three spine configurations—rigid, revolute, and prismatic—while incorporating both mechanical and complementarity constraints. Using a biologically inspired contact order, a bounding trajectory is then generated and solved through cost-function minimisation. These simulations provide insight into the mechanical feasibility and performance implications of incorporating a flexible spine into Kemba's design.

2 Method

This study examines a quadruped system with varied spinal configurations to analyse its effect on locomotion. It does so by performing multiple trajectory optimisation runs, each determining a single steady-state bound for a given spine morphology. To do this, we first create dynamic models of each spine morphology with parameters stemming from Kemba [1], as shown in Figure 3. The spine morphologies are the rigid (Figure 3.a), revolute (Figure 3.b), and prismatic (Figure 3.c). As mentioned in the previous chapter, the rigid body is the most commonly found spine morphology due to its simplicity. The revolute spine morphology mimics the flexibility of a cheetah by using a spinal joint to control the angular position of the front and hind spine bodies. In contrast, the prismatic spine leverages linear actuation to mimic the extension and retraction of a cheetah's spine.

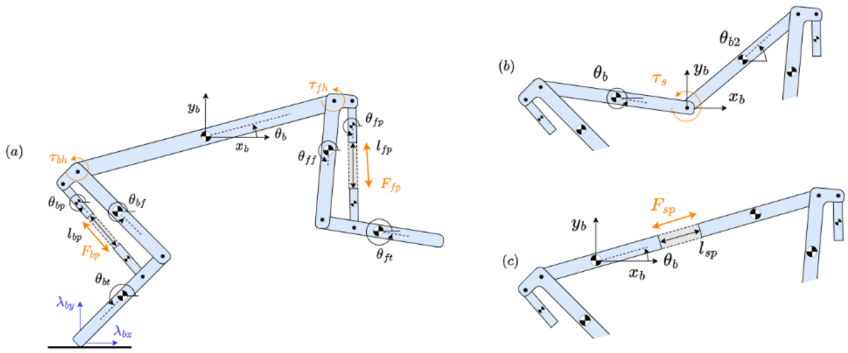

Fig. 3. Rigid (a), Revolute (b), and Prismatic (c) Spine Designs. The generalised coordinates, as well as the applied forces and torques, have been shown. All angles are absolute angles with respect to the horizontal.

Once these dynamic models have been created, constraints need to be applied to these models to ensure that certain physical limitations are adhered to. Constraints can fall under two forms:

1) *Mechanical constraints*: These constraints mimic the physical properties of the actuations.
2) *Complimentary constraints*: These constraints are mathematical relationships used to model contacts.

The contact order is defined to enforce a bounding motion. Contact order refers to a specific timing and nature of contact events between the dynamic model and the ground. In the case of this system, the contact order follows the biologically inspired contact of a cheetah during steady state bounding.

Lastly, the optimisation is set to solve and minimize a cost function. This system uses a two-step optimisation process, solving for two separate cost functions. This is later detailed below.

2.1 Dynamic model

The rigid spine model in Figure 3.a was originally developed by Mailer [1] and was updated to include the revolute and prismatic spine designs. The general coordinates are as follows:

$$\boldsymbol{q}_{rigid} = [x_b\ y_b\ \theta_b\ \theta_{ff}\ \theta_{fp}\ l_{fp}\ \theta_{ft}\ \theta_{bf}\ \theta_{bp}\ l_{bp}\ \theta_{bt}]^T \quad (1)$$

$$\boldsymbol{q}_{revolute} = [x_b\ y_b\ \theta_b\ \theta_{ff}\ \theta_{fp}\ l_{fp}\ \theta_{ft}\ \theta_{bf}\ \theta_{bp}\ l_{bp}\ \theta_{bt}\ \theta_{b2}\ ]^T \quad (2)$$

$$\boldsymbol{q}_{prismatic} = [x_b\ y_b\ \theta_b\ \theta_{ff}\ \theta_{fp}\ l_{fp}\ \theta_{ft}\ \theta_{bf}\ \theta_{bp}\ l_{bp}\ \theta_{bt}\ l_{sp}]^T \quad (3)$$

