

Parametric Models for Motion Planning and Control in Biomimetic Robotics

Goran S. Đorđević, *Member, IEEE*, Milan Rašić, *Student Member, IEEE*, and Reza Shadmehr

Abstract—We describe the design, testing and tools to build parametric models of a six-legged cockroach-like robot for velocity control without precise knowledge on the robot's geometry or its inertia. Robot legs were made by Shape Deposition Manufacturing and were compliant at the “knee.” These kinds of robots usually have a limited number of actuators and a small number of low-cost sensors. Consequently, they are difficult to control with analytic models. Our goal was to design a very fast robot that could run in a straight line over short distances at a desired velocity. We incorporated such legs into a novel body design, where position and orientation of the legs were chosen to enhance static stability. Robot design proved to be robust, as the machine did not suffer any failure in over 20 000 runs. We found that body-pitch angle was a crucial parameter in the control of running speed. To control this angle, we built a parametric model that related leg orientation to pitch angle. We experimented with various leg stiffness parameters, and built a comprehensive parametric model that quantified performance as a function of this parameter, as well as body-pitch angle, ground slope, and body mass. When these parameters were optimized, the robot consistently achieved a speed of six body lengths per second, even when pulling a large load in the form of a trailer cart.

Index Terms—Legged locomotion, mobile robot motion planning, modeling.

I. INTRODUCTION

DESIGN guidelines for small and fast robots capable of fault-tolerant action in unknown and unstructured environments are very demanding. Many of these robots are legged, leading to the control problems associated with balance and locomotion. With many legs, postural stability can be attained. On the other hand, stable locomotion cannot be solved only by body design. It requires a reconfigurable controller that can handle locomotion with variations in ground slope, payload mass, speed, etc. In this paper, we describe tools to build a controller for such a running robot, using only experimental data without precise knowledge of robot dynamics and kinematics. Of particular interest are robots that mimic locomotion and running behavior of arthropods.

Manuscript received October 23, 2003; revised April 27, 2004. This paper was recommended for publication by Associate Editor K. Lynch and Editor I. Walker upon evaluation of the reviewers' comments. This work was supported in part by the U.S. Department of Defence, Office of Naval Research, under MURI Grant N00014-98-1-0669.

G. S. Đorđević was with the Department of Biomedical Engineering, Johns Hopkins School of Medicine, Baltimore, MD 21205 USA. He is now with the Faculty of Electronics Engineering, University of Niš, 18000 Niš, Serbia and Montenegro (e-mail: gorandj@elfak.ni.ac.yu).

M. Rašić is with the University of Niš, 18000 Niš, Serbia and Montenegro (e-mail: mrasic@elfak.ni.ac.yu).

R. Shadmehr is with the Department of Biomedical Engineering, Johns Hopkins School of Medicine, Baltimore, MD 21205 USA (e-mail: shadmehr@jhu.edu).

Digital Object Identifier 10.1109/TRO.2004.833820

During two consecutive steps of a running animal, a phase relationship between vertical and horizontal components of the ground-reaction forces (GRFs) and sinusoidal horizontal force [3], independently of leg number and morphologies, result in an “M-shaped” GRF [14]. Based on an analogy with a spring-mass-damper inverted pendulum, it is thought that all running animals use a bouncing monopod strategy [6], [29]. A set of quadruped and hexapod robots have been designed, based on the idea that the leg in locomotion should act as an inverted spring-mass pendulum [1], [2], [9], [11], [26], [35]. Such robots maintain dynamic stability by engaging solely their mechanical properties [21], [23]. However, control of these systems is often difficult. In biology [14], [17], [33], arthropods typically combine dynamic stability with static stability of the body by having more than two legs contact the ground at a time. When the body is disturbed, the legs act as a passive spring-mass system [8].

Although it is clear how a biomimetic legged robot should react with its environment while running, the design of such systems remains a challenge. Available power sources and actuators are less efficient than what is observed in nature. The size-to-payload ratio of actuators, their drives, and energy storage are high, compared with muscles. One approach is to make a functional copy of the animal [5], [24], [27], [31]. Designing controllers that deliver some of the versatility inherent in the animal motor-control system [19], [21], [22], especially regarding motor learning, adaptation, and motion planning, might be a useful first step. The biomimetic principles in the design of a running insect-like robot and its controller have already proven to be a promising design guideline.

An insect of choice for building fast-running biomimetic robots is the cockroach. It has a relatively simple motor-control system and yet it displays extraordinary speed and dexterity, even over rough terrain. In fast runs, it maintains its center of gravity low to support dynamic stability. The oscillations of the pitch, yaw, and roll rotational movements of the body are modest, thus saving energy [20]. For a robust run, each leg of the tripod in the cockroach does not need firm contact with the ground. Instead, it uses kinetic energy to bridge from one firm contact to another. The controller design of a cockroach-like robot might be based on the observation that in walking and running, a cockroach uses a tripod gait with one middle leg on one side of the body along with the front and hind legs on the other side acting as virtual legs in an equivalent biped run. Even when negotiating a curve, the stable tripod gait may be a proper walking policy [20], [23]. General parameters that describe the tripod gait are stride period (SP), which is the time interval between two activations of one tripod, and duty factor (DF), which is the percentage of time, with respect to half of the SP,

that the legs are actively producing force. Another parameter that may be important in velocity control is orientation of GRFs within a tripod. The design of most successful biomimetic robots today engage these parameters.

One of the first successful autonomous legged machines, RHex [31], demonstrated mobility in general tasks, such as uneven ground and different body weights [30]. RHex is an autonomous, untethered, compliant legged hexapod robot weighing 8 kg and 0.5 m long. The robot has six independently actuated legs that rotate with one degree of freedom (DOF). The sticks that make up the legs are compliant. The three-by-three legs rotate in a clock-driven fashion, producing an alternate tripod gait [32]. The existence of the tripod ensures static mobility. Presently, it is capable of achieving five body lengths per second [30]. Its size of nearly half a meter makes it capable of running difficult terrain, even climbing stairs [10]. Whegs [25] is another robot that is half a meter long and is able to run three body lengths per second. It uses three-spoke legs driven by one motor, managing to climb up heights of one and a half of its spoke length. With six Y-shaped legs that rotate, Whegs can engage alternate tripod gait patterns [27]. One of its derivatives, Mini-Wheg, much smaller in size, can run at speeds over ten body lengths per second [38]. Both RHex and Whegs robots control their velocity by the rotation frequency of legs or whegs, and, perhaps, by the phasic relationship between the legs.

The Sprawl series of robots has several derivatives, such as Sprawlita, Sprawllette, and the newest one, fully autonomous iSprawl [34]. Sprawl depends on Shape Deposition Manufacturing (SDM) technology [11], where each leg is simultaneously machined and assembled. The leg material is urethane, and it incorporates a flexure and an air piston. Flexure in the leg provides one passive DOF, acting like an uncontrolled knee in bipedal running, whereas the piston acts like a thruster, providing one active DOF [14]. In the design of our robot, we used SDM legs similar to those incorporated in Sprawl, but redesigned the body and the controller. Many versions were designed solely for experimentation purposes as a support to iterative design. Smaller in size, usually around 0.15 m and 300 g in weight, the Sprawl batch of robots achieve locomotion velocities of five body lengths per second, similar to RHex and Whegs robots. In Sprawl, one can control velocity either by changing the stride rhythm [12], or, unlike Whegs and RHex, by changing the body posture. In fact, the ability to vary body pitch makes this robot capable of climbing over taller obstacles than other running hexapod robots.

Our effort was part of a multi-university research project to build biologically inspired robots that are cheap, fast, compliant, and fault-tolerant, using mostly off-the-shelf technology. Our approach was to make efficient models of robot-environment interactions for model-based control and locomotion planning under external disturbances, such as sloped ground and added payload mass. The other group of researchers focused on developing a mechanical model of the Sprawl robots designed in common CAD software for further study and application. The two approaches should converge as the research progresses.

Control of robots made of plastics with poor tolerances is challenging. A way to proceed is to mimic a cockroach motor-control system. Its locomotion is usually explained by reflexive, spring,

and damper-like behavior of the legs [14], responsible for rapid stabilization augmented in certain directions by its motor-control system [18]. However, present understanding of this motor-control scheme does not explain adaptation to constant external disturbances, such as sloped ground or additional payload. Instead of strictly mimicking a cockroach motor-control system, we propose a human-like motor controller that will use a feed-forward predictor, a short-latency feedback system at the level of the muscles and spinal cord, and a long-latency feedback system at the level of the brain [7]. There is evidence that such a scheme can explain to some extent adaptation that occurs as humans learn internal models of action [37]. In our robot design, we assume an analogy between the short-latency feedback system and the passive properties of the robot's legs. At this stage of design, our controller did not use online feedback. The feed-forward controller relied solely on models that related velocity to body orientation, slope, and mass. These models were built from data acquired in an extensive set of experiments that varied control parameters widely and observed their effects on performance. The control parameters were flexure stiffness, SP, DF, and body-pitch angle. To build the locomotion velocity model, we used a procedure called Successive Approximations (SA) [15], [16]. Section IV details the experimental results and model construction.

At the moment, this robot is not intended for autonomous operation. Our goal was to build a robust, fast robot, and design a controller using only experimental data with no analytic modeling of the robot. In the future, we may use the same technology when scaling up the robot with the power and the controller onboard, but the agility will be certainly reduced. Electricity may come onboard in new versions of the robot, but the air tanks that were used to power this first version can not fit on such a small body. However, steering of the robot is a possible operation mode of this kind of robot, and we expect that the human operator will carry the air tank and batteries.

The paper is organized as follows. The second section details the mechanical design of the robot and sets up a basis for determining an optimal posture of the robot. The main result of the second section is the procedure for building the body-pitch angle model. The third section provides results that quantify the robot's performance as a function of variations in leg compliance, leg orientation, SP, and DF. The fourth section relates locomotion velocity as a function of body pitch, slope of the terrain, and payload mass. The modeling methodology yields the parametric velocity model based solely on experimental data. The model enabled the robot to achieve a speed of six body lengths per second with only two air valves, a notable performance among legged robots today.

