Integrative and Comparative Biology

Integrative and Comparative Biology, volume 61, number 2, pp. 464–477
doi:10.1093/icb/icab083

Society for Integrative and Comparative Biology

SYMPOSIUM

Dynamic Modeling, Analysis, and Design Synthesis of a Reduced Complexity Quadruped with a Serpentine Robotic Tail

Yujiong Liu and Pinhas Ben-Tzvi [1]

Robotics and Mechatronics Lab, Department of Mechanical Engineering, Virginia Tech, Blacksburg, VA 24060, USA

From the symposium "An evolutionary tail: Evo-Devo, structure, and function of post-anal appendages" presented at the virtual annual meeting of the Society for Integrative and Comparative Biology, January 3–7, 2021.

[1]E-mail: bentzvi@vt.edu

Synopsis Serpentine tail structures are widely observed in the animal kingdom and are thought to help animals to handle various motion tasks. Developing serpentine robotic tails and using them on legged robots has been an attractive idea for robotics. This article presents the theoretical analysis for such a robotic system that consists of a reduced complexity quadruped and a serpentine robotic tail. Dynamic model and motion controller are formulated first. Simulations are then conducted to analyze the tail's performance on the airborne righting and maneuvering tasks of the quadruped. Using the established simulation environment, systematic analyses on critical design parameters, namely, the tail mounting point, tail length, torso center of mass (COM) location, tail–torso mass ratio, and the power consumption distribution, are performed. The results show that the tail length and the mass ratio influence the maneuvering angle the most while the COM location affects the landing stability the most. Based on these design guidelines, for the current robot design, the optimal tail parameters are determined as a length of two times as long as the torso length and a weight of 0.09 times as heavy as the torso weight.

Introduction

Tails are frequently used by animals to work alongside or in place of their legs to manipulate, propel, maneuver, and/or stabilize (Hickman 1979). For example, monkeys are observed to use their tails to grasp on branches and to balance their bodies during walking (Young et al. 2015); kangaroos are found to use their tails as additional limbs to propel their locomotion (O'Connor et al. 2014; Dawson et al. 2015); and cheetahs (Briggs et al. 2012) and kangaroo rats (Freymiller et al. 2019) are thought to use their tails to help maneuvering. Attracted by these amazing functionalities, roboticists consider augmenting legged robots with similar devices, such as the research conducted by Zeglin (1991), Jusufi et al. (2010), Libby et al. (2012), Chang-Siu et al. (2013), Kohut et al. (2013), Casarez and Fearing (2018), Patel and Boje (2015), De and Koditschek (2015), Libby et al. (2016), etc. As the first step, these researches chose the single-link pendulum as the abstraction of the animal tail and used it as an inertial adjustment device. The results showed the effectiveness of using robotic tails in mobile robot locomotion, such as helping the robot adjust its airborne orientation (Jusufi et al. 2010; Libby et al. 2012; Chang-Siu et al. 2013) or change its acceleration performance (Patel and Boje 2015).

However, most animals in nature evolved or preserved a multi-link serpentine tail structure, such as the cheetah tail in Fig. 1, where the tail could be better described as two circular arcs instead of one straight pendulum. Figure 1 also shows how the cheetah can use its tail to assist locomotion. That is, the cheetah lifts its rear legs and swings the tail counterclockwise simultaneously, to help the rear legs land on the left side of its torso. In contrast with the single-link structure, the serpentine tail can behave like a manipulator and thus can carry out multiple functions, such as supporting the body as another leg (kangaroos and jerboas), adjusting the COM, grasping branches (monkeys and chameleons), righting the body in the air, rejecting

Advance Access publication May 17, 2021

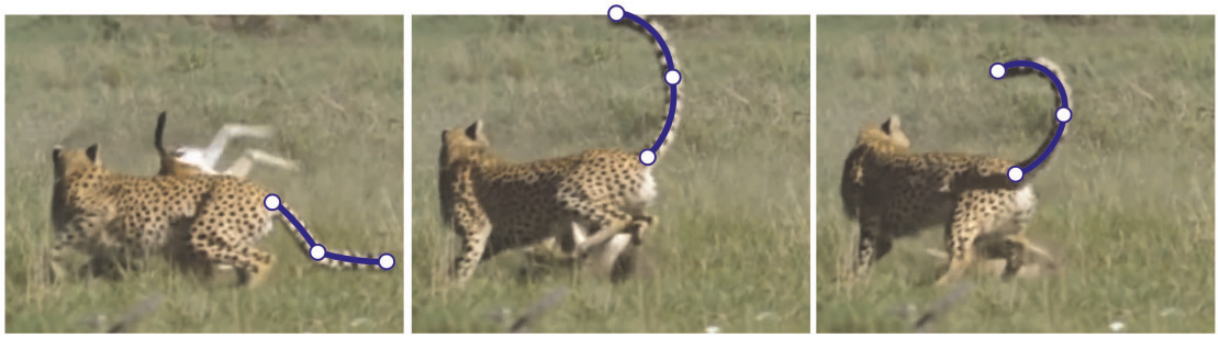

Fig. 1. The serpentine robotic tail idea is inspired from the animal kingdom.

disturbances, and assisting the ground contact behaviors. Therefore, the serpentine robotic tail idea was proposed, and several prototypes were built (Santiago et al. 2016; Rone et al. 2018; Liu et al. 2019; Saab et al. 2018a, 2019; Simon et al. 2018; Nabeshima et al. 2019; Liu and Ben-Tzvi 2020b; Wang et al. 2020). The preliminary results (Saab et al. 2018b; Rone et al. 2019; Liu and Ben-Tzvi 2021a) partially validated the above arguments but also revealed the design challenges in such systems.

Inspired by animal locomotion, one natural idea for the serpentine robotic tail research is to add it to a legged robot and investigate its usefulness on locomotion. However, a dexterous serpentine tail usually means more degrees of freedom (DOF), requiring more actuators, and thus increasing system weight. Adding such a heavy system on a general-purpose legged robot (which also requires many actuators) may severely hinder its dynamic performance, which is critical for the success of tail applications. Therefore, as the first step, a compromise is proposed that the legged robot complexity (in terms of DOF) could be reduced to compensate for the increased complexity in the tail system. Similarly, the legged robot weight is reduced and therefore allows adding a heavy tail without compromising the dynamic performance at the same time. The tradeoff is that the legged robot is no longer able to balance by itself, nor can it achieve various gaits, due to the decreased DOFs in the leg. However, since the dexterous tail introduces more control inputs, these shortcomings may be conquered by letting the tail system carry out the balancing task. This idea results in a new locomotion paradigm for the legged robots, that is, using reduced complexity legged robots (with fewer DOF in each leg) and a dexterous tail. The legs are optimized for a specific gait and are only responsible for propulsion while the tail takes the burden of balancing the body. One quadruped robot that realizes this proposed locomotion paradigm is illustrated in Fig. 2, where this robot consists of four single DOF legs and one serpentine tail that has one rolling DOF and two planar bending DOFs. Since the leg is designed to be as light as possible, its inertial loading is neglected in the dynamic model (the detailed justifications are discussed in the "Robot dynamic model" section).

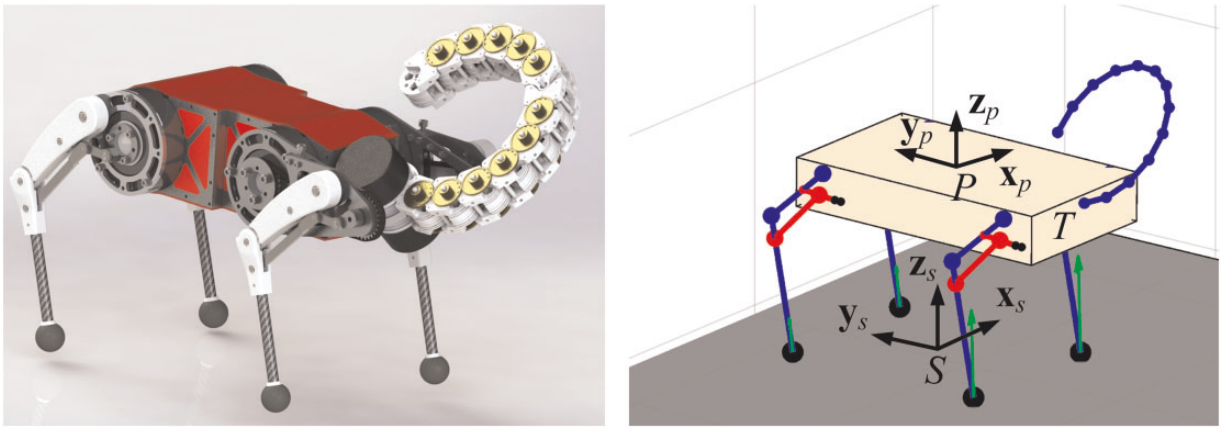

Fig. 2. Left: the initial mechanical design of a RCQ robot with a serpentine robotic tail realizing the proposed locomotion paradigm. Right: the abstract model used in this article where the green arrows on the feet indicate the GRF.