For the rigid spine model, x_b, y_b, and θ_b **,** are the position and orientation states of the spine centre of mass (CoM), while the revolute spine has a motor attached to the hind torso that controls the angular displacement of the front torso. Therefore θ_b represents the orientation of the hind torso, while θ_{b2} is the orientation of the front torso. The prismatic spine has a linear actuator between the hind and front torsos, with l_{sp} representing the displacement of the front torso. In this case, x_b, y_b, and θ_b are the position and orientation states of the hind torso. Since the leg model for each design is identical, all three models have the same leg characterisation. The orientation states of the front femur and tibia are θ_{ff} and θ_{ft}, while the back femur and tibia are θ_{bf} and θ_{bt}. The piston lengths for the front and back leg pistons are l_{fp} and l_{bp}.

The applied forces and torques from the motors and linear actuations are

$$\boldsymbol{F}_{rigid} = [F_{fp}\ F_{bp}\ ]^T\ ,\ \boldsymbol{\tau}_{rigid} = \left[\tau_{fh}\ \tau_{bh}\right]^T \quad (4)$$

$$\boldsymbol{F}_{revolute} = [F_{fp}\ F_{bp}\ ]^T\ ,\ \boldsymbol{\tau}_{revolute} = \left[\tau_{fh}\ \tau_{bh}\ \tau_s\right]^T \quad (5)$$

$$\boldsymbol{F}_{prismatic} = [F_{fp}\ F_{bp}\ F_{sp}\ ]^T\ ,\ \boldsymbol{\tau}_{prismatic} = \left[\tau_{fh}\ \tau_{bh}\right]^T \quad (6)$$

where F_{fp} and F_{bp} are the front and back piston forces, and τ_{fh} and τ_{bh} are the front and back hip motor torques. In the revolute spine model, τ_s is the spinal joint motor torque, while F_{sp} represents the linear actuator force in the spine.

The ground reaction forces (GRFs) are defined as:

$$\boldsymbol{GRF} = \left[\lambda_{fx}\ \lambda_{fy}\ \lambda_{bx}\ \lambda_{by}\right]^T \quad (7)$$

where λ_{fx} and λ_{fy} are the front foot horizontal and vertical GRFs, and λ_{bx} and λ_{by} are the back foot horizontal and vertical GRFs.

The body parameters in Table 2 (can be found in the Appendix) are taken from Kemba and adapted for the planar rigid spine model. Using Lagrange equations of motion, the EOM of the system was derived:

$$\frac{d}{dt}\left(\frac{\partial \boldsymbol{T}}{\partial \dot{\boldsymbol{q}}}\right) - \left(\frac{\partial \boldsymbol{T}}{\partial \boldsymbol{q}}\right)^T + \left(\frac{\partial \boldsymbol{V}}{\partial \boldsymbol{q}}\right)^T = \boldsymbol{Q} \quad (8)$$

where kinetic energy, $\boldsymbol{T}(\boldsymbol{q}, \dot{\boldsymbol{q}})$, was the sum of the translational and rotational energies of all moving parts, and potential energy, $\boldsymbol{V}(\boldsymbol{q})$, accounts for gravity acting on each components centre of mass. Lastly, the non-conservative forces and torques, $\boldsymbol{Q}(\boldsymbol{q}, \dot{\boldsymbol{q}})$, acting on the system are projected onto the system's generalized coordinates.

$$\boldsymbol{T}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \frac{1}{2}\dot{\boldsymbol{q}}^T \boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}} \quad (9)$$

$$\boldsymbol{V}(\boldsymbol{q}) = \sum_i m_i\ g\ y_{c,i}(\boldsymbol{q}) \quad (10)$$

$$\boldsymbol{Q}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \sum_j \left(\frac{\partial \boldsymbol{r}_j}{\partial \boldsymbol{q}}\right)^T \boldsymbol{F}_j + \sum_k \left(\frac{\partial \boldsymbol{\theta}_k}{\partial \boldsymbol{q}}\right)^T \boldsymbol{\tau}_k \quad (11)$$

