A MATLAB Framework for Efficient Gait Creation

C. David Remy, Keith Buffinton, and Roland Siegwart, *Fellow, IEEE*

Abstract—**This work introduces a framework for the creation and analysis of efficient gaits for legged systems based on the exploitation of natural dynamics. It summarizes the theory behind hybrid dynamic modeling, the identification of optimal periodic motions with single shooting and direct collocation, and the analysis of first order stability. Three examples introduce various aspects of gait creation and analysis: a stability study of a passive dynamic walker determines the ideal position of the leg's center of mass, the cost of transportation is minimized for a prismatic monopod hopper based on series elastic actuators, and a basic controller is created for the model of a bounding robot. The presented tools and examples are freely available at**
www.asl.ethz.ch/people/cremy/personal/GaitCreation.

I. INTRODUCTION

WHEN considering passive or natural dynamic effects in legged robots (i.e., motions induced by inertia, gravity, and elastic oscillations), it's often more appropriate to regard the resulting system as a complex nonlinear oscillator that propels itself forward in periodic interaction with the ground, than as a continuously controlled system. Systems using this approach include, for example, passive dynamic walkers [1, 2], the early MIT hoppers [3], the Cornell ranger [4], Denise from TU Delft [5], and many others. They can reach an efficiency similar to that of walking humans [6]. The fundamental premise is that steady motion can be achieved with small periodic inputs that produce a limit cycle which is primarily shaped by the mechanical dynamics of the system. The advantage in comparison to continuous control concepts (such as *Zero Moment Point* [7]) lies in the fact that the natural dynamic elements can accommodate the energetic fluctuations that accompany the periodic motions in walking and running. Negative actuator work is avoided and the energetic efficiency is increased. The search for limit cycles, which represent periodic motions (or simply: gaits) can be expressed as a constrained optimization problem, similar to formulations by Mombaur [8] and Stelzer [9]. Control inputs are generated numerically such that actuator effort is minimized, while a periodic motion is enforced though appropriate boundary conditions. To this end, actuator motion is encoded as a paramet-

ric excitation function that is subject to optimization. Stability is evaluated using a first-order approximation of the system on a step-to-step basis.

In the presented framework and corresponding source files, we provide tools, methods, and examples for the model-based creation and analysis of efficient gaits based on these considerations. Despite the inferior computational performance of MATLAB, we deliberately chose this script-based language as it is widely available in the academic environment and often more familiar to researchers with no explicit background in numerical methods. The purpose of the paper is three-fold: *Firstly*, it presents the theoretical basis for efficient gait creation as it was implemented in the accompanying MATLAB framework. Methods and algorithms are introduced for hybrid dynamic simulations, limit cycle identification, optimization, and stability analysis (for passive as well as for active systems). *Secondly*, a number of examples (section IV) illustrate robotic concepts that enable the exploitation of natural dynamic effects and thus yield highly efficient locomotion. They include a passive dynamic walker and two running robots based on the idea of high-compliance series elastic actuation [10]. The examples are provided with the framework where they are used to demonstrate the functionality of the software and provide a code base for further exploration. *Thirdly*, for each example, a small study emphasizes different aspects of this approach to gait creation. We examine the impact of the leg CoG on the first-order stability of a passive dynamic walker, show that similar to biological systems [11] optimal walking/running speeds exists when trying to minimize the cost of transportation (COT), and introduce a controller for a running robot that modulates the excitation to stabilize the periodic motion.

II. THEORY

A. Hybrid Dynamics

In legged locomotion, we inevitably deal with hybrid dynamic systems, i.e. systems that combine aspects of continuous and discrete dynamics [12]. Every time a foot strikes or leaves the ground, the continuous dynamics of the mechanical multibody system are interrupted and altered. Velocities change instantaneously during collisions and the mechanical configuration of the multi-body system is modified. We refer to instances when this happens as *events*. They constitute the *jump map* of the system and can be defined by the directional zero crossings of an event function

$$e(\mathbf{x}, \mathbf{p}) = 0 \text{ with } \dot{e} > 0. \qquad (1)$$

The vector $\mathbf{x}$ defines the current state of the system while $\mathbf{p}$ is used to model adjustable parameters, such as the stiffness or damping of tunable springs. For every event, a handling function defines the *jump map*; that is, the discontinuous changes in the states according to

$$\mathbf{x}^+ = g\left(\mathbf{x}^-, \mathbf{p}\right). \tag{2}$$

The superscripts – and + indicate the states right before and after the event. This could, for example, be the implementation of the impulse equations for a mechanical collision. Between events (in the *flow set*) the *continuous states* are governed by the *flow map*, a set of differential equations

$$\dot{\mathbf{x}}_{cont} = f\left(\mathbf{x}, \mathbf{p}\right) \tag{3}$$

which contain the equations of motion. Some of the states might be *discrete*, which means that their derivative is zero and they only change during events $\left(\dot{\mathbf{x}}_{disc} \equiv 0\right)$. At least one discrete state is needed in a model to describe the current system configuration (i.e. defining which legs are in the air or in ground contact). This, in turn, affects the continuous dynamics of the motion. For performance reasons, we separate the state vector $\mathbf{x} = \begin{bmatrix} \mathbf{y} & \mathbf{z} \end{bmatrix}^T$ into continuous states $\mathbf{y}$ and discrete states $\mathbf{z}$ However, there is no need for such a distinction in a theoretical consideration. It is worth mentioning that the presented formalism can model considerably more than just mechanics. In a basic implementation, the continuous elements of the vector $\mathbf{x}$ contain only the generalized coordinates $\mathbf{q}$ and generalized speeds $\dot{\mathbf{q}}$ that are used to describe the dynamics of the mechanical system. However, more information can be included and additional states can compute characteristic data such as energy consumption.

