Auton Robot
DOI 10.1007/s10514-014-9401-4

A template candidate for miniature legged robots in quasi-static motion

Konstantinos Karydis · Yan Liu · Ioannis Poulakakis · Herbert G. Tanner

Received: 13 August 2013 / Accepted: 3 July 2014

Abstract The paper introduces the Switching Four-bar Mechanism (SFM), a new low-dimensional kinematic abstraction for miniature legged robots, aimed at quasi-static motion planning in the horizontal plane. The model comprises a rigid torso and four rigid legs which engage in an alternating tetrapod gait. As the gait is executed, the torso and the legs form two switching four-bar linkages, parameterized by the leg touchdown and liftoff angles, as well as the leg angular velocity. We show that the SFM model captures on average experimentally observed behaviors of an eight-legged miniature robot crawling at low speeds quasi-statically. This work represents a first step toward a template that captures critical aspects of the kinematic behavior of miniature legged robots implementing quasi-static gaits. Such template can be used as a tool to facilitate motion planning tasks with such robots.

Keywords Miniature legged robots · Template · Motion Primitives · Planning

1 Introduction

Robots that move on multiple legs have the potential to traverse terrain where wheeled ones cannot. At a miniature scale in particular, legged systems can overcome obstacles and move in confined spaces in ways impossible for their wheeled counterparts (Mongeau et al. 2012). Such enhanced mobility capabilities can greatly benefit operations like pipe and building inspection, search-and-rescue in collapsed buildings or unstructured environments, reconnaissance, and unobtrusive wildlife monitoring.

Examples of miniature robots include the cockroach-inspired hexapod (Yumaryanto et al. 2006) which uses two piezoelectric ceramic actuators to drive its legs, the Mini-Whegs robot series (Morrey et al. 2003; Lambrecht et al. 2005) utilizing a three-spoke rimless wheel ("wheg") design, and the 3D-printed STAR robot (Zarrouk et al. 2013), which incorporates a similar leg design in addition to a mechanism that adjusts the sprawl angle of the robot. Another example is the 155 mm-long i-Sprawl (Kim et al. 2006) robot, manufactured via the Shape Deposition Manufacturing (SDM) process (Cham et al. 2002). System integration at very small scales has been made possible through the Smart Composite Microstructure (SCM) fabrication technique (Wood et al. 2008); typical examples include HAMR^2 (Baisch et al. 2010), ROACH (Hoover et al. 2008), and Medic (Kohut et al. 2011) robot weighing 2, 2.4, and 5.5 g, respectively. The SCM process has been also used to fabricate minimally actuated palm-sized walking robots, including the 100 mm-long hexapod crawlers DASH (Birkmeyer et al. 2009), DynaROACH (Hoover et al. 2010) and VelociROACH (Haldane et al. 2013), as well as the 130 mm-long eight legged robot OctoROACH (Pullin et al. 2012), which motivates our work in this paper (Fig. 1).

Developing analytical models for miniature robots is challenging. Indeed, the novel manufacturing processes and materials employed to fabricate such platforms lead to complex highly-articulated designs, and involved transmission and actuation mechanisms. These, in addition to manufacturing variabilities, can result in poor mobility per-

K. Karydis (✉) · Y. Liu · I. Poulakakis · H. G. Tanner
University of Delaware, 126 Spencer Lab, 130 Academy Street, Newark, DE, USA
e-mail: kkaryd@udel.edu

Y. Liu
e-mail: liuyan@udel.edu

I. Poulakakis
e-mail: poulakas@udel.edu

H. G. Tanner
e-mail: btanner@udel.edu

Published online: 24 July 2014

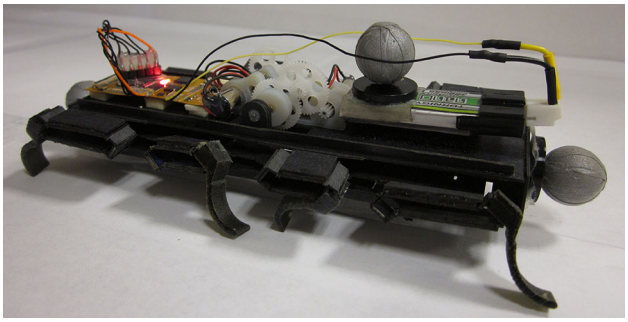

Fig. 1 The OctoRoach, designed by Pullin et al. (2012) at the University of California, Berkeley and manufactured by Motile Robotics, Inc. Its body size is 130×60×30 mm, it weights 35 g, and it can reach a maximum speed of 0.5 m/s

formance (Garcia Bermudez et al. 2012), which is not captured by traditional modeling approaches; one way of attempting to capture it is by horizontal-plane, *behavioral* models.

Research in sprawled arthropods (Blickhan and Full 1993; Cavagna et al. 1977) has motivated the introduction of reduced-order behavioral models. These models aim to capture the underlying similarities of the center-of-mass (COM) motion in animals and robots as they move through their environment (Full and Koditschek 1999). Examples of such models include the horizontal-plane Lateral Leg Spring (LLS) template (Schmitt and Holmes 2000b,a; Holmes et al. 2006) which has been successful in explaining lateral stabilization (Seipel et al. 2004), and in deriving turning strategies (Jindrich and Full 1999; Proctor and Holmes 2008) for hexapedal runners. In its typical configuration, the LLS is a conservative mechanical system composed by a rigid torso and two prismatic legs that are modeled as massless springs. Each leg represents the collective effect of a support tripod formed during the stance phase by the front and rear ipsilateral[1] (see Appendix), and the contralateral middle legs that are in contact with the ground. The Sliding Spring Leg (SSL) model (Zarrouk and Fearing 2012) incorporates the sliding effects of the leg-ground interaction for hexapedal robots operating in realistic applications.

The dynamic nature of these models may present challenges and unnecessary complexity to motion planning for low crawling speeds. Indeed, there is evidence that the motion of the robots at this scale and speed is dominated by surface forces rather than dynamic and inertia effects (Spence et al. 2010; Qian et al. 2012). We currently lack detailed ground interaction descriptions that could be incorporated into a dynamical model (Li et al. 2010). What is more, it remains unclear how to directly map the parameters of such models to robot design parameters (Hoover et al. 2010), and control parameters such as motor gains.

On the other hand, crawling at low speeds in a quasi-static operation regime is typically captured well by a kinematic model. For instance, (Hoffman and Wood 2010) proposed a kinematic model for their centipede microrobot. Although this model is capable of describing the physical platform accurately, it tends to be tailored to that particular mechanism. Thus, there appears to be a need for more general *kinematic templates*, but current literature only offers dynamic counterparts.

Once a kinematic model is available, a variety of established planning methodologies (LaValle 2006; Choset et al. 2005) can be employed to tackle complex motion planning problems. One such approach is based on *motion primitives*, with an eye at considerably simplifying the overall motion planning task. Each motion primitive corresponds to a basic action attainable by the system, such as "go straight," or "turn right," and respects motion constraints. Then, motion primitives can be concatenated to yield more complex motion plans.

The way motion primitives can be defined mainly depends on the system. Although a general formulation that facilitates portability between different platforms is still elusive, there exist some promising approaches. For example, motion primitives for acrobatic helicopters are being generated by on-board controllers in Frazzoli et al. (2005), while a different approach (Delmotte et al. 2008) uses experimental data from ants to produce motion primitives that are then being applied to drive unicycle systems. If a model for the platform exists, an alternative way for extracting motion primitives lies on optimal control design (Martin et al. 2011).

In this work we propose a new low-dimensional kinematic abstraction as a means to capture the quasi-static behavior of miniature legged robots in the horizontal plane. The model is named Switching Four-bar Mechanism (SFM), and includes a rigid torso and four stick legs moving according to an alternating tetrapod gait. Its motion is fully determined by a small set of parameters, namely the leg touchdown and liftoff angles, and the leg angular velocity. This model provides an intuitive way for mapping high-level motion planning specifications—such as path curvatures—to robot parameters, when the platform operates in a quasi-static motion regime, and at low crawling speeds.

We experimentally validate the ability of this model to describe on average the behavior of the OctoRoach robot as it is observed experimentally. We consider three curvature-parameterized motion primitives: (i) *straight line*, (ii) *clockwise turn*, and (iii) *counter-clockwise turn*, and set their parameters based on planar position and orientation measurements through motion capture. With these motion primitives at hand, we solve a constrained optimization problem to identify the particular model parameter values that lead

[1] Ipsilateral means on the same side and contralateral means on the other side. A list of the terms used in this paper is presented in the Appendix.

to model outputs matching experimentally observed behaviors of the robot. Finally, we use the primitives to realize a solution to the problem of following a square path, and compare the results predicted by our model with experimental data.

The SFM model can be viewed as a candidate for a template that captures and reproduces the horizontal-plane behavior of miniature legged robots, such as the OctoRoACH robot, when operating in quasi-static regimes and at low crawling speeds. Its parameters can be tuned to achieve various motion patterns, without relying explicitly on the detailed mechanisms by which the physical platform generates motion. The proposed kinematic template may support the development of more complex motion plans for miniature crawling robots, and provides a systematic and effective means for linking autonomous motion planning techniques with physically implementable control strategies.

The structure of the paper is as follows. Section 2 discusses the mechanical design of the OctoRoACH, and its experimentally observed behavior. Section 3 introduces the proposed template and presents its kinematic analysis, while Sect. 4 demonstrates its capabilities in simulating straight-line and curved paths. In Sect. 5 we construct various parameterized motion primitives for the robot by collecting experimental data, and we identify the model parameters that enable the model to reproduce each experimentally observed motion primitive. Then, Sect. 6 presents the problem of planning a path of square pattern using the motion primitives that were previously constructed. Section 7 concludes the paper. Some of the terminology used in the paper is explained in more detail in the Appendix.

2 The OctoRoACH robot

This section describes the design and kinematic behavior of the OctoRoACH, and highlights the aspects of this behavior that inspired the proposed model.

The robot consists of a rectangular body, two actuators, and eight legs organized so that all four legs at one side are driven by a single actuator. There is no explicit coupling between the two sides, each being driven by a single DC brushed motor through a two-stage gear transmission. As a result of its differential-drive mechanism, changing the gains of the motors at each side results in either straight-line or curved paths. The robot features onboard electronics for communication and motor control, while a 300 mAh lithium polymer (LiPo) battery powers the assembly; a detailed account of the robot's design is given by Pullin et al. (2012).

