Walking control of fully actuated robots based on the Bipedal SLIP model

Gianluca Garofalo, Christian Ott, and Alin Albu-Schäffer

Abstract— The goal of this paper is to generate and stabilize a periodic walking motion for a five degrees of freedom planar robot. First of all we will consider a biped version of the spring loaded inverted pendulum (SLIP), which shows open-loop stable behavior. Then we will control the robot behavior as close as possible to the simple model. In this way we take advantage of the open-loop stability of the walking pattern related to the SLIP, and additional control actions are used to increase the robustness of the system and reject external disturbances. To this end an upper level controller will deal with the stabilization of the SLIP model, while a lower level controller will map the simple virtual model onto the real robot dynamics. Two different approaches are implemented for the lower level: in the first one, we aim at exactly reproducing the same acceleration that a SLIP would have when put in the same condition, while in the second one, we aim at a simpler control law without exactly reproducing the aforementioned acceleration. The latter case is equivalent to considering a SLIP with additional external disturbances, which have to be handled by the upper level controller. Both approaches can successfully reproduce a periodic walking pattern for the robot.

I. INTRODUCTION

Since the interest in building humanoid robots able to interact with humans is increasing more and more during the last years, also the role of biped locomotion is gaining more importance.

The classic approaches are either to control the robot using the inverted pendulum model and the zero moment point (ZMP) [1], [2], [3] with typically complete actuation or, conversely, to build passive-dynamic walkers like robots with limited actuation efforts [4], [5], [6] which can exhibit a periodic limit cycle motion. Here, instead, the goal is to fully actuate the robot in order to reproduce, via the controller, the behavior of a passive walker for obtaining a periodic gait, as sketched in Fig. 1. In this way, we aim at utilizing the benefits of an open loop stable limit cycle for a robot, which is not a priori tailored to a specific set of periodic walking motions but capable of a variety of motion patterns.

To this end the spring loaded inverted pendulum (SLIP) model will be analyzed. For running and hopping motions, researchers usually consider a model with only one spring [7], [8], but for walking also a second spring/leg is needed [9], [10]. Since this paper is focusing on biped walking, this is the model that will be considered in the following. The main reason why it is useful to start with such a model is that it is one of the simplest models that can take into account the compliant behavior shown by humans

G. Garofalo, C. Ott and A. Albu-Schäffer are with the Institute of Robotics and Mechatronics, German Aerospace Center (DLR), Wessling, Germany. E-mail: gianluca.garofalo@dlr.de
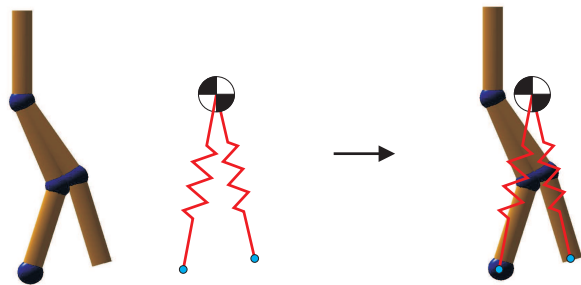
Fig. 1. The starting idea in this paper is to reshape, via the input torques, the dynamic behavior of the robot in order to make it as close as possible to the one of the simple conceptual model.

and animals during walking and running. Given the initial condition and appropriate parameter values it is possible to obtain a periodic pattern (limit cycle) for the center of mass (CoM in the following) resembling the motion and contact force profile obtained for example in human walking [9]. On the other hand due to its simplicity the SLIP model is way far from a complete dynamic description of the entire body motion. For example the effects of the swing leg are completely neglected and not taken into account in this simple model.

The SLIP model is also expected to be a useful template model for studying the effects of passive elastic elements in legged robotic walking systems. There are two main reasons for considering the presence of springs in locomotion [11]: energy storage (which means less energy consumption and unwanted heat production) and smaller force at the impact of the foot on the ground (which decrease the risk of damages).

Our aim in this paper is not to build a robot that behaves exactly like a SLIP model or some modified versions that also take into account the actuation part [12], [13], but to take advantage of the self stabilizing behavior of the model in order to obtain a desired behavior for the CoM. In other words we do not track any predefined trajectory for the CoM [14] but, conversely, we use the SLIP behavior as a target dynamic for a lower level controller. As a consequence, in the following the focus will not be put on foot steps or similar problems; these issues are automatically solved by the SLIP model.

The main contribution of this paper is to map the simple dynamics of the bipedal SLIP model to multi-body robots. In [15] a similar approach was followed for the single legged SLIP model for running motions. Herein, we focus on the
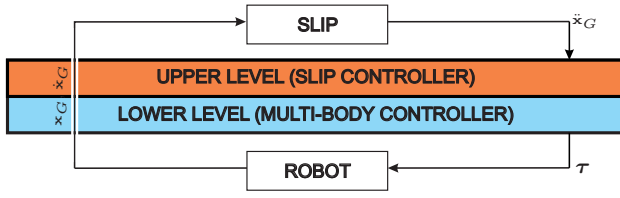
Fig. 2. Conceptual architecture of the controlled system. The controller (upper and lower level) plays the role of an interface between the virtual SLIP model and the real robot. Thanks to this interface the virtual system can be mapped onto the real one. The planning layer has been replaced requiring a desired dynamic behavior rather than desired trajectories. Knowing the actual position and velocity of the CoM of the robot, the expected acceleration is the one that a SLIP model would have had.
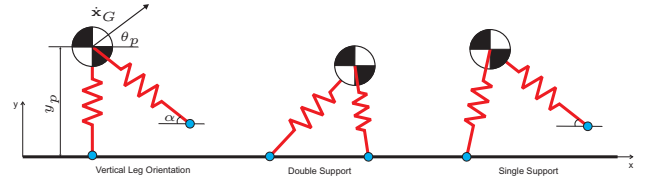

Fig. 3. SLIP model with the leg vertically oriented (also used as initial configuration), in the double support phase, and again in the single support phase (but not in the VLO condition). Note that because of the assumption of considering massless legs, the angle of attack α can instantaneously reach its desired value.

bipedal walking case. Several issues arise because of this. First of all some strategies should be considered to deal with the swing leg in terms of foot trajectory, treatment of the double support case and nonlinear terms in the dynamic model. These effects can not be neglected, but on the contrary they turn out to have a big influence on the required torques. This is one of the reasons why different kinds of control laws are analyzed. Also for the SLIP the control techniques are different than in [15], since the presence of the double support phase makes it difficult to extend the classical control approaches based on the analytic approximation of the model. As a result both control layers are modified and a complete new interface between the SLIP dynamics and the multi-body dynamics is obtained.