B. Multi Body Dynamics

The equations of motion (EoM) for a multi body system subject to constraints can be stated as:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} - \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{f} + \mathbf{J}^T(\mathbf{q})\lambda \tag{4}$$

The dynamics of the generalized coordinates $\mathbf{q}$ are given by the mass matrix $\mathbf{M}$, the differentiable force vector $\mathbf{h}$ (which includes the gravitational and coriolis forces), the generalized forces in the joints $\mathbf{f}$ (such as friction and actuation), and the contact Jacobian $\mathbf{J} = \partial\mathbf{r}/\partial\mathbf{q}$. This Jacobian is obtained by partial differentiation of the contact distance vector $\mathbf{r}$, and maps a vector of contact forces λ into the generalized coordinate space. For locomotion on flat terrain, these vectors are usually expressed in Cartesian coordinates. Depending on the number of feet that are in contact with the ground (for example defined by a set of discrete binary states), the dimensions of $\mathbf{r}$, $\mathbf{J}$, and λ vary. If all feet are in the air, the vectors and matrices are empty. For each foot that is in contact, two (three for 3D-systems) rows are added to $\mathbf{r}$. $\mathbf{J}$ and λ are expanded accordingly.

We assume that a foot that is on the ground is only allowed to perform a pure rolling motion. No slipping or sliding is modeled. For the contact distance, this means:

$$\dot{\mathbf{r}} = \mathbf{J}\dot{\mathbf{q}} = 0 \tag{5}$$

$$\ddot{\mathbf{r}} = \mathbf{J}\ddot{\mathbf{q}} + \dot{\mathbf{J}}\dot{\mathbf{q}} = 0 \tag{6}$$

must hold for the relative motion with respect to the ground. Solving (4) for $\ddot{\mathbf{q}}$ and substituting this into the constraint equation (6), allows one to solve for the contact forces λ:

$$\lambda = \left(\mathbf{J}\mathbf{M}^{-1}\mathbf{J}^T\right)^{-1}\left(-\mathbf{J}\mathbf{M}^{-1}\left(\mathbf{h} + \mathbf{f}\right) - \dot{\mathbf{J}}\dot{\mathbf{q}}\right) \tag{7}$$

These force are used in the flow map to compute the generalized accelerations $\ddot{\mathbf{q}}$ with eq. (4) and in the jump set to determine when a foot leaves the ground; that is, for a contact to open, its normal force must become negative $\lambda_i^y < 0$. The corresponding jump map only needs to register the new contact configuration (which in turn influences the computation of $\mathbf{r}$, $\mathbf{J}$, and λ); other states are not affected.

In contrast, instantaneous changes in velocities (a consequence of (5)) occur when a contact closes; that is, when a foot touches the ground. This is registered in the jump set when the vertical position of a possible contact point is about to become negative ($r^y = 0$, $\dot{r}^y < 0$). The collision that accompanies the event leads to instantaneous changes in velocities caused by external impulsive forces Λ. To compute these, the equations of motion are integrated over the duration of the collision:

$$\int_{\{t_0\}} \left\{\mathbf{M}\ddot{\mathbf{q}} - \mathbf{h} - \mathbf{f} - \mathbf{J}^T\lambda\right\} dt = \mathbf{M}\left(\dot{\mathbf{q}}^+ - \dot{\mathbf{q}}^-\right) - \mathbf{J}^T\Lambda = 0 \tag{8}$$

As the integration is performed over an infinitesimally short time span, the bounded differentiable force vectors $\mathbf{h}$ and $\mathbf{f}$ do not contribute and only the impulsive forces and the velocity changes must be taken into account. Assuming a perfectly inelastic collision with a Newtonian collision law [13], the contact points of the feet that are considered part of the collision instantaneously come to a rest (or remain motionless) ($\Lambda^y \geq 0, \ddot{\mathbf{r}}^+ = 0$). For contacts that are opening at the same time, the corresponding contact points must leave the ground right after the collision ($\Lambda = 0, \dot{r}^{y+} \geq 0$). These two alternatives are expressed in a complementary description of the collision in the normal direction:

$$\dot{\mathbf{r}}^+ - \dot{\mathbf{r}}^- = \mathbf{J} \cdot \left(\dot{\mathbf{q}}^+ - \dot{\mathbf{q}}^-\right) = \mathbf{J}\mathbf{M}^{-1}\mathbf{J}^T \cdot \Lambda$$
$$\dot{r}^{y+} \geq 0, \Lambda^y \geq 0, \dot{r}^{y+} \cdot \Lambda^y = 0 \tag{9}$$

which is solved for the post impact speeds $\dot{\mathbf{q}}^+$.