The leg drive kinematics combines a slider-crank linkage responsible for leg abduction and adduction (for definitions of these motions, see Appendix), and a parallel four-bar

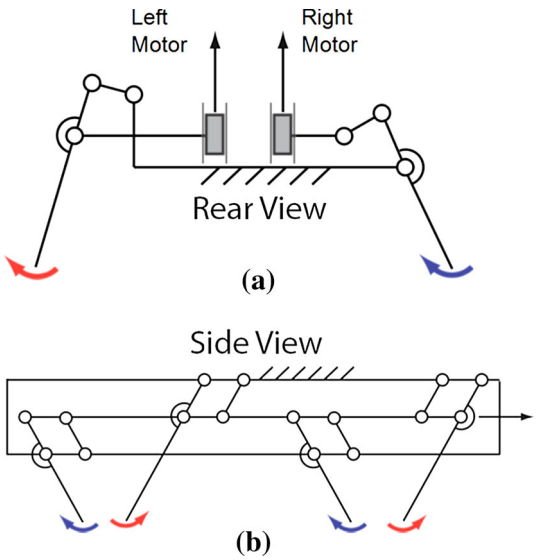

Fig. 2 Idealized robot kinematics demonstrating the kinematic coupling enabling fore-aft and in-out motion of legs. The two sides of the robot are driven independently. The motor outputs are aligned with the sagittal plane and a crank provides the vertical and fore-aft motion depicted in each figure. **a** Rear view of the ideal robot kinematics: ab- and adduction of leg pairs on each side occur out of phase when the middle member of the linkage is translated vertically. **b** Side view of the ideal robot kinematics: protraction and retraction of the legs are controlled by motion of the middle member in the fore-aft direction. The alternate pairs move approximately 180° out of phase with each other. (Courtesy of A. Pullin; reproduced from (Pullin et al. 2012) with the author's permission)

mechanism responsible for leg protraction and retraction[2], as shown in Fig. 2 (Pullin et al. 2012). Due to its mechanical coupling, the robot follows a gait whose foot-fall pattern consists of two alternating tetrapods (Fig. 3). Such metachronal gaits have been studied in the context of an eight-legged arthropod (the Ghost Crab) by Blickhan and Full (1987), and are in direct analogy with the tripod gaits commonly employed by a variety of six-legged animals and robots (Holmes et al. 2006).

Figure 4 presents typical paths of the OctoRoACH. The data depicted in Fig. 4a corresponds to inputs that should theoretically produce straight-line paths, whereas those in Fig. 4b correspond to inputs that in principle generate counter-clockwise circular paths. In different runs, the same inputs produce paths with high variability in ending points as well as path curvatures. This variability may be caused by uncertain leg-ground interaction, the effect of which tends to be more pronounced at these scales.

We report on an approach to modeling this uncertainty in Karydis et al. (2013). In this paper, instead, we focus on the

[2] Protraction (Retraction) refers to the motion of the legs on the sagittal plane bringing them farther (closer) to the center of the body; see Appendix.

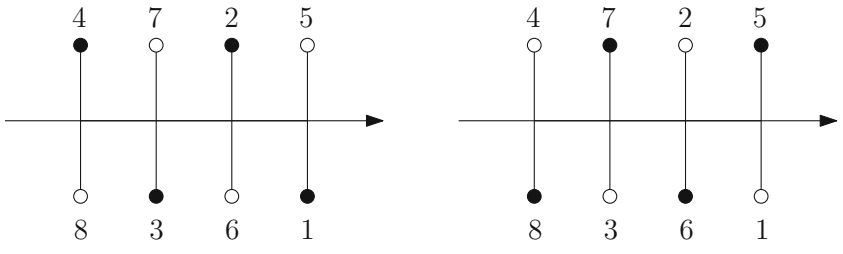

Fig. 3 The foot-fall pattern of the robot is a tetrapod gait

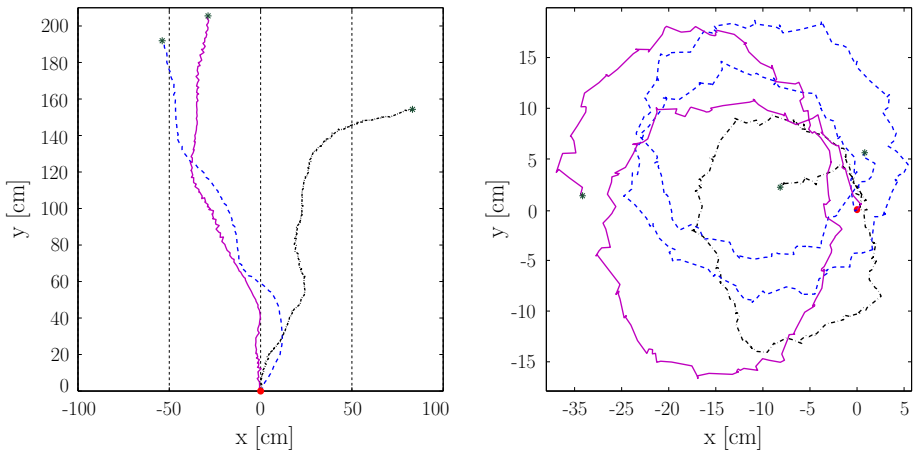

Fig. 4 Experimental results showing geometric center paths of the physical platform taken using a motion capture system. In both cases, the time duration was 20 s. All tests start at the origin (shown as a *red circle*), while the ending points are marked by a *green asterisk*. **a** Motor gains that should produce straight-line paths. The *dashed blue line* corresponds to motor gains {225, 225}, the solid magenta to {150, 150} and the *dashed-dotted black line* to {100, 100}. **b** Motor gains that should produce motion along a counter-clockwise circular path. The motors gains are set to {25, 75}, and remain unchanged. It can be readily verified that in both cases, the robot demonstrates a very high variability in its behavior (Color figure online)

proposed template, and tune its parameters according to the average of experimentally observed behaviors of the robot.

3 The Switching Four-bar Mechanism template

The SFM is a low-complexity kinematic model, intended to capture the essential features of the horizontal-plane behavior of small legged robots, like OctoRoACH, when moving in a quasi-static motion regime. The template offers a small set of parameters that can be tuned to achieve various motion patterns, and it does not rely explicitly on the detailed mechanisms by which the actual robot generates motion. For details on the terminology regarding the derivation of the SFM model equations, see Appendix.

3.1 Derivation of the template

Motivated by the foot-fall pattern of Fig. 3, we first consider an original, eight-legged horizontal-plane abstract model (Karydis et al. 2012), depicted in Fig. 5a. The model represents the robot's torso, and eight legs in the form of non-deformable links attached to the torso via hip joints.

This abstract model of Fig. 5a, in agreement with the foot-fall pattern of Fig. 3, has legs {1, 2, 3, 4} that act in unison and form one tetrapod, and legs {5, 6, 7, 8} that form the second tetrapod. The generalized coordinates for this system are the Cartesian position $\mathbf{x}_G = [x_G, y_G]^\top$ of the geometric center G of the platform's body with respect to an inertial coordinate frame O, and the angle θ between the longitudinal body-frame axis and the vertical axis of the inertial frame.

The eight-legged model of Fig. 5a can then be reduced by exploiting the synergies and symmetries that regulate the motion of the legs of the robot. Specifically, ipsilateral legs for each tetrapod are bound to touch the ground at the same instant, rotate with the same angular velocity and move in phase, thus forming the same angle with respect to the longitudinal body-frame axis at all times. For instance, legs {1, 3} in Fig. 5a form the same angle α with respect to the longitudinal body-frame axis as the system moves forward; these legs engage in a coupled in-phase motion. The complementary legs {2, 4} of the same tetrapod rotate in the same way, forming an angle β with the longitudinal body-frame axis,

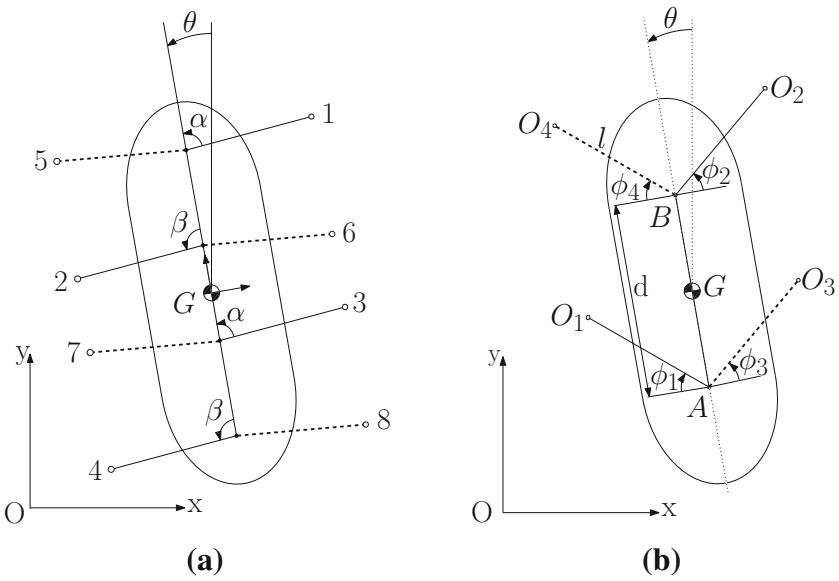

Fig. 5 **a** An eight-legged kinematic model for the OctoRoach. **b** The SFM template, where each of the two tetrapods has been replaced by a pair of legs: tetrapod {1, 2, 3, 4} corresponds to the legs $\{O_1, O_2\}$ denoted the *right pair*, and tetrapod {5, 6, 7, 8} is matched to the legs $\{O_3, O_4\}$, denoted the *left pair*. When the right pair is active, the mechanism comprising the links $\{O_1A, AB, BO_2\}$ takes the form a four-bar linkage. Similarly with the left pair active

which is not explicitly related with the angle α. The other tetrapod operates in a similar fashion.

Based on this coupling, we combine ipsilateral legs of each tetrapod into a single "virtual" leg. Depending on the chosen location of the hinge point of the virtual leg, the leg's initial angle and range of motion may have different values compared to those of the original legs in the pair; this enables the virtual leg to generate the same mechanism displacement as the pair it replaces.