The paper is organized as follows: the next section will deal with the modeling of both the SLIP model and the multi-body robot. Then the control techniques will be considered, first for the SLIP (in section IV-A) and then for the multi-body robot (in section IV-B). Finally, simulation results will be presented in section V followed by the final discussion and outline of future work in section VI.

II. MODELING

One of the key points to keep in mind is that the controller is consisting of two levels. The upper level control is directly connected to the SLIP layer and is ensuring to obtain a periodic walking pattern despite some inevitable errors and disturbances, while the lower level control is forcing the multi-body robot dynamics to behave like a SLIP model. As a result, the controller plays the role of an interface between the two dynamics, making it possible to map the dynamics of the virtual SLIP onto the dynamics of the real robot, as shown in Fig. 2.

A. Dynamic model of the SLIP

The model, shown in Fig. 3, is simply given by two massless linear springs connected to each other in the hip, where all the mass is concentrated. Due to this assumption the swing leg can instantaneously change its configuration when it takes off from the ground, to reach the desired angle of attack for the next step. Obviously this will not be true for the multi-body robot, because of the inertia of the swing leg.

Using the following notation: m for the total mass, k and l_0 for the stiffness and rest length of the springs, $\mathbf{x}_G$ for the CoM position, $\mathbf{x}_{F_i}$ with $i = \{Left, Right\}$ for the foot contact point and $\mathbf{g}_0$ for the Earth's gravitational acceleration, the system's dynamic equation can be written as:

$$\ddot{\mathbf{x}}_G = \frac{1}{m}\left(\mathbf{f}_L\left(\mathbf{x}_G\right) + \mathbf{f}_R\left(\mathbf{x}_G\right)\right) + \mathbf{g}_0, \qquad (1)$$

where $\mathbf{f}_L(\mathbf{x}_G)$ and $\mathbf{f}_R(\mathbf{x}_G)$ are the spring forces due to the left and right leg, whose expression is:

$$\mathbf{f}_i\left(\mathbf{x}_G\right) = k\left(l_0 - \|\mathbf{x}_G - \mathbf{x}_{F_i}\|\right)\frac{\mathbf{x}_G - \mathbf{x}_{F_i}}{\|\mathbf{x}_G - \mathbf{x}_{F_i}\|}, \quad i = \{L, R\}.$$

In the following we will refer to the foot contact point simply as $\mathbf{x}_F$, but obviously $\mathbf{x}_F = \mathbf{x}_{F_L}$ or $\mathbf{x}_F = \mathbf{x}_{F_R}$ depending on the stance phase.

The classic approach for studying the stability of the SLIP system is to consider a Poincaré section [16] and check the eigenvalues of the monodromy matrix. A particularly convenient section is the one corresponding to the condition of vertical leg orientation (VLO) [10]. Taking additionally the energy conservation property of the SLIP model into account, in this case the dimension of the state space is reduced from four to two. In particular it is sufficient to use the CoM height and the angle of the CoM velocity as coordinates of the Poincaré section in order to have a complete description of the system for the periodic limit cycle analysis. In fact if we are interested just in the analysis of the limit cycle the position along the forward direction is not important, then once we know the height of the CoM in the VLO condition we automatically know both the value of the potential energy and the elastic energy. From this and the total energy value it is possible to compute the kinetic energy or in other words the magnitude of the velocity; this means that the only information missing is then the angle of this vector with respect to the reference.

B. Dynamic model of the multi-body robot

A five revolute joint planar robot with massless feet (represented in Fig. 4 with the definition of the joint angles) will be considered.

1) Single support: The dynamic model for the single support situation can be written as [17]:

$$\mathbf{M}\left(\mathbf{q}\right)\ddot{\mathbf{q}} + \mathbf{C}\left(\mathbf{q}, \dot{\mathbf{q}}\right)\dot{\mathbf{q}} + \mathbf{g}\left(\mathbf{q}\right) = \boldsymbol{\tau}, \qquad (2)$$
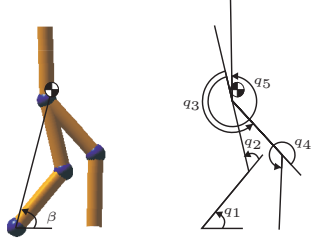
Fig. 4. Multi-body robot and definition of the joint variables accordingly to the Denavit-Hartenberg convention with the third axis coming out from the page [17]. The angle beta, instead of the time, is used to parameterize the offset of the swing foot with respect to the CoM. As a result the swing foot position is dependent on the CoM and no information about the time necessary to complete one step is needed.
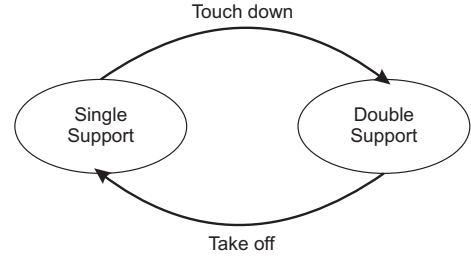

Fig. 5. The system that we want to study is called in literature a hybrid dynamical systems [19]. This means that beside the flow map, which describes the continuous dynamics of the system with a set of ordinary differential equations, also a set of discrete states (which only change because of an event) should be utilized.

where $\mathbf{q} \in \mathbb{R}^5$ are the generalized coordinates, $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{5 \times 5}$ is the inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{5 \times 5}$ is the Coriolis/centrifugal matrix, $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^5$ is the vector of the joint torques due to the gravity and $\boldsymbol{\tau} \in \mathbb{R}^5$ are the input torques. It's worth to notice that since in this paper only walking motions will be analyzed it is enough to consider a fixed base dynamic model, but the extension to the floating base case (needed for jumping and running motions) is straightforward and it is based on the same argumentation that will be followed for switching from the single to the double support phase in section II-B.2.

