

# DCM-based gait generation for walking on moving support surfaces

Johannes Engelsberger, George Mesesan, Christian Ott, Alin Albu-Schäffer

**Abstract**— This paper presents a novel walking gait generator that allows the successful traversal of moving support surfaces such as conveyor belts, moving plates and escalators. The gait generator previews all steps of a complete gait sequence, while providing efficient matrix-vector based computations. The moving support surfaces are explicitly taken into account for the trajectory design. Multiple successful simulations of walking on different non-stationary ground surfaces prove the high quality of the proposed walking gait generator.

## I. INTRODUCTION

Robotic walking is regarded as a difficult problem: walking robots have a high number of degrees of freedom (DOF), their system dynamics is nonlinear and hybrid, and – to avoid falling – contact stability constraints need to be obeyed. A widely used approach to handle this high complexity is model reduction. The most popular reduced model focuses on the robot’s CoM dynamics. In [1], Kajita et al. introduced the *linear inverted pendulum* (LIP) model that uses the zero momentum point (ZMP) as input and further constrains the CoM dynamics to a horizontal plane. Many works (including [2]–[6]) have presented successful walking gait generation and/or control based on the LIP model.

A further break-through in the field of online walking gait generation and control was the decomposition of the LIP model into a stable and an unstable component. The latter was first referred to by Pratt et al. [7] as ‘(instantaneous) Capture Point’ and by Hof et al. [8] as ‘extrapolated Center of Mass’. Takenaka et al. [9] introduced the term ‘Divergent Component of Motion’ (DCM) for it, which we use in the presented work. The DCM was successfully applied to flat floor walking, e.g. in [9]–[11]. In [12], we extended the DCM model to 3D, while Hopkins et al. [13] extended it to varying virtual pendulum heights (time-varying DCM). In [14], Mesesan et al. analyze the convex properties of CoM trajectories based on DCM and perform multi-contact locomotion in simulation. While robotic walking over flat or uneven *stationary* grounds has been successfully addressed, robotic walking across moving ground surfaces, as presented by Unhelkar et al. [15] for a wheeled mobile platform, is not covered by the state of the art.

This paper presents an awesome gait generator that extends our DCM-based planning and control framework [14], [16]–[18] to allow the traversal of moving support surfaces such as moving plates, conveyor belts and escalators. While in case of walking on stationary ground, one may argue about the number of previewed steps, a multi-step preview

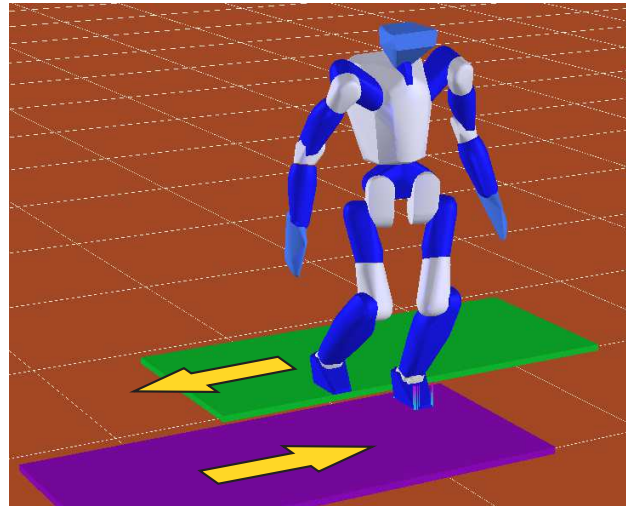


Fig. 1. Toro [19] walking over two moving plates in OpenHRP [20].

is crucial in case of moving<sup>1</sup> ground surfaces. In contrast to other gait generation methods including [2], [4], our method facilitates a preview of the *complete gait sequence* via efficient matrix-vector based computations. The resulting trajectories obey appropriate initial and terminal constraints. Even though the feet are moving with the respective support surfaces, the ground reference points are guaranteed to be feasible (assuming sufficient friction). Finally, the foot trajectories are designed to match the moving support surface motion, such that impacts with the floor are avoided. These features result in a remarkable tracking performance.