C. Periodicity

Instead of continuously monitoring the walking or running motion, we regard the dynamics of the underlying system only on a step-to-step basis; that is, we reduce them to a single function P that maps a state vector at the beginning of a stride to a vector at the end of the stride. It is rarely possible to state P analytically, but it is straightforward to compute it for a given set of initial states $\mathbf{x}^k$ by integrating the equations of

motion and evaluating the discrete changes at intermediate events.

With this point of view, steady state walking or running can be reduced to two simple requirements:

1. The transfer from state $\mathbf{x}^k$ to state $\mathbf{x}^{k+1}$ must be accompanied by some forward motion
2. All other physical states should map onto themselves at each instant, i.e. they are periodic within a subset of P: $\mathbf{x}^{k+1} = \bar{P}(\mathbf{x}^k, \mathbf{p}) = \mathbf{x}^k$

If the forward motion and periodicity requirements are met, the same motion can be carried out over and over again to continuously move the robot forward. The system exhibits a limit cycle, i.e., a closed trajectory in state space. P maps the configuration of the system at the beginning of one stride to the configuration at the beginning of the next stride. The creation of a *gait* is reduced to finding a root $\mathbf{x}^*$ of the implicit equation

$$\bar{P}(\mathbf{x}^*, \mathbf{p}) - \mathbf{x}^* = 0. \tag{10}$$

In the solution process, the parameters $\mathbf{p}$ can be tuned with respect to some optimization criterion, e.g., to maximize locomotion speed. The instants at which we observe the dynamics define a *Poincaré section* transverse to the flow of the system. $\bar{P}$ is the *Poincaré map* of the system. In legged locomotion, the Poincaré section is usually defined by a specific event in the gait cycle. We call this event *terminal*, as it defines the end of a *stride* and stops the integration. Examples of events commonly used for this purpose include apex transition [14], the ground strike of a certain leg [15], or a periodic time [8]. At such an instant the state vector is often subject to additional constraints, which can simplify the dynamics of P.

D. Stability

Stability of a motion limit cycle can be assessed via linearization of the stride-to-stride transfer function. From the Taylor series expansion of $\bar{P}$:

$$\mathbf{x}^{k+1} = \bar{P}(\mathbf{x}^k) = \bar{P}(\mathbf{x}^*) + \frac{\partial \bar{P}(\mathbf{x})}{\partial \mathbf{x}}\bigg|_{\mathbf{x}^*} \cdot (\mathbf{x}^k - \mathbf{x}^*) + \mathrm{O}(\mathbf{x}^2) \tag{11}$$

follows:

$$\Delta\mathbf{x}^{k+1} = \mathbf{x}^{k+1} - \mathbf{x}^* \approx \frac{\partial \bar{P}(\mathbf{x})}{\partial \mathbf{x}}\bigg|_{\mathbf{x}^*} \cdot \Delta\mathbf{x}^k = \mathbf{J}_x \cdot \Delta\mathbf{x}^k. \tag{12}$$

(the parameter vector $\mathbf{p}$ is omitted in (11), as it is assumed to be constant once the robot is in motion). This means a disturbance $\Delta\mathbf{x}$ of the periodic initial condition $\mathbf{x}^*$ will cause a deviation from the nominal trajectory during the next step according to: $\Delta\mathbf{x}^{k+1} \approx \mathbf{J}_x \Delta\mathbf{x}^k$. $\mathbf{J}_x$ is the *monodromy matrix* of the Poincaré map. If the magnitude of all eigenvalues (the *Floquet multipliers*) of $\mathbf{J}_x$ is smaller then 1, any disturbance will vanish over time and the limit cycle is considered stable. If at least one eigenvalue has a magnitude larger then 1, the disturbance will grow, causing the system to leave the limit

cycle. As a disturbance parallel to the flow of the system is eliminated within a single stride, one of the eigenvalues will always be zero [16].

E. Actuation

So far, we have implicitly assumed that we are dealing with purely passive systems, i.e., robots powered by gravity or simplified simulations without dissipative elements [14]. When actuation is included, this extends the physics of the system. New states must account for the additional dynamics of the actuators and might model some local servo controllers for joint-level position, velocity, or force control. The dynamic formulation is then extended to include driving terms $\mathbf{u}$ which are the *control inputs* of the system, and thus $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, \mathbf{p})$. The inputs are provided by a global controller responsible for initiating motion and stabilizing the robot. A number of reasonable assumptions are made about this controller below to allow a more systematic approach. However, they should in no way be regarded as exclusive:

- Following from the idea of passive dynamic locomotion, the controller will not enforce continuous control along predefined trajectories, but will only be used to feed energy into the system and shape its limit cycle.
- $\mathbf{u}$ defines an open-loop excitation law comparable to the spinal reflexes found in animals [17]. In the classic open-loop sense, $\mathbf{u}$ is simply a function of time but other driving terms are possible, leading to the generalized formulation $\mathbf{u}(\mathbf{x})$. This includes $\mathbf{u}(t)$ as a special case when $x_t = t$ (with $\dot{x}_t \equiv 1$) is included in the state vector $\mathbf{x}$. $\mathbf{u}$ must be periodic and might require periodicity in its derivatives, depending on the physical characteristics of the actual actuators.
- $\mathbf{u}$ is a parameterized function $\mathbf{u}(\mathbf{x}, \mathbf{s})$, with parameters $\mathbf{s}$ that define the shape of the excitation. They can be the coefficients of polynomials, splines, a Taylor or Fourier series, or define piecewise constant/linear functions as they are typically used in optimal control.