The model is complete when also the transitions between single support and double support are considered. These are caused by the two events touch down and take off of the legs, as shown in Fig. 5. Moreover, because of the impact, an energy loss will happen in this instant. This can be easily modeled considering that, at the impact, also an impulsive force is acting on the system and that it will produce an instantaneous change of the velocity (but not of the position) [18]. This means that the system will exhibit a variation of the generalized momentum in the joint space that can be expressed as:

$$\Delta(\mathbf{M}(\mathbf{q})\dot{\mathbf{q}}) = \mathbf{M}(\mathbf{q})\Delta\dot{\mathbf{q}} = \mathbf{J}_c^T(\mathbf{q})\boldsymbol{\lambda}, \qquad (3)$$

where $\mathbf{J}_c(\mathbf{q}) \in \mathbb{R}^{2 \times 5}$ is the Jacobian matrix that maps the joint velocities into the velocity along the constrained directions, while $\boldsymbol{\lambda} \in \mathbb{R}^2$ are the generalized reaction forces along the same directions.

If the impact is modeled as an inelastic impact, then the following relation must be considered:

$$\mathbf{J}_c(\mathbf{q})\dot{\mathbf{q}}^+ = \mathbf{0} \quad \Leftrightarrow \quad \mathbf{J}_c(\mathbf{q})\Delta\dot{\mathbf{q}} = -\mathbf{J}_c(\mathbf{q})\dot{\mathbf{q}}^-, \qquad (4)$$

where the superscripts $^-$ and $^+$ are used to express the quantities before and after the impact, respectively. Solving the system of equations (3) and (4) leads to:

$$\begin{aligned} \boldsymbol{\lambda} &= -\left(\mathbf{J}_c\mathbf{M}^{-1}\mathbf{J}_c^T\right)^{-1}\mathbf{J}_c\dot{\mathbf{q}}^- \\ \dot{\mathbf{q}}^+ &= (\mathbf{I} - \underbrace{\mathbf{M}^{-1}\mathbf{J}_c^T\left(\mathbf{J}_c\mathbf{M}^{-1}\mathbf{J}_c^T\right)^{-1}}_{\mathbf{J}_c^{+\mathbf{M}}}\mathbf{J}_c)\dot{\mathbf{q}}^- \end{aligned}, \quad (5)$$

wherein the matrix $\mathbf{N}(\mathbf{q}) = \mathbf{I} - \mathbf{J}_c^{+\mathbf{M}}\mathbf{J}_c$ is a projection matrix, which projects the joint velocities into the null space

of the Jacobian matrix $\mathbf{J}_c$. Therefore, (5) can be interpreted such that only the component of $\dot{\mathbf{q}}$ that is causing zero velocity at the contact point is kept after the impact. On the other hand, if we compute the energy loss we obtain:

$$\begin{aligned} 2\Delta E &= \dot{\mathbf{q}}^{+^T}\mathbf{M}\dot{\mathbf{q}}^+ - \dot{\mathbf{q}}^{-^T}\mathbf{M}\dot{\mathbf{q}}^- \\ &= \dot{\mathbf{q}}^{-^T}\left(\mathbf{N}^T\mathbf{M}\mathbf{N} - \mathbf{M}\right)\dot{\mathbf{q}}^- \\ &= -\dot{\mathbf{q}}^{-^T}\mathbf{J}_c^T\left(\mathbf{J}_c\mathbf{M}^{-1}\mathbf{J}_c^T\right)^{-1}\mathbf{J}_c\dot{\mathbf{q}}^-, \end{aligned} \quad (6)$$

where $\mathbf{J}_c(\mathbf{q})\dot{\mathbf{q}}^-$ is the velocity of the impact point right before the impact, in other words the pseudo kinetic energy related to the impact point is the energy that will be lost in the impact.

2) Double support: Assuming to have no sliding between the feet and a rigid floor, an additional constraint must be added to the model in the double support phase (just like has been previously done for modeling the impact):

$$\mathbf{J}_c(\mathbf{q})\dot{\mathbf{q}} = \mathbf{0}, \qquad (7)$$

which can also be expressed as:

$$\mathbf{J}_{H_L}(\mathbf{q}_L)\dot{\mathbf{q}}_L = \mathbf{J}_{H_R}(\mathbf{q}_R)\dot{\mathbf{q}}_R, \qquad (8)$$

where $\mathbf{J}_{H_i}(\mathbf{q}_i) \, i = \{L, R\}$ gives the velocity of the hip with respect to the correspondent foot and $\mathbf{q}_i \, i = \{L, R\}$ are the joint variables of the correspondent leg. Assuming that at least one of the two Jacobian matrices is nonsingular[1], then:

$$\dot{\mathbf{q}}_i = \mathbf{J}_{H_i}^{-1}(\mathbf{q}_i)\mathbf{J}_{H_j}(\mathbf{q}_j)\dot{\mathbf{q}}_j, \qquad (9)$$

and is possible to write:

$$\dot{\mathbf{q}} = \underbrace{\begin{bmatrix} \mathbf{J}_{H_i}^{-1}\mathbf{J}_{H_j} & \mathbf{0} \\ \mathbf{I} \end{bmatrix}}_{\boldsymbol{\Gamma}(\mathbf{q})}\underbrace{\begin{bmatrix} \dot{\mathbf{q}}_j \\ \dot{q}_5 \end{bmatrix}}_{\dot{\mathbf{q}}_a}, \qquad (10)$$

where q_5 is the joint angle of the trunk not affected by the change between single and double support.