To the best of our knowledge, the presented walking gait generator is the first of its kind, i.e., to date no other frameworks exist that would allow for comparably efficient gait generation for walking over ground surfaces that move at different speeds.

The paper is structured as follows: Section II reviews the basics of the control points used in this work. In Section III, we derive single transition phase trajectories, which are then used in Section IV as building blocks to compose multi-phase trajectories for a *complete walking sequence*. Sections V and VI provide implementation details and describe simulations, respectively, that verify the performance of our gait generator. Section VII concludes the paper.

<sup>1</sup>Note: it can be shown that walking within a reference system that is moving at a *single constant velocity* (e.g. within a train) follows the same physics as walking on stationary ground and may thus be handled by standard walking gait generators and controllers. In contrast, walking over support surfaces that move at *different velocities* (w.r.t. direction and/or magnitude) requires more sophisticated gait generation methods.

## II. REVIEW OF BASICS

The fundamental theory on the *Divergent Component of Motion* (DCM), the *enhanced Centroidal Moment Pivot* (eCMP) and the *Virtual Repellent Point* (VRP) can be found in our previous work [16], [17]. In this section, we summarize their definitions and properties.

The DCM is defined as

$$\xi = x + b \dot{x} . \quad (1)$$

Here,  $x$  and  $\dot{x}$  denote the center of mass (CoM) position and velocity, respectively, each being a three-dimensional quantity. The DCM time-constant is denoted by  $b$ , which can be derived from the average height of the CoM above the ground surface  $\Delta z_{vrp}$  as  $b = \sqrt{\Delta z_{vrp}/g}$ ,  $g$  being the gravitational constant (see [16], [17] for more details).

Reordering (1), we find the CoM dynamics

$$\dot{x} = -\frac{1}{b} (x - \xi) , \quad (2)$$

which shows that the CoM follows the DCM with a stable first order dynamics. The CoM dynamics can thus (assuming sufficient friction, see [18]) be neglected w.r.t. planning and control, which facilitates the gait design process.

Differentiating (1) and with Newton's 2<sup>nd</sup> law  $\ddot{x} = F/m$  ( $m$ : robot's total mass,  $F$ : total force acting on CoM), we find the following unstable first order dynamics for the DCM:

$$\dot{\xi} = \frac{1}{b} (\xi - v) . \quad (3)$$

Here we already inserted the definition of the Virtual Repellent Point (VRP)  $v$ . The VRP encodes  $F$  via

$$F = \frac{m}{b^2} (x - v) . \quad (4)$$

Looking at (3), we find that the DCM is pushed away from the VRP, i.e., it *diverges*. This divergent nature of the DCM can be used to design DCM reference trajectories that obey a terminal constraint (see Sec. V-B). Additionally, corresponding CoM trajectories can be derived.

Another three-dimensional point that is interesting for walking gait design and control is the so called *enhanced Centroidal Moment Pivot* (eCMP). In contrast to the Centroidal Moment Pivot (CMP, [21]), which is defined as a point on the ground surface that encodes the external force *direction*, the eCMP is a 3D point encoding both the direction and the *magnitude* of the external force  $F_{ext}$  via

$$F_{ext} = \frac{m}{b^2} (x - e) . \quad (5)$$

The eCMP can be used for three-dimensional walking gait generation, in a similar manner as the ZMP is used for flat-floor gait design. The difference between (4) and (5) represents the gravitational force, which corresponds to a VRP  $v$  at a constant height<sup>2</sup>  $\Delta z_{vrp}$  above the eCMP  $e$ , i.e.,  $v = e + [0, 0, \Delta z_{vrp}]^T$ . Due to this constant offset, considerations made for the eCMP – including the point to point interpolations from Sec. III-C – can be directly transferred to the VRP (and vice versa).