II. DESIGN OF A HEXAPEDAL ROBOT

The design of a small, inexpensive, and fast biomimetic robot started in 1999 at the Stanford Center for Design Research, Stanford, CA [4]. In this paper, we used a derivative of their original robot. We used the legs made for Sprawl, except that the flexures (the compliant area at each joint) were interchangeable, allowing us to examine the role of compliance in stable locomotion. Our robot has a compact aluminum body, different from Sprawl in size,

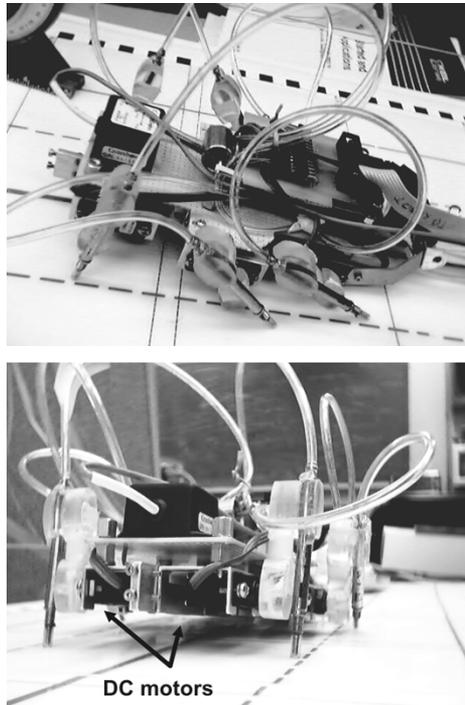


Fig. 1. Side-top and front view of the robot. Accelerometer on board measures tilt. Added weight is located just above robot's COM.

shape, material, and weight (Fig. 1). It also has six small DC servo-motors that act like hips, i.e., they control orientation of each leg in an offline fashion. The DC motors have their own servo controllers that accept serial communication for setting up the posture before the start of the run. Each leg has a one-way air piston with a reverse spring action, and a passive flexure functioning like a knee. The robot runs by alternate tripod gait. Three out of six legs make one tripod. Two tripods are controlled by two valves. Therefore, we have two control inputs. The valves are powered by a custom design interface logic connected to a parallel PC port. A PC controls the robot by commanding locomotion sequences. The air pressure comes from off board. The tripod activation is defined by SP and DF as the percentage of time that the valves are kept open during half of the SP.

Our robot, as a test platform for experimentation, has small DC motors that tether the SDM legs, similar to Sprawl robots. The motors are used only to fix the orientation of the tethered part of the legs throughout the run. We had two reasons for this. First, from a practical point of view, with the motors we had, it would be impossible to position them so quickly within the stride. Second, proper leg orientations are based on measurement of the actual body-pitch angle and ground slope. Both of them had to be measured while the robot is still, and it would take a second or two to get reliable readings from the tilt sensors. Hence, by changing the tripod gait between completion of a run and the start of another run, it is possible to download commands from the operator that steer the robot, allowing one to achieve a piecewise constant velocity. The result is a robot that runs straight ahead, pauses to change its orientation, and then continues to run. In the next two subsections, we will explain the mechanical design of the robot, and the procedure of modeling posture.

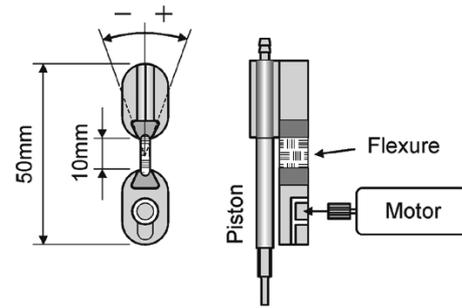


Fig. 2. Two views of leg with piston and interchangeable flexure. Lower part of the leg is tethered to the DC motor located beneath body. Legs are modular with a flexure connecting the part with a gear and the part with the piston. Flexures are made in four thicknesses, denoted Types I, II, III, and IV, ranging from 2–3 mm.

A. Mechanical Considerations and Optimal Posture

The backbone of the robot was a 2.54-mm thin aluminum bar. Six rectangular aluminum tubes were glued on two sides of the bar. Each held a DC motor (Cirrus CS-21BB) that tethered a leg. The leg design is given in Fig. 2. Two air valves (Humphrey H010E1) were placed on the back of the robot. The three-way solenoid valves were normally closed. These lightweight valves (18 g) worked under 690 kPa with low power consumption of 1.6 W on 12 V. The pistons (Festo EG-4-20-PK-2) achieved 2 cm full stroke-cycle in about 30–35 ms. Along with the flexures, they produced a force-moment couple that was transmitted to the body, producing gait. The total mass of the robot was 0.351 kg. The length of the robot was 0.144 m, denoted in Fig. 3.

The compliance of the flexure was chosen according to the mechanical properties of the body and ground. The idea was to choose a compliance that would allow the leg to bend back and return the energy while the piston was still in contact with the ground. Otherwise, the energy would be returned too soon or too late, interfering with stride rhythm. Results from bipedal walking [36], as well as studies on hexapod insects [35], suggest that the moment around the center of mass (COM) of the body should be as low as possible, in order to achieve transfer of impact energy into kinetic energy. We experimented with various flexures: Type I: 2 mm; Type II: 2.29 mm; Type III: 2.54 mm; and Type IV: 3 mm, and solved this problem empirically.

Studies on animal walking and running suggest that during locomotion, the point around which there is zero moment is kept within the footprint most of the time. That point is known as the zero-moment point (ZMP) [36]. In a tripod gait, the footprint triangle ensures stable stance, but not necessarily stable locomotion. Without an exact model of the geometry and masses, and without appropriate sensors, it is impossible to compute the location of the ZMP. Instead, we hypothesized that smooth locomotion would take place, if GRFs (in the static case) intersected at one point to form a resultant force that passed through the COM of the robot. In order to achieve this, let us briefly consider the mechanics of the robot in the static case.

The robot posture is pictured in Figs. 3 and 4. Consider one tripod, formed by legs 1, 4, and 5 in contact with the ground. The coordinate system is located at the COM (Figs. 3 and 4). Note that the COM is not located in the middle of the robot; rather it is closer to its tail. Axis Ox , an axis of longitudinal symmetry, is oriented from the tail to the head of the robot (Fig. 4). Axis Oz

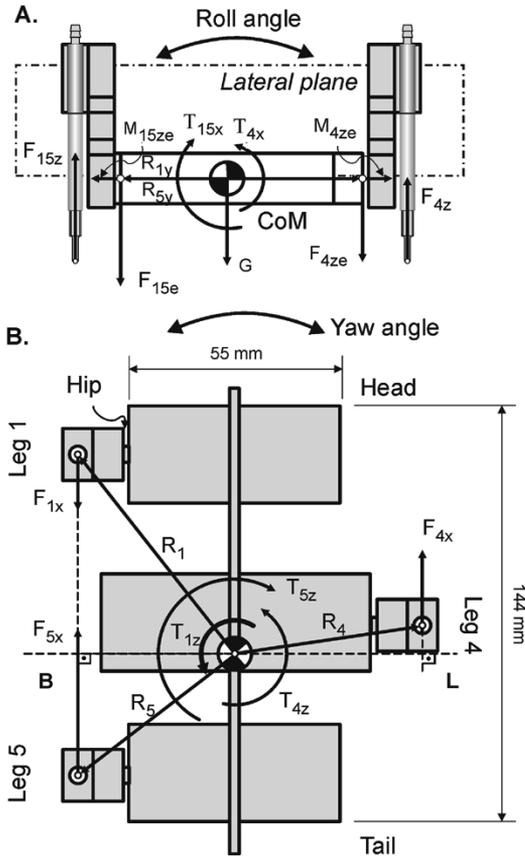


Fig. 3. (a) Rear view and (b) top view of the robot with only one tripod (legs 1, 4, and 5). The other tripod (legs 2, 3, and 6) is omitted in this drawing. GRFs are denoted as F_1 , F_4 , and F_5 .

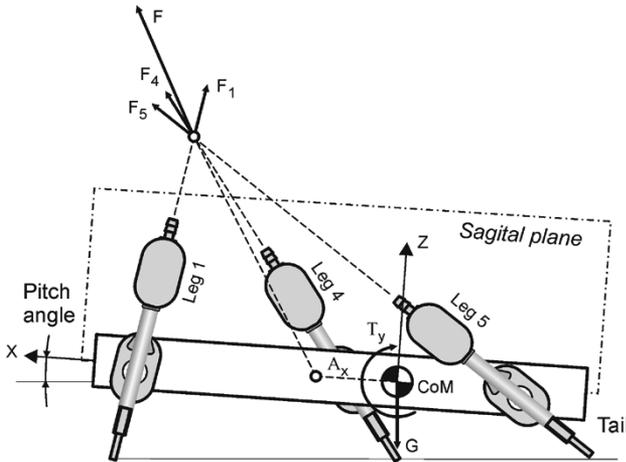


Fig. 4. Side view of the robot standing on one tripod. The angle of the resultant force determines locomotion velocity. We hypothesize that the optimal posture should have CoM on the resultant force F direction.

is perpendicular to the plane of the robot body. Axis Oy forms a right coordinate system with Ox and Oz . The xOz plane is the sagittal plane, xOy is the body plane, and yOz is the lateral plane.

Each leg in the tripod produces a GRF that forms a resultant force-moment couple. It is assumed that the GRFs reside in a plane parallel to the sagittal plane. Although the legs are compliant at the knee, the lateral compliance of the leg is neg-

ligible, compared with the forward/backward compliance; this is due to the shape of the flexure element (Fig. 2). Forces F_1 and F_5 are acting from one side of the body, while force F_4 is acting from the other side. We have plotted the projections of the GRF in the lateral plane in Fig. 3(a), and in the body plane in Fig. 3(b), whereas Fig. 4 shows projections of the GRF in the sagittal plane.