Where ***M(q)*** is the mass matrix, m_i is the individual mass of a body and $y_{c,i}$ is the vertical position of the centre of mass of that body. The external forces, $\boldsymbol{F}_j$, are the GRFs (***GRF***) and

applied forces ($\boldsymbol{F}_{rigid}/\boldsymbol{F}_{revolute}/\boldsymbol{F}_{prismatic}$) acting at the position $\boldsymbol{r}_j$. The motor torques $\boldsymbol{\tau}_k$ ($\boldsymbol{\tau}_{rigid}/\boldsymbol{\tau}_{revolute}/\boldsymbol{\tau}_{prismatic}$) are applied at the hip and body joint angles $\boldsymbol{\theta}_k$.

2.2 Mechanical constraints

Certain bounds were placed on the properties of the actuators, to reflect their physical limitations. The constraints are:

- Motor torque limits: $-\tau_{max} \leq \tau \leq \tau_{max}$
- Motor angular velocity: $-\dot{\theta}_{max} \leq \dot{\theta} \leq \dot{\theta}_{max}$
- Piston length: $0.048 \leq l_p \leq 0.048 + l_{p,max}$, where $l_p = \begin{bmatrix} l_{fp} \\ l_{bp} \end{bmatrix}$ and 0.048 is the length between the attachment points of the piston body and rod, when $l_p = 0$
- Linear actuator: $0 \leq l_{sp} \leq \frac{l_b}{2}$
- Piston force follows the piston model created by Mailer [1].

The values of the actuator limits can be found in the Appendix.

2.3 Complementarity constraints

Since there are various contacts in our system, complementarity constraints are used to help model these physical interactions [5]. These constraints are namely, the slip complementarity constraint:

$$\lambda_{i,fy}\,\beta_{i,f} = 0\,, \qquad \lambda_{i,fy} \geq 0, \qquad \beta_{i,f} \geq 0 \tag{12}$$

where $\lambda_{i,fy}$ represents the vertical GRF on either foot f, and $\beta_{i,f}$ is the horizontal foot velocity at the i^{th} node. This constraint ensures that when the foot is on the ground, the foot velocity is zero. The contact complementarity constraint:

$$\lambda_{i,fy}\,\gamma_{i,f} = 0\,, \qquad \lambda_{i,fy} \geq 0, \qquad \gamma_{i,f} \geq 0 \tag{13}$$

where $\gamma_{i,f}$ is the foot height at the i^{th} node. This constraint ensures that the GRF on foot is only applied when that foot is on the ground. The last two complementarities model the piston end stops the first is the piston complementarity constraint:

$$u_{i,jk}\left(1 - u_{i,jk}\right) = 0, \qquad 0 \leq u_{i,jk} \leq 1 \tag{14}$$

where $u_{i,jk}$ represents a piston actuation mode j (retract or extend) for a piston k at the i^{th} node. This constraint mimics the binary solenoid valve's switching behaviour [1]. The second one is the piston rebound complementarity:

$$\delta_{i,k}\,\zeta_{i,k} = 0\,, \qquad \delta_{i,k} \geq 0\,, \qquad \zeta_{i,k} \geq 0 \tag{15}$$

where $\delta_{i,k}$ denotes the distance of the piston rod at an end stop associated with the piston k at the i^{th} node. While $\zeta_{i,k}$ is the rebound force generated when the piston rod reaches that end stop.

2.4 Contact order

The contact order used follows a biologically inspired sequence derived from observations of the cheetah and greyhound gait patterns by Hudson [6]. Hudson reported the stance period of each limb of both animals during strides at various speeds. To fit the planar quadruped model, the total contact time (t_{period}) for the forelimbs were matched with the front leg of our model and total contact time of the hindlimbs were matched with the back leg.

The steady state bounding motion of the cheetah can be broken down into four distinct phases: flight, touchdown, gather, and lift-off [7]. During certain states, the GRFs are set to

zero, as seen in Table 1. The use of these GRF constraint technique is found in contact-implicit optimisation [8]. In a simplified planar model, after a full bound, all body states (except the horizontal position) of cheetah return to their original values [9]. Therefore, this analysis will take the apex transition ($\dot{y}_b = 0$) as the Poincaré section.

