Three-Dimensional Bipedal Walking Control Based on Divergent Component of Motion

Johannes Englsberger, Christian Ott, and Alin Albu-Schäffer

***Abstract*—In this paper, the concept of divergent component of motion (DCM, also called "Capture Point") is extended to 3-D. We introduce the "Enhanced Centroidal Moment Pivot point" (eCMP) and the "Virtual Repellent Point" (VRP), which allow for the encoding of both direction and magnitude of the external forces and the total force (i.e., external plus gravitational forces) acting on the robot. Based on eCMP, VRP, and DCM, we present methods for real-time planning and tracking control of DCM trajectories in 3-D. The basic DCM trajectory generator is extended to produce continuous leg force profiles and to facilitate the use of toe-off motion during double support. The robustness of the proposed control framework is thoroughly examined, and its capabilities are verified both in simulations and experiments.**

***Index Terms*—Bipedal robots, capture point, divergent component of motion, rough terrain, walking control.**

I. Introduction

The basic challenge in robotic locomotion is how to maneuver the robot's center of mass (CoM) through space. The dynamics of the CoM motion is decoupled from the rest of the robot dynamics [1], [2] and is only affected by gravitational force $\boldsymbol{F}_g$ and external forces $\boldsymbol{F}_{\text{ext}}$ (Newton's second law):

$$\ddot{\boldsymbol{x}} = \frac{1}{m}\,\boldsymbol{F} = \frac{1}{m}\,(\boldsymbol{F}_g + \boldsymbol{F}_{\text{ext}}) = \boldsymbol{g} + \frac{1}{m}\,\boldsymbol{F}_{\text{ext}} \qquad (1)$$

where m is the robot's total mass, $\ddot{\boldsymbol{x}} = [\ddot{x} \quad \ddot{y} \quad \ddot{z}]^T$ is the CoM acceleration, $\boldsymbol{g} = [0 \quad 0 \quad -g]^T$ is the gravitational acceleration vector, and g is the gravitational constant. There exists an infinite number of time-transient total forces $\boldsymbol{F}(t)$ for which the CoM moves from an initial point $\boldsymbol{x}_0$ in space to a final point $\boldsymbol{x}_{\text{end}}$. The main problem regarding bipedal locomotion is that, in order to be physically feasible, the external forces acting on the robot have to pass through the base of support (e.g., convex hull of robot's stance feet). A standard tracking controller which aims at achieving a control force of the form $\boldsymbol{F} = k_p(\boldsymbol{x}_d - \boldsymbol{x}) + k_d(\dot{\boldsymbol{x}}_d - \dot{\boldsymbol{x}}) + m\ddot{\boldsymbol{x}}_d$ disregards this physical constraint. One idea to solve this feasibility problem is the following: Instead of *checking* if the external forces intersect the base of support, a point of intersection (focus of all lines of action of external forces) is *designed* and related to a correlating force via an appropriate force law. The Linear Inverted Pendulum (LIP) model [3] follows this idea, its torque-free base joint being the mentioned focus point. Using (1) and the horizontal LIP acceleration $\ddot{\boldsymbol{x}}_{xy} = \frac{g}{z-p_z}\,(\boldsymbol{x}_{xy} - \boldsymbol{p}_{xy})$ [4], the total 3-D force acting on the CoM can be expressed as

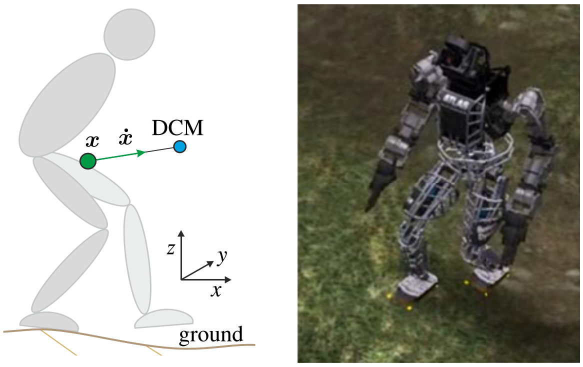

Fig. 1. Left: DCM as point in 3-D space. Right: Atlas robot [5] walking over hilly ground in Gazebo simulation (during DARPA's VRC).

$$\boldsymbol{F}_{\text{LIP}} = \underbrace{\begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix}}_{\boldsymbol{F}_g} + \underbrace{\frac{mg}{z-p_z}(\boldsymbol{x}-\boldsymbol{p})}_{\boldsymbol{F}_{\text{ext}}} = \frac{mg}{z-p_z}\begin{bmatrix} x-p_x \\ y-p_y \\ 0 \end{bmatrix} \qquad (2)$$

where $\boldsymbol{x} = [x \;\; y \;\; z]^T$ and $\boldsymbol{p} = [p_x \;\; p_y \;\; p_z]^T$ denote CoM position and LIP base joint coordinates. The vertical components of $\boldsymbol{F}_g$ and $\boldsymbol{F}_{\text{ext}}$ cancel each other so that $\ddot{z} = 0$ holds. Note: the term $\frac{mg}{z-p_z}$ is constant. Due to torque-free base and point mass assumption, all forces pass through CoM and LIP base joint, i.e., the latter can be seen as a focus point of all external forces. For walking control, the Zero Moment Point (ZMP, [6]) is often assumed to be equivalent to the LIP base joint $\boldsymbol{p}$ and is designed to be within the foot. In a real robot, resulting ZMPs generally deviate from the planned focus points due to model inaccuracies, but often not far enough to make the robot tilt and fall. This way, preplanning a feasible set of focus points (e.g., ZMPs) is a successful method for bipedal gait generation and control: Mitobe *et al.* [7] obtain smooth robot CoM manipulation based on real-time ZMP control. Stephens and Atkeson [8] present a Model Predictive Control (MPC) method for step adjustment and push recovery. Wieber [9] proposes a trajectory free linear MPC scheme, allowing for compensation of strong perturbations. Nishiwaki and Kagami [10] generate dynamically stable walking patterns by frequently updating a preview controller. Sugihara [11] introduces the Best COM-ZMP Regulator facilitating step adjustment of bipedal robots. Kajita *et al.* [12] demonstrate walking on uneven pavement.

Manuscript received December 5, 2014; accepted February 6, 2015. Date of publication March 18, 2015; date of current version April 2, 2015. This paper was recommended for publication by Associate Editor P.-B. Wieber and Editor A. Kheddar upon evaluation of the reviewers' comments. This work was supported by the Initiative and Networking Fund of Helmholtz Association through a Helmholtz Young Investigators Group under Grant VH-NG-808.

The authors are with the Institute of Robotics and Mechatronics, German Aerospace Center (DLR), 82234 Wessling, Germany (e-mail: Johannes.englsberger@dlr.de; christian.ott@dlr.de; Alin.Albu-Schaeffer@dlr.de).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TRO.2015.2405592

Several previous works [13]–[21] propose to split the CoM dynamics into a stable and an unstable part. The state variable related to the unstable part of the dynamics has been referred to as "(instantaneous) Capture Point" by Pratt and Koolen *et al.* [14]–[16], "Extrapolated Center of Mass" by Hof *et al.* [13], and "Divergent Component of Motion" (DCM) by Takenaka *et al.* [17]. Motivated by the works of Pratt *et al.*, in [18] and [19] we had used the term "Capture Point" for the DCM. Yet in [22], we depicted a significant difference between the Capture Point (point on the ground where the robot has to step to come to a stop asymptotically) and the DCM, as the DCM is not restricted to the ground plane, but is a point in 3-D [see Fig. 1 (left)]. For 2-D considerations (constant CoM height), Capture Point and DCM (projected to floor) are equivalent, but not for 3-D.

The use of the LIP model for bipedal walking control is typically restricted to horizontal motions of the CoM ($z = \text{const}$). This motivates the derivation of methods, which are not limited to constant CoM and floor height. Kajita *et al.* [4] introduce the 3-D LIP Mode, which constrains the CoM to be on a (not necessarily horizontal) plane. They present experiments for walking on spiral stairs. Zhao and Sentis [23] introduce the Prismatic Inverted Pendulum dynamics and solve it numerically to allow for 3-D foot placement planning on uneven ground surfaces. Yet, lateral foot-placement cannot be predefined, but depends on the sagittal dynamics. Also, the method is restricted to ground surfaces with laterally constant heights.

In [22], we presented a method for bipedal gait planning and control on uneven terrain that overcomes the aforementioned restrictions. Yet, that method leads to discontinuous desired leg forces and thus desired joint torques, which can cause perturbations in the actuation system. Therefore, we propose a method for generating continuous double support (CDS) trajectories, which—similar to the "Multi-Contact Transitions" in [23]—results in smooth Enhanced Centroidal Moment Pivot point (eCMP) and related leg force profiles. The use of toe-off motion facilitates energy efficient [24] and human-like [25] walking. Thus, we extend the CDS trajectory generator to a heel-to-toe (HT) trajectory generator, in which the eCMP is shifted from heel to toe during single support (SS). This allows for early toe-off motions, increasing the maximum achievable step length and height. Additionally, in this paper, we extend the study from [22] by a robustness analysis of the DCM tracking controller with respect to CoM error, constant external perturbation force, mass estimation error, and unknown force lag.

The paper is organized as follows. In Section II, we introduce the 3-D DCM, the eCMP, and VRP, which the proposed control framework is based on. In Section III, we provide methods for DCM trajectory generation, which are extended in Section IV to produce continuous double support (CDS) and heel-to-toe (HT) transitions. We present a method for 3-D DCM tracking control in Section V and analyze its robustness in Section VI. Section VII outlines a feasible mapping of the desired control forces. In Section VIII, we describe various simulations and experiments. Section IX concludes the paper.

II. Fundamentals

A. *Divergent Component of Motion*

Motivated by the performance of Capture Point control in [18] and [19] and by the definition of the Capture Point (or 2-D DCM) in [14] and [17], we introduced the 3-D DCM in [22]

$$\boldsymbol{\xi} = \boldsymbol{x} + b\dot{\boldsymbol{x}} \tag{3}$$

where $\boldsymbol{\xi} = [\xi_x, \xi_y, \xi_z]^T$ is the DCM, $\boldsymbol{x} = [x, y, z]^T$ and $\dot{\boldsymbol{x}} = [\dot{x}, \dot{y}, \dot{z}]^T$ are the CoM position and velocity and b is the time-constant of the DCM dynamics. Note the similarity of (3) to the state transformation that Slotine and Li [26] use for adaptive control. In contrast with the definition of the Capture Point in [13]–[19], in (3), the DCM is defined as a point at a certain distance in front of the CoM (see Fig. 1), i.e., it is generally not located on the ground. By reordering (3), we can derive the CoM dynamics

$$\dot{\boldsymbol{x}} = -\frac{1}{b}(\boldsymbol{x} - \boldsymbol{\xi}). \tag{4}$$

This shows that the CoM has a stable first-order dynamics for $b > 0$ (i.e., it follows the DCM). By differentiating (3) and inserting (4) and (1), we find the DCM dynamics

$$\dot{\boldsymbol{\xi}} = -\frac{1}{b}\boldsymbol{x} + \frac{1}{b}\boldsymbol{\xi} + \frac{b}{m}\boldsymbol{F} \ . \tag{5}$$

This shows that $\boldsymbol{F}$ directly influences the DCM dynamics.

Remark 1: Due to the linearity of a general (multibody) robot's CoM dynamics (1), the equations and control laws, derived in this and all following sections, hold for general free-floating robot models and not only for simplified models, such as telescopic or LIP.

B. *Enhanced Centroidal Moment Pivot Point*

Generally, a robot is subject to gravity and external forces. As proposed in Section I, we *design* external forces being appropriate for the locomotion task while fulfilling the feasibility constraint (CoP in base of support). To simplify this design process, we make use of a force-to-point transformation similar as in the LIP model. Remember that the term $\frac{mg}{z-p_z}$ in (2) is constant. This motivates the encoding of external forces in a simple repelling force law (linear dependence), based on the difference of the CoM and the so-called eCMP:

$$\boldsymbol{F}_{\text{ext}} = s\,(\boldsymbol{x} - \boldsymbol{r}_{\text{ecmp}}) \tag{6}$$

where $s > 0$ is a constant, which we determine later. The eCMP is closely related to the CMP [27] but not restricted to the foot plane or ground surface. This allows for encoding of not only the direction of the sum of external forces, but also its magnitude. The CMP is located at the intersection of the line between CoM and eCMP with the ground surface (see Fig. 2). The total force acting on the CoM is

$$\boldsymbol{F} = \boldsymbol{F}_{\text{ext}} + \boldsymbol{F}_g = s\,(\boldsymbol{x} - \boldsymbol{r}_{\text{ecmp}}) + m\boldsymbol{g}. \tag{7}$$

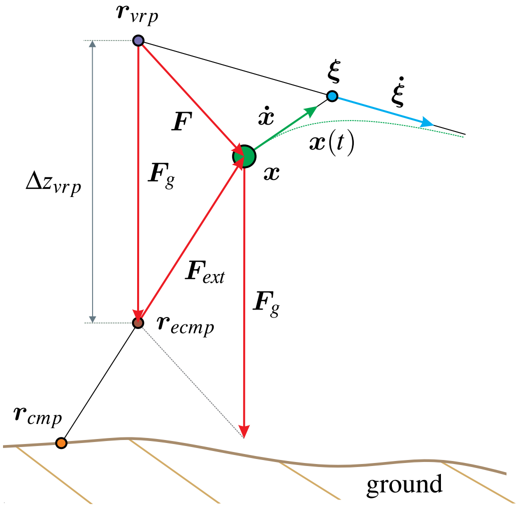

Fig. 2. Force encoding via eCMP and VRP. The figure shows a planar example case. The eCMP encodes the external forces $\boldsymbol{F}_{\text{ext}}$ via (17). The VRP encodes the sum of all forces (gravity and external forces) $\boldsymbol{F}$ via (18) and is located at a vertical offset Δz_{vrp} above the eCMP. Note that by means of appropriate scaling (length of force arrow equals $\Delta z_{\text{vrp}}/(mg)$ times force magnitude), the force vectors can geometrically be related to corresponding point distances [compare with (17)] and (18).