This article aims to analyze this robotic system through dynamic simulations, including building the dynamic model, analyzing the robot's dynamic behaviors, and determining the critical design parameters based on the dynamical analyses. The results are expected to evaluate the dynamic effects of the serpentine tail structure on the agile behaviors of the legged robot and formulate the design guidelines and determine the critical design parameters for the upcoming hardware development.

With reference to the existing literature as well as the authors' previous work (Saab et al. 2018b; Rone et al. 2019; Liu and Ben-Tzvi 2021a), the contributions of this work are summarized as follows. First, this is the first work that established a detailed dynamic model for a reduced complexity quadruped (RCQ) robot with single DOF legs and a serpentine robotic tail. Second, a novel tail controller using the partial feedback linearization (PFL) technique is formulated for airborne righting tasks. Third, for the specific tailed quadruped robot, systematic dynamic analyses were conducted to find the critical design parameters.

Materials and methods

Robot description

This section presents the necessary background information about the tailed quadruped robot in Fig. 2.

Reduced complexity quadruped

As shown in Fig. 2, the tailed quadruped robot consists of four reduced complexity legs and one serpentine tail. The coordinate systems that describe the RCQ are defined as follows. The inertial frame denoted as $\Sigma S := (S, \mathbf{x}_s, \mathbf{y}_s, \mathbf{z}_s)$ whose orientation can be found in Fig. 2. The body-fixed frame $\Sigma P := (P, \mathbf{x}_p, \mathbf{y}_p, \mathbf{z}_p)$ which is attached to the torso with its origin located at the torso COM, and the three basis vectors aligned with the three principal axes of the torso's moments of inertia.

Each reduced complexity leg has only one DOF and its mechanism is shown in Fig. 3. The leg mechanism consists of two four-bar mechanism loops, which are known as the Jansen's linkage (Nansai et al. 2015). Properly adjusting the linkage lengths as well as the joint positions results in different foot trajectories. The current design for the RCQ is optimized to mimic a trotting gait (Liu and Ben-Tzvi 2020a). Therefore, the kinematics of the leg mechanism is governed by two vector equations

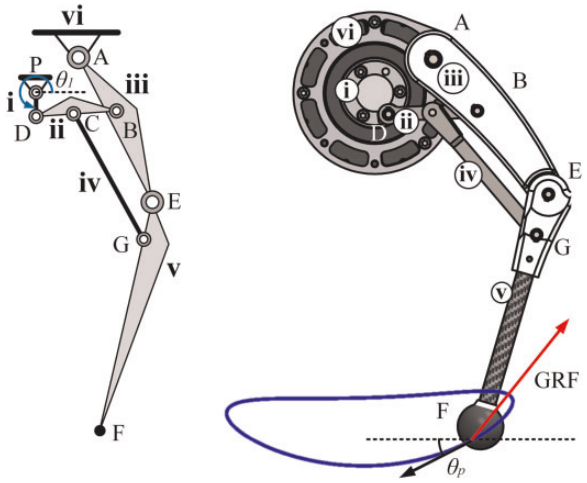

Fig. 3. Left: kinematic diagram of the single DOF leg mechanism. Right: mechanical design and foot trajectory of the single DOF robotic leg. Same label indicates the same link/joint.

$$\vec{PD} + \vec{DB} = \vec{PA} + \vec{AB} \quad (1)$$

$$\vec{CG} + \vec{GE} = \vec{CB} + \vec{BE}\,, \quad (2)$$

where the arrow above the letters indicates a vector. The foot position F is then computed using Equation (3). The Jacobian of F, $\mathbf{J}_f$, can be computed by directly differentiating Equation (3).

$$\vec{AF} = \vec{AE} + \frac{||\vec{EF}||}{||\vec{EG}||}\vec{EG}\,. \quad (3)$$

Serpentine robotic tail

The serpentine tail system is a two-segment, 13-link robotic tail named R3RT (Saab et al. 2019), which stands for the roll-revolute-revolute robotic tail. As shown in Fig. 4, the base link of the tail mounts on the torso at point T. The remaining 12 links are then serially connected to each other from the base link. The first six links form the first segment, and the remaining six links form the second segment. Inside each segment, since the adjacent links are coupled by gears, all the joints rotate at the same angle. Therefore, the tail mechanism has three DOFs in total, namely, one overall rolling DOF (defined by the base link rolling angle α with respect to the torso), one planar bending DOF for the first segment (defined by the rotation angle β_1 of any joint in the first segment), and one planar bending DOF for the second segment (defined by the rotation angle β_2 for any joint in the second segment). For convenience, body-fixed frames are defined for each link. That is, the base link frame $\sum T := (T, \mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0)$ is defined at point T, with $\mathbf{y}_0$ coinciding $\mathbf{y}_p$ and $\mathbf{x}_0$ coinciding

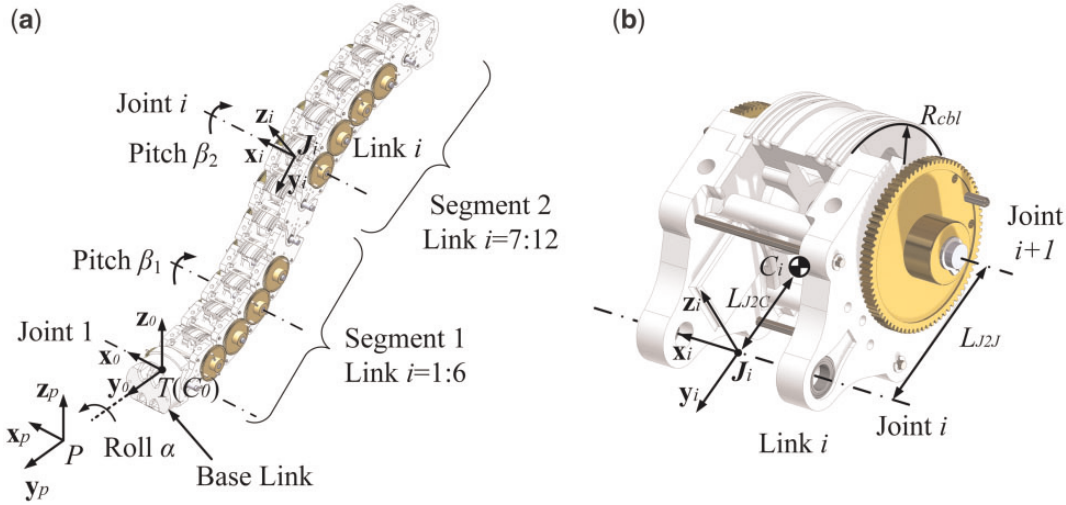

Fig. 4. The kinematic configurations for (a) the R3RT and (b) the *i*-th link.

the axis of joint 1. The *i*-th link frame $\sum J_i := (J_i, \mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i)$ is defined at the joint *i* axis, with $\mathbf{x}_i$ coinciding joint *i* axis and $\mathbf{y}_i$ pointing along with link *i*. More modeling details can be found in Liu and Ben-Tzvi (2021a).

The kinematics for the R3RT is given in the recursive form, as shown by Equations (4)–(6), where $\mathbf{R}_x$ and $\mathbf{R}_y$ are the principal rotation matrices with respect to *x* and *y* axis. ${}^S\mathbf{R}_P$ is the torso orientation and ${}^S\mathbf{R}_i$ is the orientation of the *i*-th link. $\mathbf{p}_{i,\text{com}}$, $\mathbf{p}_{i,jnt}$, $\mathbf{p}_{i,j2c}$, and $\mathbf{p}_{i,j2j}$ denote the position of C_i, J_i, the vector from J_i to C_i, and the vector from J_i to J_{i+1}, respectively. Note that the base link COM (C_0) position $\mathbf{p}_{0,\text{com}}$ is assumed to be at point *T* and thus $\mathbf{p}_{0,\text{com}} = \mathbf{p}_t$. L_{J2C} and L_{J2J} are defined in Fig. 4(b), as the distance from J_i to C_i and from J_i to J_{i+1}, respectively,

$$ {}^S\mathbf{R}_i = \begin{cases} {}^S\mathbf{R}_P\ \mathbf{R}_y(\alpha), & i = 0 \\ {}^S\mathbf{R}_P\ \mathbf{R}_y(\alpha)\mathbf{R}_x(i\beta_1), & 1 \le i \le 6 \\ {}^S\mathbf{R}_P\ \mathbf{R}_y(\alpha)\mathbf{R}_x(6\beta_1 + (i-6)\beta_2), & 7 \le i \le 12 \end{cases} \tag{4} $$

$$ \mathbf{p}_{i,\text{com}} = \mathbf{p}_{i,jnt} + \mathbf{p}_{i,j2c} \tag{5} $$

$$ \mathbf{p}_{i,jnt} = \begin{cases} \mathbf{p}_t, & i = 1 \\ \mathbf{p}_{i-1,jnt} + \mathbf{p}_{i-1,j2j}, & i > 1 \end{cases}; \begin{cases} \mathbf{p}_{i,j2c} = -L_{J2C}\mathbf{y}_i \\ \mathbf{p}_{i,j2j} = -L_{J2J}\mathbf{y}_i \end{cases}. \tag{6} $$