Table 1. Steady state bound phases.

Phase	Time	Constraint
Flight	$t_1 = 0.5 \times t_{period}$	$\begin{bmatrix}\lambda_{fx}\\ \lambda_{fy}\\ \lambda_{bx}\\ \lambda_{by}\end{bmatrix} = \mathbf{0}$
Touchdown	$t_2 = 0.14 \times t_{period}$	$\begin{bmatrix}\lambda_{bx}\\ \lambda_{by}\end{bmatrix} = \mathbf{0}$
Gather	$t_3 = 0.24 \times t_{period}$	$\begin{bmatrix}\lambda_{fx}\\ \lambda_{fy}\\ \lambda_{bx}\\ \lambda_{by}\end{bmatrix} = \mathbf{0}$
Lift-Off	$t_4 = 0.12 \times t_{period}$	$\begin{bmatrix}\lambda_{fx}\\ \lambda_{fy}\end{bmatrix} = \mathbf{0}$

2.5 Cost function

Two processes are used to find an optimal solution. The first optimisation uses the cost function $\boldsymbol{J_1}$ to ensure that a feasible solution is found. Once a feasible solution is found, the second optimisation uses cost function $\boldsymbol{J_2}$ to minimise the total time of the bound ($\sum_{i=1}^{N} \Delta \boldsymbol{t}_i$) and maximise the stride length (x_N), ultimately maximising the velocity.

$$\boldsymbol{J_1} = \rho_1 \boldsymbol{CP} + \boldsymbol{PP} \tag{16}$$

$$\boldsymbol{J_2} = \sum_{i=1}^{N} (\Delta \boldsymbol{t}_i - x_N) \tag{17}$$

$$\boldsymbol{CP} = \sum_i (\lambda_{i,fy}\, \beta_{i,f} + \lambda_{i,fy}\, \gamma_{i,f} + \delta_{i,k}\, \zeta_{i,k}) \tag{18}$$

$$\boldsymbol{PP} = \sum_i u_{i,jk} \left(1 - u_{i,jk}\right) \tag{19}$$

Where the complementarity penalty, $\boldsymbol{CP}$, is the sum of the slip and contact complementarity constraints, and the piston penalty, $\boldsymbol{PP}$, is the sum of the piston complementarity constraint. The weight $\rho_1 = 1 \times 10^5$ is used to prioritise penalty minimisation.

2.6 Solver

The problem was formulated using the second-order Radau contact-implicit direct collocation method described in [10]. It was written in a python library Pyomo [11] and uses solvers IPOPT [12] and MA86 [13] for the optimisation.

3 Results

3.1 Cost function results

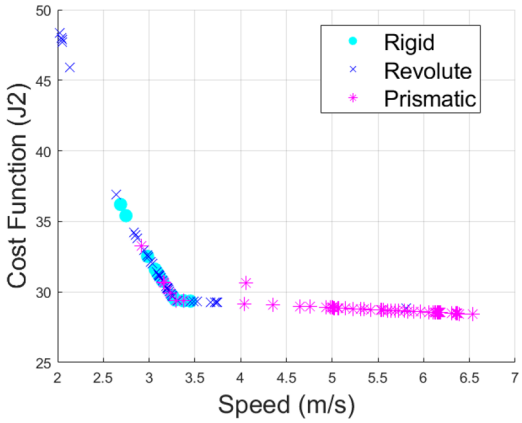

Fig. 4. Comparisons of cost function (J_2) and speed for different spine designs.

Figure 4 shows that, on average, the prismatic spine design was able to minimize and solve for a lower cost function compared to the other spine morphologies. This figure also suggests that the prismatic spine enabled higher steady-state bounding speed. The rigid and revolute spine designs were comparable to each, with some revolute solves yielding a higher steady-state bounding speed.