Inserting (7) into (5), we find the DCM dynamics

$$\dot{\boldsymbol{\xi}} = \left(\frac{bs}{m} - \frac{1}{b}\right) \boldsymbol{x} + \frac{1}{b}\,\boldsymbol{\xi} - \frac{bs}{m}\,\boldsymbol{r}_{\text{ecmp}} + b\,\boldsymbol{g}. \tag{8}$$

This shows that the states $\boldsymbol{x}$ and $\boldsymbol{\xi}$ are in general coupled. However, by choosing $s = \frac{m}{b^2}$, we can decouple the DCM dynamics from the CoM dynamics:

$$\dot{\boldsymbol{\xi}} = \frac{1}{b}\,\boldsymbol{\xi} - \frac{1}{b}\,\boldsymbol{r}_{\text{ecmp}} + b\,\boldsymbol{g}. \tag{9}$$

This decoupling of the DCM dynamics from the CoM dynamics is crucial for our controller design (see Section V), which only stabilizes the unstable DCM dynamics while leaving the naturally stable CoM dynamics unaffected. The equation for the sum of external forces becomes

$$\boldsymbol{F}_{\text{ext}} = \frac{m}{b^2}\,(\boldsymbol{x} - \boldsymbol{r}_{\text{ecmp}}). \tag{10}$$

Remark 2: To emphasize the generality of the concept of eCMP, we use the term "external force" ($\boldsymbol{F}_{\text{ext}}$) throughout the paper. For bipedal locomotion, this term may be replaced by "leg force" ($\boldsymbol{F}_{\text{leg}}$) without further adaptions.

C. Virtual Repellent Point

To simplify (9) even more, we introduce the so-called Virtual Repellent Point (VRP) as

$$\boldsymbol{r}_{\text{vrp}} = \boldsymbol{r}_{\text{ecmp}} + \begin{bmatrix} 0 & 0 & b^2 g\end{bmatrix}^T = \boldsymbol{r}_{\text{ecmp}} + \begin{bmatrix} 0 & 0 & \Delta z_{\text{vrp}}\end{bmatrix}^T. \tag{11}$$

The x- and y-components of the eCMP and the VRP are equal. Their vertical components z_{ecmp} and z_{vrp} differ by $\Delta z_{\text{vrp}} = b^2 g$. With (11), we rewrite the DCM dynamics as

$$\dot{\boldsymbol{\xi}} = \frac{1}{b}\,(\boldsymbol{\xi} - \boldsymbol{r}_{\text{vrp}}). \tag{12}$$

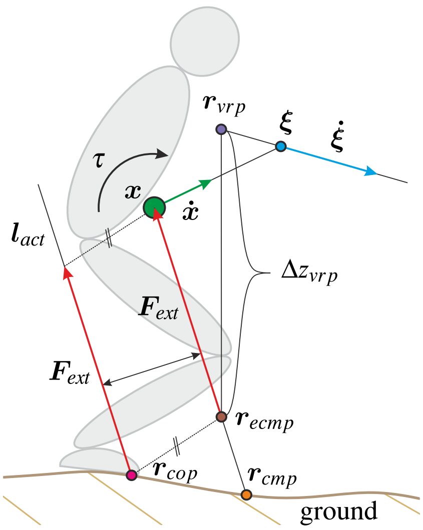

Fig. 3. Point correlations for general robot dynamics. The CMP is found at the intersection of the line CoM-to-eCMP with the ground. The line of action $\boldsymbol{l}_{\text{act}}$ of the external force can be shifted by means of a torque τ around the CoM so that the CoP does generally not coincide with the CMP.

With (7), (10), and (11), we find the correlation between the total force $\boldsymbol{F}$ acting on the CoM and the VRP as

$$\boldsymbol{F} = \frac{m}{b^2}(\boldsymbol{x} - \boldsymbol{r}_{\text{vrp}})\ . \tag{13}$$

We find that the VRP encodes the effects of gravity and all external forces in a single point.

D. Fundamental Results

In the following, we summarize the results of the previous sections and give some comments on the parameter choice. In our control framework, we use the vertical offset $\Delta z_{\text{vrp}} > 0$ between eCMP and VRP as a *design variable*. It can be approximately interpreted as average height of the CoM over the ground and is chosen based on kinematic considerations. In the whole paper, the DCM time constant can be substituted by $b = \sqrt{\Delta z_{\text{vrp}}/g}$. The DCM definition becomes

$$\boldsymbol{\xi} = \boldsymbol{x} + \sqrt{\Delta z_{\text{vrp}}/g}\ \ \dot{\boldsymbol{x}} \tag{14}$$

and the DCM dynamics (12) finally becomes

$$\dot{\boldsymbol{\xi}} = \sqrt{\frac{g}{\Delta z_{\text{vrp}}}}\ (\boldsymbol{\xi} - \boldsymbol{r}_{\text{vrp}}). \tag{15}$$

Since $\sqrt{g/\Delta z_{\text{vrp}}} > 0$, the DCM has an unstable first-order dynamics (it is "pushed" by the VRP), whereas the CoM follows the DCM with the stable first-order dynamics

$$\dot{\boldsymbol{x}} = -\sqrt{\frac{g}{\Delta z_{\text{vrp}}}}\ (\boldsymbol{x} - \boldsymbol{\xi}). \tag{16}$$

The eCMP encodes the external forces acting on the robot

$$\boldsymbol{F}_{\text{ext}} = \frac{mg}{\Delta z_{\text{vrp}}}(\boldsymbol{x} - \boldsymbol{r}_{\text{ecmp}}). \tag{17}$$

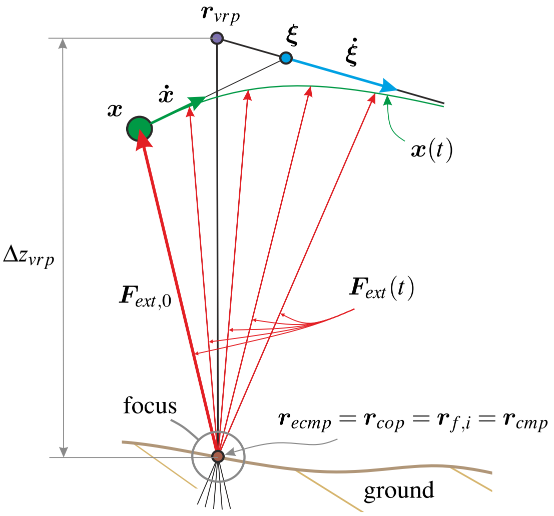

Fig. 4. For *planning*, the eCMP is designed to coincide with the point foot (or foot center) which along with the assumption $\dot{\boldsymbol{L}} = 0$ leads to a constant focus point, through which all force lines pass.

Note: in steady state (i.e., $\boldsymbol{x} = \boldsymbol{\xi} = \boldsymbol{r}_{\mathrm{vrp}} = \boldsymbol{r}_{\mathrm{ecmp}} + [0\ 0\ \Delta z_{\mathrm{vrp}}]^T$), $\boldsymbol{F}_{\mathrm{ext}} = [0\ 0\ mg]$ simply compensates for gravity. Finally, the VRP encodes the total force (gravity plus external forces) via

$$\boldsymbol{F} = \frac{mg}{\Delta z_{\mathrm{vrp}}}(\boldsymbol{x} - \boldsymbol{r}_{\mathrm{vrp}}). \tag{18}$$

Fig. 2 displays these results graphically, while Fig. 3 clarifies the correlations between the eCMP, the CMP, and the CoP for general (bipedal) robot dynamics.

III. Generation of Nominal Divergent Component of Motion Reference

Section II provided all tools that are necessary to design a method to (online) plan and track a desired DCM trajectory in 3-D. For *planning*, we first make the following assumptions:

1) A1: robot's feet are point feet (corresponding to foot centers if robot has finite-sized feet).
2) A2: changes in angular momentum $\boldsymbol{L}$ are zero ($\dot{\boldsymbol{L}} = 0$).
3) A3: instantaneous transitions between left and right SS phases [no double support (DS)].
4) A4: no impacts during support transitions.

Note: These assumptions are purely used to facilitate the planning process. In particular, they do not affect the feedback controller or robustness analysis in Sections V and VI.

We start our planning process, by placing a set of N foot positions $\boldsymbol{r}_{f,i}$ (where $i \in \{1,\ldots,N\}$) on a 3-D ground-surface (see Fig. 5). To comply with the assumptions A1 and A2, we choose $\boldsymbol{r}_{f,i}$, eCMP and CoP (see Fig. 4) to be coincident for planning. This way the corresponding forces always pass through both CoM and planned foot positions $\boldsymbol{r}_{f,i}$. This leads to a constant CoP in case of no perturbations. A desired DCM trajectory can now be derived as follows: First, given a desired eCMP-to-VRP height difference Δz_{vrp}, we find the corresponding desired VRPs (see Fig. 5) with (11) as

$$\boldsymbol{r}_{\mathrm{vrp},d,i} = \boldsymbol{r}_{f,i} + \begin{bmatrix} 0 & 0 & \Delta z_{\mathrm{vrp}} \end{bmatrix}^T. \tag{19}$$

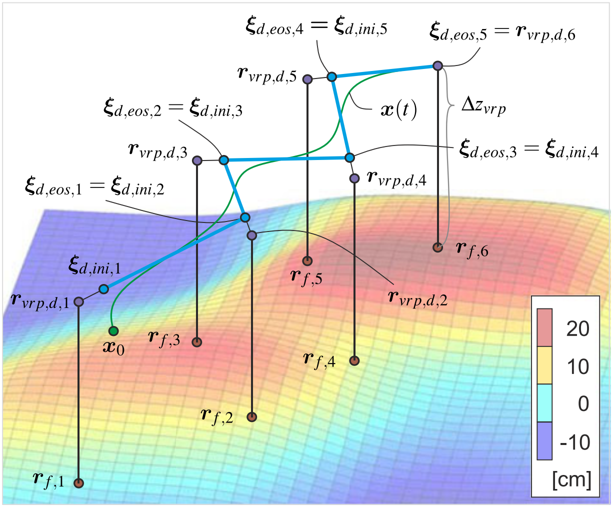
$\boldsymbol{\xi}_{d,eos,4} = \boldsymbol{\xi}_{d,ini,5}$
$\boldsymbol{\xi}_{d,eos,5} = \boldsymbol{r}_{vrp,d,6}$
$\boldsymbol{r}_{vrp,d,5}$
$\boldsymbol{x}(t)$
Δz_{vrp}
$\boldsymbol{\xi}_{d,eos,2} = \boldsymbol{\xi}_{d,ini,3}$
$\boldsymbol{r}_{vrp,d,3}$
$\boldsymbol{\xi}_{d,eos,3} = \boldsymbol{\xi}_{d,ini,4}$
$\boldsymbol{r}_{vrp,d,4}$
$\boldsymbol{\xi}_{d,eos,1} = \boldsymbol{\xi}_{d,ini,2}$
$\boldsymbol{r}_{f,6}$
$\boldsymbol{\xi}_{d,ini,1}$
$\boldsymbol{r}_{f,5}$
$\boldsymbol{r}_{vrp,d,2}$
$\boldsymbol{r}_{vrp,d,1}$
20
10
0
-10
[cm]
$\boldsymbol{x}_0$
$\boldsymbol{r}_{f,3}$
$\boldsymbol{r}_{f,4}$
$\boldsymbol{r}_{f,2}$
$\boldsymbol{r}_{f,1}$

Fig. 5. Planning of DCM trajectory over rough terrain. Note that both the DCM reference trajectory (bold blue lines) and the resulting CoM trajectory (green curve) are 3-D, i.e., not constrained to a horizontal plane (see also Fig. 15). The CoM "automatically" follows the DCM from its initial position $\boldsymbol{x}_0$ to the final equilibrium point $\boldsymbol{r}_{\mathrm{vrp},d,N}$.