With these assumptions, the system can be expressed simply as an extended form of the stride-to-stride transfer function $\mathbf{x}^{k+1} = P(\mathbf{x}^k, \mathbf{p}, \mathbf{s})$, which is parameterized via the tunable system parameters $\mathbf{p}$ and the controller parameters $\mathbf{s}$. Identical to the passive case, the creation of a gait is equivalent to finding periodic solutions of the Poincaré map

$$\bar{P}(\mathbf{x}^*, \mathbf{p}, \mathbf{s}^*) - \mathbf{x}^* = 0. \tag{13}$$

F. Optimization

Usually an infinite number of gait solutions exist, and a cost function $c(\mathbf{x}, \mathbf{p}, \mathbf{s})$ can be minimized while tuning the system parameters $\mathbf{p}$ and generating the control inputs $\mathbf{s}$. The function c can, for example, be used to express energy expendi-

ture, speed, or the average COT. The search for an optimal periodic gait can thus be stated as:

$$\min\{c(\mathbf{x},\mathbf{p},\mathbf{s})\}$$
$$s.t. \quad \overline{P}(\mathbf{x},\mathbf{p},\mathbf{s})-\mathbf{x}=0 \qquad (14)$$

This optimization problem can be solved in a number of different ways. In *single shooting*, the equations of motion are simply integrated over a full stride. Events are detected and processed along the way. Integration is stopped at the terminal event, and the cost function and the constraint violation is computed. Sequential quadratic programming is used to minimize c and drive the constraint violation to $\mathbf{0}$. Due to the long integration period, the mapping between the free parameters and the costs and constraints can be highly nonlinear, which degrades the convergence behavior of the optimization. The method can be improved by breaking the integration into a number of small segments that are evaluated separately. In *multiple shooting* the free variables are thus extended to include intermediate states at which the integration is re-started. Continuity between the individual segments is ensured by additional constraints. This increases the size of the optimization problem, but convergence is generally improved [18].

When the individual segments are reduced in length until they are equal to the step-size of the numerical solver, integration becomes unnecessary, and the constraint violations can be expressed analytically. In this method of *direct collocation* [9, 19], the intermediate states $\mathbf{x}(t_k)$ define a grid over which the solution is approximated as a piecewise cubic polynomial $\tilde{\mathbf{x}}(t)$. This ensures that the resulting trajectories are C^1 and instead of enforcing continuity over the borders of the individual segments, the constraints are introduced for the left hand side of the flow map:

$$\dot{\tilde{\mathbf{x}}}(t_k)\overset{!}{=}\dot{\mathbf{x}}(t_k)=f(\mathbf{x},\mathbf{p}) \qquad (15)$$

Since the approximation $\tilde{\mathbf{x}}(t)$ is (by definition) continuous, this method cannot reflect the discrete dynamic aspects of legged locomotion. To properly capture events and the associated jumps in the state variables, we need to define the expected event sequence a priori and break the full stride into a number of continuous intervals $\tilde{\mathbf{x}}^i(t)$ between the events. As additional constraints, the final state in each interval $\mathbf{x}^i(t_{end})$ must fulfill the event condition (1), and the state at the beginning of each interval must be equal to the outcome of the associated jump map (2) $\mathbf{x}^i(t_0)=g(\mathbf{x}^{i-1}(t_{end}),\mathbf{p})$. This also applies to the initial state $\mathbf{x}^1(t_0)$ and the outcome of the terminal jump map $g(\mathbf{x}^{end}(t_{end}),\mathbf{p})$, to create an overall periodic gait as defined in (10) and (13).

III. MATLAB FRAMEWORK

The presented concepts for modeling, simulation, gait creation, and gait analysis have been implemented in MATLAB R2010a, and can be accessed with additional material at
www.asl.ethz.ch/people/cremy/personal/GaitCreation
The framework contains methods for the simulation of hybrid dynamic systems, both passive and active. Functionality for data logging and visualization of simulations is available, and three models of different walking and running robots are provided as examples and as a starting point for further studies. Gait creation (i.e., the search for periodic solutions) and optimization is implemented with single shooting and direct collocation algorithms and functions for stability analysis of active and passive systems are provided. All concepts and functions are introduced along the provided examples.

Lagrangian mechanics are derived with the symbolic math toolbox and optimization/root search is performed using the *fsolve* and *fmincon* routines of the optimization toolbox.

IV. EXAMPLES

This section gives a brief overview of the models included in the framework. Provided here is a rather conceptual description. A more detailed explanation (including parameter choices and full state definitions) is given directly in the documentation of the source files. A number of sample problems and parameter studies were performed with the models to introduce different aspects of the framework's functionality. Their results are also reported below.