Differentiating Equations (4)–(6) and factoring out the generalized coordinates yields the Jacobian of each link, as given in Equations (7)–(9), where $\mathbf{u}_{m,n}$ is the *m* dimension unit column vector with 1 on the *n*-th entry. Similar to the position relationship, the Jacobian of the base link COM is $\mathbf{J}_{0,\text{com}} = \mathbf{J}_t$.

$$ \mathbf{J}_{i,\omega} = \begin{cases} [\mathbf{0}_{3\times3} \quad \mathbf{I}_{3\times3} \quad \mathbf{0}_{3\times3}] + \mathbf{y}_0\mathbf{u}_{9,7}^T, & i = 0 \\ \mathbf{J}_{i-1,\omega} + \mathbf{x}_0\mathbf{u}_{9,8}^T, & 1 \le i \le 6 \\ \mathbf{J}_{i-1,\omega} + \mathbf{x}_0\mathbf{u}_{9,9}^T, & 7 \le i \le 12 \end{cases}. \tag{7} $$

$$ \mathbf{J}_{i,\text{com}} = \mathbf{J}_{i,\text{jnt}} + \mathbf{J}_{i,j2c}. \tag{8} $$

$$ \mathbf{J}_{i,jnt} = \begin{cases} \mathbf{J}_t, & i = 1 \\ \mathbf{J}_{i-1,jnt} + \mathbf{J}_{i-1,j2j}, & i > 1 \end{cases}; \begin{cases} \mathbf{J}_{i,j2c} = -\tilde{\mathbf{p}}_{i,j2c}\mathbf{J}_{i,\omega} \\ \mathbf{J}_{i,j2j} = -\tilde{\mathbf{p}}_{i,j2j}\mathbf{J}_{i,\omega} \end{cases}, \tag{9} $$

To map the actuator force into the generalized space, the actuation Jacobian $\mathbf{J}_{ta}$ is also required. Let τ_α be the actuation torque for the rolling joint, $f_{\beta1}$ and $f_{\beta2}$ be the driven cable tensions for the first and second segments, respectively. The $\mathbf{J}_{ta}$ could be calculated as

$$ \mathbf{J}_{ta}^T\boldsymbol{\tau}_{ta} = \mathbf{u}_{9,7}\tau_\alpha + 6R_{cbl}\mathbf{u}_{9,8}f_{\beta1} + 6R_{cbl}\mathbf{u}_{9,9}f_{\beta2}, \tag{10} $$

where R_{cbl} is the link profile radius (referring to Fig. 4(b)).

It is worth noting that the R3RT cannot go in the direction perpendicular to the bending plane. This results in only two independent torso orientations being able to be adjusted simultaneously even though the tail has three DOFs.

System modeling

This section presents the dynamic model for the tailed quadruped robot.

Robot dynamic model

The equation of motion is established using the Inertia Matrix Method presented in Featherstone's (2014) book. That is, the system dynamics is given as

$$\mathbf{H}\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{J}_{ta}^T \boldsymbol{\tau}_{ta} + \mathbf{J}_f^T \mathbf{f}, \tag{11}$$

where $\boldsymbol{\tau}_{ta}$ is the tail actuation force, $\mathbf{f}$ is the ground reaction force (GRF), $\mathbf{H}$ is the system inertia matrix, and $\mathbf{C}$ is the joint-space bias force (including Coriolis force, centrifugal force, and gravity). The generalized coordinates $\mathbf{q}$ are selected as $[\mathbf{p}_p^T \boldsymbol{\phi}^T \alpha \beta_1 \beta_2]^T$, where $\mathbf{p}_p$ is the position vector of point P and $\boldsymbol{\phi}$ is the torso orientation (measured in the inertial frame ΣS). $\mathbf{H}$ and $\mathbf{C}$ could be obtained as

$$\begin{aligned}\mathbf{H} = {} & m_b \mathbf{J}_{b,x}^T \mathbf{J}_{b,x} + \mathbf{J}_{b,\omega}^T \mathbf{I}_b \mathbf{J}_{b,\omega} \\ & + \sum_{i=0}^{12} (m_{i,r3} \mathbf{J}_{i,com}^T \mathbf{J}_{i,com} + \mathbf{J}_{i,\omega}^T \mathbf{I}_{i,r3} \mathbf{J}_{i,\omega}).\end{aligned} \tag{12}$$

$$\mathbf{C} = \mathrm{ID}(\mathrm{model}, \mathbf{q}, \dot{\mathbf{q}}, \mathbf{0}), \tag{13}$$

where m_b, $\mathbf{I}_b$, $m_{i,r3}$, and $\mathbf{I}_{i,r3}$ are the torso mass, torso moment of inertia (in the inertial frame), R3RT link mass, and R3RT link moment of inertia (in the inertial frame), respectively. $\mathbf{J}_{b,x}$ and $\mathbf{J}_{b,\omega}$ are the Jacobians for $\mathbf{p}_p$ and $\boldsymbol{\phi}$, respectively. The "ID" stands for the inverse dynamics function defined in the inverse dynamics, as shown in the equation

$$\mathrm{ID}(\mathrm{model}, \mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) = \mathbf{J}_{ta}^T \boldsymbol{\tau}_{ta} + \mathbf{J}_f^T \mathbf{f}. \tag{14}$$

It is worth noting that due to the relatively small inertia the leg dynamics are neglected (light leg design makes the leg motion have a marginal influence on the overall quadruped motion, see discussion in Winkler et al. [2018] for more details). However, leg kinematics are still required for computing $\mathbf{f}$.

Ground contact model

The soft contact model (Azad and Featherstone 2014) is used to model the foot-ground interaction, where the collision is modeled as a nonlinear spring-damper system, as shown in the equations

$$\mathbf{f} = \mathbf{f}_n + \mathbf{f}_f = \|\mathbf{f}_n\| \mathbf{z}_s + \|\mathbf{f}_x\| \mathbf{x}_s + \|\mathbf{f}_y\| \mathbf{y}_s \tag{15}$$

$$\|\mathbf{f}_n\| = \max\{K_n z^{3/2} + D_n K_n z^{1/2} \dot{z},\ 0\}, \tag{16}$$

where $\mathbf{f}_n$ is the normal force and z is the penetration depth. K_n and D_n are the ground stiffness and collision damping coefficient, respectively. Therefore, the friction forces $\mathbf{f}_x$ and $\mathbf{f}_y$ are calculated as

$$\|\mathbf{f}_x\| = \begin{cases} \mu\|\mathbf{f}_n\|, & K_x x + D_x K_x \dot{x} > \mu\|\mathbf{f}_n\| \\ K_x x + D_x K_x \dot{x}, & \text{else} \\ -\mu\|\mathbf{f}_n\|, & K_x x + D_x K_x \dot{x} < -\mu\|\mathbf{f}_n\| \end{cases}, \tag{17}$$

where a linear spring-damper is used to model the static friction and the Coulomb friction model is used to compute the kinetic friction. $\|\mathbf{f}_y\|$ takes the same form as $\|\mathbf{f}_x\|$ except replacing x with y.

Controller architecture

To control the new tailed quadruped robot, its intrinsic locomotion paradigm requires that the controller should have two components: one simple leg controller to drive the reduced complexity leg mechanism and one tail controller that adjusts the robot orientation as needed. To achieve the overall locomotion goal, these two components need to be put under one coordination controller that synchronizes the leg motion and the tail motion. This forms the hierarchical controller structure, as shown in Fig. 5. The CPG (stands for central pattern generator) module generates the timing information and control objectives for the legs and the tail, based on the desired locomotion goal. The leg controller, which is the crank motion planning (CMP) module in Fig. 5, accepts the timing information from the CPG module and translates it into the actual crank trajectory. The tail controller accepts the control objectives from the CPG and generates the control effort accordingly. The control objectives are mainly the desired torso orientations $\boldsymbol{\phi}_d$ and angular velocity $\dot{\boldsymbol{\phi}}_d$. With the inputs from the legs and the tail, the quadruped interacts with its environment and achieves the locomotion goals. The whole system has various feedbacks (e.g., state and events) to help determine the controller inputs.