In this way, we obtain the SFM template shown in Fig. 5b. In this model, the contralateral virtual legs (e.g. $\{O_1, O_2\}$) represent the collective effect of the tetrapod they replace (e.g. {1, 2, 3, 4}). Notice that contralateral legs within a pair of virtual legs are allowed to form different angles with respect to the longitudinal body-frame axis, and retain distinct angular velocities. That is, $\phi_1 \neq \phi_2$ and $\dot{\phi}_1 \neq \dot{\phi}_2$ in general for the case of the pair $\{O_1, O_2\}$ in Fig. 5b. We make the following assumptions.

Assumption 1 Once a pair of virtual legs touches the ground (*effective pair*), the tips of the legs remain fixed until the other pair touches down (no slipping).

Assumption 2 At any given time other than the switching instant between pairs, only one pair is active (touches the ground).

Assumption 3 Within a pair, a 50 % duty factor[3] for its legs is assumed. Intuitively, both legs of each pair touch, and liftoff the ground at the same time instant.

Practically, the above assumptions suggest that the template can be viewed as a switching four-bar mechanism

[3] The duty factor refers to the percentage of the total cycle during which a leg touches the ground; see Appendix.

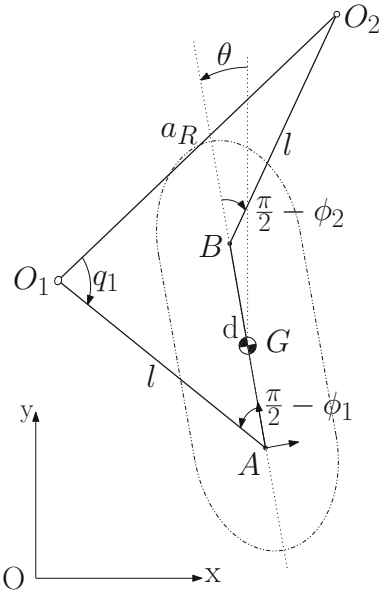

Fig. 6 Analysis of the mechanism; d is the distance between the hip points, l the leg length, and θ denotes the orientation of the model with respect to some global reference frame

(hence the name), where the kinematics of each effective pair is determined by the four-bar linkage these legs represent; (see Fig. 6).

3.2 Kinematic analysis

Due to its kinematic equivalence to a four-bar linkage, the motion of the active pair is fully determined by one degree of freedom. We use the angle ϕ_1 to describe when the right pair is active, and ϕ_3 when the left pair turns active. Notice that thanks to the symmetries of the SFM (Fig. 5b), it suffices

to analyze the motion of only one pair; the other is its mirror image.

Figure 6 illustrates the kinematic analysis of the right pair. Constant d denotes the distance between the two hip joints A and B, while l is the virtual leg length. With the notation of Fig. 6, the (position) vector-loop equation (Norton 2008)

$$\boldsymbol{R}_{AO_1} + \boldsymbol{R}_{O_1O_2} - \boldsymbol{R}_{AB} - \boldsymbol{R}_{BO_2} = 0\,, \tag{1}$$

can be expressed in complex number notation as

$$le^{j(\pi-\phi_1)} + a_R e^{j(q_1-\phi_1)} - de^{j(\pi/2)} - le^{j(\phi_2)} = 0\,. \tag{2}$$

Then, substituting $e^{\pm j\theta} = \cos\theta \pm j\sin\theta$ and separating real and imaginary parts, we write (2) as

$$\begin{aligned} &-l\cos(\phi_1) + a_R\cos(q_1-\phi_1) - l\cos(\phi_2) = 0 \\ &l\sin(\phi_1) + a_R\sin(q_1-\phi_1) - d - l\sin(\phi_2) = 0\,. \end{aligned} \tag{3}$$

Differentiation of (2) results in

$$\begin{aligned} &l\cos(\phi_1)\dot{\phi}_1 + a_R\cos(q_1-\phi_1)(\dot{q}_1-\dot{\phi}_1) - l\cos(\phi_2)\dot{\phi}_2 = 0 \\ &l\sin(\phi_1)\dot{\phi}_1 - a_R\sin(q_1-\phi_1)(\dot{q}_1-\dot{\phi}_1) + l\sin(\phi_2)\dot{\phi}_2 = 0. \end{aligned} \tag{4}$$

The configuration of the single degree of freedom system can be obtained explicitly by numerically solving (3) for a given value of ϕ_1. The angular velocities of the mechanism can be obtained from (4) in a similar fashion. Without loss of generality, we assume that the angular velocity $\dot{\phi}_1$, once set, remains *constant* throughout the stride. Due to symmetry, the left pair $\{O_3, O_4\}$ is analyzed in a completely analogous way.

Both legs in each pair touch, and lift off the ground at specific angles, which we call *touchdown* and *liftoff* angles, and denote ϕ_i^{td} and ϕ_i^{lo}, for $i = 1, \ldots, 4$, respectively. A *sweep* angle ψ_i is defined as

$$\psi_i = |\phi_i^{\text{td}}| + |\phi_i^{\text{lo}}|\,, \tag{5}$$

and quantifies the range of values for each ϕ_i. In the sequel, we set the parameter d equal to 13 cm, which is the length of the torso of the actual platform, and l equal to 3 cm, which matches the robot's half-width.

The output path from the model is parameterized by the values of the touchdown and sweep angles. Different combinations of these parameters can be used to produce very similar paths.

4 Simulations

In this section we demonstrate in simulation the capabilities of the template. The steps required for simulating one cycle are summarized in Table 1.

Table 1 Simulation process for one cycle

Step	Action
1.	Give: θ, $\mathbf{x}_G$;
2.	Give: ϕ_1^{td}, ϕ_2^{td} for the right pair;
3.	Fix points O_1 and O_2;
4.	Calculate a_R;
5.	Solve (3)–(4) as ϕ_1 and $\dot{\phi}_1$ evolve;
6.	Calculate θ, $\mathbf{x}_G$ at end of stride;
7.	Give: ϕ_3^{td}, ϕ_4^{td} for the left pair;
8.	Fix points O_3 and O_4;
9.	Calculate a_L;
10.	Solve (3)–(4) as ϕ_3 and $\dot{\phi}_3$ evolve;
11.	Calculate θ, $\mathbf{x}_G$ at end of stride.

4.1 Generating straight-line paths

In order to generate straight-line paths we select

$$\begin{aligned} &\phi_{\mathsf{SLP}}^{\text{td}} = \phi_1^{\text{td}} = \phi_2^{\text{td}} = \phi_3^{\text{td}} = \phi_4^{\text{td}} \\ &\phi_{\mathsf{SLP}}^{\text{lo}} = \phi_1^{\text{lo}} = \phi_2^{\text{lo}} = \phi_3^{\text{lo}} = \phi_4^{\text{lo}} = -\phi_{\mathsf{SLP}}^{\text{td}}\,, \end{aligned} \tag{6}$$

where the subscript SLP is used to denote straight-line path configuration.

The configuration in (6) practically suggests that in order to achieve straight-line paths we need to enforce *symmetry among the sweep angles*. Recall (5) and notice that from (6) the sweep angle is

$$\psi_{\mathsf{SLP}} = \psi_1 = \psi_2 = \psi_3 = \psi_4 = |\phi_{\mathsf{SLP}}^{\text{td}}| + |\phi_{\mathsf{SLP}}^{\text{lo}}| = 2\phi_{\mathsf{SLP}}^{\text{td}}. \tag{7}$$

The initial, and straight-line configurations are illustrated in Fig. 7a and 7b respectively.

Figure 8 depicts the evolution of the state of the model for the case of straight-line path generation, and for parameter values in accordance with the configuration given in (6) and (7). All plots correspond to a duration of one cycle, having the same initial geometric center position and body orientation. As we increase the value of the sweep angles (equivalently, $\phi_{\mathsf{SLP}}^{\text{td}}$), the model covers more ground at a single cycle, while the waving pattern of motion, associated with the kinematics of the four-bar mechanism, becomes more pronounced.

Figure 9 demonstrates how the model moves along straight-line paths. The pair of legs that is active—that is, in contact with the ground—is marked with solid line, while the respective leg tips (marked with solid disks) are assumed to remain fixed, thereby forming a hinged joint with the ground. The legs of the active pair rotate around these joints according to the kinematics of a four-bar linkage until the liftoff configuration is reached. At this instant (Fig. 9b), the opposite pair of legs becomes active while the formerly active pair

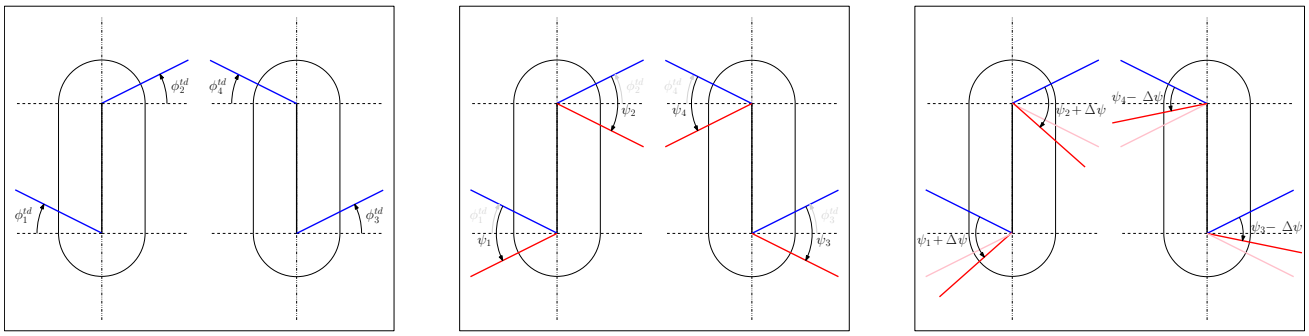

Fig. 7 Model configurations to achieve straight-line and curved paths. **a** The initial configuration comprises the four touchdown angles, $\phi_i^{td}, i = 1, \ldots, 4$. **b** All legs are configured so that touchdown and liftoff angles are symmetric with respect to the horizontal body axis (equivalently, all sweep angles, $\psi_i, i = 1, \ldots, 4$, are equal). This configuration enables straight-line paths. **c** Creating asymmetry, $\Delta\psi$, in the sweep angles enables turning of the model. In its more general form, this asymmetry is introduced between the touchdown and liftoff angles of each leg, as well as between the sweep angles of the two pairs