A. Passive Dynamic Walker

In a first example, gait creation and analysis were performed for a simple passive dynamic walking model (fig. 1). The mechanical system consisted of three segments: a main body with a point-mass m_1, and two legs with distributed mass (m_2, j_2). The legs (length l_0) ended in circular feet with radius r_{foot}. By means of an adjustable weight, the position of the center of gravity of the legs l_2 could be varied.

Since the legs are modeled as rigid bodies, one foot has to be rolling on the ground at all times while the other is swinging. This means the structure of the mechanical dynamics never changes. The dynamics are merely interrupted by the ground collision of the swing leg (when the role of stance and swing leg is also switched). A 'floating base' description (as in (4)) is thus unnecessary when stating the continuous dynamics, and the state of the system is described merely by the stance leg angle γ, the inter leg angle α, and the respective velocities. It is only for the jump map (when evaluating the collision that occurs when the swing leg hits the ground) that a floating base representation with the main body positions and speeds $x(\dot{x})$ and $y(\dot{y})$ as additional states is employed, and the post-impact velocities computed using equation (9). The mass

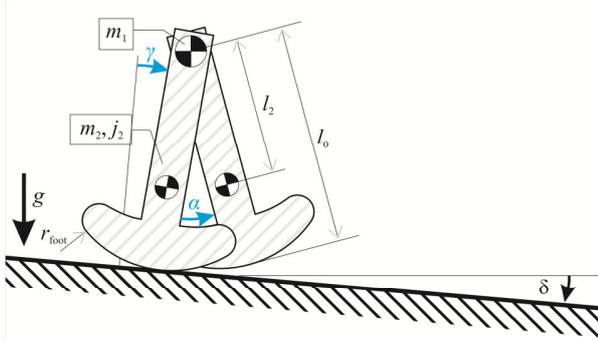matrix and differentiable force vector are derived for both cases analytically using standard Lagrangian mechanics.

Fig. 1. In the first example, the presented toolbox was used to create a steady gait of a passive dynamic walker. The leg CoG position l_2 was varied and the stability of the solution was examined.

Gaits were identified with direct shooting being used as the numerical method. Since the model is symmetric it was sufficient to evaluate only a half-stride for the computation of P. First-order stability (12) was evaluated as a function of l_2. The solution with the smallest spectral radius was obtained for $l_2 = 0.62 l_0$ where the magnitude of the largest eigenvalue was 0.77 (fig. 2). The corresponding gait had the initial conditions $(\gamma, \dot{\gamma}, \alpha, \dot{\alpha}) = (0.28, -0.48, -0.56, 0.08)$, with a step length of $0.56 l_0$ and a walking speed of 0.24.

B. Prismatic Monopod

A second example, a prismatically jointed monopod hopper with series elastic actuation (fig. 3), shows how actuation can be included in the presented framework. The model consists of a main body and two leg segments. Main body and upper leg are connected by a rotational hip joint, and the two leg segments are connected prismatically. All dimensions and system parameters are shown in figure 3.

At each joint, the motion of the two adjoining segments is coupled by a linear spring that is rigidly attached at one end to the distal segment. The other end could be moved by a servo-controlled motor. To avoid modeling of the dynamics of the actuator, it was assumed that a displacement u with velocity $\dot{u}$ can be generated. In this arrangement, joint, spring, and motor form a *series-elastic actuator* [20]. This has the great advantage that the actuation does not impede the natural dynamic hopping motion. This system is very similar to Marc Raibert's early hoppers [3] which employed a pneumatic spring that was actuated by actively increasing the air pressure during stance.
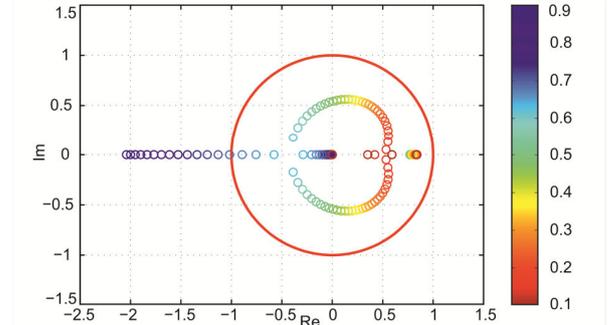

Fig. 2. Root locus plot of the Floquet multipliers of a passive dynamic walker. The Monodromy matrix was computed for various leg CoG positions l_2. The smallest spectral radius was obtained for $l_2 = 0.62$.