If we differentiate (10) with respect to time, then substitute $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ in (2), and pre-multiply everything by $\boldsymbol{\Gamma}^T(\mathbf{q})$, we get:

$$\mathbf{M}_a(\mathbf{q})\ddot{\mathbf{q}}_a + \mathbf{C}_a(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}_a + \mathbf{g}_a(\mathbf{q}) = \boldsymbol{\tau}_a, \qquad (11)$$

where $\mathbf{M}_a = \boldsymbol{\Gamma}^T\mathbf{M}\boldsymbol{\Gamma}$, $\mathbf{C}_a = \boldsymbol{\Gamma}^T\left(\mathbf{C}\boldsymbol{\Gamma} + \mathbf{M}\dot{\boldsymbol{\Gamma}}\right)$, $\mathbf{g}_a = \boldsymbol{\Gamma}^T\mathbf{g}$ and $\boldsymbol{\tau}_a = \boldsymbol{\Gamma}^T\boldsymbol{\tau}$.
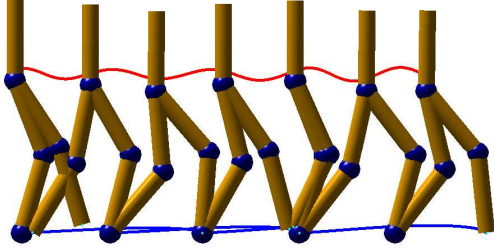
[1]If both are singular the system is in a complete rest.

Fig. 6. Overlapped configurations of the robot after consecutive steps. The red and blue lines are respectively the CoM and swing foot trajectory.

III. PLANNING

While the SLIP model defines the CoM trajectory and the target angle of attack for the next step, it is still necessary to define a trajectory for the swing leg, since this information is completely missing in the SLIP model. In order to be as close as possible to the virtual system that the robot is trying to reproduce, a good choice could be to just plan the relative position of the swing foot with respect to the CoM, ensuring that the foot will have reached the desired relative position (defined by the desired angle of attack) before the touch down event. Furthermore, we use the angle $\beta(\mathbf{x}_G)$ related to the line from the stance foot to the CoM as shown in Fig. 4 as a parameter for the trajectory. Then the desired foot position is given by

$$\mathbf{x}_{F_d} = \mathbf{x}_G + \mathbf{\Pi}\left(\beta(\mathbf{x}_G)\right) , \qquad (12)$$

where $\mathbf{\Pi}\left(\beta(\mathbf{x}_G)\right)$ is the interpolating function describing the shape of the foot trajectory relative to the CoM. Since it is also necessary to ensure the continuity of the desired trajectory, a cubic spline with desired initial and final values and two "virtual instants" [17] is used. The desired foot position cannot be used as a desired value for a conventional trajectory tracking controller since it depends on the current state. Instead, we formulate a virtual constraint [20] by defining the error term

$$\boldsymbol{\xi} = \mathbf{x}_F - \mathbf{x}_G - \mathbf{\Pi}\left(\beta(\mathbf{x}_G)\right) , \qquad (13)$$

which should be kept equal to $\boldsymbol{\xi}_d = \mathbf{0}$ by the controller.

Figure 6 shows overlapped configurations of the robot during the path. The red line represent the CoM trajectory, while the blue line the swing foot trajectory.

IV. CONTROL

A. Upper Level: Total energy control

The first control law that will be presented is the one responsible to cope with probably the biggest difference between the conservative SLIP model and the multi-body robot: the loss of energy at the impact. The energy control law acts based on the state of the SLIP model and is not only re-injecting energy in the system any time the total

level is decreasing but also dissipating energy when the level is increasing. This is different from what was done in [15], where an energy correction law is not continuously acting (based on the error between the desired level and the measured one), but intervening before the impact based on a pre-computed value of the energy loss.

The starting point is to consider the expression of the power:

$$P = \dot{\mathbf{x}}_G^T \left(m\ddot{\mathbf{x}}_G - \mathbf{f}_L - \mathbf{f}_R - m\mathbf{g}_0 \right). \qquad (14)$$

Since the power is the derivative of the energy and the desired value of the energy is constant, the problem of controlling the energy to the desired value can be solved via an energy controller that ensures:

$$P = k_E \left(E_d - E \right), \qquad (15)$$

where k_E is a positive gain and E_d is the desired constant total energy. This is possible if the resulting acceleration on the CoM is the one of a pure SLIP model plus an additional term with the following expression:

$$\mathbf{\Delta g} = \frac{1}{m} \frac{\dot{\mathbf{x}}_G}{\|\dot{\mathbf{x}}_G\|^2} k_E \left(E_d - E \right). \qquad (16)$$

This means that the controlled SLIP model is walking under the effect of a virtual state dependent gravity field. The term $\mathbf{\Delta g}$ is always in the direction of the velocity, increasing or decreasing the kinetic energy in order to keep the total energy to the desired value. Equation (1) thus is modified to

$$\ddot{\mathbf{x}}_G = \frac{1}{m} \left(\mathbf{f}_L \left(\mathbf{x}_G \right) + \mathbf{f}_R \left(\mathbf{x}_G \right) \right) + \mathbf{g}_v, \qquad (17)$$

with $\mathbf{g}_v = \mathbf{g}_0 + \mathbf{\Delta g}$.

B. Lower level: Multi-Body Control

In the following two different approaches will be presented in order to make the robot behave like the simple virtual model. The first approach is based on an exact feedback linearization of the robot dynamics and can be seen as an extension of the control law presented in [15] for an hopper to a bipedal SLIP for walking. The second approach, presented in section IV-B.2, aims at a simpler control law which requires less information about the system. Therefore, we will separately consider the different tasks that need to be executed.

1) Feedback linearization: As proposed in [15], it is possible to map the SLIP model on the real robot dynamics at an acceleration level. The idea is to ensure that the acceleration of the CoM of the real robot is the same as in the SLIP model. The control law can be derived following the same steps needed to derive the classic impedance control law with feedback linearization.

Three main tasks can be identified:

- CoM acceleration $\ddot{\mathbf{x}}_G$
- virtual constraint for the swing foot trajectory $\boldsymbol{\xi}$
- trunk orientation $\theta_T = q_1 + q_2 + q_5$.