<sup>2</sup> $\Delta z_{vrp}$  corresponds to average CoM height above the ground surface.

## III. SINGLE TRANSITION PHASE TRAJECTORIES AS BUILDING BLOCK

In this work, we use  $n_{wp}$  waypoints for VRP, DCM and CoM, respectively, and perform piecewise interpolation between them, which yields  $n_\phi = n_{wp} - 1$  *single transition phase trajectories*. These are then used as building blocks for composing the corresponding *complete* reference trajectories.

### A. General solution for DCM trajectories corresponding to polynomial VRP trajectories as input

In sections III-A and III-B, we provide explicit solutions<sup>3</sup> for both DCM and CoM trajectories for arbitrary *polynomial* VRP reference trajectories of the form

$$v^T(t) = t^T(t) P_v . \quad (6)$$

Here,  $t^T(t) = [1, t, t^2, \dots, t^{o_v}]$ ,  $o_v$  is the polynomial order of  $v(t)$  and  $P_v$  is a polynomial parameter matrix. Inserting (6) into (3) yields:

$$\dot{\xi}^T(t) = \frac{1}{b} \xi^T(t) - \frac{1}{b} \underbrace{t^T(t) P_v}_{v^T(t)} . \quad (7)$$

This ODE can be solved using the solution from appendix I-A by setting  $y(t) \rightarrow \xi^T(t)$ ,  $a \rightarrow b$  and  $p_u \rightarrow P_v$  in (43):

$$\xi^T(t) = e^{\frac{t}{b}} \left( \underbrace{\xi_0^T - t^T(0) C(b) P_v}_{\xi_e^T} \right) + t^T(t) \underbrace{C(b) P_v}_{P_\xi} . \quad (8)$$

Evaluating (8) for  $t = T$ ,  $\xi_e^T$  can be written alternatively as

$$\xi_e^T = e^{-\frac{T}{b}} \left( \xi_T^T - t^T(T) C(b) P_v \right) , \quad (9)$$

where  $\xi_T := \xi(T)$ . Replacing  $\xi_e^T$  from (9) into (8), we find

$$\xi^T(t) = \underbrace{e^{\frac{t-T}{b}} \xi_T^T}_{\gamma_\xi^T(t)} + \underbrace{\left( t^T(t) - e^{\frac{t-T}{b}} t^T(T) \right) C(b) P_v}_{c_{P_v \rightarrow \xi^T}^T} . \quad (10)$$

This is the DCM solution for a given terminal DCM  $\xi_T$  using an arbitrary *polynomial* VRP reference as input.

### B. General solution for CoM trajectories corresponding to polynomial VRP trajectories as input

In this section, we will derive an explicit solution for the CoM trajectory. By setting  $\hat{y}(t) \rightarrow x^T(t)$ ,  $\hat{a} \rightarrow -b$  and  $\hat{u}(t) \rightarrow \xi^T(t)$ , the general ODE (44) from appendix I becomes equivalent to (2):

$$\dot{x}^T(t) = -\frac{1}{b} x^T(t) + \frac{1}{b} \underbrace{\left( e^{\frac{t}{b}} \xi_e^T + t^T(t) P_\xi \right)}_{\xi^T(t)} \quad (11)$$

Here, we already inserted the DCM solution from (8) as input. By comparing  $\xi^T(t)$  from (8) to  $\hat{u}(t)$  from (45), we

<sup>3</sup>Note: for better readability, the basic math for solving the required specific ordinary differential equations (ODE) was moved to appendix I.