The continuous states of this system consist of the generalized coordinates $\mathbf{q}$, their derivatives $\dot{\mathbf{q}}$, time t, and the integrated positive mechanical actuator work (as given in the cost function) $W_{mech} = \int_0^t \max(0, F_l \dot{u}_l) + \max(0, T_\alpha \dot{u}_\alpha)\, dt$. A single binary state variable z registers whether the foot is in contact with the ground or not. The passive dynamic rebound of the system already resembles the desired hopping motion, but energy is dissipated by the plastic collisions and the damping in the spring. The initial apex height will hence not be re-reached by a purely passive hopper. To feed energy back into the system, a time based Fourier series,

$$\mathbf{u} = \mathbf{u}(t, \mathbf{s}) = \sum_i \mathbf{a}_i \sin(2\pi i f_{exct} \cdot t) + \mathbf{b}_i \cos(2\pi i f_{exct} \cdot t) \quad (15)$$

is used to control the motion of the actuators. The coefficients $\mathbf{a}_i$ and $\mathbf{b}_i$, and the excitation frequency $f_{exct} = 1/T_{exct}$, constituted the control parameter vector $\mathbf{s}$ and are determined in the optimization. Using such a series ensured that the activation function and its first derivative were periodic and (by limiting the number of terms in the series) did not exceed the closed loop position control bandwidth of the servo motors. Moreover, when looking for a periodic gait, the root search and optimization was initially performed for a small number of terms and then successively expanded. The Poincaré section (i.e., the terminal event) was defined by the excitation $e_{term}(\mathbf{x}, \mathbf{p}, \mathbf{s}) = t - T_{exct}$ to ensure that the active inputs are periodic over multiple steps. The remaining events (touch-down and lift-off) were detected and handled as described in section II.B.
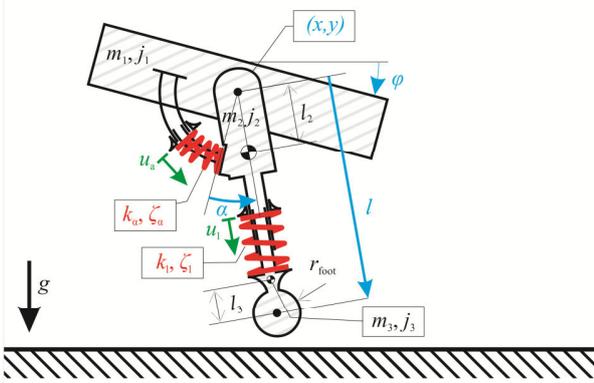
Fig. 3. To demonstrate how the framework can be used in active system, optimal periodic forward hopping was identified for the model of a planar hopping robot with series elastic actuation.

As an optimization criterion in (14), the integral of positive mechanical power was used as a simplified measure of energy consumption (a great comparison of biologically inspired optimization criteria can be found, for example, in [21]). It was evaluated as cost of transportation, i.e. normalized by the distance traveled within one stride: $c(\mathbf{x}, \mathbf{p}, \mathbf{s}) = W_{mech}/x$ (since all system parameters and states were already normalized to total mass m_0, leg length l_0, and gravity g, further normalization was unnecessary).

An optimal periodic hopping motion was created by solving the optimization problem (14) through direct collocation. With the optimal forward speed of $0.9\sqrt{l_0 g}$, the final cost of transportation was 0.098. Subsequently, the optimization was performed for pre-defined forward speeds, which were altered in the range of $0.1\sqrt{l_0 g}$ to $2.0\sqrt{l_0 g}$. Fig. 4 illustrates that similar to biological systems [11], an optimal speed exists with which the system should move when seeking to minimize energy consumption.
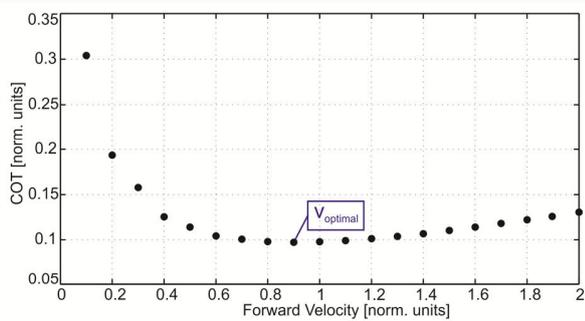

Fig 4. Cost of transportation of the one-legged hopper as a function of forward velocity. The natural passive dynamics of the hopper are best exploited for forward hopping with a velocity of $0.90\sqrt{g \cdot l_0}$.

C. Bounding Quadruped

The final example (the model of a planar bounding robot), shows how a basic control scheme can be created with the presented tools. With regard to the structure and actuation of the legs, the model (fig. 5) is identical to the hopper presented in section IV.B (it is similar to [22] but fully actuated). It is driven by series elastic actuators, which are periodically excited with parametric oscillations. The optimal forward speed was 0.7. When driven open-loop, the motion had two unstable conjugate eigenvalues with a magnitude of 1.56, and a stabilizing controller was thus introduced.