The corresponding velocities can be computed from

$$\dot{\mathbf{x}}_t = \mathbf{J}_t \left(\mathbf{q} \right) \dot{\mathbf{q}}, \qquad (18)$$

where $\dot{\mathbf{x}}_t = \begin{bmatrix} \dot{\mathbf{x}}_G^T & \dot{\boldsymbol{\xi}}^T & \dot{\theta}_T^T \end{bmatrix}^T$ is the velocity in the task space and $\mathbf{J}_t(\mathbf{q}) = \begin{bmatrix} \mathbf{J}_G^T & \mathbf{J}_\xi^T & \mathbf{J}_\theta^T \end{bmatrix}^T$ the corresponding Jacobian matrix. Considering the time derivative of (18) and substituting $\ddot{\mathbf{q}}$ from (2), we get:

$$\ddot{\mathbf{x}}_t = \mathbf{J}_t \mathbf{M}^{-1} (\boldsymbol{\tau} - \mathbf{C}\dot{\mathbf{q}} - \mathbf{g}) + \dot{\mathbf{J}}_t \dot{\mathbf{q}}. \tag{19}$$

At this point with the following choice of the torques[2]:

$$\boldsymbol{\tau} = \mathbf{M} \mathbf{J}_t^{-1} \left(\ddot{\mathbf{x}}_{t_d} - \dot{\mathbf{J}}_t \dot{\mathbf{q}} \right) + \mathbf{C}\dot{\mathbf{q}} + \mathbf{g}, \tag{20}$$

we obtain $\ddot{\mathbf{x}}_t = \ddot{\mathbf{x}}_{t_d}$. Choosing

$$\ddot{\mathbf{x}}_{t_d} := \begin{bmatrix} \frac{1}{m} \left(\mathbf{f}_L(\mathbf{x}_G) + \mathbf{f}_R(\mathbf{x}_G) \right) + \mathbf{g}_v \\ -\mathbf{K}_{P_\xi} \boldsymbol{\xi} - \mathbf{K}_{D_\xi} \dot{\boldsymbol{\xi}} \\ \ddot{\theta}_{T_d} + k_{P_T} \tilde{\theta}_T + k_{D_T} \dot{\tilde{\theta}}_T \end{bmatrix}, \tag{21}$$

where $\tilde{\theta}_T = \theta_{T_d} - \theta_T$, ensures that $\ddot{x}_G$ corresponds exactly to (17) and that $\boldsymbol{\xi} \to \mathbf{0}$ and $\theta_T \to \theta_{T_d}$. In the simulations reported in section V, the desired trunk orientation is kept vertically ($\ddot{\theta}_{T_d} = \dot{\theta}_{T_d} = 0$).

2) Multi-priority control law: From the previous control law it is possible to recognize that three different types of tasks are considered in the defined task space: The CoM task is requiring to produce a desired acceleration according to a desired dynamic. The swing foot task implements a virtual constraint. The trunk tasks corresponds to trajectory tracking task. The idea is then to separate these different types of tasks and use simpler control laws to fulfill each of them. In the end, the required torques will be added using a null space projector.

First of all a higher priority will be assigned to the CoM task, since the main goal is still to reproduce a SLIP like motion. In the end what is really important for this task is just to produce the forces of virtual springs attached to the CoM.

Once it has been understood from the SLIP model that the presence of compliant elements is useful, it is convenient to design simpler control laws than the feedback linearization while the self-stabilizing behavior will be augmented by the upper level SLIP stabilization. In other words the goal for this task is just to map the virtual spring forces in joint torques accordingly to the transpose of the Jacobian matrix that relates the joint velocity to the velocity of the CoM and without any kind of non linear compensation:

$$\boldsymbol{\tau}_G = \mathbf{J}_G^T \left(\mathbf{f}_L(\mathbf{x}_G) + \mathbf{f}_R(\mathbf{x}_G) + m \boldsymbol{\Delta} \mathbf{g} \right), \tag{22}$$

where obviously also the extra component of the virtual gravity field must be taken into account.

In case of $\dot{\theta}_{T_d} = 0$, for the other tasks simple PD control laws with gravity compensation can be used:

$$\boldsymbol{\tau}_{PD} = \begin{bmatrix} \mathbf{J}_\xi^T & \mathbf{J}_\theta^T \end{bmatrix} \begin{bmatrix} -\mathbf{K}_{P_\xi} \boldsymbol{\xi} - \mathbf{K}_{D_\xi} \dot{\boldsymbol{\xi}} + \mathbf{g}_\xi \\ k_{P_T} \tilde{\theta}_T - k_{D_T} \dot{\theta}_T + \mathbf{g}_\theta \end{bmatrix}, \tag{23}$$

---

[2]In the following we will assume to be far enough from the singular configurations. Further on in the section additional torques will be introduced in order to cope with this problem.

where $\mathbf{g}_\xi$ and $\mathbf{g}_\theta$ are the projections of the gravity torques in the corresponding space.

This control law is known to be asymptotically stable and less sensitive to model uncertainties than feedback linearization [21].

At this point adding the resulting torques will cause the two tasks to interfere with each other. To avoid this, the torques computed for the second task will be projected in the null space of the transpose of the Jacobian matrix of the higher priority task. This means that the torques produced to fulfill the second task will produce no force on the CoM which will mainly move under the effect of the gravity field and the virtual springs. Obviously this also means that the second priority task will be fulfilled as long as it is not interfering with the other one.

Before concluding this section it is worth to mention that also some torques that push the robot away from the singularity should always be considered. The classic approach is to consider the kinematic manipulability measure [22]:

$$m_{kin_i}(\mathbf{q}) = \sqrt{det\left(\mathbf{J}_{H_i}\mathbf{J}_{H_i}^T\right)} \qquad i = \{L, R\} \tag{24}$$

and then choosing a singularity avoidance potential that will be derived to obtain the torques. In addition it is possible to consider a damping term also for this task. The idea is that, considering a quadratic form for the potential, the corresponding torque will be similar to a spring like term where the transpose of $\mathbf{J}_{kin}(\mathbf{q}) = \frac{\partial m_{kin}(\mathbf{q})}{\partial \mathbf{q}}$ is responsible for projecting the force from one space to the other. At this point (23) suggests the following choice for the new torques:

$$\boldsymbol{\tau}_{SA_i} = \begin{cases} -\mathbf{J}_{kin_i}^T \left(k_P \tilde{m}_{kin_i} + k_D \mathbf{J}_{kin_i} \dot{\mathbf{q}} \right) & m_{kin_i} \leqslant m_0 \\ \mathbf{0} & m_{kin_i} > m_0 \end{cases}, \tag{25}$$

where $\tilde{m}_{kin_i} = m_{kin_i} - m_0$ and m_0 is a threshold which indicates that the robot is close enough to the singular configuration to activate the corresponding torques.