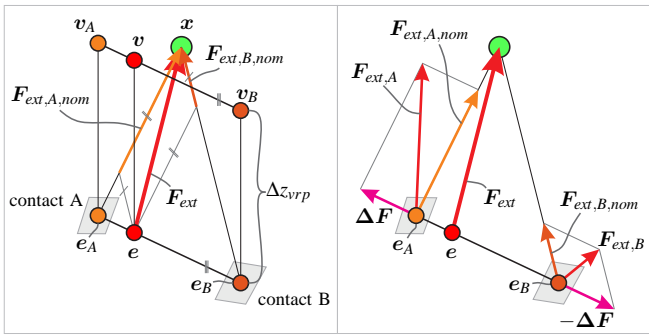


Fig. 2. Convex support point interpolation leads to feasible single foot/contact forces  $\mathbf{F}_{ext,A,(nom)}$  and  $\mathbf{F}_{ext,B,(nom)}$ . Time dependencies omitted.

infer that the solution of (11) can be found by additionally setting  $\beta \rightarrow b$ ,  $\hat{u}_e \rightarrow \xi_e^T$  and  $\mathbf{p}_{\hat{u}} \rightarrow \mathbf{P}_\xi$  in (46)

$$\begin{aligned} \mathbf{x}^T(t) &= \frac{1}{2} \underbrace{\left( e^{\frac{t}{b}} - e^{-\frac{t}{b}} \right)}_{\sinh\left(\frac{t}{b}\right)} \xi_e^T + e^{-\frac{t}{b}} \mathbf{x}_0^T \quad (12) \\ &+ \left( \mathbf{t}^T(t) - e^{-\frac{t}{b}} \mathbf{t}^T(0) \right) \mathbf{C}(-b) \mathbf{P}_\xi. \end{aligned}$$

Inserting  $\mathbf{P}_\xi$  and  $\xi_e^T$  from (8) and (9), respectively, into (12), we finally express the CoM solution as

$$\begin{aligned} \mathbf{x}^T(t) &= \underbrace{\sinh\left(\frac{t}{b}\right) e^{-\frac{t}{b}} \xi_T^T}_{\gamma_x(t)} + \underbrace{e^{-\frac{t}{b}} \mathbf{x}_0^T}_{\delta_x(t)} \quad (13) \\ &+ \underbrace{\left( \left( \mathbf{t}_t^T - e^{-\frac{t}{b}} \mathbf{t}_0^T \right) \mathbf{C}_{-b} - \sinh\left(\frac{t}{b}\right) e^{-\frac{t}{b}} \mathbf{t}_T^T \right) \mathbf{C}_b \mathbf{P}_v}_{\mathbf{c}_{P_v \rightarrow x}^T}. \end{aligned}$$

Here,  $\mathbf{C}(-b)$ ,  $\mathbf{C}(b)$ ,  $\mathbf{t}^T(t)$ ,  $\mathbf{t}^T(0)$  and  $\mathbf{t}^T(T)$  are denoted by  $\mathbf{C}_{-b}$ ,  $\mathbf{C}_b$ ,  $\mathbf{t}_t^T$ ,  $\mathbf{t}_0^T$  and  $\mathbf{t}_T^T$ , respectively.

Note: in our previous works [16]–[18] (and also for the DCM controller used in this paper), explicit CoM trajectories are not required<sup>4</sup>. We provide CoM solutions here, since other controllers may require explicit CoM reference trajectories. Also, these CoM trajectories will serve as a basis for our future research on fully (including friction constraints) dynamically feasible motion planning.

### C. Convex interpolation between moving eCMP/VRP support points

In this section, we derive feasible eCMP and VRP trajectories for traversing moving support surfaces. To this end, we attach eCMP support points to the moving support surfaces, which are thus moving relative to the inertial frame while being stationary w.r.t. a local frame attached to each respective support surface. For each single transition phase, the reference eCMP  $e(t)$  is constructed via

$$e(t) = (1 - f(t)) e_A(t) + f(t) e_B(t), \quad (14)$$

<sup>4</sup>since only unilaterality of contact forces and eCMP (similar to ZMP) constraints are guaranteed to avoid tilting of the robots feet, while friction is neglected during planning.