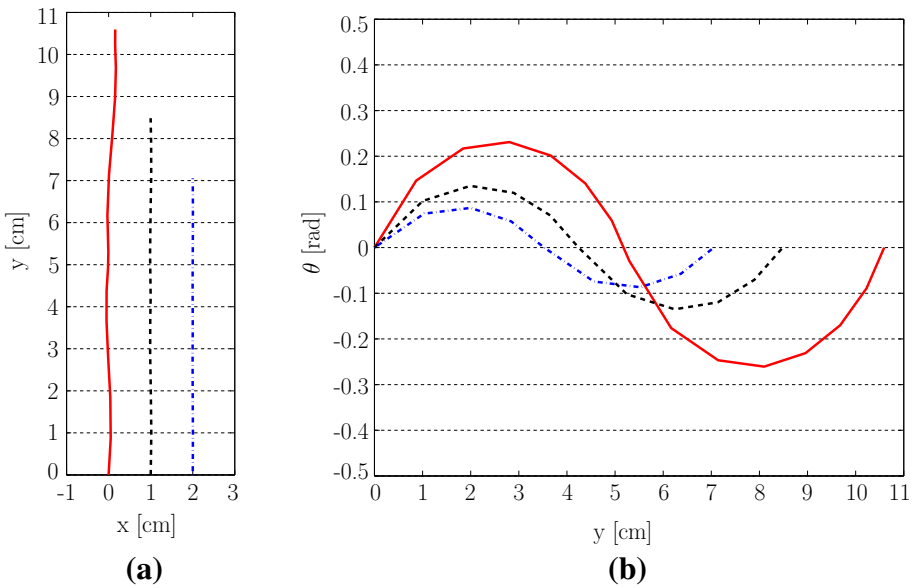

Fig. 8 Evolution of the model's state when executing straight-line paths for $\phi_{\text{SLP}}^{td} = \frac{\pi}{3}$ rad (*red solid*), $\phi_{\text{SLP}}^{td} = \frac{\pi}{4}$ rad (*black dotted*), and $\phi_{\text{SLP}}^{td} = \frac{\pi}{6}$ rad (*blue dashed-dotted*). In all cases, $\dot{\phi}_1$ (for the right pair), and $\dot{\phi}_3$ (for the left pair) are set to the common value of $\frac{\pi}{10}$ rad/s, and remain constant throughout each pair's stride. **a** Evolution of the position of the geometric center, G, of the model. Increasing ϕ_{SLP}^{td} accentuates the waving motion pattern, and enables the model to cover longer distances along the y-direction. **b** Evolution of the orientation of the model, θ, as it moves forward (Color figure online)

is reseted to its touchdown configuration. The process is then repeated (Fig. 9c).

4.2 Generating curved paths

In the previous section we saw that the symmetry of sweep angles results in straight-line paths. Curved paths can be generated by asymmetric sweep angles. Indeed, as Fig. 8b suggests, creating asymmetry in the sweep angles between the two pairs (Fig. 7c) leads to non-zero values for the model's orientation angle, θ, at the end of each stride. If the asymmetry is retained, this deviation will keep propagating, thus moving the template along a curved path. It should be emphasized that this method of generating curved paths is relevant to the SFM template, and is different from how the OctoROACH actually turns. In particular, turning in the robot is produced by applying different gains on the motors that actuate the legs on its left and right sides; this is essentially equivalent to a differential-drive steering method.

The asymmetry $\Delta\psi$ is formally defined by

$$\Delta\psi \triangleq |\min\{\psi_1, \psi_2\} - \min\{\psi_3, \psi_4\}| \ . \tag{8}$$

The $\min(\cdot)$ function above ensures the assumed 50 % duty factor for the legs of each pair (Assumption 3).

Figure 10 presents the evolution of the geometric center of the model when generating counter-clockwise curved paths for various combinations of touchdown, liftoff and sweep angles. As before, $d = 13$ cm, and $l = 3$ cm, while all paths are ten-cycle long, having the same initial position of the geometric center and body orientation. Notice that as the asymmetry $\Delta\psi$ increases, the produced paths have smaller radii.

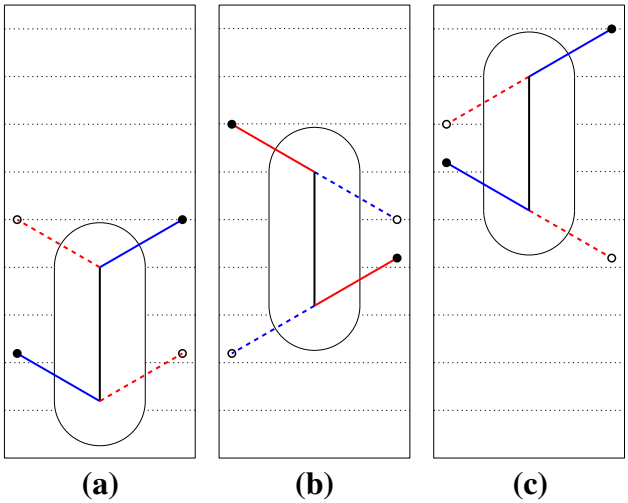

Fig. 9 The template propelling itself in straight-line paths. *Solid lines* are used to indicate the active pair at each time instant, the tips of which are marked with *solid disks*. **a** The right pair (marked in *solid blue*) begins first. Its legs rotate around their tips, which are assumed to remain fixed to the ground. **b** When the right pair reaches its liftoff configuration, the left pair (marked in *dashed red*) turns active, and its legs rotate around their hinged tips—circles are switched to *solid disks* to indicate that these tips are now fixed to the ground. At the same time, the right pair is reseted to its touchdown configuration. **c** Both pairs have completed their stride, and the model is ready to enter a second cycle (Color figure online)

The model allows for asymmetry to be generated in multiple ways, and as a result, paths of similar nature can be achieved through different combinations of touchdown, liftoff, and sweep angles. For example, consider the limiting case of right pair actuation only, depicted in Fig. 10a. In this special case, the generated paths correspond to sharper turns and have shorter length relative to their counterparts shown in Fig. 10b, in which right and left pair actuation alternate. To resolve redundancy we need a systematic way for matching path curvatures to model parameter values.

4.3 From model parameters to path curvatures

To support path and motion planning, we first need to characterize the geometric characteristics of paths produced from the model. Specifically, we are interested in relating path curvatures to model parameters such as the touchdown and liftoff angles. This is achieved by recruiting techniques from differential geometry of curves and surfaces (Do Carmo 1976; Stoker 1969).

From a general point of view, the curvature of curves on a surface can be linked to surface (Gaussian) curvatures by applying the *Gauss–Bonnet theorem*. For our planar problem, the Gaussian curvature is zero, and the Gauss–Bonnet theorem reduces to

$$\sum_{j=0}^{L} \int_{s_j}^{s_{j+1}} k(s)\mathrm{d}s + \sum_{j=0}^{L} \chi_j = 2\pi \ , \tag{9}$$

where L is the total number of model steps, s_j is the arc length of the j-th step, and $k(s)$ is the curvature of the curve associated with each step, given by

$$k(s) = \frac{x'y'' - x''y'}{((x')^2 + (y')^2)^{\frac{3}{2}}} \ .$$

The quantity χ_j denotes the instantaneous change in orientation when switching from step j to step $j+1$ (Fig. 11).

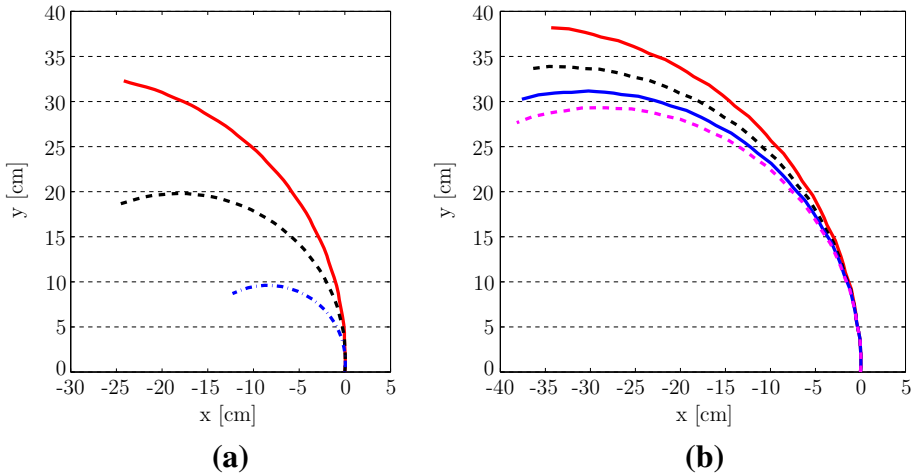

Fig. 10 Evolution of the position of the geometric center, G when executing counter-clockwise curved paths. **a** Only the right pair turns active. We select $\dot{\phi}_1 = \frac{\pi}{10}$ rad/s, set ϕ_1^{td} and ϕ_2^{td} to the common value of $\frac{\pi}{3}$ rad, and plot the model's response for $\phi_1^{lo} = \phi_2^{lo} = -\frac{\pi}{6}$ rad (*red solid*), $\phi_1^{lo} = \phi_2^{lo} = 0$ rad (*black dashed*), and $\phi_1^{lo} = \phi_2^{lo} = \frac{\pi}{6}$ rad (*blue dashed-dotted*). **b** Both pairs are active. We select $\dot{\phi}_1 = \dot{\phi}_3 = \frac{\pi}{10}$ rad/s, set $\phi_1^{td} = \phi_2^{td} = \frac{\pi}{3}$ rad, $\phi_1^{lo} = \phi_2^{lo} = \phi_3^{lo} = \phi_4^{lo} = 0$ rad, and plot the model's response for $\phi_3^{td} = \phi_4^{td} = \frac{\pi}{6}$ rad (*red solid*), $\phi_3^{td} = \phi_4^{td} = \frac{\pi}{7}$ rad (*black dashed*), $\phi_3^{td} = \phi_4^{td} = \frac{\pi}{8}$ rad (*blue solid*), and $\phi_3^{td} = \phi_4^{td} = \frac{\pi}{9}$ rad (*magenta dashed*) (Color figure online)

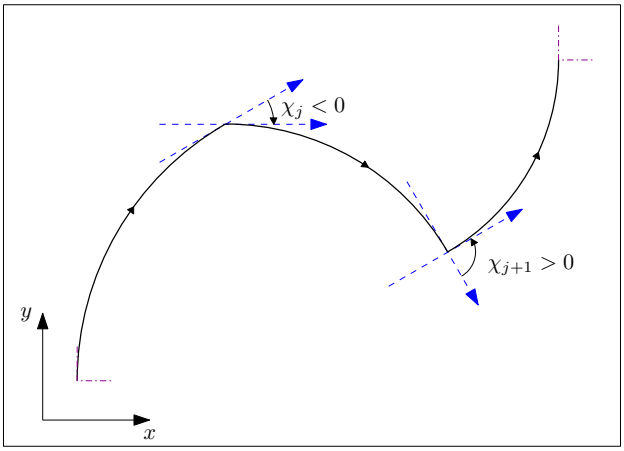

Fig. 11 Instantaneous change in orientation when switching between steps. By convention, the sign of the angle is given by the right hand rule

In essence, the integral term in (9) provides the contribution of the curvature of the path followed within a step, while the other term accounts for the instantaneous change in orientation between steps.