The final expression of the torques is then:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_{SA_L} + \boldsymbol{\tau}_{SA_R} + \boldsymbol{\tau}_G + \left(\mathbf{I} - \mathbf{J}_G^T \left(\mathbf{J}_G^T \right)^{+\mathbf{M}} \right) \boldsymbol{\tau}_{PD}. \tag{26}$$

3) Double support and redistribution of torques: As seen in II-B.2 the expression of the dynamic model during the double support is formally equal to the one obtained for the single support. This means that the control laws computed before can be directly applied also in this case provided to consider the matrices of the new dynamic model instead of the old ones and to remove the swing foot coordinates from the task space. In this way is just possible to compute the torques $\boldsymbol{\tau}_a$ and not all the torques $\boldsymbol{\tau}$. To obtain the latter the relation $\boldsymbol{\tau}_a = \boldsymbol{\Gamma}^T \boldsymbol{\tau}$ between the torques is taken into account. From this one we obtain:

$$\boldsymbol{\tau} = \left(\boldsymbol{\Gamma}^T \right)^{+\mathbf{w}} \boldsymbol{\tau}_a + \left(\mathbf{I} - \left(\boldsymbol{\Gamma}^T \right)^{+\mathbf{w}} \boldsymbol{\Gamma}^T \right) \boldsymbol{\tau}_d, \tag{27}$$

where $\mathbf{W}$ is a positive definite matrix, which induces a weighted norm for the torques. It is useful to penalize a big value of some torques (e.g. ankle torque) more than the others. In addition, we use the associated null space projector to keep the torques as close as possible to some desired value.

mass	$m = 80\,\mathrm{kg}$
angle of attack	$\alpha = 1.2\,\mathrm{rad}$
rest length	$l_0 = 1\,\mathrm{m}$
stiffness	$k = 15696\,\mathrm{N/m}$
total energy	$E = 824\,\mathrm{J}$

	shank	thigh	trunk
mass	$5\,\mathrm{kg}$	$5\,\mathrm{kg}$	$60\,\mathrm{kg}$
inertia	$0.01\,\mathrm{kg\ m^2}$	$0.01\,\mathrm{kg\ m^2}$	$1\,\mathrm{kg\ m^2}$
length	$0.48\,\mathrm{m}$	$0.48\,\mathrm{m}$	$0.48\,\mathrm{m}$

C. A remark on stability

Since we try to reproduce the SLIP behavior, also the stability analysis will be based on the same approach. In other words we will consider again the same Poincaré map for the system. To be precise it should be mentioned that a more formally correct way to analyze the system is to consider a three-dimensional Poincaré map. For the multi-body robot, in fact, the total energy is not exactly constant, but only controlled to be constant. Therefore, it should be considered as the third coordinate on the Poincaré section. Nevertheless, if the total energy control is fast enough to ensure that the total energy does not deviate far from its desired value the approximation of considering a constant energy level is reasonable.

V. SIMULATIONS

In the following simulation study, we compare the limit cycle behavior of the SLIP model with the ones of the two multi-body robot controllers. The simulation parameters of the SLIP dynamics are adopted from the simulation results in [10] and are shown in table I. The parameters of the multi-body dynamics are shown in table II. The CoM of each segment was set to the center of the segment. Moreover a CLIK algorithm [17] is used in order to ensure that both system will start with the same initial conditions and in the same configuration (VLO configuration).

The simulation environment is implemented in MATLAB using a sampling time of $0.1\,\mathrm{ms}$ and the function ode45 to integrate the equations.

Figure 7 shows the vertical motion of the CoM for the SLIP model and the controlled multi-body system with the controllers from section IV-B.1 and IV-B.2. Even the feedback linearization based controller deviates from the trajectories of the SLIP model due to the energy loss at each impact. One can see that the deviation of the simpler control law is larger, but it still results in a quasi-periodic walking motion.

The vertical ground reaction force for the SLIP model and the forces due to the virtual springs for the two controllers
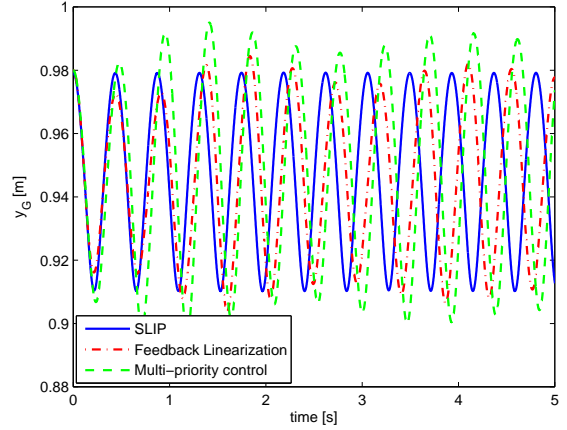

Fig. 7. Vertical motion of the CoM for the two multi-body controllers compared with the SLIP dynamics.
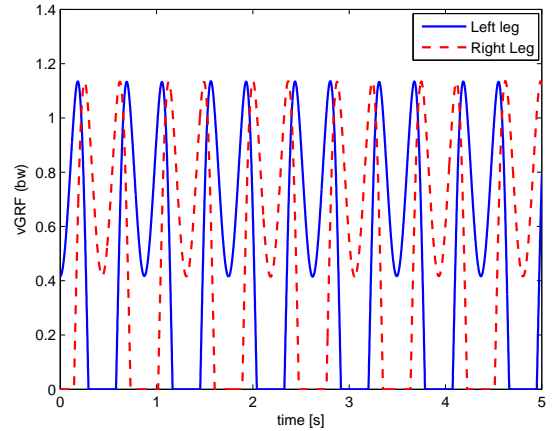

Fig. 8. Vertical ground reaction force of the SLIP model.

are shown in Fig. 8 to 10. One can see that the variations in the force are higher for the multi-body model.