The basic idea for controlling the system was to modulate the coefficients of the excitation parameters $\mathbf{s}$ in response to a disturbance in the periodic motion. By extending the stability analysis of (11) to include perturbations in the activation parameters, we can state:

$$\Delta\mathbf{x}^{k+1} \approx \left.\frac{\partial \overline{P}(\mathbf{x}, \mathbf{s})}{\partial \mathbf{x}}\right|_{\mathbf{x}^*, \mathbf{s}^*} \cdot \Delta\mathbf{x}^k + \left.\frac{\partial \overline{P}(\mathbf{x}, \mathbf{s})}{\partial \mathbf{s}}\right|_{\mathbf{x}^*, \mathbf{s}^*} \cdot \Delta\mathbf{s}^k$$

Or simply:

$$\Delta\mathbf{x}^{k+1} \approx \mathbf{J_x} \cdot \Delta\mathbf{x}^k + \mathbf{J_s} \cdot \Delta\mathbf{s}^k . \qquad (16)$$

This is a discrete, linear, first order MIMO-system, for which standardized control approaches, such as LQR [23] can be applied. Here, we employed a state regulator $\Delta\mathbf{s} = \mathbf{K} \cdot \Delta\mathbf{x}$, with a desired closed-loop response $\mathbf{J}_{des} = \mathbf{J_x} + \mathbf{J_s}\mathbf{K}$, for which we reduced the natural unstable eigenvalues to a magnitude of 0.6. As the control input $\mathbf{u}$ must eventually be generated as a position trajectory by a servo-controlled motor, it should be continuously differentiable across subsequent strides, even for the controlled system. This means that the control input $\Delta\mathbf{s}$ shouldn't change the actuator position and velocity in the Poincaré section: $\mathbf{u}(0, \mathbf{s} + \Delta\mathbf{s}) = \mathbf{u}(T_{exct}, \mathbf{s})$ and $\dot{\mathbf{u}}(0, \mathbf{s} + \Delta\mathbf{s}) = \dot{\mathbf{u}}(T_{exct}, \mathbf{s})$ for all $\Delta\mathbf{s}$. For a fixed excitation frequency f_{exct}, the input $\mathbf{u}$ is linear in the Fourier coefficients $\mathbf{a}_i$ and $\mathbf{b}_i$. When we exclude f_{exct} from the set of control parameters, we can thus state: $\mathbf{u}(t, \mathbf{s} + \Delta\mathbf{s}) = \mathbf{u}(t, \mathbf{s}) + \mathbf{u}(t, \Delta\mathbf{s})$, and we express continuity by requiring $\mathbf{u}(0, \Delta\mathbf{s}) = \dot{\mathbf{u}}(0, \Delta\mathbf{s}) = 0$. This can be expressed as a linear constraint in the modulation of the excitation parameters $\Delta\mathbf{s}$:

$$\begin{pmatrix} 1 & 0 & \cdots & & 0 \\ 0 & 0 & \cdots & 0 & 1 & \cdots & 1 \\ 0 & 1 & \cdots & i & 0 & \cdots & 0 \end{pmatrix} \cdot \begin{pmatrix} \Delta f_{stride} \\ \Delta\mathbf{a}_1 \\ \vdots \\ \Delta\mathbf{a}_i \\ \Delta\mathbf{b}_1 \\ \vdots \\ \Delta\mathbf{b}_i \end{pmatrix} = \mathbf{C} \cdot \Delta\mathbf{s} = \mathbf{0} \qquad (17)$$

with the constraint matrix $\mathbf{C}$. This equation $\left(\mathbf{C}\Delta\mathbf{s} = \mathbf{C}\mathbf{K} \cdot \Delta\mathbf{x} = \mathbf{0} \mid \forall\mathbf{x}\right)$ must hold for all disturbances and

we thus require $\mathbf{CK} = \mathbf{0}$. When the number of control parameters $\mathbf{s}$ is large enough, the two requirements $\mathbf{J_s K} = \mathbf{J}_{des} - \mathbf{J_x}$ and $\mathbf{CK} = \mathbf{0}$ are fulfilled while minimizing the controller effort $\Delta\mathbf{s}$ using the pseudo inverse

$$\mathbf{K} = \begin{pmatrix} \mathbf{J_s} \\ \mathbf{C} \end{pmatrix}^{+} \begin{pmatrix} \mathbf{J}_{des} - \mathbf{J_x} \\ \mathbf{0} \end{pmatrix} \qquad (18)$$

which creates a stably controlled system (fig. 6) with all Floquet multipliers being smaller or equal to 0.6.
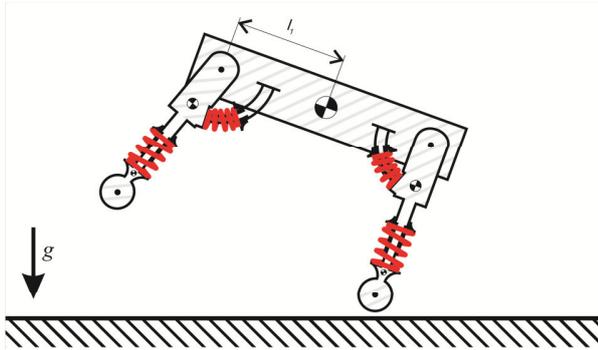

Fig. 5. To show a more complex multi-legged system, optimal bounding was created for the model of a planar robot. Structure and actuation are identical to the model introduced in section IV.B.