Horizontal projections of the GRFs form moments around the Oz axis. Vertical projections of the GRFs form the resultant moments around the Ox and Oy axes. Moments around the Oz axis, T_{1z} , T_{4z} , and T_{5z} , are shown in the body plane in Fig. 3(b). A nonzero resultant moment gives rise to the yaw angle. Since tripods are symmetric and act alternatively, it is reasonable to expect that a nonzero resultant moment will produce a change of yaw angle that would be cancelled by the opposing tripod. However, due to the flexures in the legs, this cancellation is never perfect, making the robot veer from a straight path, as was experimentally observed. Similarly, the net moment around the Ox axis is the result of the vertical component of the GRFs. Moments are denoted by T_{15x} and T_{4x} [Fig. 3(a)]. The resultant moment changes the roll angle during a run. This produces a wobbling from left to right. In order to keep wobbling as low as possible, it is desirable to push the middle legs further away from robot's longitudinal axis, because it is the middle leg that produces the thrust against the two legs on the other side. This also helps reduce yaw oscillations.

The net moment around the Oy axis is also a consequence of the vertical projections of GRFs (Fig. 4). Three concurrent force vectors, F_1 , F_4 , and F_5 form a resulting vector F that causes a net moment T_y around the Oy axis. If the net moment is not zero, locomotion is affected via the pitch angle, causing a flutter in this angle. For smooth locomotion, it is desirable to make the net moment around the Oy axis as low as possible. This can be achieved if the resulting force vector passes through the CoM, i.e., if distance A_x equals zero. Therefore, we placed the middle legs closer to the rear legs. We expected that the angle of the resulting force relative to the ground should determine the length of the jump, thus locomotion velocity. For example, to produce a fast run, we should set the angle near 45° . These ideas were based on a static case and might not hold during locomotion. However, further experiments showed that if we chose appropriate SP and DF, the robot ran smoothly with small deviations in the pitch angle. This suggested that the resultant force vector indeed passed near the CoM. This embraces our assumption that we can use the static condition of the optimal posture.

In the static case, the body exhibits passive stability, i.e., the body stays in balance if we have at least one tripod in contact with the ground. Even if pushed by hand, the body quickly returns to the same posture under gravity load. However, due to flexures in the legs, we do not know the exact body orientation. Tripods in alternating tripod runs must provide a uniform body transition from stride to stride, which can be defined as dynamic stability in running. Insect mechanics suggest that dynamic stability is important in hexapod locomotion, and often originates from statically stable solutions of posture [35]. Dynamic stability complements static stability at the point of transition between strides. Existence of dynamic stability provides a flow of

kinetic energy that helps when no ground contact is available during the stride. While the static stability can be attained by tripods, achieving and maintaining dynamic stability in hopping by alternate tripod activation is much more challenging, if the legs of the robot are too stiff. Embedded flexures help maintain dynamic stability, but due to unknown inertial parameters and frequency characteristics of the legs, we cannot determine sufficient conditions on dynamically stable locomotion. Furthermore, flexures might change their properties across different production batches as well as in time, temperature, payload, and other factors. A possible solution might be an empirical, nonanalytic, feed-forward model that will provide posture close to the statically optimal case, in the sense of reducing resultant torque around the COM.

B. Modeling the Pitch Angle of the Robot

The relationship between leg orientation and body-pitch angle under gravity load is a nonlinear map, mostly due to flexures in the legs. In general, six legs will produce six controllable DOFs of the robot's body. This problem is augmented by engaging synergies. First, we have three-by-three legs forming tripods, and second, they have symmetrical orientation. We simplified the search for a pitch-angle model by keeping the front legs in the fixed position (10° from vertical, tilting backward). While varying the middle and rear legs, due to compliant legs, we ensured that two constraints were satisfied: all six legs were on the ground, and the legs were oriented in such a way that the resultant torque around Oy axis at COM was near zero. The flexures used in determining the pitch-angle model were chosen among available Types III-I-II, III-II-I, and III-II-II, for front-middle-rear legs, respectively.

The experimental procedure was as follows. First, middle legs were positioned randomly, and then rear legs were positioned in accordance with constraints. Next, body-pitch angle was measured by a tilt sensor (ADXL 202JC Evalboard, Crossbow). Measured leg-orientation angles as well as body-pitch angle were recorded for each trial. The data was then divided into two groups. One was selected for modeling and the other was used for cross-validation. Next, results were sorted according to body pitch in ascending order, producing an array of body-pitch angles. Experiment ordinal numbers were rearranged accordingly. For fitting with polynomials, we mapped a rearranged vector of experiment ordinals to the symmetrical interval by centering it at zero mean, and scaled it to the unit standard deviation. The new independent variable, denoted η , is called the normalized experiment index.

Body-pitch angles between 3.5° and 7.0° were used in modeling. Data for cross-validation included pitch angles ranging from 3° to 7.75° .

The pitch-angle data for flexure combination III-I-II are shown with circles in Fig. 5, whereas the leg-angle data, given in numbered steps from the neutral position, are given in Fig. 6. As the front legs did not change orientation, only data for the middle and rear legs are given. Note that abscissae in Figs. 5 and 6 represent the normalized experiment index η . That means that during the experiment, we know the exact orientations of the legs, as well as the corresponding body-pitch angle. This knowledge can be generalized as follows. Since the body-pitch

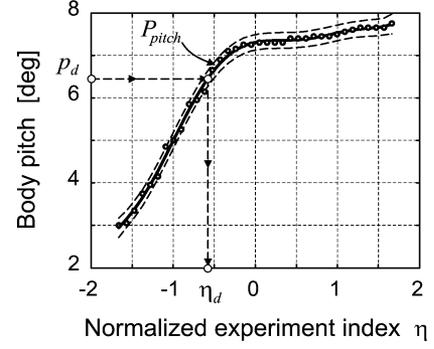


Fig. 5. Measured pitch angles, dark circles, and fitting polynomial P_{pitch} with confidence limits (dashed lines).

angle characteristic is a monotonic function of η , given the desired pitch angle, we can determine from what experiment it originates, i.e., what leg orientations are needed to achieve such a pitch angle. Even more, we can assume that mapping is continuous, meaning that leg-orientation values that lie between any two experiments will lead to body-pitch angle also between the two corresponding angles. Therefore, fitting of the experimental data on the pitch angle and middle and rear leg angles with respect to experiment number η was performed by polynomials. The polynomial that fits the pitch-angle data is given by

$$P_{Pitch} \triangleq 0.084\eta^7 + 0.003\eta^6 - 0.594\eta^5 + 0.179\eta^4 + 1.381\eta^3 - 1.227\eta^2 + 0.396\eta + 7.305. \quad (1)$$

The polynomials that fit data from the legs orientation, with respect to a normalized experiment number η and confidence limits, shown in Fig. 6, are given by

$$\begin{aligned} P_{leg3} &\triangleq 0.195\eta^3 + 1.052\eta^2 - 16.202\eta + 79.186 \\ P_{leg4} &\triangleq -0.647\eta^3 - 0.661\eta^2 + 16.052\eta + 169.601 \\ P_{leg5} &\triangleq 0.486\eta^3 + 0.392\eta^2 - 14.855\eta + 84.660 \\ P_{leg6} &\triangleq -0.596\eta^3 - 0.299\eta^2 + 14.978\eta + 163.291. \end{aligned} \quad (2)$$

Front legs, left and right, were at fixed position and were not included in modeling.

The resulting model can be used in two ways. First, for a desired pitch, we can determine leg orientations. Given the desired pitch angle, p_d , shown on the pitch-angle axis in Fig. 5, we calculated the corresponding experiment index by finding root η_d of the $P_{Pitch} - p_d = 0$ on the given interval, also shown in Fig. 5. Calculating the values P_{leg3} through P_{leg6} of the polynomials P_{leg3} through P_{leg6} at η_d , we obtained desired leg angles in steps from the neutral position, shown in Fig. 6. Second, we computed body-pitch angle given the leg orientations. The only limitations are that legs should be on the ground and oriented so to maintain the optimal posture. Note that normalized experiment index is used to match the two types of submodels, one polynomial that fits body-pitch data (1), and four polynomials that fit leg orientation data (2).

Next, we tested the pitch-angle model against cross-validation data. This data set was collected along with the data used for modeling, but on a different grid of body-pitch angles. The model computed leg angles for arbitrary body pitch, shown by a

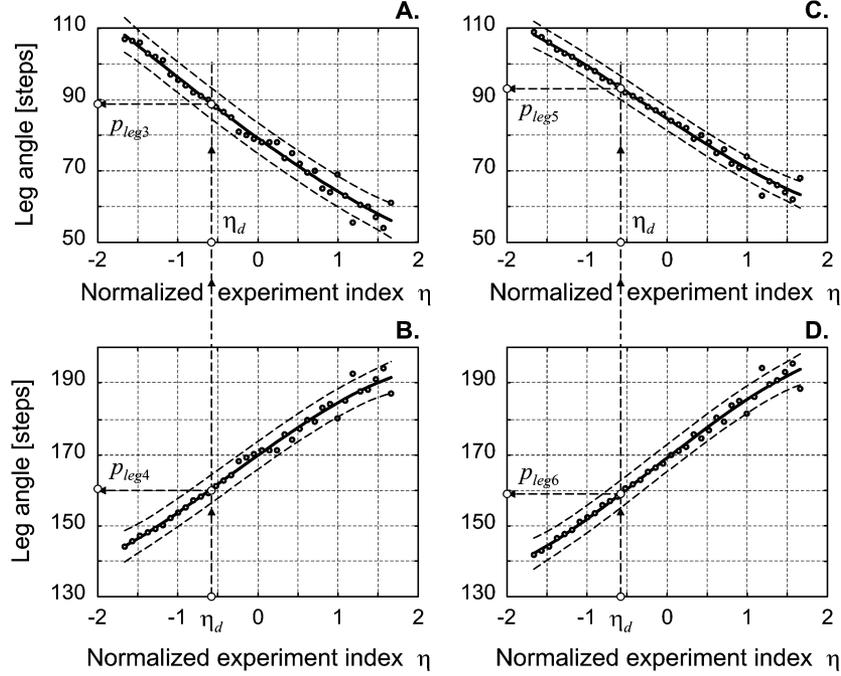


Fig. 6. Measured leg angles, dark circles, and fitting polynomials with confidence intervals [(a): Third leg. (b): Fourth leg. (c): Fifth leg. (d): Sixth leg]. Leg angles are measured by motor-controller steps (0.72° each) from the neutral leg position, the 128th step.

solid line in Fig. 7. On the same diagrams we showed cross-validation data (dark circles). The model captured the nonlinear nature of the inverse kinematics well, along with the flexures in the legs. Furthermore, the model extrapolated well within a certain range of pitch angles. At the interval of modeling, the polynomial that approximated the pitch angle had a positive first derivative. It means that we did not have multiple leg-orientation solutions for one body-pitch angle. This pitch-angle model of the robot was used in the remaining experiments.