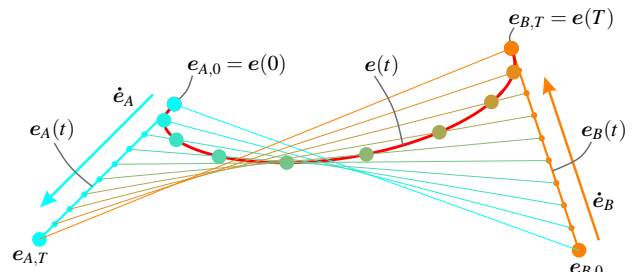


Fig. 3. Moving eCMP (VRP equivalent) support point interpolation

i.e., via linear interpolation between a *preceding eCMP support point*  $e_A(t)$  and a *leading eCMP support point*  $e_B(t)$  using a temporal interpolation function  $f(t)$ . Furthermore, if  $0 \leq f(t) \leq 1 \forall t \in [0, T]$  ( $T$  denoting the duration of the interpolation), (14) forms a *convex* interpolation between the two points [14]. Inserting (14) into (5) we find

$$\mathbf{F}_{ext}(t) = \frac{m}{b^2} \left( \mathbf{x} - \underbrace{\left( (1 - f(t)) e_A(t) + f(t) e_B(t) \right)}_{e(t)} \right), \quad (15)$$

which can be split into a pair of single external forces

$$\mathbf{F}_{ext,A,nom}(t) = (1 - f(t)) \frac{m}{b^2} \left( \mathbf{x}(t) - e_A(t) \right) \quad (16)$$

and

$$\mathbf{F}_{ext,B,nom}(t) = f(t) \frac{m}{b^2} \left( \mathbf{x}(t) - e_B(t) \right). \quad (17)$$

These nominal single external forces  $\mathbf{F}_{ext,A,nom}(t)$  and  $\mathbf{F}_{ext,B,nom}(t)$  are proportional to the offsets of  $e_A(t)$  and  $e_B(t)$  from the CoM, respectively. Therefore, they automatically pass through both the respective eCMP support points and the CoM (see Fig. 2 (left)) and are thus feasible with regard to unilaterality and non-tilting<sup>5</sup> constraints by *design*.

While the support points move w.r.t. the spatial reference system, they are stationary w.r.t. their respective support surfaces (i.e., in the moving reference systems). That way tilting of any one of the supporting feet/contacts is avoided. Friction cone constraints may be

- fulfilled already.
- obtained at the planning stage by force modulation in the nullspace of (15), i.e., by adding forces  $\pm \Delta \mathbf{F}$  to the respective nominal contact forces  $\mathbf{F}_{ext,A,nom}$  and  $\mathbf{F}_{ext,B,nom}$  that cancel each other and thus produce no CoM forces or torques (see Fig. 2 (right)).
- enforced via a quadratic program (QP) at a higher level by making use of the full robot dynamics instead of the simple support point dynamics considered for planning.

After verifying the feasibility of external forces, we now specify the support point motions. In this paper, we limit our analysis to support points moving at *constant velocities*<sup>6</sup>:

$$e_A(t) = e_{A,0} + t \dot{e}_A \quad (18)$$

$$e_B(t) = e_{B,0} + t \dot{e}_B$$

<sup>5</sup>given that  $e_A(t)$  and  $e_B(t)$  are within the “single contact support polygons” of the (potentially moving) contacts A and B, respectively.

<sup>6</sup>Our method could handle arbitrary *polynomial* support point trajectories.

$o_f$	$\mathbf{p}_f^T$	boundary conditions
1	$[0, \frac{1}{T}]$	$f(0) = 0, f(T) = 1$
3	$[0, 0, \frac{3}{T^2}, -\frac{2}{T^3}]$	$f(0) = 0, f(T) = 1$ $\dot{f}(0) = 0, \dot{f}(T) = 0$
5	$[0, 0, 0, \frac{10}{T^3}, -\frac{15}{T^4}, \frac{6}{T^5}]$	$f(0) = 0, f(T) = 1$ $\dot{f}(0) = 0, \dot{f}(T) = 0$ $\ddot{f}(0) = 0, \ddot{f}(T) = 0$

TABLE I  
VARIOUS POSSIBLE INTERPOLATION SCHEMES.