To find the desired DCM trajectory from given VRPs, the method presented in [19] is extended to 3-D. For a constant VRP, the solution of (15) is

$$\boldsymbol{\xi}(t) = \boldsymbol{r}_{\mathrm{vrp}} + e^{\sqrt{\frac{g}{\Delta z_{\mathrm{vrp}}}}\, t}\,(\boldsymbol{\xi}_0 - \boldsymbol{r}_{\mathrm{vrp}}) \tag{20}$$

where $\boldsymbol{\xi}_0 = \boldsymbol{\xi}(0)$. The "internal" step time t is reset at the beginning of each step, i.e., $t \in [0, t_{\mathrm{step},i}]$ ($t_{\mathrm{step},i}$ is the duration of the ith step). With (20) and the constant desired VRPs from (19), we find the desired DCM locations at the end of each step (see Fig. 5) via recursion:

$$\begin{aligned}\xi_{d,eos,i-1} = \xi_{d,\mathrm{ini},i} = {} & \boldsymbol{r}_{\mathrm{vrp},d,i} + e^{-\sqrt{\frac{g}{\Delta z_{\mathrm{vrp}}}}\, t_{\mathrm{step},i}} \\ & \times (\boldsymbol{\xi}_{d,eos,i} - \boldsymbol{r}_{\mathrm{vrp},d,i}).\end{aligned} \tag{21}$$

$\xi_{d,\mathrm{ini},i}$ is the ith initial desired DCM. For planning, we assume that the DCM will come to a stop over the final previewed foot position, i.e., $\boldsymbol{\xi}_{d,eos,N-1} = \boldsymbol{r}_{vrp,d,N}$. All $\boldsymbol{\xi}_{d,eos,i}$ are computed via (21). Now, based on (20), the reference trajectories for the DCM position (bold blue lines in Fig. 5) can be computed as

$$\boldsymbol{\xi}_{d,i}(t) = \boldsymbol{r}_{\mathrm{vrp},d,i} + e^{\sqrt{\frac{g}{\Delta z_{\mathrm{vrp}}}}\,(t - t_{\mathrm{step},i})}(\boldsymbol{\xi}_{d,eos,i} - \boldsymbol{r}_{\mathrm{vrp},d,i}). \tag{22}$$

The corresponding DCM velocity trajectories are found via derivation of (22)

$$\dot{\boldsymbol{\xi}}_{d,i}(t) = \sqrt{\frac{g}{\Delta z_{\mathrm{vrp}}}}\; e^{\sqrt{\frac{g}{\Delta z_{\mathrm{vrp}}}}\,(t - t_{\mathrm{step},i})}(\boldsymbol{\xi}_{d,eos,i} - \boldsymbol{r}_{\mathrm{vrp},d,i}). \tag{23}$$

That way all future desired DCM trajectories (i.e., for $i = 1\ldots N$) can be computed. Yet, our feedback controller (see Section V) only takes the DCM reference of the current step into account. Thus, we evaluate (22) and (23) only for $i = 1$ (index "1" always indicates the current step).

In practice, we limit ourselves to the use of the current ($\boldsymbol{r}_{\mathrm{vrp,d,1}}$) and next three desired VRPs ($\boldsymbol{r}_{\mathrm{vrp,d,2}}$, $\boldsymbol{r}_{\mathrm{vrp,d,3}}$,

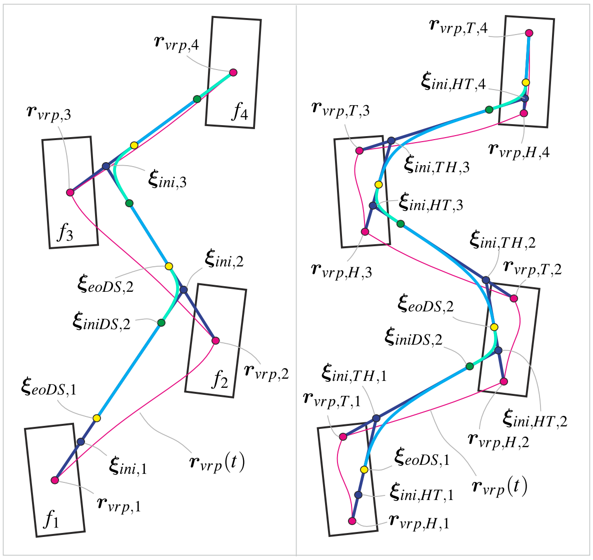
$\boldsymbol{r}_{vrp,4}$
f_4
$\boldsymbol{r}_{vrp,3}$
$\boldsymbol{\xi}_{ini,3}$
f_3
$\boldsymbol{\xi}_{ini,2}$
$\boldsymbol{\xi}_{eoDS,2}$
$\boldsymbol{\xi}_{iniDS,2}$
f_2
$\boldsymbol{r}_{vrp,2}$
$\boldsymbol{\xi}_{eoDS,1}$
$\boldsymbol{\xi}_{ini,1}$
$\boldsymbol{r}_{vrp}(t)$
$\boldsymbol{r}_{vrp,1}$
f_1
$\boldsymbol{r}_{vrp,T,4}$
$\boldsymbol{\xi}_{ini,HT,4}$
$\boldsymbol{r}_{vrp,T,3}$
$\boldsymbol{r}_{vrp,H,4}$
$\boldsymbol{\xi}_{ini,TH,3}$
$\boldsymbol{\xi}_{ini,HT,3}$
$\boldsymbol{\xi}_{ini,TH,2}$
$\boldsymbol{r}_{vrp,H,3}$
$\boldsymbol{r}_{vrp,T,2}$
$\boldsymbol{\xi}_{eoDS,2}$
$\boldsymbol{\xi}_{iniDS,2}$
$\boldsymbol{\xi}_{ini,TH,1}$
$\boldsymbol{r}_{vrp,T,1}$
$\boldsymbol{\xi}_{ini,HT,2}$
$\boldsymbol{r}_{vrp,H,2}$
$\boldsymbol{\xi}_{eoDS,1}$
$\boldsymbol{\xi}_{ini,HT,1}$
$\boldsymbol{r}_{vrp}(t)$
$\boldsymbol{r}_{vrp,H,1}$

Fig. 6. CDS (left) and HT (including CDS; right) DCM reference trajectories. Depicted for three previewed future footsteps.

and $\boldsymbol{r}_{\mathrm{vrp,d},4}$) to calculate the DCM tracking reference (see Fig. 6). This reduces the computational effort while the deviation from a trajectory generated with all future VRPs is marginal.

Remark 3: The most significant property of the DCM trajectories derived in this section is that they correspond to constant CoPs (e.g., in foot centers), such that the resulting force profiles are likely to be feasible. Since the natural DCM dynamics is unstable, it needs to be stabilized (see Section V).

IV. Continuous Double Support and Heel-to-Toe Trajectories [28]

A. Continuous Double Support Trajectories

The planning method presented in Section III is very powerful, as it allows for the design of walking trajectories over unstructured (3-D) ground surfaces in real time. Yet, as it primarily only considers single support phases (and instantaneous transitions between them), there is the drawback of a discontinuous VRP (and eCMP) reference and thus discontinuous desired external forces at the support transitions (see F_{ext} in Fig. 9). This leads to discontinuities in the commanded joint torques, which can be unfeasible for a physical robot due to its limited actuator dynamics and might excite unmodeled structural elasticities. This motivates the derivation of DCM trajectories that lead to continuous eCMP (and corresponding external force) transitions. Therefore, we use a walking state machine (see Section IV-C) and consider the anticipated DS phases for the generation of a modified DCM reference trajectory.

Fig. 6 (left) shows the outline used for the generation of DCM trajectories with continuous double support (CDS) transitions. The CDS trajectory generation builds on the method proposed in Section III, which leads to a zigzag trajectory (light blue lines in Fig. 5). The edges around the initial DCMs $\boldsymbol{\xi}_{\mathrm{ini},i}$ are now "rounded" (turquoise curves in Fig. 6) to guarantee continuity of eCMP and corresponding external forces during DS. For a desired DCM position and velocity, the correlating VRP is obtained from (15) as

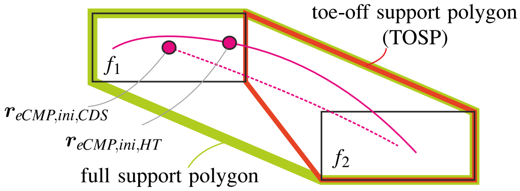
toe-off support polygon (TOSP)
f_1
f_2
$\boldsymbol{r}_{eCMP,ini,CDS}$
$\boldsymbol{r}_{eCMP,ini,HT}$
full support polygon

Fig. 7. Full support polygon and TOSP.

$$\boldsymbol{r}_{\mathrm{vrp}} = \boldsymbol{\xi} - \sqrt{\frac{\Delta z_{\mathrm{vrp}}}{g}}\,\dot{\boldsymbol{\xi}}. \tag{24}$$

This means that a reference trajectory with continuous DCM position and velocity results in a continuous VRP trajectory and corresponding continuous eCMP [see (11)] and external force [see (17)] trajectories. This motivates the use of a third-order polynomial interpolation to "round" the edges of the preliminary DCM reference trajectory (corresponding to smooth transition during DS). Given a desired DS duration t_{DS}, the idea is to compute two points

$$\boldsymbol{\xi}_{\mathrm{iniDS},i} = \boldsymbol{r}_{\mathrm{vrp},i-1} + e^{-\sqrt{\frac{g}{\Delta z_{\mathrm{vrp}}}}\,\Delta t_{\mathrm{DS,ini}}}\left(\boldsymbol{\xi}_{\mathrm{ini},i} - \boldsymbol{r}_{\mathrm{vrp},i-1}\right) \tag{25}$$

("iniDS" stands for "initial double support") and

$$\boldsymbol{\xi}_{\mathrm{eoDS},i} = \boldsymbol{r}_{\mathrm{vrp},i} + e^{\sqrt{\frac{g}{\Delta z_{\mathrm{vrp}}}}\,\Delta t_{\mathrm{DS,end}}}\left(\boldsymbol{\xi}_{\mathrm{ini},i} - \boldsymbol{r}_{\mathrm{vrp},i}\right) \tag{26}$$

("eoDS" stands for "end of double support") on the preliminary DCM trajectory [see Fig. 6 (left)], where the DCM would be $\Delta t_{\mathrm{DS,ini}} = \alpha_{\mathrm{DS,ini}}\, t_{\mathrm{DS}}$ before and $\Delta t_{\mathrm{DS,end}} = (1 - \alpha_{\mathrm{DS,ini}})\, t_{\mathrm{DS}}$ after the instantaneous support transition. In this study, we chose the parameter $\alpha_{\mathrm{DS,ini}}$ as 0.5. $\boldsymbol{\xi}_{\mathrm{iniDS},i}$ and $\boldsymbol{\xi}_{\mathrm{eoDS},i}$ (and the corresponding DCM velocities) are used as boundary conditions for the interpolation polynomial.

For initial and final DCM position and velocity boundary conditions, a polynomial parameter matrix can be found as[1]

$$\boldsymbol{P} = \begin{bmatrix} 2/T_s^3 & 1/T_s^2 & -2/T_s^3 & 1/T_s^2 \\ -3/T_s^2 & -2/T_s & 3/T_s^2 & -1/T_s \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_{\mathrm{ini}}^T \\ \dot{\boldsymbol{\xi}}_{\mathrm{ini}}^T \\ \boldsymbol{\xi}_{\mathrm{end}}^T \\ \dot{\boldsymbol{\xi}}_{\mathrm{end}}^T \end{bmatrix}, \tag{27}$$

where T_s denotes the total duration of the transition and $[\boldsymbol{\xi}_{\mathrm{ini}}, \dot{\boldsymbol{\xi}}_{\mathrm{ini}}, \boldsymbol{\xi}_{\mathrm{end}}, \dot{\boldsymbol{\xi}}_{\mathrm{end}}]^T$ are the boundary conditions on the initial and final DCM position and velocity. With $\boldsymbol{P}$, for any time $t \in [0, T_s]$ the DCM position and velocity can be computed as

$$\begin{bmatrix} \boldsymbol{\xi}^T(t) \\ \dot{\boldsymbol{\xi}}^T(t) \end{bmatrix} = \begin{bmatrix} t^3 & t^2 & t & 1 \\ 3t^2 & 2t & 1 & 0 \end{bmatrix} \boldsymbol{P} \tag{28}$$

where t denotes the time in the current state (see Section IV-C). This polynomial interpolation [using $\boldsymbol{\xi}_{\mathrm{iniDS},i}$, $\boldsymbol{\xi}_{\mathrm{eoDS},i}$, and the corresponding DCM velocities as boundary conditions in (27)]

[1] The transpose denotes a row vector.

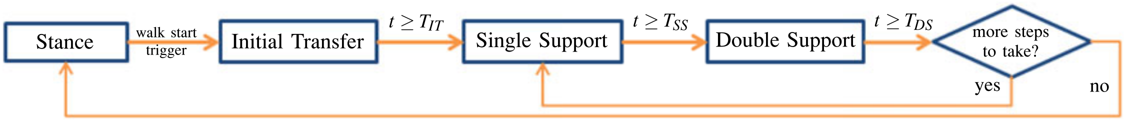

Fig. 8. Walking state machine: state ∈ [Stance, IT, SS, DS], state durations: T_{IT}, T_{SS}, T_{DS}.

is used during DS (turquoise curves in Fig. 6). The VRP, eCMP, and $\boldsymbol{F}_{\mathbf{ext}}$ can be computed via (24), (11), and (17).

During SS, exponential interpolation (22) is used (light blue line segments). Fig. 6 (left) also shows the continuous VRP trajectories ($\boldsymbol{r}_{\mathrm{vrp}}(t)$, pink curves) during DS. During SS, the VRP positions $\boldsymbol{r}_{\mathrm{vrp},i}$ are constant. Note that Fig. 6 shows a top view of the trajectories. Yet, just like in Fig. 5, the resulting DCM, VRP, and eCMP trajectories are 3-D in general.