The identical procedure resulted in pitch-angle models for other types of flexure combinations. For example, Type III-II-I has the following pitch-angle model:

$$\begin{aligned}
 P_{\text{PITCH}} &\triangleq -0.128\eta^3 + 0.0301\eta^2 + 1.562\eta + 4.982 \\
 P_{\text{LEG3}} &\triangleq -1.244\eta^2 - 5.578\eta + 97.130 \\
 P_{\text{LEG4}} &\triangleq 1.244\eta^2 + 5.578\eta + 153.870 \\
 P_{\text{LEG5}} &\triangleq -6.633\eta + 109 \\
 P_{\text{LEG6}} &\triangleq +6.6332\eta + 140
 \end{aligned} \quad (3)$$

whereas the Type III-II-II flexure combination has model

$$\begin{aligned}
 P_{\text{PITCH}} &\triangleq -0.234\eta^3 - 0.198\eta^2 + 1.382\eta + 5.982 \\
 P_{\text{LEG3}} &\triangleq -6.116\eta + 106.889 \\
 P_{\text{LEG4}} &\triangleq 6.1162\eta + 152.111 \\
 P_{\text{LEG5}} &\triangleq -5.477\eta + 111 \\
 P_{\text{LEG6}} &\triangleq 5.4772\eta + 138.
 \end{aligned} \quad (4)$$

The pitch-angle models (1)–(4), by maintaining the optimality condition with respect to resultant force and the robot’s COM, improves the conception of Sprawlita robot control. By using these models later in experiments to determine leg orientations, given the desired pitch angles for the specific

flexure combinations, we nearly doubled reported locomotion performance of the original Sprawl robot.

III. FROM EXPERIMENTS TO PARAMETRIC MODEL-BASED CONTROL OF ROBOT LOCOMOTION

A number of factors can affect locomotion velocity. Some of these factors are leg orientations with respect to the body, elasticity of joint flexures, SP, ground slope, and payload mass. In order to arrive at a comprehensive mathematical model, we need precise data on both the robot and the environment. When using this type of model, we need additional sensors on any of those parameters, which significantly increase the complexity and the cost of the robot. Furthermore, deriving precise mathematical models is probably equally as time consuming as brute-force experimentation. Instead, we propose a modeling tool that will compile experimental data resulting in a very small and computationally inexpensive model of body-pitch angle under gravitational load, as well as a model of locomotion velocity. Our experimentation methodology follows a stepwise approach to understanding the robot’s performance and to revealing crucial control parameters. We ran an extensive set of experiments to observe how these parameters affected locomotion. The robot and the track are shown in Fig. 8. The track was a 2.5×0.35 m board, padded with a silicon film to increase friction. To measure velocity, the robot dragged a trailer with a PC mouse. The robot and the mouse were connected with one revolute joint.

A. Probing the Robot and Choosing Flexures

We quantified performance of the robot as a function of SP, DF, body-pitch angle, slope of the track, and payload mass. SP was $\{100, 175, 250\}$ ms; DF was $\{20, 30, 40, 50\}\%$; body-pitch angle was $\{0.5, 1.5, 2.5, 3.5\}^\circ$; slope of the

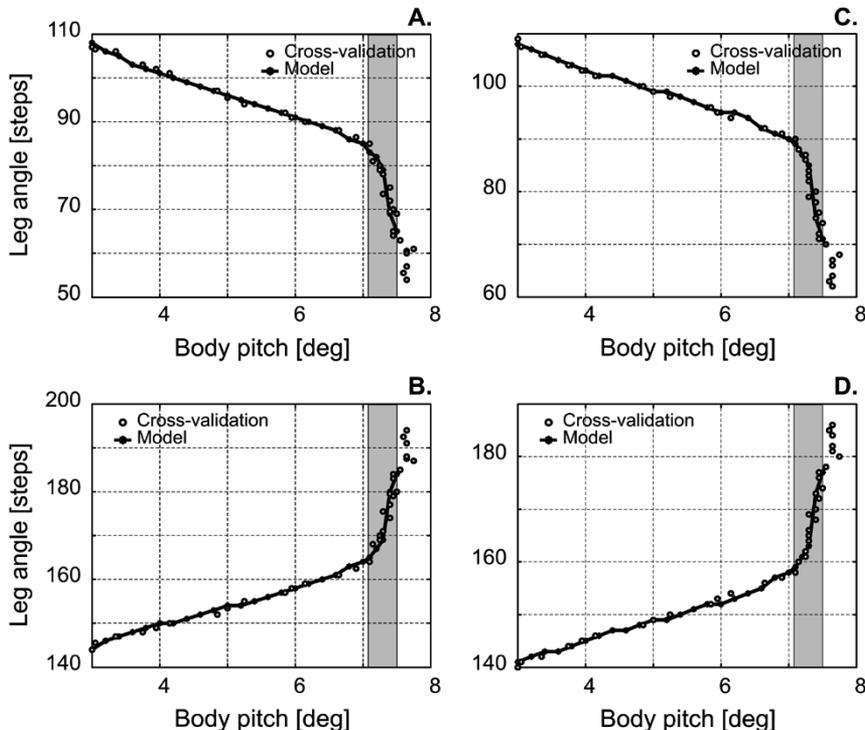


Fig. 7. Cross-validation test for direct kinematics model [(a): Third leg. (b): Fourth leg. (c): Fifth leg. (d): Sixth leg]. Outputs of the model for legs 3–6 were calculated in the range $3\text{--}7^\circ$ with step 0.2° , straight-line segments. The data from the experiment, dark circles, belong to the same range. We also verified extrapolation in the range $7.1\text{--}7.5^\circ$ with step 0.1° , gray regions.

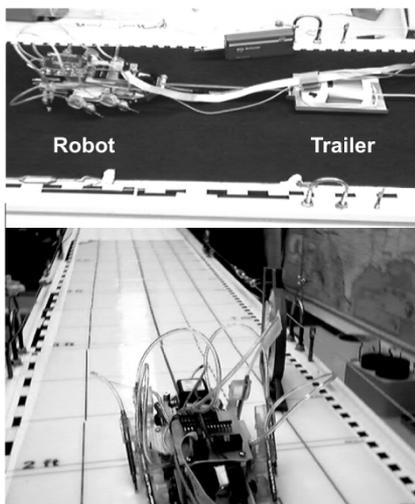


Fig. 8. Robot on the track. Tilt angle of the track could be up to $\pm 30^\circ$ along longer dimension. The trailer of the robot had a position sensor.

track was $\{0, 2.5, 5.0, 7.5, 10\}^\circ$; and payload mass was $\{0, 25, 50, 75\}$ g, located just above the COM. Two consecutive runs were performed with each combination of the factors, producing approximately 2000 runs.

The fastest run was three body lengths per second, i.e., 0.42 m/s, achieved with 3.5° of body-pitch angle. We found that the pitch angle of the body and a combination of SP and DF were two factors that significantly affected the velocity of the robot. We also found that the body-pitch angle was the factor that most significantly affected velocity. Other factors that had a significant effect were SP and DF. Near-optimal SP was in the in-

terval $[100, 150]$ ms, with the DF 35% per tripod and pitch angle greater than 3° .

We next examined how the stiffness of the flexures influenced performance. In cockroaches, front legs are stiffer than other legs. This prevents tumbling. Rear legs are softer, and this allows storage of impact energy. As shown in Fig. 2, the legs have interchangeable flexures. In this experiment, robot had to run for 3 s. To get a close look at the robot's dynamics, we removed the mouse trailer so as to not interfere with the motion of the robot. Rather, we measured only the final position achieved by the robot, and assumed the robot maintained a constant velocity.

For each set of factors, we performed three runs. From the results of the previous experiment, we chose the body-pitch angle to be 3.5° because it produced the fastest run. In this condition, the front legs were oriented 9° toward middle legs; the middle and rear legs were rotated toward the forward legs at 15° and 37° , respectively. These angles are with respect to a neutral position. A neutral position of the legs assumes that they are parallel to each other and orthogonal to the body. We considered the effect of all combinations of flexure Types I-IV: $\{2, 2.29, 2.54, 3\}$ mm per one tripod ($4^3 = 64$ combinations in total), given the following factors: SP was $\{100, 175, 250\}$ ms; DF per tripod was $\{30, 40, 50\}\%$. Both tripods had the same flexure combinations.