The effect of the energy control approach (16) is presented in Fig. 11 with the feedback linearization based controller and different values of the controller gain k_E. One can see that without the energy controller (i.e. for $k_E = 0$) the multi-body system instantaneously loses energy at each step due to the impact. The red dash-dotted line and the green dashed line show the total energy for two different controller gains. This evaluation verifies that the proposed energy control method efficiently compensates for the energy loss due to the impacts and the energy is recovered according to the time constant in (15). Figure 12 shows the results with the simpler control law from section IV-B.2. Even if there is a larger variation in the total energy now due to the uncompensated nonlinear terms in the closed loop dynamics, also for this controller the energy loss at the impact is compensated.

Finally, Fig. 13 and 14 show the behavior of the two controllers on the Poincaré section. One can see that both controllers converge to a fixed point, which however is
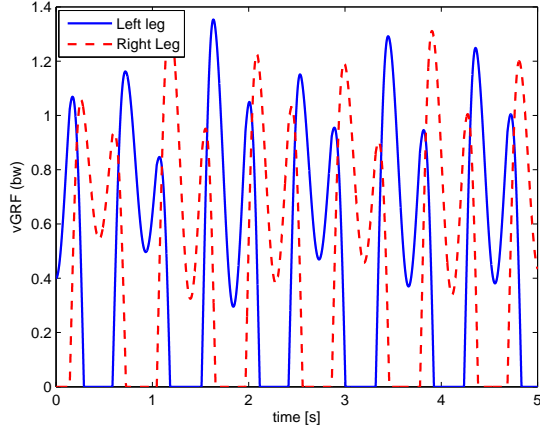
Fig. 9. Virtual springs forces of the multi-body system with the feedback linearization based controller.
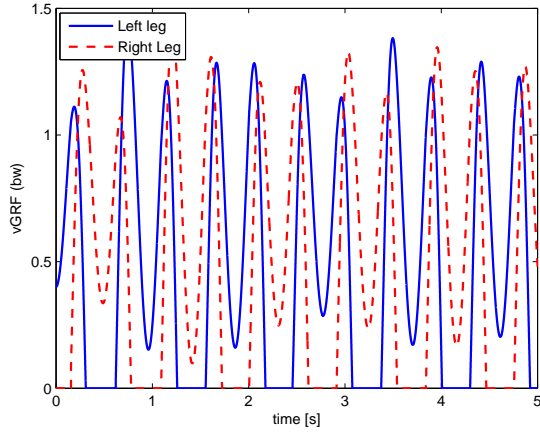

Fig. 10. Virtual springs forces of the multi-body system with the simplified control law.
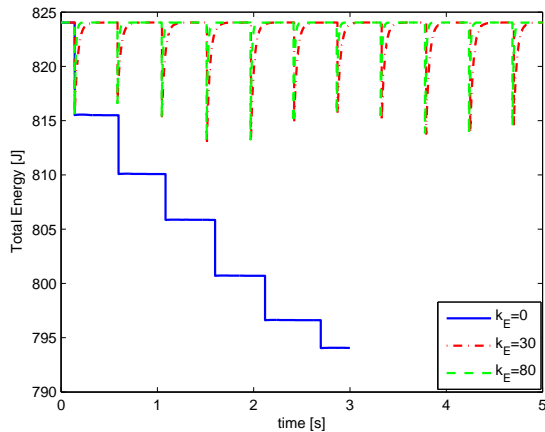

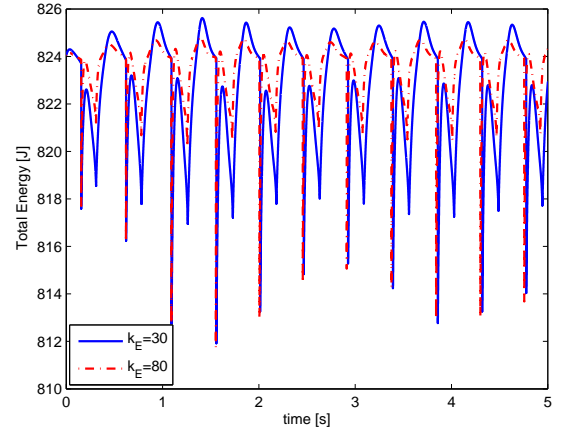Fig. 11. Energy control in the multi-body model using the feedback linearization based controller.

Fig. 12. Energy control in the multi-body model using the simplified controller.
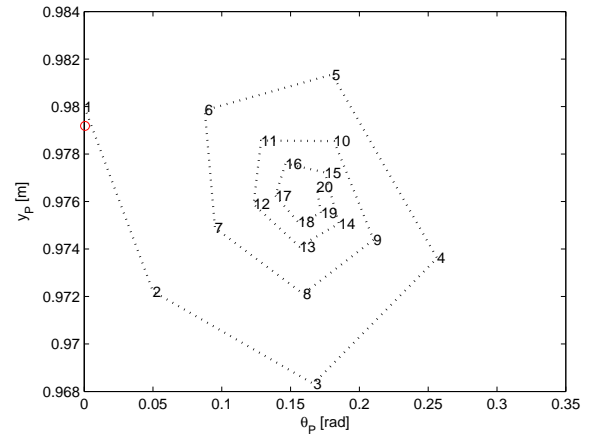

Fig. 13. Poincaré section of the multi-body model with the feedback linearization based controller. The red circle is the fixed point for the original SLIP model.

different to the original fixed point of the SLIP model, represented with a red circle in the figures.