Then, selecting an appropriate number of steps allows the template to transcribe a closed circular curve, for which

$$2\pi R = \int_c \mathrm{d}s \ . \tag{10}$$

Combining (9) and (10) yields

$$R = \frac{\int_c \mathrm{d}s}{\sum_{j=0}^{L} \int_{s_j}^{s_{j+1}} k(s)\mathrm{d}s + \sum_{j=0}^{L} \chi_j} \ . \tag{11}$$

The path curvature produced by specific values for the touchdown and liftoff angles is given by taking the inverse of (11), that is, $k_{\mathsf{path}} = 1/R$.

5 Motion primitives for the OctoRoACH

Our approach to linking the template's parameters to the OctoRoACH is data-driven; we collect data corresponding to different trajectories of the robot, and then identify the model parameters resulting in trajectories that remain close to the experimental average.

5.1 Generation of motion primitives

We define three motion primitives: (i) *straight line* (SL), (ii) *clockwise turn* (CW), and (iii) *counter-clockwise turn* (CCW). These motion primitives are parameterized by the leg touchdown and liftoff angles, as well as the leg angular velocity, or, equivalently, the path curvature and running time. We choose a constant running time of 3 s,[4] and consider nine curvature-parameterized instances of these primitives (Table 2); four instances for each turn type (CW and CCW), and one for the SL type.

Table 2 A library of parameterized motion primitives for the OctoRoACH

	Description	Motor Gains $\{K_L, K_R\}$	Target Curvature (cm^{-1})	Target Heading (deg)
1.	Clockwise U-Turn	{80, 0}	0.250	$\theta \lesssim -150°$
2.	Clockwise Sharp turn	{70, 10}	0.125	$\theta \simeq -135°$
3.	Clockwise 90° Turn	{60, 20}	0.063	$\theta \simeq -90°$
4.	Clockwise Mild turn	{50, 30}	0.031	$\theta \simeq -45°$
5.	Straight-line	{40, 40}	≤ 0.01	$\theta \simeq 0°$
6.	Counter-clockwise Mild turn	{30, 50}	0.031	$\theta \simeq 45°$
7.	Counter-clockwise 90° Turn	{20, 60}	0.063	$\theta \simeq 90°$
8.	Counter-clockwise Sharp turn	{10, 70}	0.125	$\theta \simeq 135°$
9.	Counter-clockwise U-Turn	{0, 80}	0.250	$\theta \gtrsim 150°$

The two motor gains, $\{K_L, K_R\}$ are inputs to the robot and correspond to the left and right side motor respectively. For all instances of motion primitives, the sum of the two motor gains remains constant,

$$|K_L|_i + |K_R|_i = \text{constant}, \ \forall i = 1, \ldots, 9 \ .$$

To acquire the target curvatures, we first estimate experimentally the minimum turning radius of the robot. The tightest turn is achieved when the legs of one only side turn active; we experimentally identified a minimum turning radius of 4 cm.[5] Therefore, the U-Turn CW and CCW primitives need to follow paths of radius of 4 cm, or equivalently, have a curvature of 0.25 cm^{-1}.

This maximum turning curvature defines the range of motion, which we then partition so that our primitives cover it with adequate resolution for our planning purposes; these values are shown in Table 2. Note further that the Straight-Line (SL) type motion primitive should, in theory, have zero curvature. Measurement noise, as well as various random

[4] Later in Sect. 5.2 we provide insight on choosing this duration; see also Fig. 12.

[5] This value depends on the mechanical properties of the test surface, and may slightly vary among different OctoRoACH robots.

effects associated with ground contact render this infeasible. Therefore, we relax this restriction by accepting as SL type paths those that have curvatures less than 0.01 cm^{-1}.

The target orientations are not related to experimental data; instead, they are defined so that they cover the range of orientations $\theta = [-180°, 180°]$ with sufficient resolution. We adopt the convention that the initial orientation of the robot is $\theta = 0°$, and that positive changes in the orientation correspond to counter-clockwise angles θ; see Fig. 5b. With respect to Table 2, and due to the uncertainty inherent to ground contact as well as measuring noise, the actual final orientation of single trajectories deviates from this target value. However, the targeted value is achieved on average, as it will be readily verified in the sequel.

5.2 Experimental setup and data collection

To construct the parameterized motion primitives we generate collections of *open-loop* planar position and orientation measurement data, through experiments in which the robot is configured according to the values of motor gains shown in Table 2. In our experiments, the maximum rate of leg rotation is approximately 2 Hz, which roughly corresponds to a maximum (nominal) forward velocity of 0.05 m/s. The data is obtained with the use of a motion capture system at a rate of 100 Hz, and is written to text files at a rate of 20 Hz. This reduction is necessary due to hardware restrictions, but does not affect the quality of the captured data since the maximum rate of leg rotation is an order of magnitude lower. Each run lasts for 3 sec, and we collect data from a total of 250 paths for each case. The measured states are the planar position of the geometric center of the robot $[x_G, y_G]$, as well as its orientation θ, the angle formed with respect to the longitudinal body-fixed axis and the y-inertial axis; see Fig. 5b.

All experiments are conducted on a rubber floor mat surface; as part of future work we plan to investigate the behavior of the robot on other terrain types as well. The initial position and orientation of the robot were manually set inside a designated area. The initial pose errors that are produced through this inexact procedure can be bounded using data statistics; we calculate the sample mean and standard deviation for the initial position along the x and y directions, as well as the same statistics for the initial orientation. The results are summarized in Table 3. The reported error bounds accommodate for measurement noise as well.

As mentioned above, we have chosen a 3 s duration for our data collections and the generated motion primitives. Figure 12 provides insight on this choice. The histograms show how individual experimental paths disperse at different time instances. For clarity purposes, we only show the histograms of three primitives: 90° CW and CCW Turns, and SL. After the 3 s period, there is a very high dispersion around the experimental averages shown with black thick curves in Fig. 13. As a result, constructing motion primitives that last longer is not meaningful; after this time duration the variance in the experimental data becomes unacceptably high. On the other hand, shorter execution times may increase significantly the computational complexity when concatenating primitives, an asset in the context of motion planning.

Table 3 Generating primitives: Initial pose error statistics

	Measurement	Mean (cm/cm/deg)	Standard Deviation (cm/cm/deg)
1.	Position along x-axis:	−0.097	0.079
	Position along y-axis:	−0.006	0.042
	Orientation:	0.75	0.61
2.	Position along x-axis:	−0.031	0.283
	Position along y-axis:	−0.058	0.213
	Orientation:	0.26	2.23
3.	Position along x-axis:	−0.156	0.177
	Position along y-axis:	−0.041	0.141
	Orientation:	1.23	1.37
4.	Position along x-axis:	−0.136	0.113
	Position along y-axis:	−0.053	0.174
	Orientation:	1.06	0.88
5.	Position along x-axis:	−0.007	0.234
	Position along y-axis:	0.027	0.054
	Orientation:	0.06	1.81
6.	Position along x-axis:	−0.245	0.190
	Position along y-axis:	−0.052	0.188
	Orientation:	1.90	1.49
7.	Position along x-axis:	−0.322	0.156
	Position along y-axis:	−0.012	0.130
	Orientation:	2.50	1.23
8.	Position along x-axis:	−0.258	0.216
	Position along y-axis:	−0.044	0.172
	Orientation:	2.01	1.67
9.	Position along x-axis:	−0.234	0.186
	Position along y-axis:	−0.112	0.148
	Orientation:	1.83	1.46

Figure 13 depicts random samples of 50 paths (plotted in magenta) for each instance of motion primitive as well as the average out of the whole data set of 250 paths, marked by black thick lines. Due to its relative low-precision production, the platform tends to veer to the right, as shown in Fig. 13e.

Table 4 shows the average final state of the robot, for each instance, while Table 5 contains the target and observed values for path curvatures and final orientations, on average. Both path curvatures, as well as final orientations, for all cases are very close to their target values on average, although these metrics may deviate significantly for individual paths.

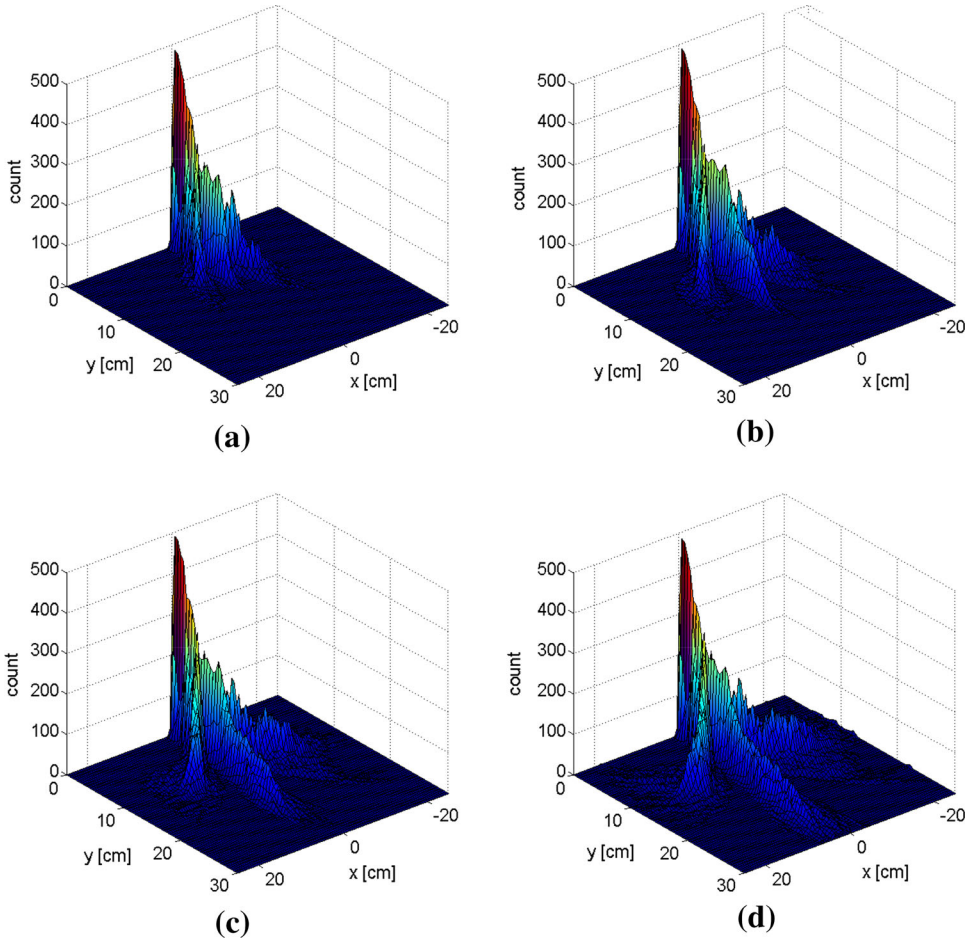

Fig. 12 Path dispersion at **a** 2 s, **b** 2.5 s, **c** 3 s, **d** 3.5 s for three primitives: 90° CW and CCW Turns, and the SL primitive. All primitives start at the origin, and are largely dispersed at the end of the 3 s trial as shown in (**c**). The z axis counts how many paths are inside a particular grid square. Due to the selected grid size, some paths may appear more than once inside a square. Distinguish CW paths, which steer to the left of the page, from CCW paths steering to the right