High-level locomotion control

The high-level controller refers to the CPG module, where it is designed as a finite state machine (FSM). The FSM is a directed graph that consists of finite states (node) and the transitions (edges) between states. The transitions are usually triggered by specific events in the state. For the tailed quadruped system, the states could be the different contact cases, such as when the quadruped is airborne or when the front feet are on the ground. The transitions could be triggered by a specific event, such as the touchdown event for a specific foot or when the torso reaches a specific pitch angle. Using the FSM framework, the locomotion patterns and the switches between patterns could be designed more clearly and

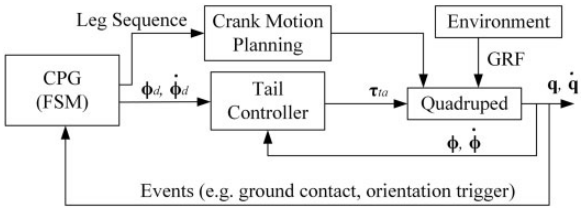

Fig. 5. Controller structure for the new-tailed quadruped robot.

efficiently. For example, referring to Fig. 6, one step jumping motion for the tailed quadruped could be designed as follows. The FSM consists of four states: stance, standup, righting, and landing, which corresponds to four feet, rear feet, no foot, and front feet on the ground, respectively. The transitions between states and the actions during each state are presented in the figure too. It is worth noting that the state definitions in Fig. 6 are pretty general for the proposed tailed quadruped robot. The same FSM could be used to design other locomotion patterns. For example, closing the loop between the landing state and the stance state yields a bounding gait and a pronking gait. Preliminary results on this kind of locomotion control could be found in Liu and Ben-Tzvi (2021b), which uses a pendulum tail instead of a serpentine tail.

Low-level tail control

The purpose of the tail controller is to generate the control efforts on the tail joints that drive the torso to the desired orientation. To achieve this goal, the existing approaches (using pendulum tails) mainly include momentum-based trajectory planning (Patel and Boje 2015) and momentum-guided nonlinear feedback control (Chang-Siu et al. 2013). Here, we introduce another control approach based on the PFL technique (Spong 1994) since the dynamic we are interested in is only the torso orientation ϕ, which is one part of the full state $\mathbf{q}$. This approach avoids the manual process of designing tail trajectories and thus is faster and more general, which is suitable for dynamics-based analysis and synthesis. The tradeoff is that it is difficult to incorporate advanced constraints (e.g., limiting tail motions so that it does not collide with ground or other parts of the robot) in the controller design thereby making the controller impractical for a hardware robot. However, since our purpose is to analyze the overall effects of different tail parameters (such as the tail length) on the quadruped locomotion and use the results as design guidance for the hardware development, this shortcoming is thought to be acceptable in comparison with its benefits.

To derive the PFL controller, the system output is constructed as

$$\mathbf{y} = \mathbf{q}_s - \mathbf{q}_d(t), \tag{18}$$

where $\mathbf{q}_s = \mathbf{S}\mathbf{q}$ are the partial states to be linearized and s indicates its dimension. $\mathbf{q}_d$ is the reference trajectory and $\mathbf{S} = \partial\mathbf{q}_s/\partial\mathbf{q}$ represents the selection matrix. Then, the output dynamics are constructed as a spring-damper system, which is known to be exponentially stable:

$$\ddot{\mathbf{y}} + \mathbf{K}_d\dot{\mathbf{y}} + \mathbf{K}_p\mathbf{y} = 0, \tag{19}$$

where $\mathbf{K}_d = K_d\mathbf{I}_{s\times s}$ and $\mathbf{K}_p = K_p\mathbf{I}_{s\times s}$ with $K_d, K_p > 0$. Assuming that the tail controller only acts when the quadruped is in the air, solving Equation (11) for $\ddot{\mathbf{q}}$ and using $\mathbf{q}_s = \mathbf{S}\mathbf{q}$ yields

$$\ddot{\mathbf{q}}_s = \mathbf{S}\mathbf{H}^{-1}\left(\mathbf{J}_{ta}^T\boldsymbol{\tau}_{ta} - \mathbf{C}\right). \tag{20}$$

Substituting Equation (20) into Equation (19) and solving for $\boldsymbol{\tau}_{ta}$, the tail controller is obtained as

$$\boldsymbol{\tau}_{ta} = \mathbf{X}^+(\mathbf{S}\mathbf{H}^{-1}\mathbf{C} + \ddot{\mathbf{q}}_d + \mathbf{K}_d(\dot{\mathbf{q}}_d - \dot{\mathbf{q}}_s) + \mathbf{K}_p(\mathbf{q}_d - \mathbf{q}_s)) \tag{21}$$

in which $\mathbf{X} = \mathbf{S}\mathbf{H}^{-1}\mathbf{J}_{ta}^T$ and $\mathbf{X}^+$ is the Moore–Penrose inverse of $\mathbf{X}$.

When the quadruped is stably standing on the ground, the tail controller in Equation (21) is no longer necessary. For these scenarios, the tail should simply go back to its natural position. That is, the tail controller is a pure damping system.

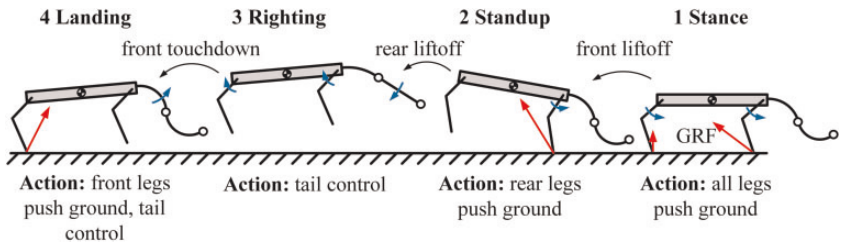

Fig. 6. Illustration of the motion sequence for one-step jumping.

Table 1. The initial RCQ-R3RT properties

Var.	Value	Var.	Value
g	9.8ms^{-2}	L_{J2J}	40 mm
$\|PA\|$	40 mm	L_{J2C}	32.7 mm
$\angle PA$	30°	R_{cbl}	14 mm
$\|PD\|$	15 mm	$m_{0,r3}$	1000g
$\|DC\|$	26 mm	${}^{0}\mathbf{I}_{0,r3}$	diag([2.77 2.29 1.07]) gm^2
$\|DB\|$	52.5 mm	$m_{\{1-12\},r3}$	41.7g
$\|AB\|$	40.15 mm	${}^{i}\mathbf{I}_{\{1-12\},r3}$	diag([0.014 0.01 0.009]) gm^2
$\|AE\|$	105 mm	Torso length (hip to hip)	300 mm
$\|EG\|$	25 mm	Torso width	200 mm
$\|EF\|$	151.5 mm	Torso COM Offset	150 mm
$\|CG\|$	95 mm	$\|PT\|$	190 mm
K_n	5E4 Nm^{-1}	m_b	6000 g
D_n	0.75	${}^{P}\mathbf{I}_b$	diag([45 20 65]) gm^2
K_x,K_y	3E4 Nm^{-1}	K_p	2500
D_x,D_y	0.01	K_d	100
μ	1	K_{d1},K_{d2}, K_{d3}	1

$$\boldsymbol{\tau}_{ta} = -[K_{d1}\dot{\alpha}\ K_{d2}\dot{\beta}_1\ K_{d3}\dot{\beta}_2]^T \tag{22}$$

where $K_{d1}, K_{d2}, K_{d3} > 0$ are the damping coefficients. Since the pure damping consumes energy, the established stability will not be violated.

Simulation setting

All numerical computations are carried out in Matlab. For numerical integration, the built-in function *ode45* was used with an absolute tolerance of 1E-6 and a relative tolerance of 1E-8. Note that the *ode45* uses a variable step scheme to control integration error. The actuator saturations are set as 30 Nm for the rolling joint torque and 200 N for the cable tensions. The joint motion ranges of the R3RT are set as $\beta_1, \beta_2 \leq 25°$. All the robot-related properties are listed in Table 1, where $|XY|$ means the length of a line segment XY. $\angle PA$ is the angle of PA away from the horizontal line in Fig. 3. The "torso COM offset" is measured from the torso COM to the line connecting the rear hips. The program is executed on a typical desktop with an Intel Core i7-7700 CPU (Central Processing Unit, 3.6 GHz) and one 32 GB random-access memory. Under such a computational environment, the meantime to evaluate one-time dynamics (including calculating the controller) is around 1.05 ms.