The total number of runs in this experiment was 576. The distance traveled for a 3-s run represented the performance. We observed that the robot was faster with thinner, softer flexures. Although the softer flexures produced faster runs, they often caused unrecoverable jamming of the front legs that prevented further runs. The better choice seemed to be thicker flexures in the front legs to ensure stability, despite a reduction in speed. The choice of the flexures in the middle and rear legs followed

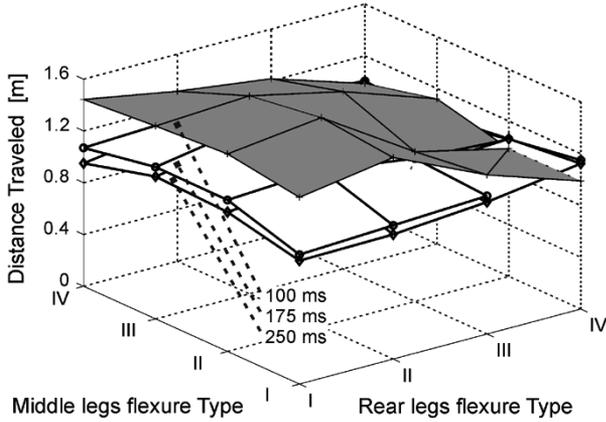


Fig. 9. Set of experimental data for Type III flexures in front legs. Three planes are shown for three SPs: 100, 175, and 250 ms. Middle and rear leg flexures are denoted on the horizontal lines as Types I, II, III and IV. Distance in meters traveled in each run is on the vertical axis.

TABLE I
VELOCITY MEAN AND STANDARD DEVIATION ACHIEVED BY THREE MOST SUCCESSFUL FLEXURE COMBINATIONS

Types:	III-I-II	III-II-I	III-II-II
Vel \pm SD [m/s]	0.485 \pm 0.018	0.501 \pm 0.019	0.517 \pm 0.017

the observation that the robot performed irregularly if the differences between flexure thicknesses were too high. One of the best sets of runs was with a Type III flexure in the front legs. In this condition, we found variation in performance when changing SPs and middle and rear leg flexures from Type I to Type IV. The best result was for shorter SPs (Fig. 9). We ran additional tests with 100 ms SP only. The results with the three flexures are given in Table I. The first ordinal number stands for front legs, the second ordinal stands for middle legs, and the last one stands for rear legs. Performance was calculated in ten consecutive runs. For the flexure combination of Type III-II-II the robot ran 0.517 m/s, i.e., 3.6 of its body lengths per second, and the difference was not significant compared with the other two sets of flexures.

B. Experimentation Methodology

Having a model of inverse kinematics, and a reasonable understanding of what flexures should be used with each leg, we proceeded with experiments to make a model that related velocity of the robot as a function of three major control parameters: body-pitch angle, slope of the ground, and the weight of the robot. We also tried to understand whether different flexures were particularly well-suited for running uphill or carrying loads.

We organized the final experiment in two steps. First, on a coarse grid of the parameter space, we found the region where performance appeared to peak. Second, we carried out a detailed experiment on that region of parameter space. A typical run, with the optimal posture and properly selected flexures on a 100 ms SP is shown in Fig. 10. In each trial, the robot ran for 2 s. The run was repeated three times for each combination of the parameters. In all, we ran 15 840 trials. We estimated that during

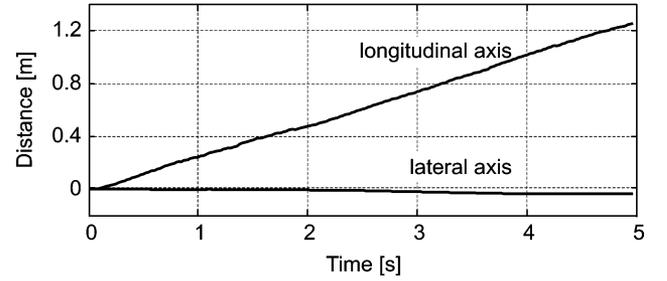


Fig. 10. Typical run for 5 s recorded along longitudinal and lateral axes.

these trials, the legs were exposed to more than 400 000 impact cycles. This certainly challenged the SDM design of the legs.

Initially, we varied combinations of flexure Types III-I-II, III-II-I, III-II-II, track slope $\{0, 6, 9\}^\circ$, weight $\{10, 20, 30, 40, 50\}$ g, and SP $\{100, 110, 120, 130, 140, 150\}$ ms, while DF was kept at 35%. We also varied the body-pitch angle at the interval $[3.5, 6.5]^\circ$. Altogether, we ran 8910 experiments, by repeating three times each combination of the parameters. As we had three flexure combinations, we made three different models of kinematics and locomotion velocity. Based on models in (1)–(4), we found the leg orientations for a number of desired body-pitch angles. In all three combinations, the peak locomotion velocity was nearly 0.8 m/s, which is approximately 5.5 body lengths per second. Note that this velocity is significantly higher than in our earlier results. Therefore, tuning body-pitch angle and applying the best combination of flexures had a significant impact on performance.

Repeatability of performance was close to one body length on each eight body lengths traveled, i.e., less than 10% of the length of the run. We noticed that on the flat track, all leg combinations performed similarly. However, major differences were apparent in uphill running. On the slope of 6° and 9° , the body-pitch angle, as well as SP and the weight, significantly influenced velocity. In general, softer flexures were more sensitive in the higher slope, degrading the optimal tripod orientation calculated from the model. We also observed that on the horizontal track, payload did not significantly influence velocity of motion, as it did on the slope. After evaluating the data, we observe that the second combination of flexures, Type III-II-II, provided the most consistent results. With this combination, the robot ran smoothly, without tumbling. The high body-pitch angle also embraced the natural body posture of the robot under heavier payloads.

Next, we ran the robot with Type III-II-II flexures only. To derive the final velocity model with the payload mass, the body-pitch angle, and slope angle as parameters, we collected data by varying those parameters on a grid. Payload varied as $\{10, 20, 30, 40, 50\}$ g, as well as ground-slope angles $\{0, 2, 4, 6, 8, 9, 10\}^\circ$. The body-pitch angle was chosen among $\{3.5, 3.8, 4.1, \dots, 6.5\}^\circ$. This step was limited by the resolution of the DC motors that orient the legs. Finally, the SP was $\{100, 110, 120, \dots, 150\}$ ms. The DF remained at 35%. We again had three trials for each set of parameters, a total of 6930 trials. Two trials were used for model construction, whereas the third set of data was used for cross-validation. Data to be used in the modeling are stored in the four-dimensional (4-D) matrix

of $[5 \times 7 \times 11 \times 6]$ dimension. Each of these dimensions corresponds to the number of parameter variations. We observed a slightly increased maximum velocity of 0.86 m/s, i.e., six body lengths per second. This was attained at 5° body-pitch angle, with 10 g of payload mass on the horizontal track.

IV. MODELING OF LOCOMOTION VELOCITY

With the robot optimally tuned for running, and with the body-pitch angle as control parameter, we make a parametric model of locomotion velocity regarding both control parameter and two task parameters: ground slope and payload weight. Using collected data from Section III-B, and a brief description of the SA in Section IV-A, we describe the parametric modeling procedure to compute expected robot velocity given a set of control parameters in Section IV-B, followed by the discussion on interpolation properties of the model as well as potential usage of the model.

A. SA Formalism

Our approach was to use a modified SA algorithm [15] to build locomotion velocity models. In our previous work, we used SA to model continuous trajectories in the joint space of a robotic arm. Here, SA was used to model locomotion velocity. The result of the modeling here is a continuous model of a discrete performance set.

We developed a model with the following parameters: payload mass w , ground slope α , and pitch angle β . There exists a performance measure that is a function of these parameters

$$\wp \triangleq \wp(w, \alpha, \beta), \quad \wp : \mathbb{R}^3 \mapsto \mathbb{R}. \quad (5)$$

The performance measure is the robot's velocity. This measure was recorded by consecutive repetitions of the experiments with different values of the three parameters. The goal was to find a model

$$\mathfrak{M}_\wp^3 = [c_{k_0 k_1 k_2}], \quad k_0 = 0, \dots, n_0 \\ k_1 = 0, \dots, n_1, \quad k_2 = 0, \dots, n_2 \quad (6)$$

where n_0, n_1 , and n_2 represent degrees of fitting polynomials. The polynomial degrees were chosen by the authors explicitly, based on the complexity of the relationship between $\{w, \alpha, \beta\}$ and \wp . The algorithm finds a set of coefficients for each fixed value of w and α to fit

$$\wp(w, \alpha, \beta) = \sum_{k_0=0}^{n_0} c_{k_0, k_1}(w, \alpha) \cdot \beta^{k_0} \quad (7)$$

as a polynomial in β . It then finds a second set of coefficients for each value of $c_{k_0, k_1}(w, \alpha)$ so that

$$c_{k_0, k_1}(w, \alpha) = \sum_{k_1=0}^{n_1} c_{k_0, k_1, k_2}(w) \cdot \alpha^{k_1} \quad (8)$$

is now a polynomial in α . Similarly, the algorithm solves for c_{k_0, k_1, k_2} to fit as a polynomial in w

$$c_{k_0, k_1, k_2}(w) = \sum_{k_2=0}^{n_2} c_{k_0, k_1, k_2} \cdot w^{k_2}. \quad (9)$$

Coefficients c_{k_0, k_1, k_2} are arranged in the 3-D matrix \mathfrak{M}_\wp^3 , called the locomotion velocity model. In this sense, each set of coeffi-

cients found in the previous step is successively fitted with the new set of the coefficients. The model shrinks in size by successive fitting by the product of the ratios of the total number of experiment variations, according to the parameter and the degree of fitting polynomial used in those steps.

This modeling procedure results in a multidimensional set of coefficients, resembling a lookup table. Unlike in a lookup table, here the coefficients are associated with a basis function used for fitting. This inherently allows for interpolation and a small amount of extrapolation. The most important feature of the SA is that we can randomly address the model to gain behavior completely different from the one used in modeling. When the full model was fit, we tested its quality using different statistical methods.