B. Continuous Heel-to-Toe Trajectories

When a walking robot is restricted to full foot support (no toe-off) during DS, its maximum step length and height is limited. This is mainly due to singularities related to the stretched knee of the hind leg in the end of DS. To overcome this problem, a robot can use its toes or allow the rear foot to tilt around its front edge. This "toe-off" motion leads to an increased effective hind leg length during DS. While the singularity problem is eased, a toe-off introduces a new problem: To avoid tilting of the robot's feet, the desired CoP has to stay inside the base of support. During toe-off, the support polygon is instantaneously decreased to the "toe-off support polygon" (TOSP, see Fig. 7). For the DCM reference generation, we assume the eCMP to coincide with the CoP (i.e., no torques around CoM considered, see Figs. 3 and 4). Thus, if the reference eCMP lies outside the TOSP, it has to be projected to it, resulting in a discontinuous eCMP. In our method for CDS (see Section IV-A), the eCMP trajectories start from the foot-centers (see $\boldsymbol{r}_{\mathrm{eCMP,ini,CDS}}$ in Fig. 7), such that it takes comparably long until the eCMP reaches the TOSP. In this section, we propose a heel-to-toe (HT) shift of the eCMP during SS to achieve an earlier entry of the corresponding eCMP in the TOSP so that the robot can start its toe-off motion earlier. Therefore, we consider two eCMPs—one close to the heel and one close to the toe—instead of only one central eCMP for each foot. As before, the corresponding heel- and toe-VRPs ($\boldsymbol{r}_{\mathrm{vrp},H,i}$ and $\boldsymbol{r}_{\mathrm{vrp},T,i}$) are Δz_{vrp} above the heel- and toe-eCMPs. Instead of one central VRP in each foot, we now use both heel- and toe-VRPs in each foot to first derive the preliminary DCM trajectory [dark blue lines in Fig. 6 (right)], assuming instantaneous transitions. As in Section III, this is done via backward recursion (21), starting with the final previewed toe-VRP and iterating backward in time over all heel- and toe-VRPs until the current one. That way, all future initial HT DCMs $\boldsymbol{\xi}_{\mathrm{ini,HT,i}}$ and initial toe-to-heel DCMs $\boldsymbol{\xi}_{\mathrm{ini,TH,i}}$ are computed. The main difference is that in (21) instead of $t_{\mathrm{step},i}$ we use $\Delta t_{\mathrm{HT}} = \alpha_{\mathrm{HT}}\, t_{\mathrm{step},i}$ (HT transition) and $\Delta t_{\mathrm{TH}} = (1 - \alpha_{\mathrm{HT}})\, t_{\mathrm{step},i}$ (toe-to-heel transition) as the duration of transition. In this study, we chose the parameter α_{HT} as 0.5. The edges around the initial HT-DCMs are now rounded via third-order polynomial splines [turquoise curves in Fig. 6 (right)] to achieve a continuous VRP transition from toe to heel during DS. Therefore, the initial and final DS DCM positions can be computed via

$$\xi_{\mathrm{iniDS},i} = \boldsymbol{r}_{\mathrm{vrp},T,i-1} + e^{-\sqrt{\frac{g}{\Delta z_{\mathrm{vrp}}}}\,\Delta t_{\mathrm{DS,ini}}} \times (\xi_{\mathrm{ini,HT},i} - \boldsymbol{r}_{\mathrm{vrp,T},i-1}) \tag{29}$$

$$\xi_{\mathrm{eoDS},i} = \boldsymbol{r}_{\mathrm{vrp},H,i} + e^{\sqrt{\frac{g}{\Delta z_{\mathrm{vrp}}}}\,\Delta t_{\mathrm{DS,end}}} \times (\xi_{\mathrm{ini,HT},i} - \boldsymbol{r}_{\mathrm{vrp,H},i}). \tag{30}$$

$\boldsymbol{\xi}_{\mathrm{iniDS},i}$, $\boldsymbol{\xi}_{\mathrm{eoDS},i}$ and the according DCM velocities as used as boundary conditions in (27) to compute the polynomial parameters for the DS phases. Unlike in Section IV-A, the SS DCM reference trajectory is not computed via exponential interpolation (22) but again via the polynomial interpolation (28) with $\xi_{\mathrm{eoDS},i-1}$ as initial and $\xi_{\mathrm{iniDS},i}$ as final boundary condition in (27), which results in the light blue curves in Fig. 6 (right). The pink curves illustrate the continuous VRP transitions during SS (from $\boldsymbol{r}_{\mathrm{vrp,H,i}}$ to $\boldsymbol{r}_{vrp,\mathrm{T},i}$) and DS (from $\boldsymbol{r}_{\mathrm{vrp,T},i}$ to $\boldsymbol{r}_{\mathbf{vrp},\mathrm{H},i+1}$).

The following table gives an overview on the use of the interpolation methods in the different trajectory generators.

Trajectory generator		**DCM interpolation method**
discontinuous		exponential (22)
conti. DS	double support	polynomial (28)
	single support	exponential (22)
HT	double support	polynomial (28)
	single support	polynomial (28)

Remark 4: The nominal eCMP computed from the DCM references in Sections IV-A and IV-B is not guaranteed to lie within the (BoS). A nominal eCMP outside the BoS increases the likelihood of infeasible desired forces and thus the need for feasible force mapping (see Section VII).

C. Walking State Machine

We use a walking state machine (see Fig. 8) to keep track of and thoroughly switch between walking states and provide the DCM interpolators with the necessary inputs (such as previewed foot positions and the current time in state t). In total, we distinguish four states: stance, initial transfer (IT), SS, DS. Stance can be seen as the robot idle state, where no walking trajectories are commanded and it is simply balancing in place. During IT, the robot shifts its DCM from between its stance feet to the final DS DCM position $\boldsymbol{\xi}_{\mathrm{eoDS},2}$ of the upcoming step (see Fig. 6). SS and DS state correlate with the phases described in Section IV. A timer associated with the walking state machine provides the exponential and polynomial interpolators (22) and (28) with the time in state t. It increases t, until the duration of the current state is reached ($t \in [0, T_s]$, where $T_s \in \{T_{\mathrm{IT}}, T_{\mathrm{SS}}, T_{\mathrm{DS}}\}$), which triggers a state transition and resets t ($t = 0$). After each

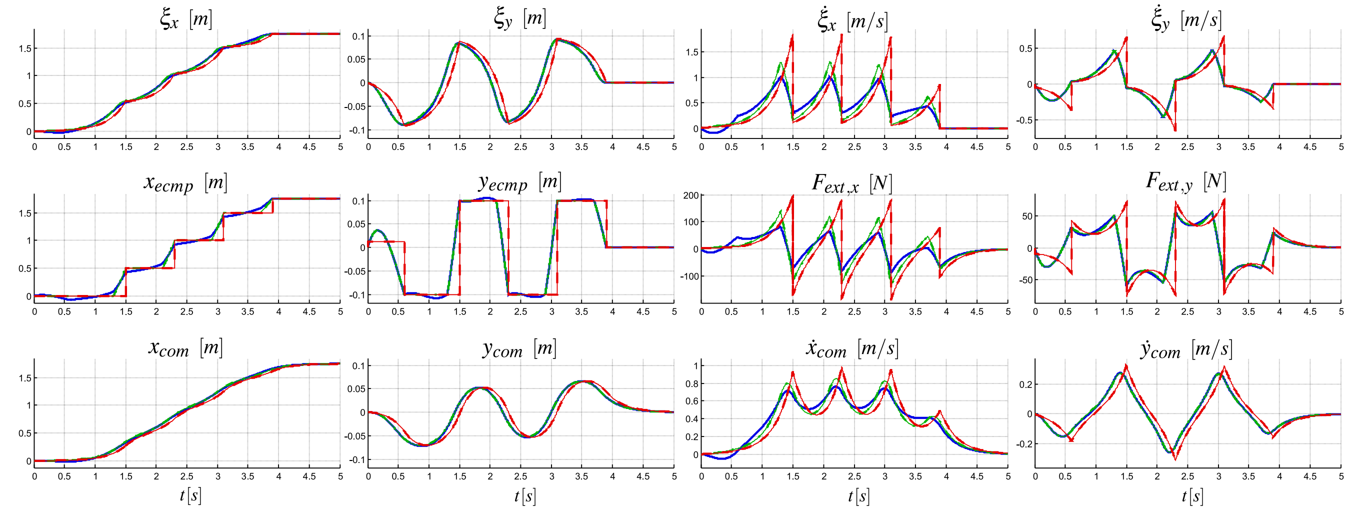

Fig. 9. Comparison of discontinuous (red dashed), CDS (green dash-dotted) and HT (blue solid) trajectory generators. Shown trajectories correspond to a walk with the following parameters: $t_{\text{step}} = 0.8$ s, $t_{\text{DS}} = 0.2$ s, step length 0.5 m, step width 0.2 m, foot length 0.15 m.

DS phase, the state machine checks whether there are more steps to take and then decides to switch to stance or the next SS phase.

As mentioned in Section III, in our implementation, we distinguish between the list of preplanned footsteps (e.g., 100 preplanned footsteps for walking from one room to another) and the list of previewed steps. The latter contains only a number of steps (typically four, as in Fig. 6), which are considered for the DCM reference trajectory. In the beginning of each SS phase, the list of previewed steps is updated, and all reference points (see Section IV) for the current SS and upcoming DS are computed. In addition, the polynomial parameter matrix from (27) is precomputed. For advanced applications (e.g., online foot-adjustment), these computations might be done every control cycle, otherwise a once-per-step precomputation is sufficient.

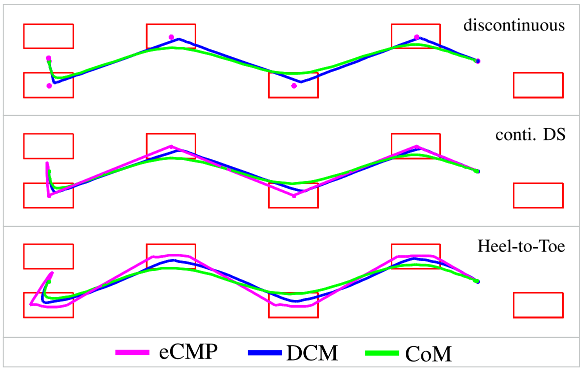

Fig. 10. Comparison (top view) of VRP (pink), DCM (blue), and CoM (green) reference trajectories. Top: discontinuous [22], middle: CDS (Section IV-A), bottom: HT (Section IV-B) trajectories.

D. Resulting Divergent Component of Motion Reference Trajectories

Fig. 9 shows a comparison of the discontinuous, CDS, and HT trajectory generators for a four-step sample walk of a bipedal robot (find details of the walking task in the caption). For the CDS and HT trajectories, the force profiles are continuous (as compared with discontinuous force profiles for the original method from Section III) and the maximum absolute of the external forces is strongly decreased, resulting in smaller minimum required joint torques.

The maximum DCM velocity (especially in sagittal direction x) is drastically reduced (see quantitative comparison in the table below). As the term "DCM" indicates, the DCM velocity is a measure for how fast one of the robot states (the DCM) is diverging. Although we make use of this divergence to allow the robot to dynamically walk ("controlled divergence"), it comes with the inherent danger of loss of control if the actual interactions of the robot system with its environment differ from the predicted ones. For instance in case of a delayed landing of the swing foot, the DCM during SS may overshoot out of the previewed (DS) support polygon, which may result in unrecoverable instability of the robot. Thus, the lower DCM velocities observed for the CDS and HT trajectories decrease the risk of loss of control. One can also observe lower maximum CoM velocities for the CDS and HT trajectories, which correlates with lower required joint speeds or higher maximum walking speed for given maximum joint velocities.

The table below shows a quantitative comparison of the achieved sagittal forces, DCM velocities, and CoM velocities.

Algorithm	$F_{\text{ext},\,x\,,\text{max}}$ $[N]$		$\dot{\xi}_{x\,,\text{max}}$ $[\frac{m}{s}]$		$\dot{x}_{\text{com, max}}$ $[\frac{m}{s}]$	
	abs.	perc.	abs.	perc.	abs.	perc.
discont. DS	190	100%	1.86	100%	0.98	100%
cont. DS	136	71.6%	1.31	70.4%	0.85	86.7%
cont. DS/HT	90.2	47.5%	1.01	54.3%	0.76	77.6%

Fig. 10 shows a top view for the three different DCM trajectory generators (discontinuous, CDS, and HT). Once again note the discontinuous and continuous trajectories of the VRPs (pink). In addition, note that the DCM finally comes to a stop at the center of the supporting feet. Thus, the recursion from Section III starts from $\boldsymbol{\xi}_{d,eos,N-1} = (\boldsymbol{r}_{\text{vrp},d,N-1} + \boldsymbol{r}_{\text{vrp},d,N})/2$, when less than four steps are previewed.