3.2 Spine trajectory results

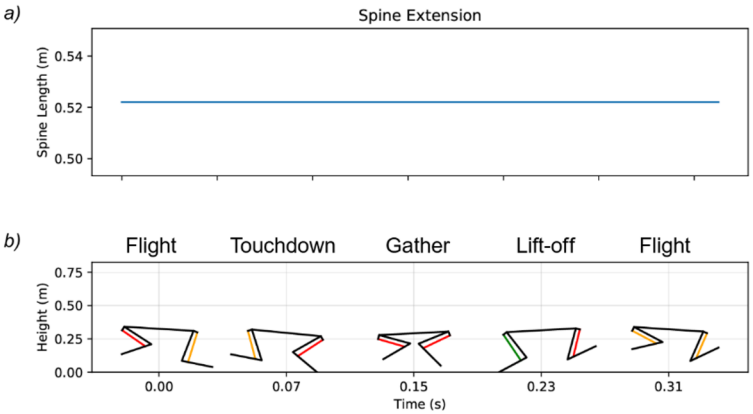

Fig. 5. Rigid spine results: (a) spine extension and (b) bounding phase with foot contacts.

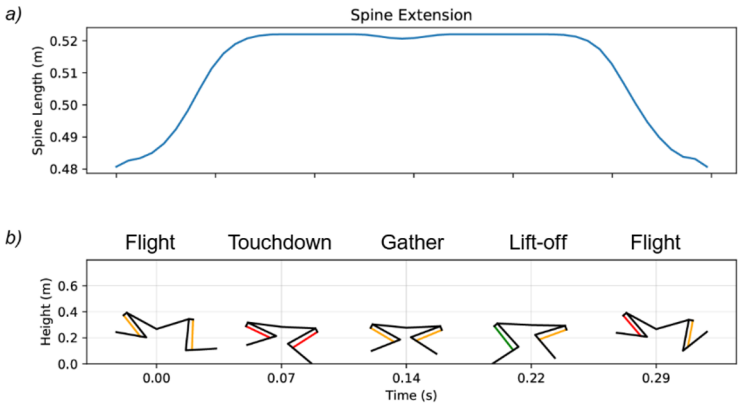

Fig. 6. Revolute spine results: (a) spine extension and (b) bounding phase with foot contacts.

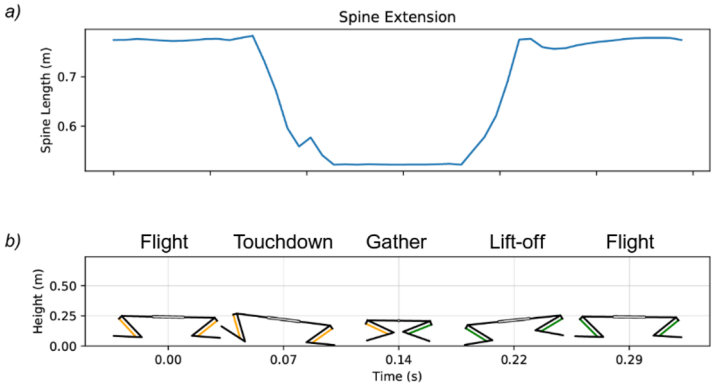

Fig. 7. Prismatic spine results: (a) spine extension and (b) bounding phase with foot contacts.

The figures above show the spine trajectory and bounding phases for a randomly selected optimal run from each spine morphology. In Figure 5.a, we can see that the spine length is constant throughout the bound as this is the rigid body spine design. The bounding phases in Figure 5.b, reflect the contact prescribed in the methodology. Figure 6 shows an optimal run for the revolute spine design, where we can see the maximum spine length occurs during the touchdown, gather and lift-off phases, while the spinal joint is contracting in the flight phase. Lastly, Figure 7 shows an optimal solve for the prismatic spine design. The spine profile shows that the prismatic spine design has its maximum length during the flight phase and its minimum length during the gather phase.