Here,  $e_{A,0} := e_A(0)$  and  $e_{B,0} := e_B(0)$ . The support point velocities  $\dot{e}_A$  and  $\dot{e}_B$  are equal to the corresponding support contact velocities, i.e.,  $e_A$  and  $e_B$  are stationary w.r.t. the respective contact frames. Due to its convex construction via (14), the interpolated eCMP  $e(t)$  always lies on the line connecting the two support points  $e_A(t)$  and  $e_B(t)$  (see Fig. 3). Note the similarity to Bézier curves, the main distinctive feature being that in our setup the start position  $e_{B,0}$  of the leading support point is not constrained to coincide with the end position  $e_{A,T} = e_A(T)$  of the preceding support point.

Unlike other methods (e.g. [13], [22], [23]) that use a time-varying virtual pendulum length and thus parameter  $b$ , in our work the *constant* average CoM height  $\Delta z_{vvp}$  (corresponding to a time *constant*  $b$ ) leads to an equivalent usage of eCMP  $e$  and VRP  $v$  for both planning and control<sup>7</sup>. Any corresponding eCMP/VRP points and trajectories are simply offset vertically by  $\Delta z_{vvp}$ . Therefore, for the eCMP support point interpolation (14) a corresponding equivalent VRP support point interpolation exists:

$$\mathbf{v}(t) = (1 - f(t)) \mathbf{v}_A(t) + f(t) \mathbf{v}_B(t), \quad (19)$$

with the *preceding VRP support point*  $\mathbf{v}_A(t)$  and *leading VRP support point*  $\mathbf{v}_B(t)$ , which are obtained via

$$\mathbf{v}_A(t) = \mathbf{v}_{A,0} + t \dot{\mathbf{v}}_A \quad (20)$$

$$\mathbf{v}_B(t) = \mathbf{v}_{B,0} + t \dot{\mathbf{v}}_B \quad (21)$$

where  $\mathbf{v}_{A,0} := \mathbf{v}_A(0)$  and  $\mathbf{v}_{B,0} := \mathbf{v}_B(0)$ . The VRP support points are  $\Delta z_{vvp}$  above the eCMP support points.

In this work, we choose the temporal interpolation function  $f(t)$  to be a *polynomial* that is expressed as

$$f(t) = \mathbf{t}_{o_f}^T \mathbf{p}_f. \quad (22)$$

The vector  $\mathbf{t}_{o_f}^T = [1, t, t^2, \dots, t^{o_f}]$  and  $o_f$  denotes the polynomial order of  $f(t)$ . The polynomial parameter vector  $\mathbf{p}_f$  is designed to satisfy given continuity requirements of the VRP reference trajectories, which translate to boundary conditions for  $f(t)$  (see table I and Fig. 4). Now, combining (19), (20),

<sup>7</sup>Note: from here on, our formulation is purely based on the VRP  $\mathbf{v}$ .