V. Three-Dimensional Divergent Component of Motion Tracking Control

To track the desired DCM trajectory, we aim at a closed-loop dynamics of the form

$$\underbrace{\dot{\boldsymbol{\xi}} - \dot{\boldsymbol{\xi}}_{\boldsymbol{d}}}_{\dot{\boldsymbol{e}}_\xi} = -k_\xi \, \underbrace{(\boldsymbol{\xi} - \boldsymbol{\xi}_{\boldsymbol{d}})}_{\boldsymbol{e}_\xi} \tag{31}$$

which is stable for $k_\xi > 0$ (DCM error $\boldsymbol{e}_\xi$ converges asymptotically). As the DCM error $\boldsymbol{e}_\xi$ converges asymptotically, also the commanded VRP $\boldsymbol{r}_{\mathrm{vrp}}$ and its corresponding eCMP $\boldsymbol{r}_{\mathrm{ecmp}}$ converge to their desired values. This desired dynamics is realized exactly by a DCM tracking control law of the form

$$\boldsymbol{r}_{\mathrm{vrp}} = \boldsymbol{\xi} + k_\xi \sqrt{\frac{\Delta z_{\mathrm{vrp}}}{g}} \; (\boldsymbol{\xi} - \boldsymbol{\xi}_{\boldsymbol{d}}) - \sqrt{\frac{\Delta z_{\mathrm{vrp}}}{g}} \; \dot{\boldsymbol{\xi}}_{\boldsymbol{d}} \; . \tag{32}$$

Note that (32) only stabilizes the unstable DCM dynamics, without influencing the naturally stable CoM dynamics (16). The closed-loop dynamics has the form

$$\begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{\xi}} \end{bmatrix} = \underbrace{\begin{bmatrix} -\sqrt{\frac{g}{\Delta z_{\mathrm{vrp}}}}\, \boldsymbol{I} & \sqrt{\frac{g}{\Delta z_{\mathrm{vrp}}}}\, \boldsymbol{I} \\ \boldsymbol{0} & -k_\xi \, \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\xi} \end{bmatrix}}_{\text{feedback}} + \underbrace{\begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ k_\xi \, \boldsymbol{I} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_{\boldsymbol{d}} \\ \dot{\boldsymbol{\xi}}_{\boldsymbol{d}} \end{bmatrix}}_{\text{feedforward}} . \tag{33}$$

Here, $\boldsymbol{I}$ and $\boldsymbol{0}$ denote 3×3 unit and zero matrices. Since the eigenvalues of the system matrix are stable for $k_\xi > 0$ and the feedforward terms $\boldsymbol{\xi}_d$ and $\dot{\boldsymbol{\xi}}_d$ are bounded (see Fig. 9), the preconditions for BIBO stability are fulfilled. The system matrix is time-invariant, which is an advantage compared with [19]. This stability analysis neglects physical limitations such as limited base of support (see Section VII).

With (32) and (17), we find the desired external force as

$$\boldsymbol{F}_{\mathrm{ext},d} = \frac{mg}{\Delta z_{\mathrm{vrp}}} (\boldsymbol{x} - \underbrace{(\boldsymbol{r}_{\mathrm{vrp}} - \begin{bmatrix} 0 & 0 & \Delta z_{\mathrm{vrp}} \end{bmatrix}^T)}_{\boldsymbol{r}_{\mathrm{ecmp},\,c}}). \tag{34}$$

Note that the only equations that are finally needed are (21)–(23) (or the corresponding equations from Section IV) for 3-D DCM trajectory generation and (32) and (34) for force-based DCM tracking control. They can be computed in real time on any computer. The main design parameters are “average CoM height” Δz_{vrp}, DCM tracking gain k_ξ and time per step t_{step}.

Remark 5: For constant desired VRPs, the desired DCM velocity with (15) becomes $\dot{\boldsymbol{\xi}}_d = \sqrt{\frac{g}{\Delta z_{\mathrm{vrp}}}} \, (\boldsymbol{\xi}_d - \boldsymbol{r}_{\mathrm{vrp},d,1})$. Using this, the DCM tracking control law (32) becomes

$$\boldsymbol{r}_{\mathrm{vrp}} = \boldsymbol{r}_{\mathrm{vrp},d,1} + (1 + k_\xi) \sqrt{\frac{\Delta z_{\mathrm{vrp}}}{g}})\underbrace{(\boldsymbol{\xi} - \boldsymbol{\xi}_d)}_{\boldsymbol{e}_\xi}. \tag{35}$$

This shows that the commanded VRP deviates from the nominal VRP $\boldsymbol{r}_{\mathrm{vrp},d,1}$ in case of perturbations. Equation (35) will be used in the method for step adjustment (see Fig. 13).

VI. Analysis of Robustness of Divergent Component of Motion Control

In Section V, we introduced a nominally asymptotically stable DCM tracking controller. In the following sections, we examine its robustness with respect to:

1) CoM error,
2) constant external perturbation (force),
3) unknown force lag,
4) error in total mass estimation.

By combining (18) and (32), the total commanded force acting on the CoM can be expressed as

$$\boldsymbol{F}_c = \frac{\hat{m}}{b^2} \, (\hat{\boldsymbol{x}} \; - \; \hat{\boldsymbol{\xi}} \; - k_\xi \; b \, (\hat{\boldsymbol{\xi}} - \boldsymbol{\xi}_d) + b \, \dot{\boldsymbol{\xi}}_d) \tag{36}$$

where $\hat{m}$, $\hat{\boldsymbol{x}}$, and $\hat{\boldsymbol{\xi}}$ denote the estimated mass, CoM position, and DCM position, respectively. Note that throughout this section, we substituted the term $\sqrt{\Delta z_{\mathrm{vrp}}/g}$ by b (as in the derivations) to keep the equations clear and compact. Equation (36) returns the desired (total, i.e., sum of gravitational and external) force $\boldsymbol{F}_c$ (“c” stands for “control”), which ideally results in the desired tracking behavior of the DCM. The commanded external force can be found as $\boldsymbol{F}_{\mathrm{ext},c} = \boldsymbol{F}_c - \boldsymbol{F}_g$. In the following, we will assume $\hat{m} = m$, $\hat{\boldsymbol{x}} = \boldsymbol{x}$, and $\hat{\boldsymbol{\xi}} = \boldsymbol{\xi}$ (perfect model/measurements), if not stated differently.

Guaranteeing system stability is the most important goal of a controller. Yet, for some underactuated systems, such as a bipedal walker, it can be disadvantageous to have imaginary parts of the system eigenvalues and thus overshooting behavior of the controller. Therefore, we will—where needed—also examine this issue and provide design rules.

1) Robustness w.r.t. CoM Error: Here, we examine the influence of an error in the CoM estimate on the stability of the proposed DCM controller (36). Such an error can occur, when the robot model is inaccurate or when the robot is carrying a load of unknown mass properties. The measured CoM position is

$$\hat{\boldsymbol{x}} = \boldsymbol{x} + \boldsymbol{\Delta x} \tag{37}$$

where $\boldsymbol{\Delta x}$ denotes the CoM error. As the CoM position error $\boldsymbol{\Delta x}$ may not be constant, the estimated CoM velocity becomes $\dot{\hat{\boldsymbol{x}}} = \dot{\boldsymbol{x}} + \boldsymbol{\Delta}\dot{\boldsymbol{x}}$. Using the definition of the DCM (3), we find the following correlation between the estimated and real DCM:

$$\hat{\boldsymbol{\xi}} = \boldsymbol{\xi} + \Delta \boldsymbol{x} + b \, \Delta \dot{\boldsymbol{x}}. \tag{38}$$

Inserting (37) and (38) into the control law (36) and setting $\boldsymbol{F} = \boldsymbol{F}_c$ in (5) results in the following closed-loop dynamics:

$$\begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{\xi}} \end{bmatrix} = \begin{bmatrix} -\frac{1}{b} \, \boldsymbol{I} & \frac{1}{b} \, \boldsymbol{I} \\ \boldsymbol{0} & -k_\xi \, \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\xi} \end{bmatrix} + \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ k_\xi \, \boldsymbol{I} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_{\boldsymbol{d}} \\ \dot{\boldsymbol{\xi}}_{\boldsymbol{d}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ -k_\xi \, \boldsymbol{I} & -(k_\xi \; b + 1) \, \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \Delta \boldsymbol{x} \\ \Delta \dot{\boldsymbol{x}} \end{bmatrix}. \tag{39}$$

The dynamics matrix is unchanged compared with (33). Thus, for bounded CoM errors $\Delta \boldsymbol{x}$ and $\Delta \dot{\boldsymbol{x}}$ and DCM references $\boldsymbol{\xi}_d$ and $\dot{\boldsymbol{\xi}}_d$, the preconditions for BIBO stability are fulfilled.

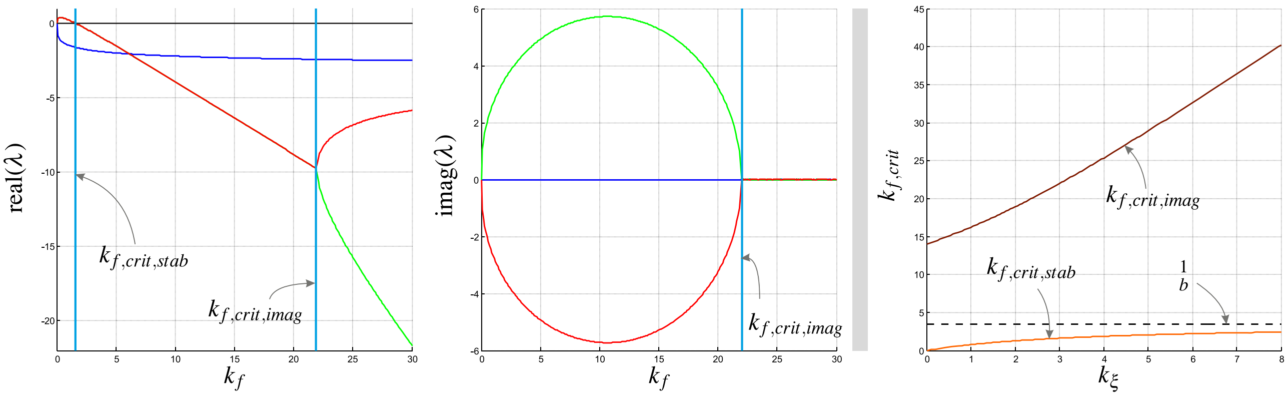

Fig. 11. Left and mid: Typical course of eigenvalues (real/imaginary parts), Vertical lines denote critical values of k_f. right: Critical values for k_f, as a function of the DCM gain k_ξ. Plotted for $b = 0.2856$ s (corresponds to $\Delta z_{\text{vrp}} = 0.8$ m).

If a constant CoM offset $\Delta\boldsymbol{x}$ is assumed, the dynamics of the DCM error $\boldsymbol{e}_\xi = \boldsymbol{\xi} - \boldsymbol{\xi}_d$ has the following solution

$$\boldsymbol{e}_\xi(t) = -\Delta\boldsymbol{x} + \underbrace{e^{-k_\xi\, t}}_{\to 0}(\boldsymbol{e}_{\xi,0} + \Delta\boldsymbol{x}) \tag{40}$$

where $\boldsymbol{e}_{\xi,0}$ denotes the initial DCM error. Then, $\boldsymbol{e}_\xi$ tends toward a stationary error $\boldsymbol{e}_{\xi,\text{stat}} = -\Delta\boldsymbol{x}$ for $t \to \infty$.

2) Robustness w.r.t. Unknown External Perturbation: In this section, we examine the effect of an unknown constant external force $\Delta\boldsymbol{F}$ onto the closed-loop dynamics. The total force $\boldsymbol{F}$, acting on the robot's CoM, is composed of the control force $\boldsymbol{F}_c$ and the external perturbation:

$$\boldsymbol{F} = \boldsymbol{F}_c + \Delta\boldsymbol{F}. \tag{41}$$

With (4), (5), (41), and (36), the closed-loop dynamics becomes

$$\begin{bmatrix}\dot{\boldsymbol{x}}\\ \dot{\boldsymbol{\xi}}\end{bmatrix} = \begin{bmatrix}-\frac{1}{b}\,\boldsymbol{I} & \frac{1}{b}\,\boldsymbol{I}\\ \boldsymbol{0} & -k_\xi\,\boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{\xi}\end{bmatrix} + \begin{bmatrix}\boldsymbol{0} & \boldsymbol{0}\\ k_\xi\,\boldsymbol{I} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{\xi_d}\\ \dot{\boldsymbol{\xi}}_{\boldsymbol{d}}\end{bmatrix} + \begin{bmatrix}\boldsymbol{0}\\ \frac{b}{m}\,\boldsymbol{I}\end{bmatrix}\Delta\boldsymbol{F}. \tag{42}$$

The dynamics matrix is unchanged compared with (33); therefore, the eigenvalues are stable for $k_\xi > 0$ and $b > 0$. The dynamics of the DCM error $\boldsymbol{e}_\xi = \boldsymbol{\xi} - \boldsymbol{\xi}_d$ has the solution

$$\boldsymbol{e}_\xi(t) = \frac{b\,\Delta\boldsymbol{F}}{m\,k_\xi} + \underbrace{e^{-k_\xi\, t}}_{\to 0}\left(\boldsymbol{e}_{\xi,0} - \frac{b\,\Delta\boldsymbol{F}}{m\,k_\xi}\right) \tag{43}$$

$\boldsymbol{e}_{\xi,0}$ is the initial DCM error. $\boldsymbol{e}_\xi$ tends toward a stationary error $\boldsymbol{e}_{\xi,\text{stat}} = b\,\Delta\boldsymbol{F}/(m\,k_\xi)$ for $t \to \infty$. Then, the effect of (36) is equivalent to a linear spring of stiffness $c_\xi = \frac{m\,k_\xi}{b}$.