Results and discussion

Airborne righting

The first set of simulations is to evaluate the effectiveness of the closed-loop tail controller for the airborne righting task, for which the control objective is to adjust the torso orientation from an arbitrary gesture back to the horizontal gesture. Therefore, the partial states are selected as $\mathbf{q}_s = [\phi_x\ \phi_y]^T$ and the controller objective is $\mathbf{q}_d(\mathrm{t}) = [0\ 0]^T$. The quadruped is initialized in the air with $\mathbf{p}_p(0) = [0\ 0\ 0.4]^T$ meters and falling on the ground with a non-horizontal orientation. The tail initial condition is set to zero for all joints and all initial velocities are zero. Under these conditions, the following tests were conducted. First, a case study with an initial orientation of $\boldsymbol{\phi}(0) = [-0.2\ -0.2\ 0]^T$ radians is performed. The corresponding state response, the normal GRF, and the tail controller efforts are plotted in the top row of Fig. 7, where the "FR," "RL," "FL," and "RR" in the GRF subfigure stand for "front right leg," "rear left leg," and so on. The horizontal lines for β_1 and β_2 indicate that the tail joints reach their maximal range. The rapid change in the actuation force is due to the rapid change in the state during the ground contact events. In practice, this could be avoided by setting a low pass filter for the actuator outputs.

To further test the robustness of the controller in terms of model uncertainty (e.g., frictional losses), a gaussian white noise (generated by the Matlab function *wgn* using 1 W power, and then scaled to 10%

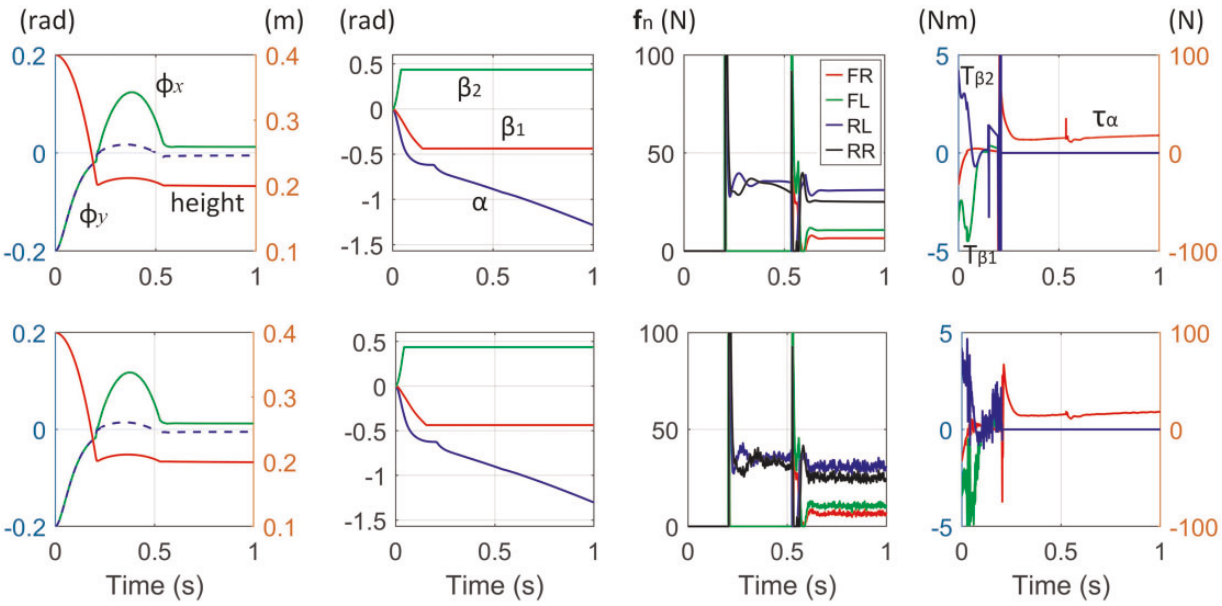

Fig. 7. Response plots for the airborne righting case study. The top row shows the deterministic results and the bottom row shows the stochastic results. The tail–torso length ratio is 1.6, the tail–torso mass ratio is 0.07, and the torso COM offset is 150 mm.

of the deterministic signal) is added to the system bias force $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ directly. The results are presented in the bottom row of Fig. 7. From the comparison, the controller is found to be able to absorb the uncertainty that appeared in the system dynamics and thus generate a similar simulation output. Since the added noise does not affect the numerical robustness significantly, the following simulations only use deterministic system dynamics. It is also worth noting that the variable step ordinary differential equation (ODE) solver that we used in this work is not able to handle the added noise directly since in this case, the system dynamics becomes a stochastic differential equation (SDE), which is significantly different from the ODE. To handle this problem, we generated the noise in advance and then approximated the noise signal using spline interpolation. Similarly, for each numerical integration, the noise is deterministic and continuous (among running, it is still stochastic). This technique allows us to efficiently simulate the stochastic signal using the variable step ODE solver.

The airborne righting case study is also simulated without using the tail controller ($\boldsymbol{\tau}_{ta} = \mathbf{0}$). The results are presented in Fig. 8, where the top row shows the snapshots that correspond to the airborne righting motion in Fig. 7, and the bottom row shows the snapshots for the case without using the tail controller. The comparison visually shows how the tail's reactive action helped the airborne righting of the quadruped, and the tail controller automatically generated the required motion for this task.

However, during the simulation, we found that for certain initial torso orientations the tail is not able to right the quadruped in time and thus fails the landing. This is thought of as the physical limitation of the tail system, which is closely related to its workspace configuration and the actuation saturation values. The former term affects the overall angles that the torso could be rotated (conservation of momentum) while the latter affects how fast the righting could be done. Note that we cannot increase the release height to alleviate the time constraint because the height is determined by the quadruped leg capability. To find out the physical limitation for the initial tail design, the torso initial orientations are incremented manually until the system fails. A failure is determined as the quadruped tipping over after landing. The results are plotted in Fig. 9, where the green circles represent the successful trials and the red crosses indicate the failed cases. The limitation region is roughly estimated by sketching the boundary of the green circles. It can be found that the limitation region is symmetric in roll angle but asymmetric in pitch angle. This is due to the rear part of the robot being heavier than the front part, that is, tipping over backward is naturally easier and thus harder to control. In addition, the limitation region in the roll direction is found to be much larger than that of the pitch direction. This is due to the unlimited tail roll rotation, α, mainly responsible for torso roll adjustment, and the limited tail bending, β_1 and β_2, mainly responsible for torso pitch adjustment. Figure 9 also presents the torso orientation trajectories $\phi_x(t)$ and $\phi_y(t)$ to visually show

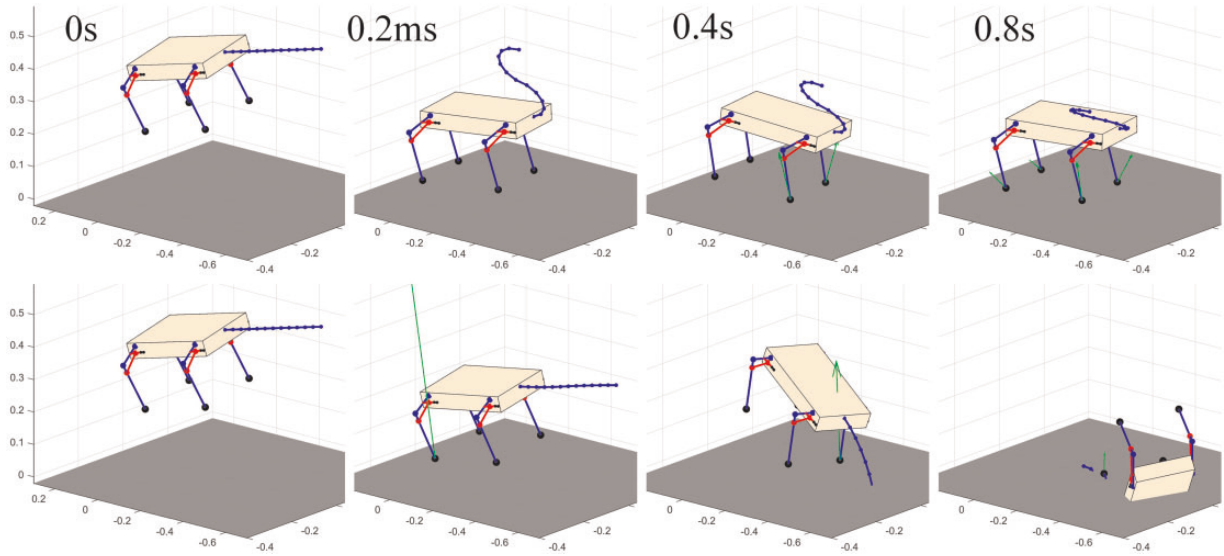

Fig. 8. Snapshots for the airborne righting case study with (top row) and without (bottom row) tail controller. The same column has the same time. The tail–torso length ratio is 1.6, the tail–torso mass ratio is 0.07, and the torso COM offset is 150 mm. A video for these motions is available at https://youtu.be/s6EubHGq-5c.