5.3 Nominal parameter identification

Next, we identify the template parameters that correspond to each parameterization of motion primitive. The parameters estimated are

$$\zeta = [\phi_1^{\mathrm{td,n}}, \phi_2^{\mathrm{td,n}}, \phi_3^{\mathrm{td,n}}, \phi_4^{\mathrm{td,n}}, \phi_1^{\mathrm{lo,n}}, \phi_2^{\mathrm{lo,n}}, \phi_3^{\mathrm{lo,n}}, \phi_4^{\mathrm{lo,n}}, \dot{\phi}_{\mathrm{RL}}] \,,$$

where the superscript $(^{\mathrm{n}})$ stands for "nominal," $(^{\mathrm{td}})$ is used to indicate the model's touchdown angles, while $(^{\mathrm{lo}})$ marks liftoff angles. The angular velocity $\dot{\phi}_{\mathrm{RL}}$ is assumed to be the same for both pairs,[6] since turning is achieved by imposing the asymmetry $\Delta\psi$ in the sweep angles of the two pairs, as per (8). The parameter d is set to 13 cm, which is the length of the torso of the actual platform, while the leg length is l is set to 3 cm.

Let $\mathcal{W}_i$, for $i = 1, \ldots, 9$ denote the collections of all experimentally observed planar paths of the robot for each instance of motion primitives, and use w_i to denote an element in these sets. The averages of all elements for each set are indicated by w_i^{ave}. Let $p_i(\zeta)$ represent the trajectory generated by the model corresponding to a particular value of the parameter vector ζ.

Given sets of paths $\mathcal{W}_i$ realized by the robot, the parameter vector ζ for each parameterization can then be selected as the solution of a constrained least-squares optimization problem

$$\min_{\zeta \in Z} \| p_i(\zeta) - w_i^{\mathrm{ave}} \|^2, \; i = 1, \ldots, 9 \,, \tag{12}$$

where $||\cdot||$ denotes the L_2 norm. Due to Assumption 3, the legs of each pair have to complete their swing phase at the same time instant. This constraint is incorporated in (12) by coupling the liftoff angles for each pair.

The solution ζ^{n} makes the model produce trajectories shown in Fig. 13 with blue thick dashed curves.[7] The numerical values of the components of ζ^{n} are given in Table 6. These values allow the model to capture *on average* the behavior of the system, as shown in Fig. 13.

[6] Only when both pairs are actuated. Otherwise it corresponds to the actuated pair.

[7] It is emphasized here that all model trajectories also correspond to a duration of 3 s.

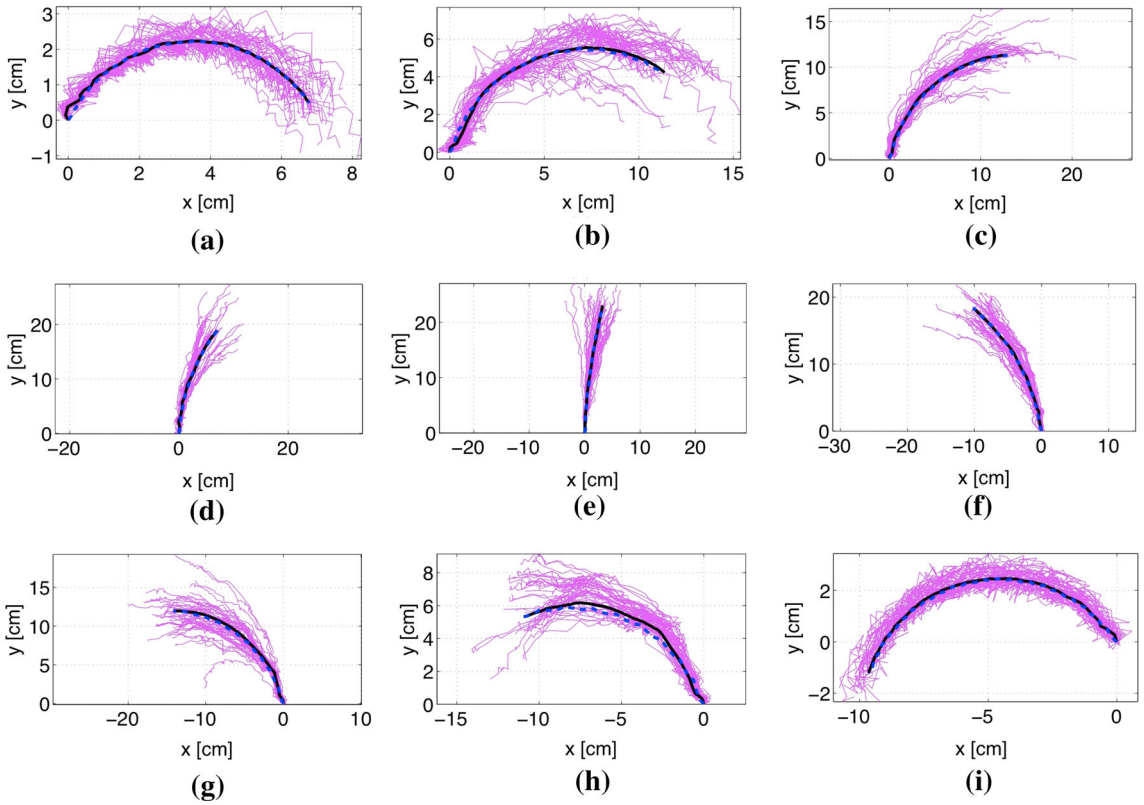

Fig. 13 Experimental data for the parameterized motion primitives contained in Table 2 and the respective nominal template output counterparts. The Figure reads from *left* to *right*, and from *top* to *bottom*. List of primitives: **a** CW U-Turn. **b** CW Sharp Turn. **c** CW 90° Turn. **d** CW Mild Turn. **e** SL. **f** CCW Mild Turn. **g** CCW 90° Turn. **h** CCW Sharp Turn. **i** CCW U-Turn. In all cases, both the robot and the model start at the origin. Out of a total of 250 paths for each parameterization, 50 randomly selected paths of the robot's geometric center are shown in *magenta*, and the *black thick curves* correspond to their experimental average out of the whole data set of 250 paths for each case. The *blue thick dashed curves* depict the output of the model parameterized by the nominal parameter values of Table 6. Note the differences in scale among different cases, and the robot veering off to the right; this is more clearly seen in (**e**) (Color figure online)

Table 4 Generating primitives: Average values for final pose

	X-axis Position (cm)	Y-axis Position (cm)	Orientation (deg)
1.	6.781	0.489	−140.55
2.	11.359	4.220	−129.91
3.	12.928	11.307	−85.02
4.	7.040	18.883	−39.34
5.	3.246	23.044	−10.62
6.	−10.182	18.393	43.26
7.	−14.183	11.960	90.90
8.	−10.939	5.320	101.75
9.	−9.646	−1.211	160.54

Table 5 Average values for curvature and final orientation

	Target Curvature (cm^{-1})	Observed Curvature (cm^{-1})	Target Orientation (deg)	Observed Orientation (deg)
1.	0.250	0.256	$\lesssim -150°$	−140.55
2.	0.125	0.133	$\simeq -135°$	−129.91
3.	0.063	0.066	$\simeq -90°$	−85.02
4.	0.031	0.030	$\simeq -45°$	−39.34
5.	≤ 0.01	0.009	$\simeq 0°$	−10.62
6.	0.031	0.041	$\simeq 45°$	43.26
7.	0.063	0.068	$\simeq 90°$	90.90
8.	0.125	0.127	$\simeq 135°$	101.75
9.	0.250	0.185	$\gtrsim 150°$	160.54

6 Primitives-based path planning

We demonstrate the potential of the SFM in predicting the behavior of the robot in more complex scenarios. In particular, we consider a primitives-based path planning approach, where various behaviors result from suitable concatenations of motion primitives.