B. Parametric Modeling of Experimental Data

The initial 4-D data set for modeling was reduced to 3-D data with the SP variation data excluded, as we wanted to design a model that only had one control parameter, body-pitch angle. We modeled the 3-D matrix with dimension $[5 \times 7 \times 11]$, taking only the SP combined with DF that resulted in the best performance across all experiments, i.e., 110 ms of SP and 35% of DF. For each slope and weight parameter, we perform the polynomial fit, taking body-pitch angle β as an independent variable and pitch angle as the dependent variable. The degree of the polynomial is n_0 . The results are the coefficients stored in a 3-D matrix denoted $\mathbf{C}_P[5 \times 7 \times (n_0 + 1)]$. In the second step, we fit the data with a polynomial of n_1 degree, taking the slope angle α as the independent variable and the slope angle from \mathbf{C}_P as the dependent variable. Resulting polynomial coefficients are stored in the 3-D matrix denoted $\mathbf{C}_{PS}[5 \times (n_1 + 1) \times (n_0 + 1)]$. In the third step, we fit the coefficients of the second approximation, taking payload weights as the independent variable and the weight vector from \mathbf{C}_{PS} as the dependent variable. The degree of the polynomial is n_2 . The result, a new set of coefficients, is stored in the 3-D matrix \mathbf{C}_{PSW} .

With the final result being a matrix $\mathbf{C}_{PSW}[(n_2 + 1) \times (n_1 + 1) \times (n_0 + 1)]$ that represents the parametric model of the velocity with slope, pitch, and weight as parameters. The size of the model is very small. With $n_0 = 3$, and $n_1 = n_2 = 2$, we have a velocity model of only 36 coefficients (288 B of memory). The model is ten times smaller than the initial data used in modeling, and interpolation of the model and its ability to extrapolate due to the polynomials used in the SA is of importance.

Addressing of the model is straightforward. Given the desired weight w_d , desired slope α_d , and desired pitch angle β_d , the procedure follows simple backward computation on the \mathbf{C}_{PSW} data. First, for the weight parameter w_d , we compute the polynomials of (7). The results are coefficients of the new polynomials of the model (6) that are computed at the desired slope α_d . Resulting coefficients form the new polynomials, and (5) will be computed at the desired pitch angle β_d . It finally gives the velocity from the model for the desired values of the parameters. This simple procedure requires only two operations: multiplication and addition. Hence, it is particularly suitable for realization on embedded, onboard robot controllers.

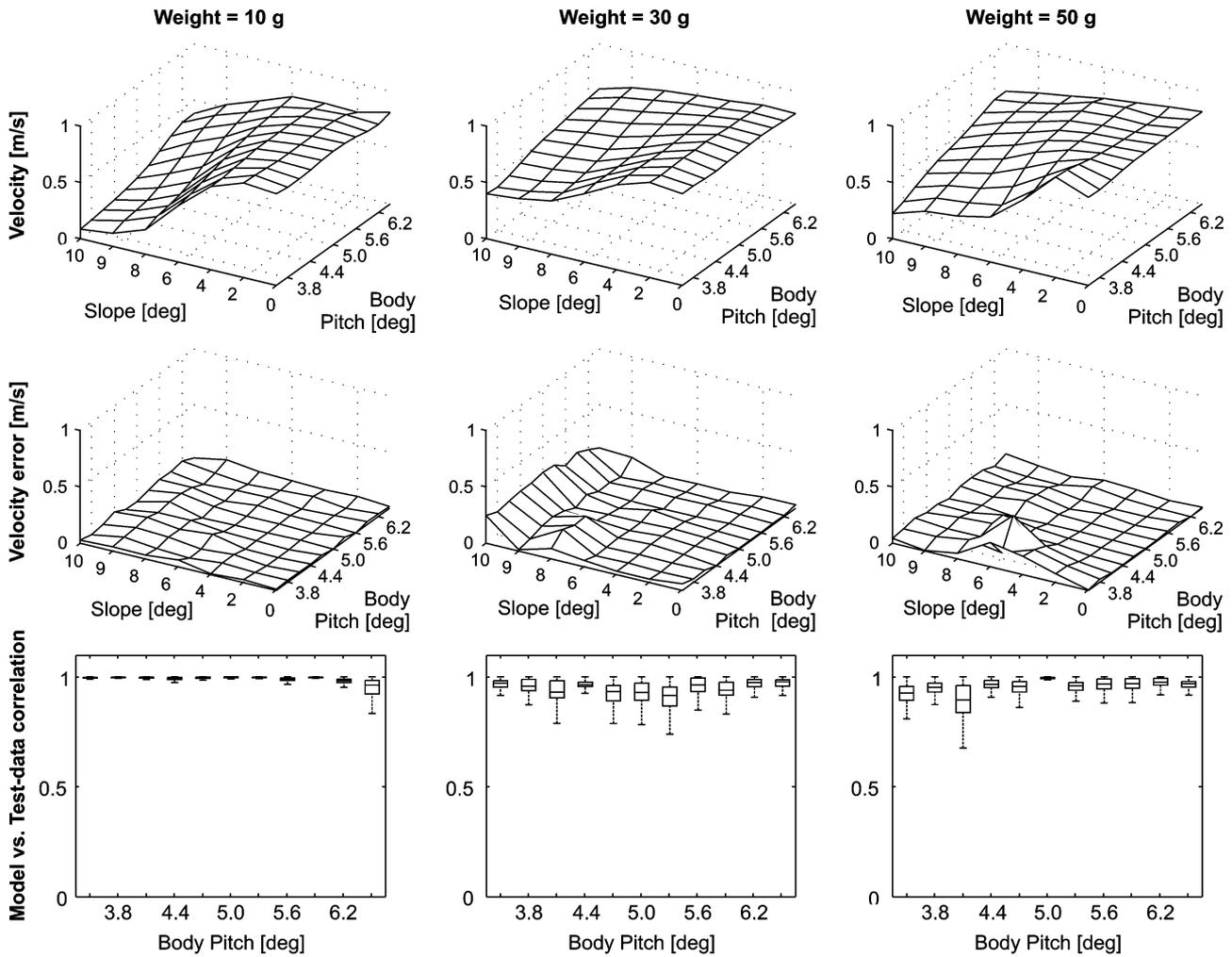


Fig. 11. Velocity, errors, and statistical merit of the velocity model as function of three parameters: payload, slope, and pitch angle. First column is for payload of 10 g, second column for payload of 30 g, and third column is for payload of 50 g. Upper row shows data from the velocity model with body pitch 3.5° to 6.5° , and ground slope 0° , 2° , 4° , 6° , 8° , 9° , and 10° , as parameters, middle row shows the errors relative to the test data, the absolute value of the difference between the model and the test data, and the lower row shows how velocities from model correlates with test data as we vary slope and body pitch. Boxes are plotted for each body-pitch angle. The box has lines at the lower quartile, median, and upper quartile values. The whiskers are lines extending from each end of the box to show the extent of the rest of the data.

The velocity model was cross-validated with the third measurement set on the same grid of parameters. Although we occasionally ran experiments on a different grid in setting up the protocol, it was not our intention to make a model that needs to extrapolate velocity. If a specific portion of the parameter space is of interest, we would include it in modeling rather than leaning on extrapolation properties. Nevertheless, if it happens that the model is addressed by the parameters outside of the training region but not far away, we may expect that the model with polynomials as basis functions will respond reasonably well. The same property of the SA was used in generalizing inverse kinematics of a redundant robot outside of a training region [16].

On the surface plots of Fig. 11, we show the velocity from the model, and in the same plot, the error between the model and the test data. The error is given as the absolute value of the difference between the model and the test data. It is obvious that the velocity model, as well as the test data, is rather smooth, and therefore, we would not expect the model to produce scattered outliers. However, we ran a bootstrap approach to generate data

for cross-validation to quantify how well the model interpolates in the cross-validation data.

The last row in Fig. 11, three box plots, one per each of three payloads {10, 30, 50} g, shows how pairs of model and test data randomly selected by bootstrapping correlate, on average, for all the slopes. A mean correlation of almost 1.0 accross all body-pitch angles, in the low-payload case, proves that the pitch-angle model, along with further parametric modeling, interpolated all the parametric space of interest really well. With added mass, the mean correlation for all ground slopes shows that the model interpolates not so well, yet keeps correlation above 0.8. This is mostly due to the fact that the body-pitch model did not take into account ground slope as a parameter.

The velocity model performed best at minimal weight and degraded as we increased payload. This was due to the fact that we did not have body-pitch angle models for different weights. For softer flexures, the weight of 50 g represents almost 15% of total body weight, and definitely changes body posture as well as its dynamic in hopping. Despite this, the model correlates well to the test data in all cases.

The model shows decreasing velocity in running uphill. It was observed that the robot had constant performance on zero slope (top row in Fig. 11). Practically, it turned out that the pitch angle was not a control parameter in this case. However, on nonzero slopes, it became an important control parameter. At the lower boundary (3.5°) of the pitch-angle interval, the robot achieved its highest velocity. However, the interval we examined, from 3.5° to 6.5° , was of general interest on the runs. Maximal velocity from the model, 0.8991 m/s, was found for a horizontal track, body pitch of 3.5° , and 50 g of payload. Added mass had two effects. First, it changed the posture by increasing body-pitch angles when deflecting softer legs were used (rear and middle), leading to an appropriate pitch angle. Second, more efficient running results from increased friction between the feet of the rear legs and the track. From Fig. 11, we observe that uphill velocity is increasing if robot carries some mass (the best result happened with 30 g) and it might be a hint for redesign of the robot in future.

To conclude, experiments and modeling showed that the behavior of the robot was a highly complex function of all parameters: slope, weight, SP, and body pitch. Generally, there was an interval of body-pitch angle where increased pitch resulted in faster runs. Similarly, we found an optimal SP. Those two control parameters could be integrated in one model, or they could be used concurrently, with online adaptation of the SP after using model-based control of the body posture. In sum, the pitch-angle model is indeed a valuable control parameter, once we set up the best dynamic performance, choosing the SP, duty cycle, and flexures in the legs with given task parameters such as ground slope and added mass.