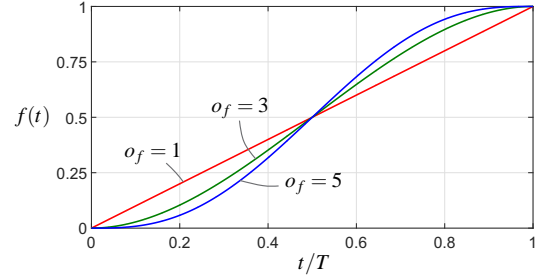


Fig. 4. Illustration of particular functions  $f(t)$  from table I

(21), and (22) in a single equation we find

$$\begin{aligned} \mathbf{v}^T(t) &= \mathbf{t}_{o_f}^T(t) \left( \left( \begin{bmatrix} 1 \\ \mathbf{0}_{o_f \times 1} \end{bmatrix} - \mathbf{p}_f \right) \mathbf{v}_A^T(t) + \mathbf{p}_f \mathbf{v}_B^T(t) \right) \quad (23) \\ &= \mathbf{t}_{o_f+1}^T(t) \left( \underbrace{\left( \begin{bmatrix} 1 \\ \mathbf{0}_{(o_f+1) \times 1} \end{bmatrix} - \begin{bmatrix} \mathbf{p}_f \\ 0 \end{bmatrix} \right)}_{\mathbf{P}_{v_{A,0}}} \mathbf{v}_{A,0}^T + \underbrace{\begin{bmatrix} \mathbf{p}_f \\ 0 \end{bmatrix}}_{\mathbf{P}_{v_{B,0}}} \mathbf{v}_{B,0}^T \right. \\ &\quad \left. + \underbrace{\left( \begin{bmatrix} 0 \\ 1 \\ \mathbf{0}_{o_f \times 1} \end{bmatrix} - \begin{bmatrix} 0 \\ \mathbf{p}_f \end{bmatrix} \right)}_{\mathbf{P}_{\dot{v}_A}} \dot{\mathbf{v}}_A^T + \underbrace{\begin{bmatrix} 0 \\ \mathbf{p}_f \end{bmatrix}}_{\mathbf{P}_{\dot{v}_B}} \dot{\mathbf{v}}_B^T \right). \end{aligned}$$

Note: In the second row of (23), we use a time vector  $\mathbf{t}_{o_f+1}^T(t)$  that is augmented by one additional time exponential with respect to  $\mathbf{t}_{o_f}^T(t)$  from the first row. In the remainder of the paper, we will denote  $\mathbf{t}_{o_f+1}^T(t)$  simply by  $\mathbf{t}^T(t)$  for brevity, i.e.,  $\mathbf{t}^T(t) = [1, t, t^2, \dots, t^{o_f+1}]$ . Thus, we rewrite (23) as

$$\mathbf{v}^T(t) = \mathbf{t}^T(t) \left( \underbrace{\mathbf{p}_{v_{A,0}} \mathbf{v}_{A,0}^T + \mathbf{p}_{v_{B,0}} \mathbf{v}_{B,0}^T + \mathbf{p}_{\dot{v}_A} \dot{\mathbf{v}}_A^T + \mathbf{p}_{\dot{v}_B} \dot{\mathbf{v}}_B^T}_{\mathbf{P}_v} \right), \quad (24)$$

which returns the three-dimensional reference VRP<sup>8</sup> as a function of time and  $\mathbf{P}_v$ .

Having a close look at Fig. 3, we observe that, as compared to formulation (21), a reformulation of the leading VRP support point  $\mathbf{v}_B(t)$  via its terminal point  $\mathbf{v}_{B,T} = \mathbf{v}_B(T)$  is advantageous because  $\mathbf{v}_B(T)$  equals  $\mathbf{v}(T)$ . In Sec. IV, we exploit this equality for a compact formulation via VRP waypoints. Therefore, we evaluate (21) for  $t = T$  which yields  $\mathbf{v}_{B,0} = \mathbf{v}_{B,T} - T \dot{\mathbf{v}}_B$ . Inserting  $\mathbf{v}_{B,0}$  back into (24) yields

$$\mathbf{P}_v = \mathbf{p}_{v_{A,0}} \mathbf{v}_{A,0}^T + \mathbf{p}_{v_{B,0}} \mathbf{v}_{B,T}^T + \mathbf{p}_{\dot{v}_A} \dot{\mathbf{v}}_A^T + (\mathbf{p}_{\dot{v}_B} - T \mathbf{p}_{v_{B,0}}) \dot{\mathbf{v}}_B^T. \quad (25)$$

The final usage of this reformulation is exemplified in Fig. 5 (in the foot trace of contact 4) for transition phase  $\varphi = 7$ , where  $\mathbf{v}_{A,0,7} = \mathbf{v}_{0,7} = \mathbf{v}_{wp,7}$  and  $\mathbf{v}_{B,T,7} = \mathbf{v}(T) = \mathbf{v}_{wp,8}$ .