V. CONCLUSION

This paper introduced a framework for the creation of efficient running and walking gaits with robotic devices that are able to exploit passive/natural dynamics. The complete code, including sources for the presented examples and studies is freely available for further research and contains additional documentation. Videos of the models and gaits accompany this paper and are available online.
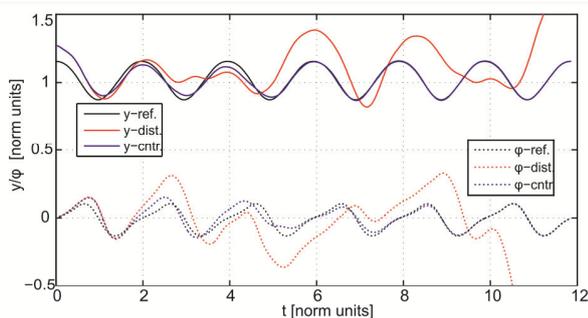

Fig. 6. When all states of the bounding model were disturbed by 10%, the controlled system returns to the original limit cycle (shown for vertical height y and pitch angle φ) within a few steps, while the uncontrolled system diverges and eventually falls.

REFERENCES

[1] T. McGeer, "Passive Dynamic Walking," *Int. J. Robot. Res.,* vol. 9, pp. 62-82, 1990.

[2] S. Mochon and T. A. McMahon, "Ballistic walking: an improved model," *Math. Biosci.,* vol. 52, pp. 241-60, 1980.

[3] M. H. Raibert, *Legged robots that balance*. Cambridge, Mass.: MIT Press, 1986.

[4] P. Bhounsule and A. Ruina, "Cornell Ranger: Energy-Optimal Control," presented at the Dynamic Walking 2009, Vancouver, Canada, 2009.

[5] M. Wisse, *et al.*, "Passive-Based Walking Robot," *Robot. & Autom. Mag.,* vol. 14, pp. 52-62, 2007.

[6] S. Collins, *et al.*, "Efficient Bipedal Robots Based on Passive-Dynamic Walkers," *Science,* vol. 307, pp. 1082-85, 2005.

[7] M. Vukobratovic and B. Borovac, "Zero-Moment Point — Thirty Five Years of its Life," *International Journal of Humanoid Robotics,* vol. 1, pp. 157-73, 2004.

[8] K. D. Mombaur, *et al.*, "Open-loop stability - a new paradigm for periodic optimal control and analysis of walking mechanisms," in *Robotics, Automation and Mechatronics, 2004 IEEE Conference on*, 2004, pp. 704-09 vol.2.

[9] M. Stelzer, *et al.*, "Efficient Dynamic Modeling, Numerical Optimal Control and Experimental Results for Various Gaits of a Quadruped Robot," in *International Conference on Climbing and Walking Robots CLAWAR*, Catania, Italy, 2007.

[10] M. Hutter, *et al.*, "Design of an Articulated Robotic Leg with Nonlinear Series Elastic Actuation," in *International Conference on Climbing and Walking Robots CLAWAR*, Istanbul, Turkey, 2009.

[11] D. F. Hoyt and C. R. Taylor, "Gait and the energetics of locomotion in horses," *Nature,* vol. 292, pp. 239-40, 1981.

[12] R. Goebel, *et al.*, "Hybrid dynamical systems," *Control Systems Magazine, IEEE,* vol. 29, pp. 28-93, 2009.

[13] C. Glocker, "On frictionless impact models in rigid-body systems " *Phil. Trans. R. Soc. A,* vol. 359, pp. 2385-04, 2001.

[14] A. Seyfarth, *et al.*, "A movement criterion for running," *J. Biomech.,* vol. 35, pp. 649-55, 2002.

[15] C. D. Remy, *et al.*, "Stability Analysis of Passive Dynamic Walking of Quadrupeds," *Int. J. Robot. Res.,* vol. 29, pp. 1173-85, 2010.

[16] E. H. Jackson, *Perspectives of Nonlinear Dynamics*: Cambridge University Press, 1990.

[17] T. G. Brown, "Studies in the Physiology of the Nervous System. VIII. Neural Balance and Reflex Reversal, with a Note on Progression in the Deceberate Guine-Pig," *Experimental Physiology,* vol. 4, pp. 273-88, 1911.

[18] H. G. Bock and K. J. Plitt, "A multiple shooting algorithm for direct solution of optimal control problems " in *9th IFAC World Congress*, Budapest, Hungary, 1985, pp. 242-47.

[19] O. Von Stryk, "Numerische Loesung optimaler Steuerungsprobleme: Diskretiesierung, Parameteroptimierung und Berechnung der adjungierten Variablen," PhD, TU München, München, 1994.

[20] D. W. Robinson, *et al.*, "Series elastic actuator development for a biomimetic walking robot," in *International Conference on Advanced Intelligent Mechatronics AIM*, Atlanta, USA, 1999, pp. 561-68.

[21] M. Srinivasan, "Fifteen observations on the structure of energy-minimizing gaits in many simple biped models," *Journal of The Royal Society Interface,* vol. 8, pp. 74-98, 2011.

[22] I. Poulakakis, *et al.*, "Modeling and experiments of untethered quadrupedal running with a bounding gait: The Scout II robot," *Int. J. Robot. Res.,* vol. 24, pp. 239-56, 2005.

[23] G. F. Franklin, *et al.*, *Digital Control of Dynamic Systems*, third ed. Menlo Park, CA: Addison Wesley Longamn, 1997.