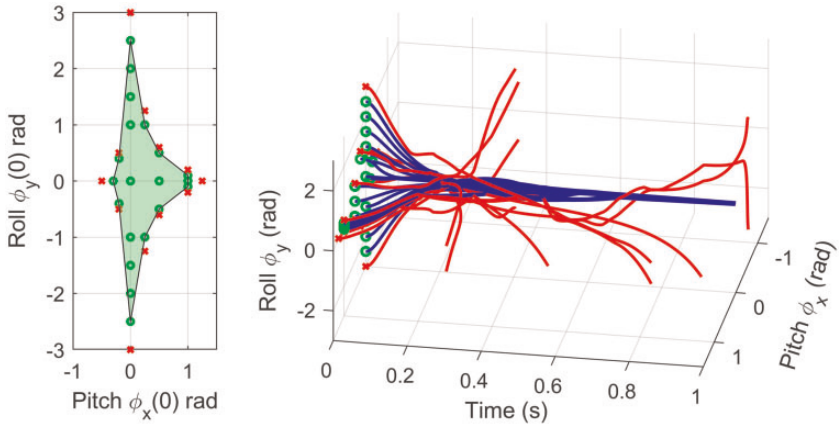

Fig. 9. Estimated limitation region (left) and the torso orientation trajectories $[\phi_x(t)\ \phi_y(t)]^T$ with different initial conditions (right). The tail–torso length ratio is 1.6, the tail–torso mass ratio is 0.07, and the torso COM offset is 150 mm.

the successful landings (blue trajectories) and the failed landings (red trajectories). Therefore, for design synthesis purpose, the general guidelines are that (1) the tail design should increase its actuation capability as well as the workspace and (2) the leg and torso design should make sure that the quadruped orientation will not go beyond its limitation region after lifting from the ground.

Maneuvering

The maneuvering motion combines the tail motion with the leg motion. That is, the robot is initialized on the ground and is lifted in to the air by the leg motion. When the robot is in the air, the tail is swung to change its heading direction. Since the tail can only adjust two torso orientations at the same time, the partial state is selected as $\mathbf{q}_s = [\phi_y\ \phi_z]^T$ and the controller objective is $\mathbf{q}_d(t) = [0\ \delta]^T$ where δ is the desired yaw maneuvering angle. However, this arrangement will leave the pitch motion uncontrolled, which may cause the landing to be unstable. Therefore, adjustment of the torso COM location is needed to make sure that the torso pitch motion evolves around the horizontal plane during the jumping. This observation brings up the design synthesis issue in the "Mass distribution" section to determine the optimal torso COM location. The initial condition for the tail roll joint (α) is also changed to 90°. The leg crank speeds are all set to be the same 40 rad/s, which generate a jumping height of around 100 mm. One case study was simulated and captured in Fig. 10. For the initial tail design

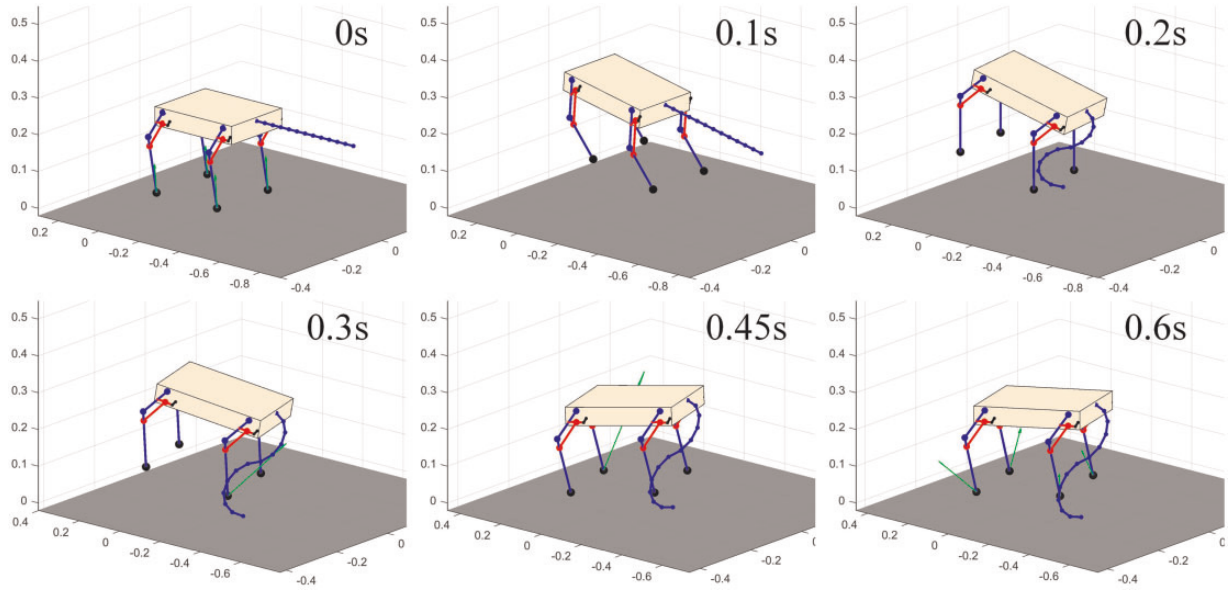

Fig. 10. Snapshots for the maneuvering case study. The tail–torso length ratio is 1.6, the tail–torso mass ratio is 0.07, and the torso COM offset is 240 mm. A video for this motion is available at https://youtu.be/s6EubHGq-5c.

with an adjusted torso COM offset of 240 mm, the robot maneuvered 20.88° for a desired δ of 30°. This inconsistency reflects the physical limit of the tail system, that is, the maximal maneuver angle (denoted as δ_{lim}) that the robot can achieve by swinging its tail. More systematic trials were conducted but are put into the next section for the design synthesis discussion.

Design analysis and synthesis

Using the developed simulation environment, this section generalizes the analyses from previous sections and determines the critical design considerations for the quadruped-tail system. Since the current robot is mainly designed for performing maneuvering tasks, the maneuvering motion is selected as the benchmark scenario. Note that although the R3RT is limited for adjusting only two torso orientations simultaneously (referring to the "Serpentine robotic tail" section for details), its dexterity is sufficient for design synthesis since, for maneuvering tasks, only two orientations (pitch and roll) are needed.

Tail parameters

This section aims to investigate the performances of different tail parameters, namely, the tail mounting point and the tail length. It is thought that mainly these two tail parameters affect the maneuver angle and the control effort. Therefore, the performance evaluation criteria are selected as the maximal maneuver angle δ_{lim} and the total control effort $\tau_c = \int \boldsymbol{\tau}_{ta}^T \boldsymbol{\tau}_{ta} \mathrm{d}t$. The tail mounting point results are presented in Fig. 11, where the green circle marks the initial design result, and the red cross marks the failed test (unsuccessful maneuver). As shown in the figures, tail mounting location is found to have a marginal influence on the overall performance. Note that the mounting location is also constrained by other factors, such as the tail workspace and the collisions with other parts of the robot. Therefore, considering the actual design feasibility, the initial tail mount point is deemed to be good.

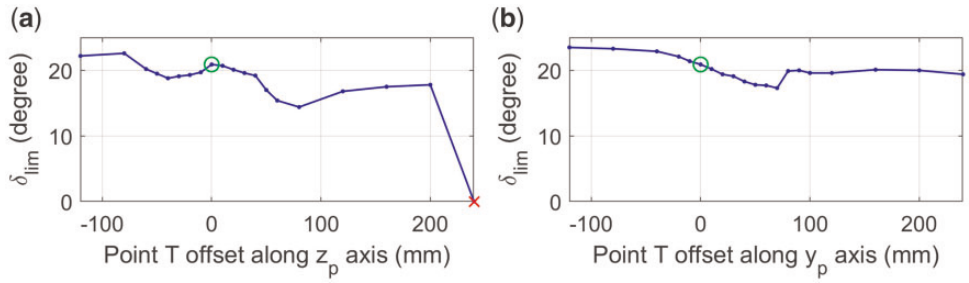

Fig. 11 Performance trends of adjusting the tail mounting point *T* from its initial design position (a) up and down and (b) forward and backward.

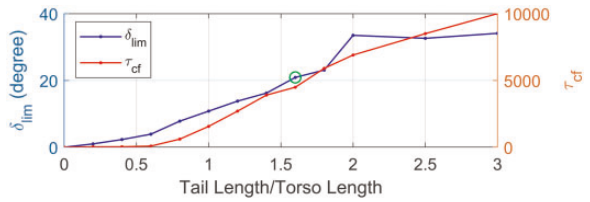

Fig. 12. Performance trends of different tail–torso length ratio.