3) Robustness w.r.t. Unknown Force Lag: In real world robotic systems, the commanded force can often not be created instantaneously, but is subject to a lag. To examine this effect, we model the force dynamics as

$$\dot{\boldsymbol{F}} = -k_f\,(\boldsymbol{F} - \boldsymbol{F}_c) \tag{44}$$

where $\boldsymbol{F}$ and $\boldsymbol{F}_c$ denote the real and commanded forces, and $k_f > 0$. Combining (16), (5), and (44) and applying the DCM control law (36), we find the following closed-loop dynamics:

$$\begin{bmatrix}\dot{\boldsymbol{x}}\\ \dot{\boldsymbol{\xi}}\\ \dot{\boldsymbol{F}}\end{bmatrix} = \begin{bmatrix}-\frac{1}{b}\,\boldsymbol{I} & \frac{1}{b}\,\boldsymbol{I} & \boldsymbol{0}\\ -\frac{1}{b}\,\boldsymbol{I} & \frac{1}{b}\,\boldsymbol{I} & \frac{b}{m}\,\boldsymbol{I}\\ \frac{m\,k_f}{b^2}\,\boldsymbol{I} & -\frac{m\,k_f\,(1+k_\xi\,b)}{b^2}\,\boldsymbol{I} & -k_f\,\boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{\xi}\\ \boldsymbol{F}\end{bmatrix} + \begin{bmatrix}\boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{0}\\ \frac{m\,k_\xi\,k_f}{b}\,\boldsymbol{I} & \frac{m\,k_f}{b}\,\boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{\xi_d}\\ \dot{\boldsymbol{\xi}}_{\boldsymbol{d}}\end{bmatrix}. \tag{45}$$

We provide analytic expressions for two critical values of k_f. The first one, $k_{f,\text{crit,stab}}$, is the limit for asymptotic stability of the system (all eigenvalues stable for $k_f > k_{f,\text{crit,stab}}$). The second one, $k_{f,\text{crit,imag}}$, is a limit which assures all eigenvalues are real for $k_f > k_{f,\text{crit,imag}}$.

Fig. 11 (left and mid) shows the typical variation of the three eigenvalues for different force-dynamics constant k_f. One eigenvalue is stable for $k_f > 0$, while the other two are unstable for $k_f < k_{f,\text{crit,stab}}$. We find $k_{f,\text{crit,stab}}$ as

$$k_{f,\text{crit,stab}} = \frac{k_\xi}{1 + k_\xi\,b}\,. \tag{46}$$

Fig. 11 (right) correlates $k_{f,\text{crit,stab}}$ with k_ξ.

Remark 6: $k_{f,\text{crit,stab}}$ converges to an upper limit of $\frac{1}{b}$ for $k_\xi \to \infty$; therefore, for every choice of k_ξ the system is guaranteed to be stable if $k_f > \frac{1}{b}$. Otherwise, it has to be assured that $k_\xi < k_{\xi,\text{max,stable}} = \frac{k_f}{1-k_f\,b}$. Thus, for each $k_f > 0$ a small enough DCM gain k_ξ can be found such that the system is stable. Higher DCM gains lead to instability.

Remark 7: In case all eigenvalues of system (45) are stable, there is no steady-state error (i.e., $\boldsymbol{x} = \boldsymbol{\xi} = \boldsymbol{\xi}_d$).

We will now present the second critical value for k_f, such that for $k_f > k_{f,\text{crit,imag}}$ all eigenvalues are real. For fast force dynamics (high k_f), three real eigenvalues exist. Below a critical value ($k_f < k_{f,\text{crit,imag}}$), only one real eigenvalue exists. For the limit case $k_f = k_{f,\text{crit,imag}}$ the system has one single and one double real eigenvalue [see also Fig. 11 (left)]. With some

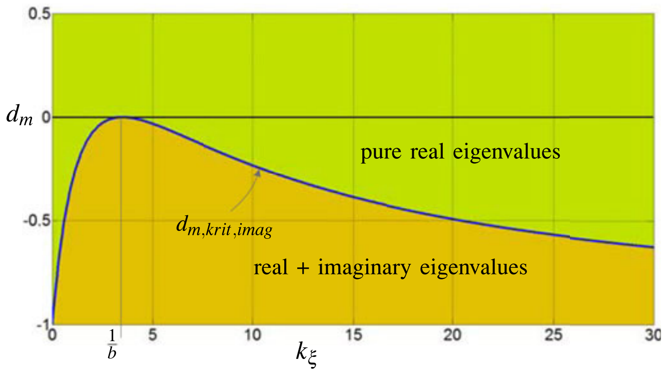

Fig. 12. Influence of mass estimate error on eigenvalues (real versus imag.).

algebraic analysis, we find the critical value as

$$k_{f,\mathrm{crit,imag}} = \frac{2 - 3d - 3d^2 + 2d^3 + 2w}{b(1-d)^2} \tag{47}$$

where $d = k_\xi\, b$ and w stands for

$$w = \sqrt{1 - 3d + 6d^2 - 7d^3 + 6d^4 - 3d^5 + d^6}. \tag{48}$$

In Fig. 11 (right), $k_{f,\mathrm{crit,imag}}$ is plotted over k_ξ.

Remark 8: The analysis presented in this section may also be of interest when designing a (first order) filter for the controller implementation, which might be beneficial in real-world control applications suffering from severe noise. The critical values $k_{f,\text{crit,stab}}$ and $k_{f,\mathrm{crit,imag}}$ can be used to estimate how much filtering is allowed without producing the mentioned undesirable effects (instability and oscillations).

4) Robustness w.r.t. Error in Mass Estimate: In this section, we examine how an inaccurate estimation of the robot's mass affects the stability of the DCM controller (36). For a mass error $\Delta m = d_m\, m$, the estimated mass is

$$\hat{m} = m + \Delta m = (1 + d_m)\, m. \tag{49}$$

Combining (16) and (5) and using (49) in the DCM control law (36), we find the following closed-loop dynamics:

$$\begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{\xi}} \end{bmatrix} = \begin{bmatrix} -\frac{1}{b}\,\boldsymbol{I} & \frac{1}{b}\,\boldsymbol{I} \\ \frac{d_m}{b}\,\boldsymbol{I} & \left(-k_\xi - \left(k_\xi + \frac{1}{b}\right) d_m\right) \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\xi} \end{bmatrix} + \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ (1 + d_m)\, k_\xi\, \boldsymbol{I} & (1 + d_m)\, \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi_d} \\ \dot{\boldsymbol{\xi}}_{\boldsymbol{d}} \end{bmatrix}. \tag{50}$$

The corresponding eigenvalues are

$$\lambda_{1,2} = -\frac{1}{2b}\left(a \pm \sqrt{a^2 - 4\, k_\xi\, b\, (1 + d_m)}\right) \tag{51}$$

where $a = (1 + d_m)(1 + k_\xi b)$. Simple analysis shows that the eigenvalues can only become positive if $k_\xi < 0$ or $d_m < -1$ (total estimated mass smaller than zero) holds. For a positive DCM gain and any estimation of a positive mass (self-evident), the system is guaranteed to be stable.

In order for $\lambda_{1,2}$ to have imaginary parts (see Fig. 12), the radicand of (51) has to be negative, which is fulfilled by

$$d_m < d_{m,\text{krit,imag}} = -\left(\frac{1 - k_\xi\, b}{1 + k_\xi\, b}\right)^2. \tag{52}$$

Equation (52) has a maximum of $d_{m,\text{krit,imag}} = 0$ for $k_\xi = \frac{1}{b}$, i.e., for this choice of k_ξ, the eigenvalues have imaginary parts even for a slightly underestimated mass.

VII. Force Mapping and Feasibility Assurance

This section is meant to provide an outline of how the desired external force from (34) can be mapped to feasible command outputs. In the unperturbed case, our control framework produces desired external forces which intersect with the nominal eCMP and the CoM at all times so that they comply with the contact constraints very well. Yet, in case of perturbations, the desired forces for DCM tracking are not guaranteed to be feasible. This poses the challenge of finding the optimal control outputs, which best fulfill the desired tasks (e.g. contact force control) while being feasible. One option for finding optimal control outputs is quadratic programming (QP) [29], [30]. In the following, one possible QP formulation for mapping desired forces to feasible control outputs is outlined:[2]

$$\begin{aligned} \min_{\ddot{\boldsymbol{q}},\boldsymbol{\tau},\boldsymbol{\rho}}\quad & \|\Delta \boldsymbol{F}_{\text{ext}}\|^2_{\boldsymbol{W_F}} + \|\Delta \boldsymbol{r}(\ddot{\boldsymbol{q}},\boldsymbol{\tau},\boldsymbol{\rho})\|^2_{\boldsymbol{W_r}} \\ s.t.\quad & \boldsymbol{M}\ddot{\boldsymbol{q}} + \boldsymbol{n} = \boldsymbol{S}^T\boldsymbol{\tau} + \boldsymbol{J_c}^T\boldsymbol{W_c} \qquad \text{(eq. of motion)} \\ & \underbrace{\boldsymbol{J_c}\,\ddot{\boldsymbol{q}} + \dot{\boldsymbol{J}}_{\boldsymbol{c}}\,\dot{\boldsymbol{q}}}_{\ddot{\boldsymbol{x}}_{\boldsymbol{c}}} = \boldsymbol{0} \qquad \text{(contact constraint)} \\ & \boldsymbol{W_c} = \boldsymbol{Q}\,\boldsymbol{\rho} \\ & \underline{\boldsymbol{\tau}} \le \boldsymbol{\tau} \le \overline{\boldsymbol{\tau}} \\ & \boldsymbol{0} \le \boldsymbol{\rho} \le \overline{\boldsymbol{\rho}}. \end{aligned} \tag{53}$$

The base link and joint accelerations $\ddot{\boldsymbol{q}}$, the joint torques $\boldsymbol{\tau}$, and the contact force magnitudes $\boldsymbol{\rho}$ are used as optimization variables. As in [30], the friction cones in the robot's feet are approximated by base vectors, such that the contact wrenches $\boldsymbol{W}_c$ can be computed from $\boldsymbol{\rho}$ via the adjoint $\boldsymbol{Q}$. Equation (53) minimizes the norm of the force error $\Delta\boldsymbol{F}_{\text{ext}}$ and all other task errors $\Delta\boldsymbol{r}(\ddot{\boldsymbol{q}},\boldsymbol{\tau},\boldsymbol{\rho})$ (tasks may include, e.g., desired spatial accelerations of the feet and desired body posture), while feasibility is assured. The diagonal matrices $\boldsymbol{W}_F$ and $\boldsymbol{W}_r$ allow to weight the different tasks. The external force error $\Delta\boldsymbol{F}_{\text{ext}}$ can be computed as

$$\Delta \boldsymbol{F}_{\text{ext}} = \underbrace{\boldsymbol{A}\,\boldsymbol{W_c}}_{\boldsymbol{F}_{\text{ext}}} - \boldsymbol{F}_{\text{ext},d}. \tag{54}$$

Here, $\boldsymbol{A}$ denotes the adjoint mapping of the contact wrenches $\boldsymbol{W}_c$ to the external force $\boldsymbol{F}_{\text{ext}}$ acting on the CoM. $\boldsymbol{F}_{\text{ext},d}$ is the desired external force, provided by (34). The first equality constraint in (53) denotes the robot's equation of motion. $\boldsymbol{M}$

[2] IHMC uses a similar approach to control Atlas [29] (see Section VIII-B). QP-based whole body control for TORO [31] is part of our ongoing research.

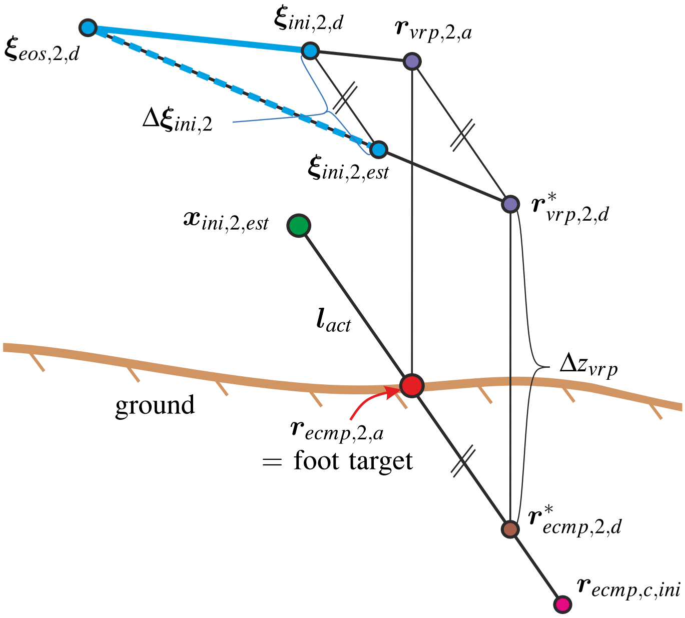
$\boldsymbol{\xi}_{eos,2,d}$
$\boldsymbol{\xi}_{ini,2,d}$
$\boldsymbol{r}_{vrp,2,a}$
$\Delta\boldsymbol{\xi}_{ini,2}$
$\boldsymbol{\xi}_{ini,2,est}$
$\boldsymbol{x}_{ini,2,est}$
$\boldsymbol{r}^*_{vrp,2,d}$
$\boldsymbol{l}_{act}$
Δz_{vrp}
ground
$\boldsymbol{r}_{ecmp,2,a}$
= foot target
$\boldsymbol{r}^*_{ecmp,2,d}$
$\boldsymbol{r}_{ecmp,c,ini}$

Fig. 13. Appropriate step adjustment on 3-D ground surface.