6.1 An illustrative example: closing a square

Suppose we want the robot to follow a counter-clockwise square path. Assume that some path planning methodology that takes as input the constructed primitives yields

Table 6 Nominal template parameters

	$\phi_1^{\text{td,n}}$ (deg)	$\phi_2^{\text{td,n}}$ (deg)	$\phi_3^{\text{td,n}}$ (deg)	$\phi_4^{\text{td,n}}$ (deg)	$\phi_1^{\text{lo,n}}$ (deg)	$\phi_2^{\text{lo,n}}$ (deg)	$\phi_3^{\text{lo,n}}$ (deg)	$\phi_4^{\text{lo,n}}$ (deg)	$\dot{\phi}_{\text{RL}}$ (deg/s)
1.	−50.74	−64.69	0	0	−56.75	−70.70	0	0	7.45
2.	−24.66	−44.92	0	0	−38.37	−58.63	0	0	6.86
3.	32.72	4.19	0	0	−55.28	−83.81	0	0	7.56
4.	55.67	43.29	0	0	−72.15	−84.53	0	0	8.59
5.	56.72	45.84	68.18	70.47	−51.78	−62.66	−57.70	−55.41	9.00
6.	0	0	44.69	29.61	0	0	−67.60	−82.68	8.64
7.	0	0	26.28	1.37	0	0	−55.66	−80.57	8.65
8.	0	0	−14.32	−35.59	0	0	−48.00	−69.28	7.59
9.	0	0	−35.09	−61.67	0	0	−41.36	−67.94	8.59

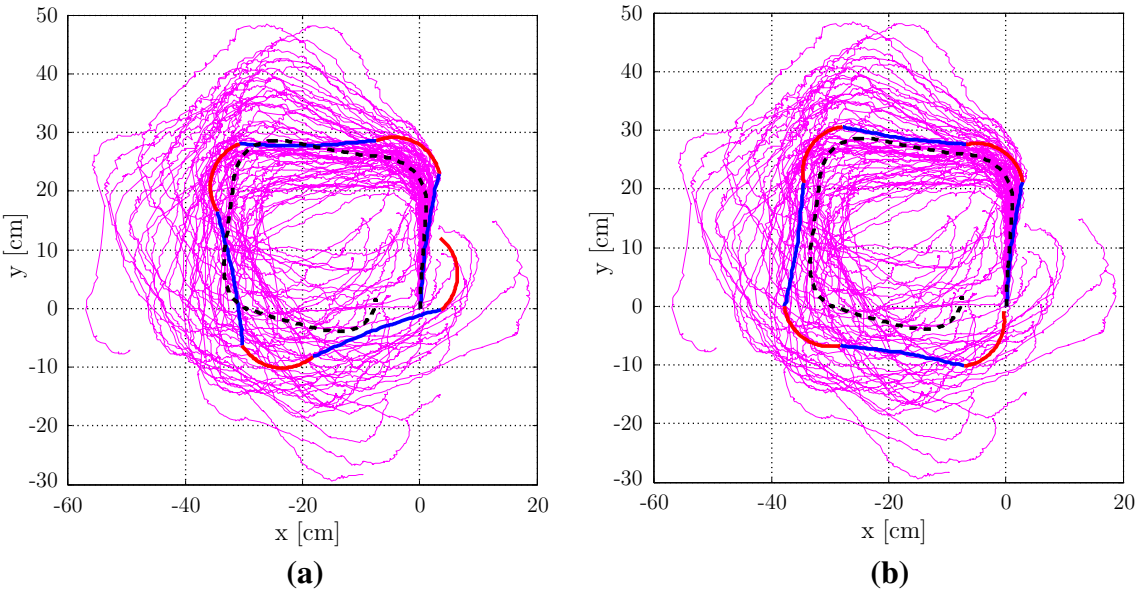

Fig. 14 Following a square path by concatenating Straight-Line and Counter-Clockwise Sharp Turn parameterizations. Model output paths are shown with *red* and *blue thick curves*, and are plotted against the experimentally observed paths, shown in *magenta*. We collected data from a total of 60 paths, captured with a motion capture system, at a rate of 100 Hz. Although individual paths are characterized by a very high variability, their experimental average, plotted with a *black dashed curve*, forms a square pattern. **a** Model paths where each motion primitive had an execution time of 3 s. The output results in a distorted square, and the error between initial and final pose is large. **b** Model paths where each motion primitive had a reduced execution time equal to 2.85 s. In this case, the model output follows a square pattern minimizing the error between the initial and final pose, while staying close to the experimental average (Color figure online)

the solution as the following sequence of parameterized primitives:

$$\mathcal{P} = \{\text{SL},\ \text{CCW},\ \text{SL},\ \text{CCW},\ \text{SL},\ \text{CCW},\ \text{SL},\ \text{CCW}\}\,, \tag{13}$$

where the CCW primitive is the one associated with a sharp turn (to accommodate for the biased behavior of the robot toward the right).

The output of the planner yields the paths plotted as a concatenation of red and blue thick curves in Fig. 14. We first consider an execution time $t_{\text{ex}} = 3\,\text{s}$, similarly to Sect. 5. Figure 14a presents the outcome of this concatenation. The resulting path is rather inaccurate, and has the form of a distorted square. Deviations from the desired square path are primarily because of improper orientation at the end of each primitive. The error propagates and keeps increasing, resulting to a mismatch between final and initial poses.

To suppress the effect of this error, we treat the execution time of each primitive as a parameter to be determined rather than as a fixed value at 3 s. In particular, let $\mathcal{P}(t_{\text{ex}})$ denote a plan (13), where all of its components are parameterized by the execution time $t_{\text{ex}} \in [0, 3]\,\text{s}$, assumed to be the same for all primitives. The value of t_{ex} is then selected as the solution of a constrained least-squares optimization problem

$$\min_{t_{\text{ex}} \in [0,3]} \|\boldsymbol{r}^{\text{fin}}(t_{\text{ex}}) - \boldsymbol{r}^{\text{ini}}(t_{\text{ex}})\|^2\,, \tag{14}$$

Table 7 Closing a square: Initial pose error statistics

Measurement	Mean (cm/cm/deg)	SD (cm/cm/deg)
X-axis Position:	−0.030	0.060
Y-axis Position:	−0.017	0.022
Orientation:	0.23	0.46

Table 8 Closing a square: Average values for final pose

Measurement	Final Value (cm/cm/deg)
X-axis Position:	−7.522
Y-axis Position:	1.409
Orientation:	1.32

where $||\cdot||$ denotes the L_2 norm, and $\boldsymbol{r}^{\text{fin}}$ and $\boldsymbol{r}^{\text{ini}}$ denote the final and initial model-predicted poses produced by a plan $\mathcal{P}(t_{\text{ex}})$. Solving (14) reduces the execution time of each motion primitive instance to $t_{\text{ex}} = 2.85$ s. The resulting path is shown in Fig. 14b, with the red and blue thick curve. The shape of the path has thus been improved by only varying slightly the primitives' execution time.

This observation suggests that a direct concatenation of motion primitives may not always produce the desired result, and optimizing (tuning) the execution time may help in alleviating such discrepancies.

6.2 Experimental results

The next step is to execute the sequence in (13) on the robot. We generated *open-loop* planar position and orientation measurement data, through experiments in which the robot is configured to the respective sequence of motor gains

$$\begin{aligned}&\{\{40,40\},\ \{10,70\},\ \{40,40\},\ \{10,70\},\\ &\{40,40\},\ \{10,70\},\ \{40,40\},\ \{10,70\}\}\ .\end{aligned} \tag{15}$$

Similarly to the data collections in Sect. 5, each primitive is executed for 3 s. We collect data from a total of 60 paths, and the measured states are the planar position of the geometric center of the robot $[x_G, y_G]$ and its orientation θ. The experiments are conducted on the same rubber floor mat surface, and the initial position and orientation of the robot are manually set. Table 7 contains the initial pose error statistics.

The results are superimposed on Fig. 14. The black dashed curve denotes the average out of all 60 paths. The average final error pose is relatively small, although individual paths (plotted in magenta) can deviate significantly. Table 8 contains the final pose error statistics. The deviations in the position along the y-axis, and in the orientation remain small.

As a final remark, it can be readily verified from Fig. 14b that the SFM template is able to predict fairly well the average of the experimental data when we add the primitives' execution time as an additional model parameter. Yet, individual paths demonstrate very high variability, which is not captured by the parameters identified in Table 6. One way to capture the variability in individual paths is to use a stochastic extension of the SFM template, by treating some of its parameters as random variables. To estimate these parameters, we have proposed in Karydis et al. (2013) a general framework for probabilistically validating (stochastic) models based on available experimental data.

7 Conclusions

The Switching Four-bar Mechanism (SFM) is a low-complexity kinematic model able to capture and reproduce the horizontal-plane behavior of small legged robots, such as the OctoRoACH. It comprises four stick legs rotating in pairs, forming an alternating pair of four-bar linkages, each of which can be fully described by a single degree of freedom. Simulations show that the model is able to create path profiles that are in accordance with the oscillatory path profiles observed in the bio-inspired OctoRoACH robot while operating at low crawling speeds.

Generating various trajectories is achieved by properly tuning a small set of physically intuitive model parameters. Given a path produced by the robot, we are able to reproduce it with the proposed model by solving a constrained optimization problem which identifies the model parameter values for achieving the best match between an observed behavior of the robot and the output of the model.

The above procedure allowed us to *construct a table that links robot inputs to model inputs*. We considered three types of motion primitives comprising straight-line paths, clockwise and counter-clockwise turns, constructed nine curvature-parameterized instances of them, and established the correspondence between the motor gains of the actual platform and model parameters. We further showed that the template is able to predict the motion of the robot when it follows a square path consisting of concatenations of the constructed primitives. Its low complexity and good fit to the experimental data support that the proposed model has the potential of generalizing well in predicting the robot's behavior in more involved planning scenarios.

The proposed model cannot capture dynamic modes of operation—like bounding gaits—as well as high-speed behaviors. In these operation regimes, dynamic templates such as the SLIP or LLS models appear to be more appropriate. The hypothesis that the SFM model can serve as a template for miniature legged robots under specific operational conditions is empirically refutable. The efficacy of the proposed

model as a template needs to be tested further against data from other miniature legged robots, like OctoRoACH.

Acknowledgments This work is supported in part by NSF under Grants IIS-1350721, CMMI-1130372, and CNS-1035577, and by ARL MAST CTA # W911NF-08-2-0004.

Appendix: Terminology

For the convenience of the reader, we present here a collection of the terms used in this paper.

Abduction	Motion of the legs on the coronal plane that moves them farther from the center of the body
Adduction	Motion of the legs on the coronal plane that brings them closer to the center of the body
Contralateral	On opposite sides
Coronal	Vertical plane dividing a body into the front and back halves
Cycle	Periodic motion of the legs
Duty factor	Percentage of the total cycle during which a particular leg touches the ground
Gait	Pattern of movement of the legs
Ipsilateral	On the same side
Protraction	Motion of the legs on the sagittal plane that moves them further away from the center of the body
Retraction	Motion of the legs on the sagittal plane that brings them closer to the center of the body
Sagittal	Vertical plane dividing a body into a right and left half
Stance phase	Portion of the cycle during which a particular leg touches the ground
Swing phase	Portion of the cycle during which a particular leg is lifted off the ground and moves forward

References

Baisch, A. T., Sreetharan, P., & Wood, R. J. (2010). Biologically-inspired locomotion of a 2g hexapod robot. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, (pp. 5360–5365).