Figure 12 presents the performance trend with respect to different tail lengths. It is not surprising that the longer the tail, the larger the maneuvering angle and the larger the required control effort. However, this trend does not go forever due to the physical limitations of the actuator. In fact, when the tail is longer than two times the torso length, the performance improvement is very limited. Therefore, the tail length of the robot should be determined as two times as long as the torso length.

Mass distribution

The torso COM location and the tail–torso mass ratio are found to have a significant influence on the locomotion performance, in which the former mainly affects the locomotion stability and the latter mainly affects the maneuvering angle. This section investigates this design issue by changing the torso COM location and the mass ratio. The moment of inertia is assumed to be the same for different COM locations and the new tail length is used. The results are collected in Fig. 13 where the tail mass is defined by the total mass of the 12 links (excluding the base link) in the R3RT. From Fig. 13(a), the torso COM location is found to have little influence on the maneuvering angle but has a significant influence on the motion success. Therefore, for maneuvering tasks, the torso COM offset should be between 180 and 330 mm. The green circle indicates the value used in previous maneuvering simulations. Figure 13(b) shows that a larger tail–torso mass ratio helps to increase the maneuvering angle but reaches saturation after 0.1. This is found due to the actuator saturation and the limited airborne period, that is, the tail actuators are not able to drive the heavy tail to the desired position in the limited period. The failed cases (ratio > 0.3) are due to the heavier tail shifting the overall COM backward and thus destabilizing the landing. Therefore, the best tail–torso mass ratio is determined to be between 0.08 and 0.1.

Power consumption

Through previous analyses, it is found that different design configurations require different control efforts for the same motion. This section investigates these configurations further from the perspective of power consumption, that is, finding the total work and the peak power for the maneuvering motion. The former criterion helps to determine the battery capacity while the latter helps to determine the actuator size. Since the leg inertia is neglected, only the tail consumption is calculated. The actuator power, actuator peak power, and the total work are calculated using the equations

$$p_\alpha = \tau_\alpha \dot{\alpha}, p_{\beta 1} = 6R_{cbl} f_{\beta 1} \dot{\beta}_1, p_{\beta 2} = 6R_{cbl} f_{\beta 2} \dot{\beta}_2. \quad (23)$$

$$p_{\max} = \max\{|p_\alpha|, |p_{\beta 1}|, |p_{\beta 2}|\}, W_{ta} = \int (p_\alpha + p_{\beta 1} + p_{\beta 2}) \mathrm{d}t. \quad (24)$$

Since previous sections determine that the most significant factors affecting the maneuvering performance are the tail–torso length ratio and tail–torso mass ratio, these two factors are used as the main design variables for the power consumption calculation. The results are presented in Fig. 14 where the work distribution shows a uniformly descending trend as the tail becomes longer and heavier. However, this trend is violated in the peak power distribution that more and more local peaks appear. This is found due to the higher impact forces in τ_α that are induced by the non-controllable, chaotic landing events. Therefore, the best tail design is determined to have a length of two times as long as the torso length and have a weight of 0.09 times as heavy as the torso weight. This design combination

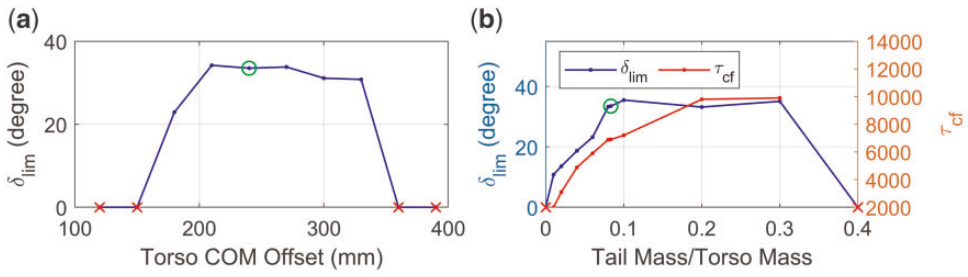

Fig. 13 Performance trends of adjusting the (a) torso COM offset and (b) the tail–torso mass ratio.

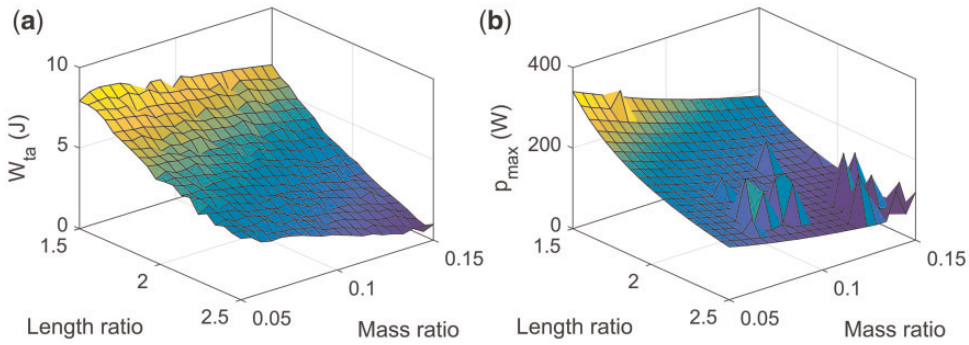

Fig. 14. Power consumptions for maneuvering 15°: (a) total work distribution and (b) peak power distribution.

consumes 4.5 J energy for the maneuvering motion and requires actuators with at least 130 W peak power.

Discussion

As stated in the "Introduction" section, the current robotic system was selected as the start point of this research because it is the simplest platform that combines a serpentine tail with a quadruped body. However, as shown in the analyses, this simplicity inevitably brings some limitations to the platform. This section discusses these limitations and the potential solutions that may lead to the final goal of a general legged robot with a general serpentine tail.

The first limitation comes from the fact that the tail system, the R3RT, can only control two torso orientations simultaneously. This means that for different control tasks (e.g., airborne righting or maneuvering), the tail has to move to different ready positions in advance. This switching of working modes breaks the continuous locomotion into discrete tasks. To alleviate this limitation, omnidirectional robotic tails may be used, such as the mechanisms proposed in Rone et al. (2018) and Liu et al. (2019). However, it is worth noting that as the dexterity increases, the actuation requirements of these tails also increase. That is, the final system may become heavier and thus undermine the dynamic performance of the robot. It is also worth noting that the optimal design results in this article may not fit the final hardware implementation, due to the model simplifications and the specific robot configurations. If this happens, the obtained design guidelines in "Design analysis and synthesis" section will be used to adjust the design parameters.

Another limitation comes from the reduced complexity legs which limit the mobility of the quadruped. Since the legs do not have enough mobility, the quadruped cannot balance itself by planning the footholds. More importantly, based on the Grübler–Kutzbach criterion (Gogu 2005), the quadruped legs may experience motion confliction if more than two feet are on the ground (mobility equals zero). Therefore, if motion confliction is prohibited, the only possible gaits for the RCQ are pronking, bounding, trotting, and galloping, which are all dynamic gaits. The solution for this limitation is simply to use the general-purpose quadruped platform, such as the MIT mini cheetah (Katz et al. 2019). However, considering the heavy tail system to be added, significant design changes would be required to make sure that the tailed robot is still capable of conducting dynamic motions.

As for the tail controller, since the PFL formulation computes torque irrespective of physical joint torque limits, this control scheme is not able to handle control limited systems. In the simulation, the joint torque/cable tension is manually set to the designated saturation value (referring to the "Simulation setting" section) whenever the computed torque exceeds this limit. To include advanced constraints such as the control limit or path constraints, optimal control techniques (Rao 2009) may be used. Indirect methods such as the differential dynamic programming-based controller (Tassa et al. 2014) could be used as the online controller due to its efficiency. Direct methods such as direct collocation (Kelly 2017) or orthogonal collocation (Garg et al. 2010) could be used to prepare offline trajectories.

It is also worth noting that in this work, the bending shape for each tail segment is abstracted as a circular arc. This is not naturally true. The main reason for this choice is that the circular arc bending shape facilitates mechanical design and manufacturing. In addition, the circular arc bending shape showed good approximation accuracy for the hyper-redundant robots and continuum robotic manipulators (Webster and Jones 2010). However, investigations on other bending shapes (e.g., a shape that minimizes actuator demands for generating the same moment) are also valuable for future development of serpentine robotic tails.