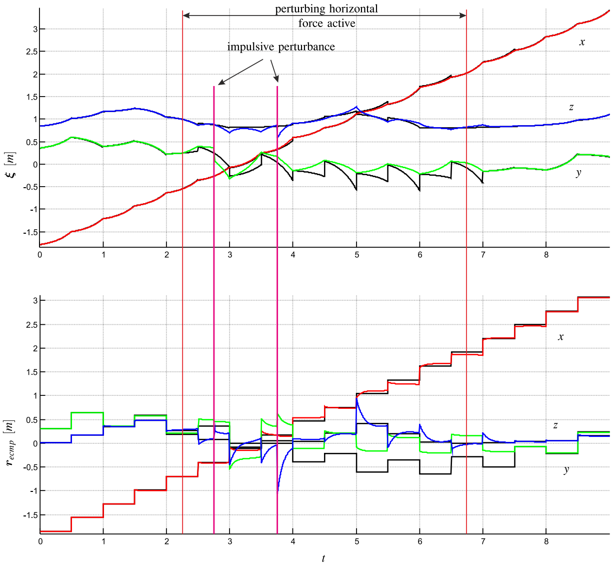

Fig. 14. Prismatic inverted pendulum simulation. Top: desired (black) and real DCMs, Bottom: desired (point foot position, black) and real eCMPs.

denotes the inertia matrix, the vector $\boldsymbol{n}$ encodes Corioli, centrifugal and gravitational torques, $\boldsymbol{S}$ is the actuation matrix, and $\boldsymbol{J}_c$ is the contact Jacobian. The second equality constraint enforces the contact constraint, i.e., $\ddot{\boldsymbol{x}}_{\boldsymbol{c}} = 0$. The operators $\underline{\bullet}$ and $\overline{\bullet}$ in the inequality constraints denote lower and upper limits. The contact force magnitudes $\boldsymbol{\rho}$ are constraint to be positive, such that the unilaterality constraint is ensured. The resulting optimal joint accelerations $\ddot{\boldsymbol{q}}^*$ or torques $\boldsymbol{\tau}^*$ may be commanded to the robot. Note: due to the inequality constraints, global stability, as shown for the underlying control law (32), cannot be guaranteed for underactuated robots.

VIII. Simulations and Experiments

A. Prismatic Inverted Pendulum Simulations (Point Mass)

To demonstrate the performance of the methods proposed in Sections III and V, we carried out simulations based on the Prismatic Inverted Pendulum model [23] (robot is modeled as point mass with two massless point feet). We assume instantaneous impact-free transitions between SS phases (no DS). Fig. 15 shows a screenshot of the simulation. First, we preplan desired footprints (also applicable for online planned foot locations) on a 3-D ground surface of known geometry. The methods for planning and real-time control are exactly the ones described in Sections III and V. Due to the model assumptions (point mass and point foot), only the direction CoM-to-foot is feasible for the external forces, while the other two spatial directions are unactuated (robot follows its unstable natural dynamics). Therefore, the desired eCMP $\boldsymbol{r}_{\text{ecmp},c}$ has to be projected via

$$\boldsymbol{r}_{\text{ecmp},f} = \boldsymbol{x} + \boldsymbol{u}_{x,f}\boldsymbol{u}_{x,f}^T(\boldsymbol{r}_{\text{ecmp},c} - \boldsymbol{x}) \tag{55}$$

where $\boldsymbol{r}_{\text{ecmp},f}$ is the best feasible eCMP and $\boldsymbol{u}_{x,f}$ is the unit vector pointing from the CoM to the point foot. To comply with the constraint that the feet can only push the CoM off, desired eCMPs "above" the CoM are projected onto the CoM itself, resulting in zero external force (i.e., free-falling robot).

To compensate for the lack of controllability, the robot adjusts its target location for the subsequent step. Due to the nonlinearity of (55), the current state is numerically integrated forward to provide estimates for the initial DCM $\boldsymbol{\xi}_{\text{ini},2,\text{est}}$ and CoM $\boldsymbol{x}_{\text{ini},2,\text{est}}$ at the step transition. Reordering (21), we compute the ideal VRP $\boldsymbol{r}^*_{\text{vrp},2,d}$ (and the corresponding eCMP $\boldsymbol{r}^*_{\text{ecmp},2,d}$), which would shift the DCM from the estimated initial DCM $\boldsymbol{\xi}_{\text{ini},2,\text{est}}$ to the final desired DCM $\boldsymbol{\xi}_{eos,2,d}$ of that step within the next stepping time (dashed blue line in Fig. 13). The intersection $\boldsymbol{r}_{\text{ecmp},2,a}$ of the line $\boldsymbol{x}_{\text{ini},2,\text{est}}$-to-$\boldsymbol{r}^*_{\text{ecmp},2,d}$ with the ground is chosen as *target location* for the adjusted step. It is used as current foot location $r_{f,1}$ (see Section III) at the support transition (see step updater block in Fig. 16). The adjusted nominal VRP $\boldsymbol{r}_{\text{vrp},2,a}$ is found Δz_{vrp} further above. With $\boldsymbol{r}_{\text{vrp},2,a}$ and $\boldsymbol{\xi}_{eos,2,d}$, the desired initial DCM $\boldsymbol{\xi}_{\text{ini},2,d}$ is computed via (21). Now, the adjusted DCM reference (solid blue line in Fig. 13) is computed via (22) and (23).

This step adjustment has two beneficial properties: 1) Since the initial commanded eCMP $\boldsymbol{r}_{\text{ecmp},c,\text{ini}}$ [computed from the initial DCM error $\Delta\boldsymbol{\xi}_{\text{ini},2}$ via (35) and (11)] is perfectly in the actuated direction (line CoM-to-foot), good convergence in the beginning of the subsequent step is assured. 2) If the DCM error converges to zero, the commanded eCMP becomes consistent with the point-foot constraint.

In the simulation shown in Figs. 14 and 15, the stepping time was set to 0.5 s. The modeled mass was 60 kg. The approximate "average height" of the CoM over the ground was set to $\Delta z_{\text{vrp}} = 0.8$ m. The surface height varied between plus 50 cm and minus 10 cm. An unknown lateral disturbance of 58.86 N (10% of robot's weight)—active from $t = 2.25$ s till $t = 6.75$ s—was perturbing the robot. Additionally, at $t = 2.75$ s and $t = 3.75$ s, the robot suffered lateral and vertical pushes, each resulting in a sudden velocity change of 1 m/s. The robot was well able to compensate for the perturbations. After the perturbations, the

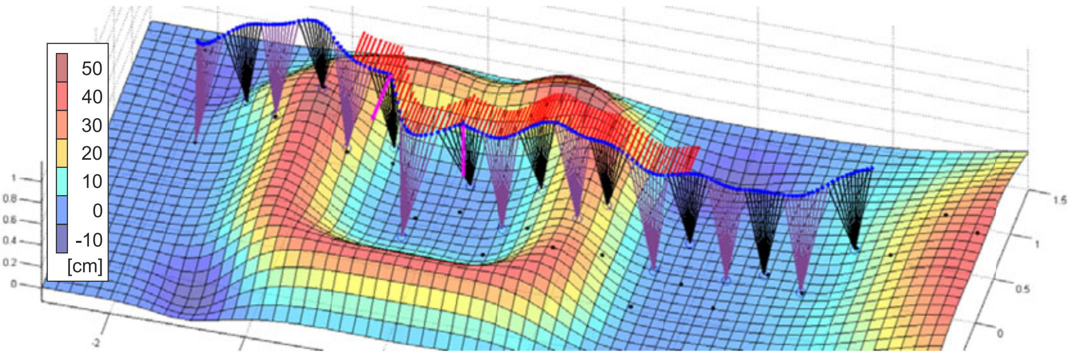

Fig. 15. Simulation of a prismatic inverted pendulum [23]. Walking direction: left to right. The biped is modeled as point mass with two point feet. The red lines denote a lateral disturbance corresponding to 10% of the robot's weight. The pink lines denote a horizontal and a vertical push, each corresponding to a sudden velocity change of 1 m/s. The black points on the ground denote the preplanned footstep locations.

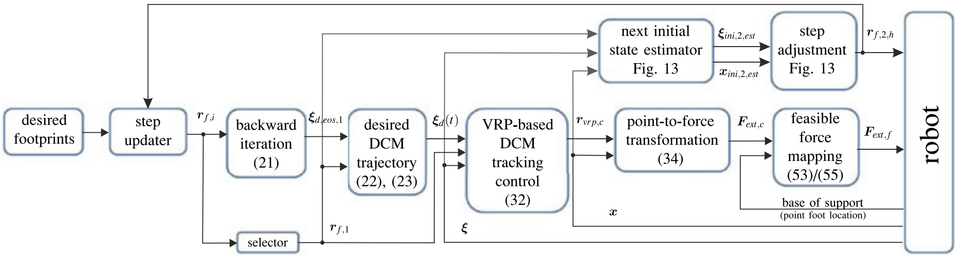

Fig. 16. Overall control framework (step adjustment is only implemented for prismatic inverted pendulum simulation).

desired foot locations were tracked very well after a couple of steps. Note that in this simulation neither a finite-sized foot nor torques around the CoM were available, showing the robustness of the proposed control laws and step adjustment method.

Other simulation setups including (unknown) constant and impulsive perturbations in different directions as well as mass estimation errors also showed a very robust performance of the simulated robot. The robustness increases for shorter times per step t_{step} and bigger DCM gains k_ξ. Fig. 16 shows the overall control framework used for the simulation.

B. Whole-Body Simulations and Experiments

The DCM trajectory generators from Section IV and the associated DCM feedback control (32) were tested in numerous simulations and experiments with DLR's humanoid robot TORO [31] and IHMC's Atlas robot [5].

We carried out simulations in OpenHRP3 [32] in which TORO walks over a set of stairs (see Fig. 17). The desired and real (three-dimensional) DCM and eCMP trajectories are shown in Fig. 18. The DCM trajectories are tracked accurately, and the eCMP transitions are smooth during DS.

During the preparations for DARPA's Virtual Robotics Challenge (VRC) [29], the CDS trajectory generator (Section IV-A) was first developed and subsequently evolved into the HT trajectory generator (Section IV-B). Based on the latter, the simulated Atlas robot could achieve a maximum step length of up to 70 cm [see Fig. 19 (left)] in both IHMC's simulation environment [33] and the official VRC simulator Gazebo [34]. The algorithms described in this paper were also used for the DARPA Robotics Challenge Trials 2013 [35]. Amongst others, IHMC's Atlas robot accomplished to get full marks for the terrain task, which was composed of ramps, stairs, and inclined stairs [see Fig. 19 (right)]. Fig. 20 shows an experimental result of Atlas walking on flat ground while using the CDS trajectory generator (from Section IV-A). The plot shows a series of seven steps. It displays estimated and desired eCMP and DCM trajectories in x (forward) and y (sideward) direction. The estimated DCM and eCMP were attained via a Kalman filter [29] and by mapping joint torques to ground reaction forces. Apart from deviations related to the foot touch-downs, both eCMP and DCM trajectories are tracked well. The eCMP transition during DS is continuous, resulting in continuous leg forces and joint torques.

IX. Conclusion and Future Work

We have demonstrated a control framework for bipedal walking on uneven terrain. We extended the concept of "DCM" [17] to 3-D and used it to derive methods for planning and tracking of 3-D DCM trajectories in real-time. We proposed two DCM trajectory generators for CDS and HT transitions, which lead to continuous leg forces (and thus joint torques) and allow for toe-off motion during DS. An asymptotically stable DCM tracking controller was proposed, and its robustness with respect to model uncertainties was thoroughly examined. Additionally, we outlined a possible QP formulation which assures feasibility of the finally commanded control outputs. Moreover, we introduced a method for step adjustment on uneven terrain, which results in robust walking of highly underactuated robot models. The capabilities of the control framework were demonstrated in simulations and experiments. The derivation of the trajectory generators and controllers was facilitated by the simplicity

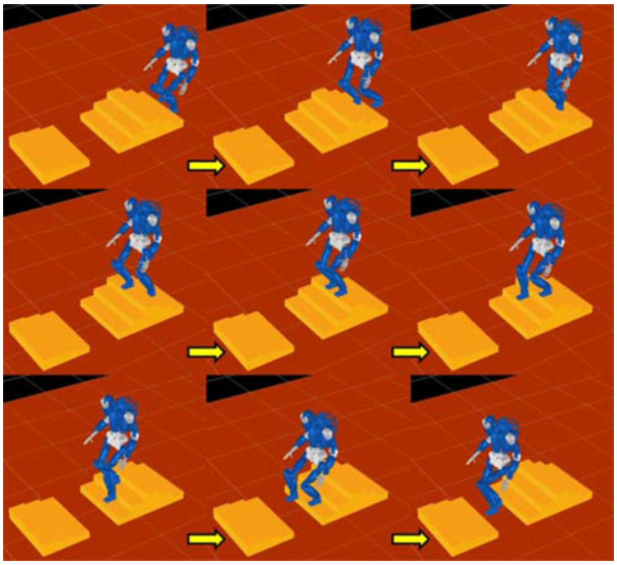

Fig. 17. OpenHRP3 simulation [32] of DLR's humanoid robot TORO. Walking over a set of stairs of variable height. (step height differences: $[+12, +12, +12, -12, -12, -12, +10, +5, +3, -18]$ cm).