4 Discussion

The prismatic spine consistently outperformed both the rigid and revolute designs in steady-state bounding, which can be attributed to its ability to increase effective stride length. Since stride frequency was held constant across all three designs, a longer stride directly translated into greater forward velocity. This mechanical advantage highlights the added utility of the prismatic degree of freedom, echoing findings by Fisher [3], who similarly noted the performance benefits of spine extension for acceleration.

Beyond stride length, the prismatic spine demonstrated greater adaptability in spine kinematics. As shown in Figure 7, the spine extended during flight and contracted during the

gather phase, effectively behaving as a compliant spring-mass system. This behaviour suggests potential for energy storage and return, reducing locomotion cost and improving bounding efficiency. Such a mechanism aligns with observations in biological systems where elastic energy storage contributes to sustained high-speed locomotion [2,6]. However, experimental validation is needed to quantify these energetic benefits in hardware and therefore future work should look into an energy analysis the spine.

The revolute spine trajectory shows an inverted response to the prismatic spine profile. This inversion also contrasts what is seen in nature, where the spine of a cheetah contracts during the gather phase and extend during the flight phase. The actuator bounds are most likely what limited the revolute spine design from producing a spine trajectory that mimics that of the cheetah. Future work will look at the selection of the actuator limits and whether the revolute spine design for Kemba can produce the desired spine profile.

Finally, several limitations should be noted. The optimisation framework was restricted to steady-state bounding, with stride frequency fixed across morphologies. Different frequencies or transient accelerations may yield additional insights into the role of spine morphology. Furthermore, while actuator constraints were included in the model, real-world hardware will introduce inaccuracies such as compliance, backlash, and control delays. Therefore, these optimisations may work in simulation, but they will not directly translate on a real robot.

5 Conclusion

This study investigated the effect of spine morphology on steady-state bounding performance in quadruped robots. By modelling rigid, revolute, and prismatic spine configurations within the constraints of Kemba's actuators, we showed that the prismatic spine consistently achieved higher velocities through increased stride length and improved alignment of ground reaction forces with the direction of travel.

When compared to the cheetah's natural spine profile, the prismatic design most closely replicates the extension–contraction dynamics observed during flight and gather phases [2]. In contrast, the revolute spine introduced angular flexibility but did not substantially improve performance relative to the rigid spine, as it lacked the translational motion necessary to extend the effective stride. This finding reinforces the importance of incorporating both translational and angular dynamics to approach the biomechanical efficiency seen in nature.

More broadly, our results highlight the performance gap between rigid and flexible spine designs. While rigid spines offer mechanical simplicity, they limit stride modulation and velocity potential. Flexible morphologies—particularly prismatic designs—introduce new opportunities for enhancing speed, efficiency, and potentially adaptability to uneven terrain.

Future work will focus on determining the optimal placement of the prismatic joint and developing actuators capable of reproducing the required force profiles. Experimental implementation on Kemba will provide the next step in validating the simulation results and quantifying the true energetic benefits of flexible spines.

6 Appendix

Table 2. Model parameters.

Parameter	Symbol	Value	Units
Body Mass	m_b	2.886	kg
Body Inertia	I_b	0.1236	$\mathrm{kg \cdot m^2}$
Body Length	l_b	0.522	m
Femur Mass	m_{femur}	0.335	kg
Femur Inertia	I_{femur}	0.003	$\mathrm{kg \cdot m^2}$
Femur Length	l_{femur}	0.242	m
Tibia Mass	m_{tibia}	0.159	kg
Tibia Inertia	I_{tibia}	0.0013	$\mathrm{kg \cdot m^2}$
Tibia Length	l_{tibia}	0.234	m
Femur Piston Offset	$l_{p,femur}$	0.031	m
Tibia Piston Offset	$l_{p,tibia}$	0.048	m
Piston Body Mass	$m_{p,body}$	0.23	kg
Piston Body Inertia	$I_{p,body}$	0.0008	$\mathrm{kg \cdot m^2}$
Piston Rod Mass	$m_{p,rod}$	0.071	kg
Piston Rod Inertia	$I_{p,rod}$	0.0002	$\mathrm{kg \cdot m^2}$
Rotor Inertia	I_{rotor}	0.0001	$\mathrm{kg \cdot m^2}$
Actuator Limits			
Max Piston Length	$l_{p,max}$	0.07	m
Max Motor Torque	τ_{max}	25	$\mathrm{N \cdot m}$
Max Motor Angular Velocity	$\dot{\theta}_{max}$	30	rad/s
Rigid and Prismatic Spine			
Body Mass	m_{b1}, m_{b2}	1.443	kg
Body Inertia	I_{b1}, I_{b2}	0.0303	$\mathrm{kg \cdot m^2}$
Body Length	l_{b1}, l_{b2}	0.261	m