Conclusion

This article conducted the theoretical analysis for a new RCQ with a serpentine robot tail. A floating base dynamic model was established, and a hierarchical locomotion control framework was proposed. Based on the established model, numerical experiments on the airborne righting and maneuvering behaviors of the robot were conducted. The results validated the proposed locomotion paradigm (simplified legs plus a dexterous tail) and verified the effectiveness of the proposed locomotion controller. Using the established simulation environment, systematic investigations on the critical tail parameters, system mass properties, and power consumptions were also performed. The results reveal that the most significant factors affecting the robot performance are the torso COM location, tail–torso mass ratio, and the tail–torso length ratio. Based on the analysis results, the optimal tail design was determined to be two times as long and 0.09 times as heavy as the torso. This design enables the robot to accomplish the maneuvering motion using 4.5 J energy and 130 W tail actuators.

Data availability statements

The data underlying this article will be shared on reasonable request to the corresponding author.

Supplementary data

Supplementary data are available at *ICB* online.

Funding

This work was supported by the National Science Foundation (1906727).

References

Azad M, Featherstone R. 2014. A new nonlinear model of contact normal force. IEEE Trans Robot 30:736–9.

Briggs R, Lee J, Haberland M, Kim S. 2012. Tails in biomimetic design: analysis, simulation, and experiment. IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura, Portugal, October 7–12. New York (NY): IEEE. p. 1473–80.

Casarez CS, Fearing RS. 2018. Steering of an underactuated legged robot through terrain contact with an active tail. IEEE/RSJ International Conference on Intelligent Robots and Systems, Madrid, Spain, October 1–5. New York (NY): IEEE. p. 2739–6.

Chang-Siu E, Libby T, Brown M, Full RJ, Tomizuka M. 2013. A nonlinear feedback controller for aerial self-righting by a tailed robot. IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, May 6–10. New York (NY): IEEE. p. 32–9.

Dawson RS, Warburton NM, Richards HL, Milne N. 2015. Walking on five legs: investigating tail use during slow gait in kangaroos and wallabies. Aust J Zool 63: 192–200.

De A, Koditschek DE. 2015. Parallel composition of templates for tail-energized planar hopping. IEEE International Conference on Robotics and Automation, Seattle, WA, May 26–30. New York (NY): IEEE. p. 4562–9.

Featherstone R. 2014. Rigid body dynamics algorithms. Berlin: Springer.

Freymiller GA, Whitford MD, Higham TE, Clark RW. 2019. Escape dynamics of free-ranging desert kangaroo rats (rodentia: heteromyidae) evading rattlesnake strikes. Biol J Linn Soc 127:164–72.

Garg D, Patterson M, Hager WW, Rao AV, Benson DA, Huntington GT. 2010. A unified framework for the numerical solution of optimal control problems using pseudospectral methods. Automatica 46:1843–51.

Gogu G. 2005. Mobility of mechanisms: a critical review. Mech Mach Theory 40:1068–97.

Hickman GC. 1979. The mammalian tail: a review of functions. Mamm Rev 9:143–57.

Jusufi A, Kawano DT, Libby T, Full RJ. 2010. Righting and turning in mid-air using appendage inertia: reptile tails, analytical models and bio-inspired robots. Bioinspir Biomim 5:045001.

Katz B, Di Carlo J, Kim S. 2019. Mini cheetah: a platform for pushing the limits of dynamic quadruped control. IEEE International Conference on Robotics and Automation, Montreal, Canada, May 20–24. New York (NY): IEEE. p. 6295–301.

Kelly M. 2017. An introduction to trajectory optimization: how to do your own direct collocation. SIAM Rev 59:849–904.

Kohut NJ, Pullin AO, Haldane DW, Zarrouk D, Fearing RS. 2013. Precise dynamic turning of a 10 cm legged robot on a low friction surface using a Tail. IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, May 6–10. New York (NY): IEEE. p. 3299–306.

Libby T, Johnson AM, Chang-Siu E, Full RJ, Koditschek DE. 2016. Comparative design, scaling, and control of appendages for inertial reorientation. IEEE Trans Robot 32:1380–98.

Libby T, Moore TY, Chang-Siu E, Li D, Cohen DJ, Jusufi A, Full RJ. 2012. Tail-assisted pitch control in lizards, robots and dinosaurs. Nature 481:181–4.

Liu Y, Ben-Tzvi P. 2020a. An articulated closed kinematic chain planar robotic leg for high speed locomotion. J Mech Robot 12:041003.

Liu Y, Ben-Tzvi P. 2020b. Design, analysis, and integration of a new two-degree-of-freedom articulated multi-link robotic tail mechanism. J Mech Robot 12:021101.

Liu Y, Ben-Tzvi P. 2021a. Dynamic modeling, analysis, and comparative study of a quadruped with bio-inspired robotic tails. Multibody Syst Dyn 51:195–219.

Liu Y, Ben-Tzvi P. Forthcoming 2021b. Feedback control of the locomotion of a tailed quadruped robot. ASME 2021 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Virtual, Online, August 17–20.

Liu Y, Wang J, Ben-Tzvi P. 2019. A cable length invariant robotic tail using a circular shape universal joint mechanism. J Mech Robot 11:051005.

Nabeshima J, Saraiji MY, Minamizawa K. 2019. Prosthetic tail: artificial anthropomorphic tail for extending innate body functions. In: Proceedings of the 10th Augmented Human International Conference. New York (NY): ACM. p. 1–4.

Nansai S, Rojas N, Elara MR, Sosa R, Iwase M. 2015. On a jansen leg with multiple gait patterns for reconfigurable walking platforms. Adv Mech Eng 7:168781401557382.

O'Connor SM, Dawson TJ, Kram R, Donelan JM. 2014. The kangaroo's tail propels and powers pentapedal locomotion. Biol Lett 10:20140381.

Patel A, Boje E. 2015. On the conical motion of a two-degree-of-freedom tail inspired by the cheetah. IEEE Trans Robot 31:1555–60.

Rao AV. 2009. A survey of numerical methods for optimal control. Adv Astronaut Sci 135:497–528.

Rone WS, Saab W, Ben-Tzvi P. 2018. Design, modeling, and integration of a flexible universal spatial robotic tail. J Mech Robot 10:041001.

Rone WS, Saab W, Kumar A, Ben-Tzvi P. 2019. Controller design, analysis, and experimental validation of a robotic serpentine tail to maneuver and stabilize a quadrupedal robot. J Dyn Syst Meas Control 141:081002.

Saab W, Rone WS, Ben-Tzvi P. 2018a. Discrete modular serpentine robotic tail: design, analysis and experimentation. Robotica 36:994–1018.

Saab W, Rone WS, Kumar A, Ben-Tzvi P. 2019. Design and integration of a novel spatial articulated robotic tail. IEEE/ASME Trans Mechatron 24:434–46.

Saab W, Yang J, Ben-Tzvi P. 2018b. Modeling and control of an articulated tail for maneuvering a reduced degree of freedom legged robot. IEEE/RSJ International Conference on Intelligent Robots and Systems, Madrid, Spain, October 1–5. New York (NY): IEEE. p. 2695–700.

Santiago JLC, Godage IS, Gonthina P, Walker ID. 2016. Soft robots and kangaroo tails: modulating compliance in continuum structures through mechanical layer jamming. Soft Robot 3:54–63.

Simon B, Sato R, Choley JY, Ming A. 2018. Development of a bio-inspired flexible tail system. 12th France-Japan and 10th Europe-Asia Congress on Mechatronics, Tsu, Japan, September 10–12. New York (NY): IEEE. p. 230–5.

Spong MW. 1994. Partial feedback linearization of underactuated mechanical systems. IEEE/RSJ International Conference on Intelligent Robots and Systems, Munich, Germany, September 12–16. New York (NY): IEEE. p. 314–21.

Tassa Y, Mansard N, Todorov E. 2014. Control-limited differential dynamic programming. IEEE International Conference on Robotics and Automation, Hong Kong, China, May 31–June 7. New York (NY): IEEE. p. 1168–75.

Wang X, Ren H, Kumar A, Ben-Tzvi P. 2020. Design and analysis of a variable inertia spatial robotic tail for dynamic stabilization. Biomimetics 5:55.

Webster RJ, Jones BA. 2010. Design and kinematic modeling of constant curvature continuum robots: a review. Int J Robot Res 29:1661–83.

Winkler AW, Bellicoso CD, Hutter M, Buchli J. 2018. Gait and trajectory optimization for legged systems through phase-based end-effector parameterization. IEEE Robot Autom Lett 3:1560–7.

Young JW, Russo GA, Fellmann CD, Thatikunta MA, Chadwell BA. 2015. tail function during arboreal quadrupedalism in squirrel monkeys (*Saimiri boliviensis*) and Tamarins (*Saguinus oedipus*). J Exp Zool Part A Ecol Genet Physiol 323:556–66.

Zeglin GJ. 1991. Uniroo—a one legged dynamic hopping robot [dissertation]. Boston (MA): Massachusetts Institute of Technology.