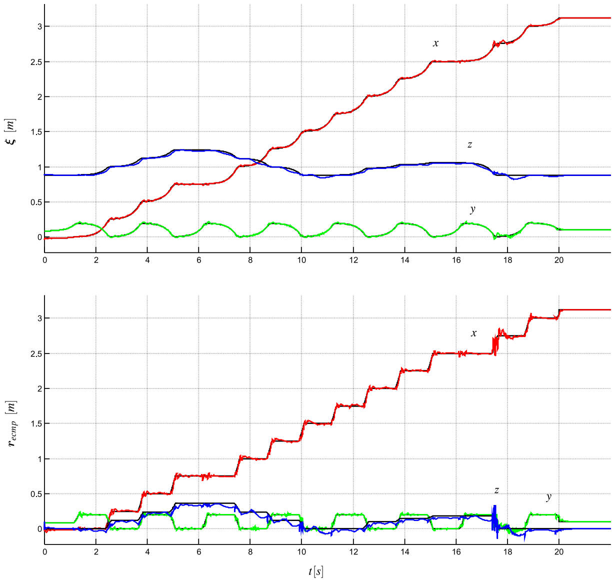

Fig. 18. OpenHRP3 simulation of humanoid robot TORO. Top: desired (black) and real DCMs, Bottom: desired (black) and real eCMPs.

of the DCM dynamics (first order). The proposed DCM-based walking control framework stands out with regard to comprehensibleness and low computational cost.

Our future work includes the use of the proposed control strategies on DLR's humanoid TORO [31]. In particular, we plan to embed the proposed DCM-based walking controllers into an inverse dynamics-based whole-body control framework and to examine its robustness. That way our currently admittance-based walking will evolve to force-controlled walking (TORO can be torque-controlled), which we presume to be more robust due to its compliant behavior. Also, we plan to work on walking on uneven ground and push recovery.

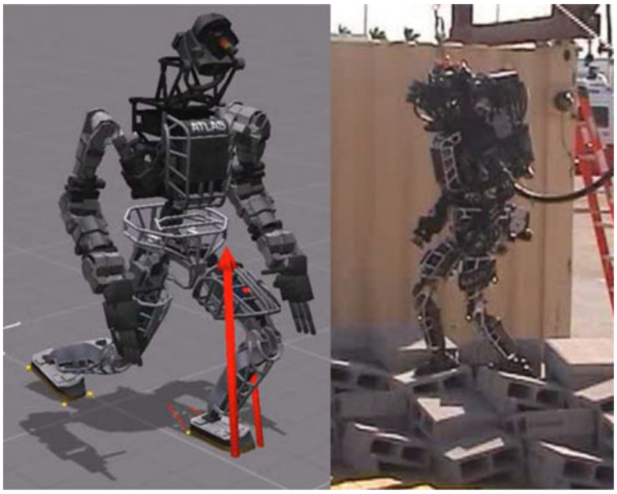

Fig. 19. Left: Atlas robot [5] doing 70 cm steps in IHMC's simulation environment (SimulationConstructionSet [33]) using toe-off motion. Right: Physical Atlas robot walking over cinder blocks during DRC Trials.

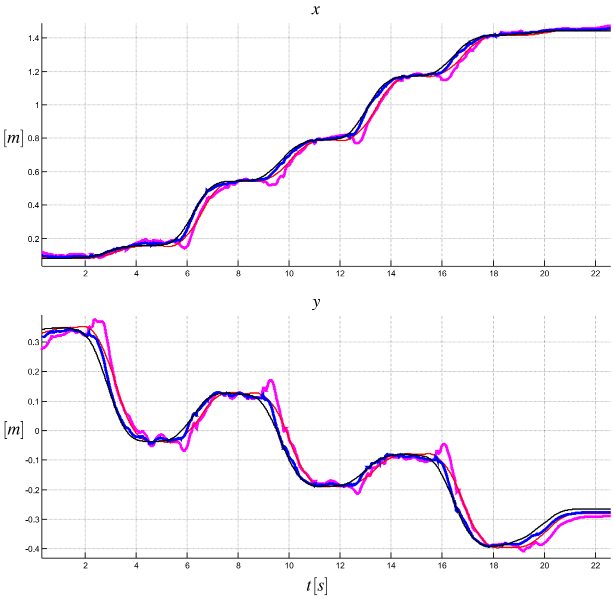

Fig. 20. Experimental results of IHMC's Atlas robot walking during DRC Trials. Pink/red: estimated/desired eCMP, blue/black: estimated/desired DCM.

Acknowledgment

J. Englsberger would like to thank J. Pratt and his colleagues at IHMC's robotics laboratory for welcoming him for a three-month research visit during DARPA's VRC in 2013.

References

[1] S. Dubowsky and E. Papadopoulos, "The kinematics, dynamics, and control of free-flying and free-floating space robotic systems," *IEEE Trans. Robot. Autom.*, vol. 9, no. 5, pp. 531–543, Oct. 1993.

[2] P.-B. Wieber, "Holonomy and nonholonomy in the dynamics of articulated motion," *Fast Motions in Biomechanics and Robotics* (Lecture Notes in Control and Information Sciences), vol. 340. Berlin, Germany: Springer, 2006, pp. 411–425.

[3] T. Sugihara, Y. Nakamura, and H. Inoue, "Realtime humanoid motion generation through ZMP manipulation based on inverted pendulum control," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2002, pp. 1404–1409.

[4] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, "Biped walking pattern generation by using preview control of zero-moment point," in *IEEE Int. Conf. Robot. Autom.*, 2003, pp. 1620–1626.

[5] Atlas—the agile anthropomorphic robot. (2015). [Online]. Available: http://www.bostondynamics.com/robot_Atlas.html

[6] M. Vukobratovic and Y. Stepanenko, "On the stability of anthropomorphic systems," *Math. Biosci.*, vol. 15, pp. 1–37, 1972.

[7] K. Mitobe, G. Capi, and Y. Nasu, "Control of walking robots based on manipulation of the zero moment point," *Robotica*, vol. 18, no. 6, pp. 651–657, 2000.

[8] B. Stephens and C. Atkeson, "Push recovery by stepping for humanoid robots with force controlled joints," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2010, pp. 52–59.

[9] P.-B. Wieber, "Trajectory free linear model predictive control for stable walking in the presence of strong perturbations," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2006, pp. 137–142.

[10] K. Nishiwaki and S. Kagami, "Online walking control system for humanoids with short cycle pattern generation," *Int. J. Robot. Res.*, vol. 28, no. 6, pp. 729–742, 2009.

[11] T. Sugihara, "Standing stabilizability and stepping maneuver in planar bipedalism based on the best COM-ZMP regulator," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2009, pp. 1966–1971.

[12] S. Kajita, M. Morisawa, K. Miura, S. Nakaoka, K. Harada, K. Kaneko, F. Kanehiro, and K. Yokoi, "Biped walking stabilization based on linear inverted pendulum tracking," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, pp. 4489–4496.

[13] A. L. Hof, "The 'extrapolated center of mass' concept suggests a simple control of balance in walking," *Human Movement Sci.*, vol. 27, pp. 112–125, 2008.

[14] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture point: A step toward humanoid push recovery," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2006, pp. 200–207.

[15] T. Koolen, T. D. Boer, J. Rebula, A. Goswami, and J. E. Pratt, "Capturability-based analysis and control of legged locomotion. Part 1: Theory and application to three simple gait models," *Int. J. of Robot. Res.*, vol. 31, no. 9, pp. 1094–1113, 2012.

[16] J. E. Pratt, T. Koolen, T. D. Boer, J. Rebula, S. Cotton, J. Carff, M. Johnson, and P. Neuhaus, "Capturability-based analysis and control of legged locomotion, Part 2: Application to m2v2, a lower-body humanoid," *Int. J. Robotics Res.*, vol. 31, no. 10, pp. 1117–1133, 2012.

[17] T. Takenaka, T. Matsumoto, and T. Yoshiike, "Real time motion generation and control for biped robot, 1st report: Walking gait pattern generation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2009, pp. 1084–1091.

[18] J. Englsberger, C. Ott, M. A. Roa, A. Albu-Schäffer, and G. Hirzinger, "Bipedal walking control based on capture point dynamics," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2011, pp. 4420–4427.

[19] J. Englsberger and C. Ott, "Integration of vertical COM motion and angular momentum in an extended capture point tracking controller for bipedal walking," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2012, pp. 183–189.

[20] M. Morisawa, S. Kajita, F. Kanehiro, K. Kaneko, K. Miura, and K. Yokoi, "Balance control based on capture point error compensation for biped walking on uneven terrain," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2012, pp. 734–740.

[21] K. Seo, J. Kim, and K. Roh, "Towards natural bipedal walking: Virtual gravity compensation and capture point control," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2012, pp. 4019–4026.

[22] J. Englsberger, C. Ott, and A. Albu-Schäffer, "Three-dimensional bipedal walking control using divergent component of motion," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 2600–2607.

[23] Y. Zhao and L. Sentis, "A three dimensional foot placement planner for locomotion in very rough terrains," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2012, pp. 726–733.

[24] A. D. Kuo, "Energetics of actively powered locomotion using the simplest walking model," *J. Biomech. Eng.*, vol. 124, pp. 113–120, 2002.

[25] Y. Ogura, K. Shimomura, H. Kondo, A. Morishima, T. Okubo, S. Momoki, H. ok Lim, and A. Takanishi, "Human-like walking with knee stretched, heel-contact and toe-off motion by a humanoid robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2006, pp. 3976–3981.

[26] J. Slotine and W. Li, "On the adaptive control of robot manipulators," *Int. J. Robot. Res.*, vol. 6, no. 3, pp. 49–59, 1987.

[27] M. B. Popovic, A. Goswami, and H. Herr, "Ground reference points in legged locomotion: Definitions, biological trajectories and control implications," *Int. J. Robot. Res.*, vol. 24, no. 12, pp. 1013–1032, 2005.

[28] J. Englsberger, T. Koolen, S. Bertrand, J. Pratt, C. Ott, and A. Albu-Schäffer, "Trajectory generation for continuous leg forces during double support and heel-to-toe shift based on divergent component of motion," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 4022–4029.

[29] T. Koolen, J. Smith, G. Thomas, S. Bertrand, J. Carff, N. Mertins, D. Stephen, P. Abeles, J. Englsberger, S. McCrory, *et al.*, "Summary of team ihmc's virtual robotics challenge entry," in *Proc. Int. Conf. Humanoid Robots*, 2013, pp. 307–314.

[30] L. Righetti, J. Buchli, M. Mistry, M. Kalakrishnan, and S. Stefan, "Optimal distribution of contact forces with inverse dynamics control," *Int. J. Robot. Res.*, vol. 32, no. 3, pp. 280–298, 2013.

[31] J. Englsberger, A. Werner, C. Ott, B. Henze, M. A. Roa, G. Garofalo, R. Burger, A. Beyer, O. Eiberger, K. Schmid, and A. Albu-Schffer,"Overview of the torque-controlled humanoid robot toro," in *IEEE-RAS Int. Conf. Humanoid Robots*, pp. 916–923, 2014.

[32] F. Kanehiro, K. Fujiwara, S. Kajita, K. Yokoi, K. Kaneko, H. Hirukawa, Y. Nakamura, and K. Yamane, "Open architecture humanoid robotics platform," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2002, pp. 24–30.

[33] "Simulation construction set—IHMC's simulation environment for humanoid robots. (2015). [Online]. Available: http://ihmc.us/groups/scs/

[34] Gazebo—a multi-robot simulator for outdoor environments. (2015). [Online]. Available: http://gazebosim.org

[35] "Darpa robotics challenge trials 2013. (2015). [Online]. Available: http://www.theroboticschallenge.org/

Johannes Englsberger received the Dipl.-Ing. degree in mechanical engineering from Technical University of Munich, Munich, Germany, in 2009. Since then, he has been working toward the Ph.D. degree with the German Aerospace Center (DLR), Institute of Robotics and Mechatronics, Wessling, Germany.

His research interests include bipedal walking and running control, humanoid whole-body control, and mechanical design.

Christian Ott received the Dipl.-Ing. degree in mechatronics from Johannes Kepler University, Linz, Austria, in 2001 and the Dr.-Ing. degree in control engineering from Saarland University, Saarbrcken, Germany, in 2005.

From 2001 to 2007, he was with DLR, Institute of Robotics and Mechatronics, Wessling, Germany. From May 2007 to June 2009, he was Project Assistant Professor with the Department of Mechano-Informatics, University of Tokyo, Tokyo, Japan. Since then, he has been the Team Leader with the Helmholtz Young Investigators Group for "Dynamic Control of Legged Humanoid Robots" at DLR. In January 2014, he became the Head of the Department of Analysis and Control of Advanced Robotic Systems at DLR. His research interests include nonlinear robot control, flexible joint robots, impedance control, and humanoid manipulation.

Alin Albu-Schäffer received the Graduation degree in electrical engineering from Technical University of Timisoara, Timisoara, Romania, in 1993 and the Ph.D. degree in automatic control from the Technical University of Munich (TUM), Munich, Germany, in 2002.

Since 2012, he has been the Head of the DLR's Institute of Robotics and Mechatronics, Wessling, Germany. Moreover, he is a Professor at TUM, holding the Chair for "Sensor based Robotic Systems and Intelligent Assistance Systems" at the Computer Science Department. His research interests include robot design, modeling and control, flexible joint, and variable compliance robots for manipulation and locomotion, physical human–robot interaction, bioinspired robot design.