Birkmeyer, P., Peterson, K., & Fearing, R. S. (2009). DASH: A dynamic 16g hexapedal robot. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Saint Louis, MO, (pp. 2683–2689).

Blickhan, R., & Full, R. J. (1987). Locomotion energetics of ghost crab: II. Mechanics of the center of mass during walking and running. *Journal of Experimental Biology*, *130*(1), 155–174.

Blickhan, R., & Full, R. J. (1993). Similarity in multilegged locomotion: Bouncing like a monopode. *Journal of Comparative Physiology A: Neuroethology, Sensory, Neural, and Behavioral Physiology*, *173*, 509–517.

Cavagna, G. A., Heglund, N. C., & Taylor, C. R. (1977). Mechanical work in terrestrial locomotion: Two basic mechanisms for minimizing energy expenditure. *American Journal of Physiology- Regulatory, Integrative and Comparative Physiology*, *233*, 243–261.

Cham, J. G., Bailey, S. A., Clark, J. E., Full, R. J., & Cutkosky, M. R. (2002). Fast and robust: Hexapedal robots via shape deposition manufacturing. *The International Journal of Robotics Research*, *21*(10–11), 869–882.

Choset, H., Burgard, W., Hutchinson, S., Kantor, G., Kavraki, L. E., Lynch, K., et al. (2005). *Principles of robot motion: Theory, algorithms, and implementation*. Cambridge, MA: MIT Press.

Delmotte, F., Mehta, T., & Egerstedt, M. (2008). A software tool for hybrid control. *IEEE Robotics Automation Magazine*, *15*(1), 87–95.

Do Carmo, M. P. (1976). *Differential geometry of curves and surfaces*. Englewood Cliffs, NJ: Prentice-Hall.

Frazzoli, E., Dahleh, M. A., & Feron, E. (2005). Maneuver-based motion planning for nonlinear systems with symmetries. *IEEE Transactions on Robotics*, *21*(6), 1077–1091.

Full, R., & Koditschek, D. (1999). Templates and anchors: Neuromechanical hypotheses of legged locomotion on land. *Journal of Experimental Biology*, *202*, 3325–3332.

Garcia Bermudez, F., Julian, R., Haldane, D., Abbeel, P., & Fearing, R. (2012). Performance analysis and terrain classification for a legged robot over rough terrain. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura, Algarve, Portugal, (pp. 513–519).

Haldane, D. W., Peterson, K. C., Bermudez, F. L. G., & Fearing, R. S. (2013). Animal-inspired design and aerodynamic stabilization of a hexapedal millirobot. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, (pp. 3279–3286).

Hoffman, K., Wood, R. (2010). Towards a multi-segment ambulatory microrobot. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, (pp. 1196–1202).

Holmes, P., Full, R. J., Koditschek, D. E., & Guckenheimer, J. (2006). The dynamics of legged locomotion: Models, analyses, and challenges. *SIAM Review*, *48*(2), 207–304.

Hoover, A. M., Steltz, E., Fearing, R. S. (2008). RoACH: An autonomous 2.4g crawling hexapod robot. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nice, France, (pp. 26–33).

Hoover, A. M., Burden, S., Fu, X. Y., Sastry, S., & Fearing, R. S. (2010). Bio-inspired design and dynamic maneuverability of a minimally actuated six-legged robot. In: *Proceedings of the IEEE International Conference on Biomedical Robotics and Biomechatronics*, Tokyo, Japan, (pp. 869–876).

Jindrich, D., & Full, R. (1999). Many-legged maneuverability: Dynamics of turning in hexapods. *The Journal of Experimental Biology*, *202*, 1603–1623.

Karydis, K., Poulakakis, I., & Tanner, H. G. (2012). A switching kinematic model for an octapedal robot. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura, Algarve, Portugal, (pp. 507–512).

Karydis, K., Poulakakis, I., & Tanner, H. G. (2013). Probabilistic validation of a stochastic kinematic model for an eight-legged robot. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, (pp. 2562–2567).

Kim, S., Clark, J. E., & Cutkosky, M. R. (2006). iSprawl: Design and tuning for high-speed autonomous open-loop running. *The International Journal of Robotics Research*, *25*(9), 903–912.

Kohut, N., Hoover, A., Ma, K., Baek, S., & Fearing, R. (2011). MEDIC: A legged millirobot utilizing novel obstacle traversal. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, Shanghai, China, (pp. 802–808).

Lambrecht, B., Horchler, A. D., Quinn, R. (2005) A small, insect-inspired robot that runs and jumps. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, (pp. 1240–1245).

LaValle, S. M. (2006). *Planning algorithms*. Cambridge: Cambridge University Press.
Li, C., Hoover, A. M., Birkmeyer, P., Umbanhowar, P. B., Fearing, R. S., Goldman, D. I. (2010). Systematic study of the performance of small robots on controlled laboratory substrates. In: *Proceedings of the SPIE Conference on Micro- and Nanotechnology Sensors, Systems, and Applications II*, Orlando, FL, (vol. 7679, pp. 76,790Z–13).
Martin, P., Johnson, E., Murphey, T., & Egerstedt, M. (2011). Constructing and implementing motion programs for robotic marionettes. *IEEE Transactions on Automatic Control*, *56*(4), 902–907.
Mongeau, J.-M., McRae, B., Jusufi, A., Birkmeyer, P., Hoover, A. M., Fearing, R., et al. (2012). Rapid inversion: Running animals and robots swing like a pendulum under ledges. *PLoS ONE*, *7*(6), e38003.
Morrey, J. M., Lambrecht, B., Horchler, A. D., Ritzmann, R. E., & Quinn, R. D. (2003). Highly mobile and robust small quadruped robots. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Las Vegas, NV, (vol. 1, pp. 82–87).
Norton, R. (2008). *Design of machinery*. New York, NY: McGraw Hill.
Proctor, J., & Holmes, P. (2008). Steering by transient destabilization in piecewise-holonomic models of legged locomotion. *Regular and Chaotic Dynamics*, *13*(4), 267–282.
Pullin, A., Kohut, N., Zarrouk, D., & Fearing, R. (2012). Dynamic turning of 13 cm robot comparing tail and differential drive. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, Saint Paul, MN, (pp. 5086–5093).
Qian, F., Zhang, T., Li, C., Masarati, P., Hoover, A., Birkmeyer, P., Pullin, A., Fearing R., Goldman D.I. (2012). Walking and running on yielding and fluidizing ground. In: *Proceedings of Robotics: Science and Systems*, Sydney, Australia.
Schmitt, J., & Holmes, P. (2000a). Mechanical models for insect locomotion: Dynamics and stability in the horizontal plane-II application. *Biological Cybernetics*, *83*(6), 517–527.
Schmitt, J., & Holmes, P. (2000b). Mechanical models for insect locomotion: Dynamics and stability in the horizontal plane I theory. *Biological Cybernetics*, *83*(6), 501–515.
Seipel, J. E., Holmes, P. J., & Full, R. J. (2004). Dynamics and stability of insect locomotion: A hexapedal model for horizontal plane motions. *Biological Cybernetics*, *91*(2), 76–90.
Spence, A. J., Revzen, S., Seipel, J. E., Mullens, C., & Full, R. J. (2010). Insects running on elastic surfaces. *Journal of Experimental Biology*, *213*, 1907–1920.
Stoker, J. J. (1969). *Differential geometry*. New York, NY: Wiley-Interscience.
Wood, R., Avadhanula, S., Sahai, R., Steltz, E., & Fearing, R. (2008). Microrobot design using fiber reinforced composites. *Journal of Mechanical Design*, *130*(5), 1–11.
Yumaryanto, A., Ana, J., & Lee, S. (2006). A cockroach-inspired hexapod robot actuated by LIPCA. In: *Proceedings of the IEEE Conference on Robotics, Automation and Mechatronics*, Bangkok, Thailand, (pp. 1–6).
Zarrouk, D., & Fearing, R. (2012). Compliance-based dynamic steering for hexapods. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura, Algarve, Portugal, (pp. 3093–3098).
Zarrouk, D., Pullin, A., Kohut, N., & Fearing, R. (2013). STAR—a sprawl tuned autonomous robot. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, (pp. 20–25).

Konstantinos Karydis received with honors his Eng. Diploma in mechanical engineering from the National Technical University of Athens, Greece, in 2010. He is currently a PhD Candidate in the Department of Mechanical Engineering at the University of Delaware. His research interests include modeling and control of nonlinear and stochastic systems, motion planning, as well as design and fabrication of robotic legged platforms. Mr. Karydis is the recipient of the 2010 Helwig Fellowship offered by the Department of Mechanical Engineering at the University of Delaware, and is a student member of IEEE.

Yan Liu received her BSc in mechanical engineering Harbin Institute of Technology, China, in 2010. In 2011 she joined the graduate program at the University of Delaware where she spend two years as a research assistant. Her research interests are in the area of robotic legged locomotion modeling and control.

Ioannis Poulakakis received his Ph.D. in Electrical Engineering (Systems) from the University of Michigan, MI in 2009. From 2009 to 2010 he was a post-doctoral researcher with the Department of Mechanical and Aerospace Engineering at Princeton University, NJ. Since September 2010 he has been with the Department of Mechanical Engineering at the University of Delaware, where he is currently an Assistant Professor. His research interests lie in the area of dynamical systems and control with applications to robotic systems, particularly to dynamically dexterous legged robots. Part of his recent work deals with aspects of networked decision making. Dr. Poulakakis received the National Science Foundation Career Award in 2014 for his research on robotic legged locomotion.

Herbert G. Tanner received his Eng. Diploma and his Ph.D. in mechanical engineering from the National Technical University of Athens, Greece, in 1996 and 2001, respectively. From 2001 to 2003 he was a post doctoral fellow with the Department of Electrical and Systems Engineering at the University of Pennsylvania. From 2003 to 2005 he was an assistant professor with the Department of Mechanical Engineering at the University of New Mexico, and he held a secondary appointment with the Electrical and Computer Engineering Department at UNM. In 2008 he joined the Department of Mechanical Engineering at the University of Delaware, where he is currently an associate professor. Since 2012 he is serving as a director of the graduate certificate program in cognitive science at the University of Delaware. Dr. Tanner received the 2005 National Science Foundation Career award. He is a member of the ASME, and a senior member of IEEE. He has served in the editorial boards of the IEEE Robotics and Automation Magazine and the IEEE Transactions on Automation Science and Engineering, and the conference editorial boards of both IEEE Control Systems and IEEE Robotics and Automation Societies.