Author S. Abiodun wants to thank MathWorks for the funding they provided for this research.

References

1. C. Mailer, S. Shield, R. Govender, A. Patel, Getting Air: Modelling and Control of a Hybrid Pneumatic-Electric Legged Robot. IEEE Int. Conf. Robot. Autom. (ICRA) **2023**, 9973, 9979 (2023). https://doi.org/10.1109/ICRA48891.2023.10160737
2. M. Hildebrand, Motions of the running cheetah and horse. J. Mammal. **40**, 481 (1959). https://doi.org/10.2307/1376265
3. C. Fisher, S. Shield, A. Patel, The effect of spine morphology on rapid acceleration in quadruped robots. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS) **2017**, 2121, 2127 (2017). https://doi.org/10.1109/IROS.2017.8206028
4. P. Eckert, A. Spröwitz, H. Witte, A. J. Ijspeert, Comparing the effect of different spine and leg designs for a small bounding quadruped robot. IEEE Int. Conf. Robot. Autom. (ICRA) **2015**, 3128, 3133 (2015). https://doi.org/10.1109/ICRA.2015.7139629

5. M. Kelly, An introduction to trajectory optimization: How to do it, what it is, and why it works. SIAM Rev. **59**, 849 (2017). https://doi.org/10.1137/16m1062569
6. P. E. Hudson, S. A. Corr, A. M. Wilson, High speed galloping in the cheetah (Acinonyx jubatus) and the racing greyhound (Canis familiaris): Spatio-temporal and kinetic characteristics. J. Exp. Biol. **215**, 2425 (2012). https://doi.org/10.1242/jeb.066720
7. T. Kamimura, Y. Ambe, S. Aoi, F. Matsuno, Body flexibility effects on foot loading based on quadruped bounding models. Artif. Life Robot. **20**, 270 (2015). https://doi.org/10.1007/s10015-015-0223-z
8. A. Patel, S. L. Shield, S. Kazi, A. M. Johnson, L. T. Biegler, Contact-implicit trajectory optimization using orthogonal collocation. IEEE Robot. Autom. Lett. **4**, 2242 (2019). https://doi.org/10.1109/LRA.2019.2900840
9. Z. Gan, Z. Jiao, C. D. Remy, On the dynamic similarity between bipeds and quadrupeds: A case study on bounding. IEEE Robot. Autom. Lett. **3**, 3614 (2018). https://doi.org/10.1109/LRA.2018.2854923
10. S. Shield, A contact-implicit direct trajectory optimization scheme for the study of legged maneuverability. Ph.D. thesis, University of Cape Town, Faculty of Engineering and the Built Environment (2022)
11. W. E. Hart, C. D. Laird, J.-P. Watson, D. L. Woodruff, G. A. Hackebeil, B. L. Nicholson, J. D. Siirola, Pyomo-optimization modeling in python (Springer, Berlin, 2017)
12. A. Wächter, L. T. Biegler, On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Math. Program. 106, 25 (2006)
13. T. Rees, HSL. A collection of Fortran codes for large scale scientific computation (2022)
14. S. Bhattacharya, A. Singla, D. Dholakiya, S. Bhatnagar, B. Amrutur, A. Ghosal, S. Kolathaya, Learning active spine behaviors for dynamic and efficient locomotion in quadruped robots. IEEE Int. Conf. Robot. Human Interact. Commun. (RO-MAN) **2019**, 1, 6 (2019). https://doi.org/10.1109/RO-MAN46459.2019.8956332