We could also use the velocity model as an inverse model. That is, if we knew the first two parameters plus desired velocity, we could compute the value of the third parameter. This is useful in computing desired body-pitch angle if we need to achieve desired velocity on a known slope with known mass. For example, if we know the weight $w_\delta = 45$ g, the slope $\alpha_\delta = 4.7^\circ$, and the desired velocity $v = 0.717$ m/s, we can calculate pitch angle at the definition range of the model [$3.5^\circ, 6.5^\circ$]. The procedure is similar to regular addressing of the model. The only difference happens at the last step, where we take the final polynomial and, instead of computing its value for a pitch angle to get velocity, we find the root of the polynomial that equals desired velocity. That root is, in fact, the desired pitch angle of robot's body. In a particular case, we computed the last polynomial as

$$-0.0197v^3 + 0.2478v^2 - 0.8811v + 1.3978 = 0.717. \quad (10)$$

One of three roots of (10), $v_{1,2,3} \in \{6.578; 4.936; 1.064\}^\circ$, 4.936° , belongs to the desired pitch-angle range [$3.5^\circ, 6.5^\circ$], as shown in Fig. 12. Hence, we conclude that 4.936° of body-pitch angle will provide, under given conditions, the desired velocity of 0.717 m/s. By running the robot three times with the desired combination of parameters, we observe that the average velocity of 0.702 ± 0.167 m/s closely matches the result from the model.

A very useful property of this modeling procedure is extrapolation. For example, if we want to compute the whole family of locomotion velocities on a slope varying from 0° to 10° , but with payload mass of 65 g, which is by 30% more than the maximum mass used in experimentation, simple addressing of the model

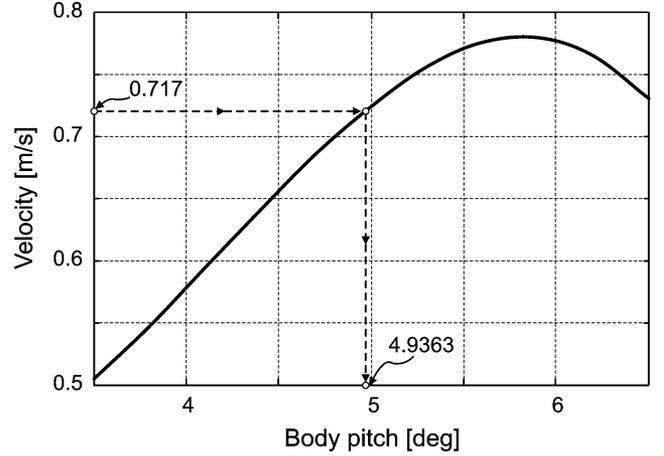


Fig. 12. Example of inverse addressing of the velocity model. Given payload mass of 45 g and a slope angle of 4.7° , the model gives velocity, shown by a solid line. For the desired locomotion velocity of 0.717 m/s, we calculate a 3.7987° body-pitch angle to be necessary.

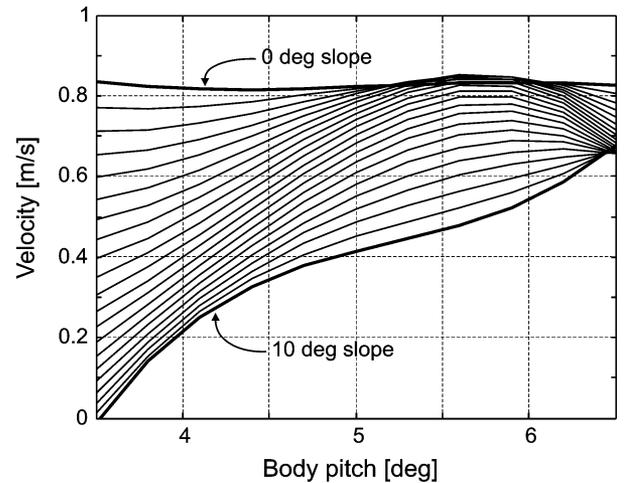


Fig. 13. Locomotion velocity for a variety of parameters, of which added mass exceeds the maximum used in experimentation by 30% , i.e., 65 g. The slope parameter was addressed at the 0.5° rate on the interval $[0, 10]^\circ$.

for desired parameters produces a family of curves, shown in Fig. 13.

The mass of 65 g is more than we had for modeling. Therefore, on the flat ground, with the Type III-II-II flexures, the bigger mass forces the body-pitch angle to a constant, no matter how we commanded it. The velocity increases with higher pitch angles, but it is a nonlinear function of slope and body-pitch angle. This suggests possible improvement of the body-pitch angle model with one more parameter, the ground-slope angle.

V. CONCLUSION

We presented an approach to the design of a small six-legged robot. The robot we used is a derivative of the Sprawl [13] series of robots made by SDM. Our robot had legs like Sprawl, but the body and sensors were different. This air-powered robot had six pistons, one per leg, three per tripod, with two valves, one per tripod. Each leg had interchangeable passive flexures connecting tethered parts of the leg with the part that embodied the piston. Small DC motors beneath the body orient the legs in an

offline fashion. The robot moved by hopping from one tripod to another.

Our design of robot's body differs from its Sprawl predecessors in two major aspects. The Sprawl robot had middle legs positioned in the middle of the robot, between front and rear legs. All six legs had the same type of flexure, softer than ones in the Sprawlita robot. In addition, leg orientations did not follow a specific procedure that might lead to successful running. In our robot, middle legs were pushed backward, closer to the rear legs, shifting the COM backward, compared with the Sprawl. Several combinations of flexures were tested, leading to the conclusion that front legs require higher stiffness, whereas middle and rear legs require softer flexures.

We chose leg orientations based on the static analysis, supported by experiments later on, following the constraint that all GRFs of one tripod should intersect in a single point. In this way, GRFs from the tripods did not produce the resultant moment at the COM of the robot. We also hypothesized that locomotion velocity may be controlled by changing the angle of the remaining resultant forces. Both of these ideas appeared to pan out with the experimental data.

Our goal was to apply parametric modeling tools to build a feed-forward locomotion velocity model with typical task parameters: ground slope and payload mass. The only parameter in the model that we used to control the robot's performance was body-pitch angle. Body-pitch angle is set up by leg orientation at the beginning of the run. Our kinematic model used polynomials to approximate experimental data. Once formed, the model allowed us to compute how the legs should be oriented in order to achieve desired body pitch, and consequently, locomotion velocity. The pitch-angle model embodies information on flexures in each leg. The model is small, 192 B, and simple to calculate on embedded microcontrollers.

Our modeling used a modified version of the SA algorithm. We believe that such parametric modeling, due to the polynomials used in the approximation, may be particularly appropriate for control with onboard microcontrollers. The procedure supports random addressing, interpolation, and extrapolation to some extent. It also enables refinement of the model as interaction with the environment changes over time and through experience. For example, in our task for uphill running with a mass, we built a 3-D velocity model with the body-pitch angle as the single control parameter. The size of the velocity model, only 288 B, was determined by as few as 36 coefficients. The computation of the model required only addition and multiplication, making it suitable for a small on-board computer.

We performed approximately 20 000 trials with the robot, spanning several months. Our experience with this robot suggested that the robot design was robust. Neither the robot nor sensors suffered a failure. Indeed, not a single screw fell off. The legs and body remained in excellent condition. This robustness suggested that the technology applied in the design can produce robots that are genuinely durable.

Stepwise experimentation gradually revealed the robot's performance and the crucial control parameters in the task. Peak performance of three body lengths per second with an initial guess on the flexure types revealed an optimal SP, DF, and pitch angle. Once we found SP/DF combinations that led to smooth

running, we ran the experiment to find the best possible combination of flexures with a constant pitch angle. We discovered which flexures produce the best performance of 3.6 body lengths per second. We then discovered that variation of pitch angle doubled the robot's performance up to almost six body lengths per second. This is double the speed of Sprawl with the same number of valves [13]. Although it is hard to compare the performance of our robot with truly autonomous ones, our robot still managed to run very fast, even with a considerable burden in the form of a trailer that had one-third of the robot's weight.

Altered environment conditions and minor changes in robot design parameters may also benefit from the proposed parametric modeling procedure. For example, robots equipped with models from laboratory conditions will require additional fine-tuning to improve their performance in altered environments. Aging and wear of the robot, particularly its feet and flexures, also cause remodeling. It is expected that cheap robots made of low-quality materials will differ from each other due to poor manufacturing tolerances. Finally, many of their successors will be manufactured with minor differences in geometry and inertia. All these issues can be resolved by running experiments in the lab in order to recognize salient geometric and inertia variations of the robot. Then, taking the SA procedure with recurrent fitting, we may "personalize" these models to the particular robot. The adaptation property of the SA procedure, achieved by incremental least-squares algorithms within the SA, has been applied to incremental modeling of inverse kinematics of a walking hexapod [28].

ACKNOWLEDGMENT

The authors thank Prof. M. Cutkosky and the Stanford Center for Design Research for providing us with legs, actuators, and parts of the experimental setup. Parts of the experiments were run by S. Anđelković and D. Stojilković, University of Niš, and by C. Lee, Johns Hopkins University, Baltimore, MD. We also thank the anonymous reviewers for their valuable contributions. We are grateful to J. T. Francis for proofreading.