VI. CONCLUSION AND FUTURE WORKS

In this paper we utilize the dynamics of the bipedal SLIP model as a template model for the control of a multi-body robot. The controller acts on two levels. In order to compensate for the energy loss that is inherent to the impacts at the foot touch down, an energy control algorithm for the SLIP model is proposed. In contrast to previous solutions in the literature, this controller is based on a continuous regulation of the energy and does not need any prediction of the energy loss for the next impact. The controlled SLIP dynamics is then used as a desired dynamic for a lower level controller designed for the multi-body robot dynamics. Two solutions of different complexity are proposed and compared. A multi-body robot controller based on feedback linearization results in a very close emulation of the SLIP behavior. Moreover, a simpler control approach is discussed, which avoids the
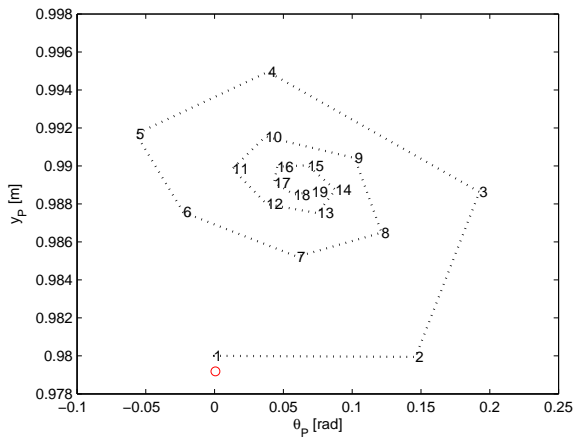
Fig. 14. Poincaré section of the multi-body model with the simplified controller. The red circle is the fixed point for the original SLIP model.

computational burden and parameter sensitivity problems of the feedback linearization based controller. While from a computational point of view the implementation of the proposed controllers is not more difficult than the usual ones used for example in manipulation tasks, it should be mentioned that the additional constrain of the ZMP has to be taken into account when a finite support polygon is considered.

The proposed controllers have been shown in simulations to produce a stable limit cycle behavior similar to the SLIP dynamics. Due to the energy loss at the foot touch down and the nonlinearities in the dynamics, the fixed point on the Poincaré section, however, differs from the original fixed point of the SLIP model. Therefore, one of our future works will be to investigate how the resulting limit cycle and the corresponding fixed point on the Poincaré section can be actively controlled. Moreover, we plan to investigate on the use of passive elastic components in order to reduce the control effort in the realization of SLIP based walking patterns.

ACKNOWLEDGMENTS

REFERENCES

[1] M. Vukobratovic and B. Borovac, "Zero-moment point - thirty five years of its life," *I. J. Humanoid Robotics*, vol. 2, no. 2, pp. 225–227, 2005.

[2] S. Kajita and K. Tani, "Study of dynamic biped locomotion on rugged terrain-derivation and application of the linear inverted pendulum mode."

[3] P.-B. Wieber, "Trajectory free linear model predictive control for stable walking in the presence of strong perturbations," *IEEE-RAS International Conference on Humanoid Robots*, 2006.

[4] T. McGeer, "Passive dynamic walking," *Int. Journal of Robotics Research*, vol. 9, no. 2, pp. 62–82, 1990.

[5] S. Collins, A. Ruina, R. Tedrake, and M. Wisse, "Efficient bipedal robots based on passive-dynamic walkers," *Science*, vol. 307, no. 5712, pp. 1082–1085, 2005.

[6] K. Sreenath, H.-W. Park, and J. W. Grizzle, "Embedding active force control within the compliant hybrid zero dynamics to achieve stable, fast running on MABEL," *The International Journal of Robotics Research*, 2011, submitted.

[7] R. Blickhan, "The spring-mass model for running and hopping," *Journal of Biomechanics*, vol. 22, pp. 1217–1227, 1989.

[8] A. Seyfarth, H. Geyer, M. Gnther, and R. Blickhan, "A movement criterion for running." *Journal of Biomechanics*, vol. 35, no. 5, pp. 649–55, 2002.

[9] H. Geyer, A. Seyfarth, and R. Blickhan, "Compliant leg behavior explains basic dynamics of walking and running," *Proc. R. Soc. B*, vol. 273, no. 1603, p. 28612867, 2006.

[10] J. Rummel, Y. Blum, and A. Seyfarth, "Robust and efficient walking with spring-like legs," *Bioinspiration and Biomimetics*, 2010.

[11] R. M. Alexander, "Three uses for springs in legged locomotion," *Int. Journal of Robotics Research*, vol. 9, no. 2, pp. 53–61, 1990.

[12] M. M. Ankarali and U. Saranli, "Analysis and control of a dissipative spring-mass hopper with torque actuation," *Robotics: Science and Systems*, 2010.

[13] D. Koepl, K. Kemper, and J. Hurst, "Force control for spring-mass walking and running," *IEEE Conference on Advanced Intelligent Mechatronics*, 2010.

[14] M. Ahmadi and M. Buehler, "Controlled passive dynamic running experiments with the arl-monopod ii," *IEEE Transactions on Robotics*, vol. 22, no. 5, p. 974986, 2006.

[15] M. Hutter, D. Remy, M. A. Höpflinger, and R. Siegwart, "Slip running with an articulated robotic leg," *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pp. 4934–4939, 2010.

[16] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2002.

[17] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*, 1st ed. Springer Publishing Company, Incorporated, 2008.

[18] X. Mu and Q. Wu, "On impact dynamics and contact events for biped robots via impact effects." *IEEE transactions on systems man and cybernetics Part B Cybernetics a publication of the IEEE Systems Man and Cybernetics Society*, vol. 36, no. 6, pp. 1364–1372, 2006.

[19] R. Goebel, R. Sanfelice, and A. Teel, "Hybrid dynamical systems," *IEEE Control Systems Magazine*, vol. 29, no. 2, pp. 28–93, 2009.

[20] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, *Feedback Control of Dynamic Bipedal Robot Locomotion*. CRC Press, 2007.

[21] B. Paden and R. Panja, "A globally asymptotically stable 'pd+' controller for robot manipulators," *International Journal of Control*, vol. 47, no. 6, pp. 1697–1712, 1988.

[22] T. Yoshikawa, *Foundations of Robotics: Analysis and Control*. Cambridge, MA: MIT Press, 1990.