REFERENCES

- [1] M. Ahmadi and M. Buehler, "Stable control of a simulated one-legged running robot with hip and leg compliance," *IEEE Trans. Robot. Autom.*, vol. 13, pp. 96–104, Feb. 1997.
- [2] R. McN. Alexander, "Three uses for springs in legged locomotion," *Int. J. Robot. Res.*, vol. 9, no. 2, pp. 53–61, 1990.
- [3] J. J. Abbas and R. J. Full, "Neuromechanical interaction in cyclic movements," in *Biomechanics and Neural Control of Posture and Movement*, J. M. Winters and P. E. Crago, Eds. New York: Springer-Verlag, 2000.
- [4] S. A. Bailey, J. G. Cham, M. R. Cutkosky, and R. J. Full, "Biomimetic robotic mechanisms via Shape Deposition Manufacturing," in *Robotics Research: 9th International Symposium*, J. Hollerbach and D. Koditschek, Eds. London, U.K.: Springer-Verlag, 2000.
- [5] —, "Comparing the locomotion dynamics of the cockroach and a Shape Deposition Manufactured biomimetic hexapod," in *Proc. ISER*, Honolulu, HI, Dec. 2000, pp. 239–248.
- [6] R. Blickhan and R. J. Full, "Similarity in multilegged locomotion: Bouncing like a monopode," *J. Comput. Phys.*, vol. 173, pp. 509–517, 1993.
- [7] N. Bhushan and R. Shadmehr, "Computational nature of human adaptive control during learning of reaching movements in force fields," *Biol. Cybern.*, vol. 81, no. 1, pp. 39–60, 1999.
- [8] C. Bartling and J. Schmitz, "Reaction to disturbances of a walking leg during stance," *J. Exp. Biol.*, vol. 203, pp. 1211–1233, 2000.

- [9] M. Buehler, U. Saranli, D. Papadopoulos, and D. Koditschek, "Dynamic locomotion with four and six-legged robots," in *Proc. Int. Symp. Adapt. Motion Animals, Mach.*, Montreal, QC, Canada, Aug. 2000, pp. 1–2.
- [10] D. Campbell and M. Buehler, "Stair descent in the simple hexapod RHex," in *Proc. IEEE Int. Conf. Robot. Autom.*, Taipei, Taiwan, Sep. 2003, pp. 1380–1385.
- [11] J. G. Cham, S. A. Bailey, J. E. Clark, R. J. Full, and M. R. Cutkosky, "Fast and robust: Hexapedal robots via Shape Deposition Manufacturing," *Int. J. Robot. Res.*, vol. 21, no. 10, pp. 869–882, Oct. 2002.
- [12] J. G. Cham, J. K. Karpick, and M. R. Cutkosky, "Stride period adaptation for a biomimetic running hexapod," *Int. J. Robot. Res.*, vol. 23, no. 2, pp. 141–153, Feb. 2004.
- [13] J. E. Clark, J. G. Cham, S. A. Bailey, E. M. Froehlich, P. K. Nahata, R. J. Full, and M. R. Cutkosky, "Biomimetic design and fabrication of a hexapedal running robot," in *Proc. IEEE Int. Conf. Robot. Autom.*, Seoul, Korea, 2001, pp. 3643–3649.
- [14] M. H. Dickinson, C. T. Farley, R. J. Full, M. A. R. Koehl, R. Kram, and S. Lehman, "How animals move: An integrative view," *Science*, vol. 288, pp. 100–106, 2000.
- [15] G. S. Dorđević, M. Rašić, D. Kostić, and V. Potkonjak, "Motion control skills in robotics," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 30, pp. 219–238, May 2000.
- [16] G. S. Dorđević, M. Rašić, D. Kostić, and D. Šurdilović, "Learning of inverse kinematics behavior of redundant robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, Detroit, MI, May 1999, pp. 3165–3170.
- [17] R. J. Full and D. E. Koditschek, "Templates and anchors: Neuromechanical hypotheses of legged locomotion on land," *J. Exp. Biol.*, vol. 202, pp. 3325–3332, 1999.
- [18] R. J. Full, T. Kubow, J. Schmitt, P. Holmes, and D. Koditschek, "Quantifying dynamic stability and maneuverability in legged locomotion," *Integr. Computat. Biol.*, vol. 42, pp. 149–157, 2002.
- [19] H. Komsuoglu, D. McMordie, U. Saranli, N. Moore, M. Buehler, and D. E. Koditschek, "Proprioception-based behavioral advances in a hexapod robot," in *Proc. IEEE Int. Conf. Robot. Autom.*, Seoul, Korea, 2001, pp. 3650–3655.
- [20] R. Kram, B. Wong, and R. J. Full, "Three-dimensional kinematics and limb kinetic energy of running cockroaches," *J. Exp. Biol.*, vol. 200, pp. 1919–1929, 1997.
- [21] T. T. Kubow and R. J. Full, "The role of the mechanical system in control: A hypothesis of self-stabilization in hexapedal runners," *Philos. Trans. R. Soc. Lond. B.*, vol. 354, pp. 849–861, 1999.
- [22] B. Klaassen, R. Linnemann, D. Spennberg, and F. Kirchner, "Biomimetic walking robot scorpion: Control and modeling," in *Proc. 9th Int. Symp. Intell. Robot. Syst.*, 2001, pp. 101–108.
- [23] T. T. Lee, C.-M. Liao, and T. K. Chen, "On the stability properties of hexapod tripod gait," *IEEE Trans. Robot. Autom.*, vol. RA-4, pp. 427–434, Aug. 1988.
- [24] A. Martin-Alvarez, W. de Weert, J. Hillebrand, P. Putz, A. Matthyssen, and J. F. de Weerd, "Walking robots for planetary exploration missions," in *Proc. 2nd World Autom. Congr.*, Montpellier, France, May 1996.
- [25] G. M. Nelson, R. D. Quinn, R. J. Bachmann, W. C. Flannigan, R. E. Ritzmann, and J. T. Watson, "Design and simulation of a cockroach-like hexapod robot," in *Proc. IEEE Int. Conf. Robot. Autom.*, Albuquerque, NM, Apr. 1997, pp. 1106–1111.
- [26] R. D. Quinn, D. A. Kingsley, J. T. Offi, and R. E. Ritzmann, "Improved mobility through abstracted biological principles," in *Proc. IEEE Int. Conf. Intell. Robots, Syst.*, Lausanne, Switzerland, 2002, pp. 2652–2657.
- [27] R. D. Quinn, G. M. Nelson, R. J. Bachmann, D. A. Kingsley, J. T. Offi, T. J. Allen, and R. E. Ritzmann, "Parallel complementary strategies for implementing biological principles into mobile robots," *Int. J. Robot. Res.*, vol. 22, no. 3, pp. 169–186, Apr. 2003.
- [28] M. Rašić and G. S. Dorđević, "Modeling of kinematic redundancy by incremental successive approximations," in *Proc. IEEE 6th TELSIKS*, Niš, Serbia and Montenegro, Oct. 2003, pp. 791–794.
- [29] M. H. Raibert, *Legged Robots That Balance*. Cambridge, MA: MIT Press, 1986.
- [30] *RHex: The Compliant Hexapod Robot*. [Online]. Available: <http://www.rhex.net/>
- [31] U. Saranli, M. Buehler, and D. E. Koditschek, "RHEX: A simple and highly mobile hexapod robot," *Int. J. Robot. Res.*, vol. 20, no. 7, pp. 616–631, Jul. 2001.
- [32] U. Saranli and D. Koditschek, "Template-based control of hexapedal running," in *Proc. IEEE Int. Conf. Robot. Autom.*, Taipei, Taiwan, Sep. 2003, pp. 1374–1379.
- [33] J. Schmitt, M. Garcia, R. C. Razo, P. Holmes, and R. J. Full, "Dynamic and static stability of legged locomotion in the horizontal plane: A test case using insects," *Biol. Cybern.*, vol. 86, no. 5, pp. 343–353, 2002.
- [34] *Sprawl Robots: Fast, robust hexapods functionally inspired by nature*. [Online]. Available: <http://www-cdr.stanford.edu/biomimetics/documents/sprawl/>
- [35] L. H. Ting, R. Blickhan, and R. J. Full, "Dynamic and static stability in hexapedal runners," *J. Exp. Biol.*, vol. 197, pp. 251–269, 1994.
- [36] M. Vukobratovic, B. Borovac, D. Surla, and D. Stokic, *Biped Locomotion: Dynamics, Stability, Control, and Application*, ser. Scientific Fundamentals of Robotics 7. New York: Springer-Verlag, 1990.
- [37] T. Wang, G. S. Dorđević, and R. Shadmehr, "Learning the dynamics of reaching movements results in the modification of arm impedance and long-latency perturbation responses," *Biol. Cybern.*, vol. 85, no. 6, pp. 437–448, 2001.
- [38] *Whigs Series Robots*. [Online]. Available: <http://biorobots.cwru.edu/projects/whigs/>



Goran S. Dorđević (M'95) was born in Niš, Serbia, in 1963. He received the E.E., M.Sc., and Ph.D. degrees in electronics engineering in 1988, 1992, and 1995, respectively, from the University of Niš, Niš, Serbia.

He was with The Johns Hopkins University, Baltimore, MD, from 2000 through 2002. Currently, he is with the Faculty of Electronics Engineering, University of Niš. His major teaching areas are robotics and control engineering. He is the author or coauthor of one book, 20 journal papers, and more than 50 conference papers, mostly in robotics.

His research interests include biomimetics, human-robot interaction, intelligent robot control, legged locomotion, modeling, service robotics, and redundancy resolution.



Milan Rašić (S'00) was born in Niš, Serbia, in 1972. He received the E.E. and M.Sc. degrees in electronics engineering in 1996 and 2000, respectively, from the University of Niš, Niš, Serbia.

He is currently a Teaching Assistant with the Faculty of Electronics Engineering, University of Niš. He is the author or coauthor of one book, five journal papers, and more than ten conference papers, mostly in robotics. His research interests include embedded control systems, intelligent robot control, modeling, mobile robotics, and redundancy resolution.



Reza Shadmehr was born in Tehran, Iran, in 1963. He received the Ph.D. degree in computer science and robotics in 1991 from the University of Southern California (USC), Los Angeles.

From 1991–1994, he was with the Massachusetts Institute of Technology, Cambridge, where he received postdoctoral training in human motor control. He has been with Johns Hopkins University Biomedical Engineering, Baltimore, MD, since 1995.

Dr. Shadmehr received an IBM Graduate Fellow award while at USC, and a McDonnell-Pew Fellowship while at MIT.