<sup>8</sup>Note: in contrast to cases with time-varying VRP support points  $\mathbf{v}_A(t)$  and  $\mathbf{v}_B(t)$  (as treated in this paper, see Fig. 7), for stationary support points  $\mathbf{v}_A, \mathbf{v}_B = \text{const}$ , the interpolated VRP  $\mathbf{v}(t)$  moves on a spatial *line* spanned by these two support points (see [14]).

#### D. Specific single transition phase trajectories for walking on moving grounds

For a specific transition phase  $\varphi$ , we can evaluate (25) for  $\mathbf{v}_{A,0} = \mathbf{v}_{0,\varphi}$ ,  $\mathbf{v}_{B,T} = \mathbf{v}_{T,\varphi}$ ,  $\dot{\mathbf{v}}_A = \dot{\mathbf{v}}_{0,\varphi}$  and  $\dot{\mathbf{v}}_B = \dot{\mathbf{v}}_{T,\varphi}$ :

$$\begin{aligned} \mathbf{P}_{v,\varphi} &= \mathbf{p}_{v_{A,0}} \mathbf{v}_{0,\varphi}^T + \mathbf{p}_{v_{B,0}} \mathbf{v}_{T,\varphi}^T \\ &+ \mathbf{p}_{\dot{v}_A} \dot{\mathbf{v}}_{0,\varphi}^T + (\mathbf{p}_{\dot{v}_B} - T \mathbf{p}_{v_{B,0}}) \dot{\mathbf{v}}_{T,\varphi}^T. \end{aligned} \quad (26)$$

Here,  $\mathbf{v}_{0,\varphi}$ ,  $\mathbf{v}_{T,\varphi}$ ,  $\dot{\mathbf{v}}_{0,\varphi}$  and  $\dot{\mathbf{v}}_{T,\varphi}$  denote the initial and final VRP positions and velocities, respectively. For a given local time in transition phase  $t_\varphi \in [0, T_\varphi]$ , the VRP reference trajectory from (6) can be written as

$$\mathbf{v}_\varphi^T(t) = \mathbf{t}^T(t_\varphi) \mathbf{P}_{v,\varphi}. \quad (27)$$

Similarly, the DCM solution (10) becomes

$$\begin{aligned} \xi_\varphi^T(t) &= \gamma_{\xi,\varphi}(t) \xi_{T,\varphi}^T + \mathbf{c}_{P_v \rightarrow \xi(t),\varphi}^T \mathbf{P}_{v,\varphi} \\ &= \gamma_{\xi,\varphi}(t) \xi_{T,\varphi}^T + \underbrace{\mathbf{c}_{P_v \rightarrow \xi(t),\varphi}^T \mathbf{p}_{v_{A,0}}}_{\alpha_{\xi,\varphi}(t)} \mathbf{v}_{0,\varphi}^T + \underbrace{\mathbf{c}_{P_v \rightarrow \xi(t),\varphi}^T \mathbf{p}_{v_{B,0}}}_{\beta_{\xi,\varphi}(t)} \mathbf{v}_{T,\varphi}^T \\ &+ \underbrace{\mathbf{c}_{P_v \rightarrow \xi(t),\varphi}^T \mathbf{p}_{\dot{v}_A}}_{\bar{\alpha}_{\xi,\varphi}(t)} \dot{\mathbf{v}}_{0,\varphi}^T + \underbrace{\mathbf{c}_{P_v \rightarrow \xi(t),\varphi}^T (\mathbf{p}_{\dot{v}_B} - T \mathbf{p}_{v_{B,0}})}_{\bar{\beta}_{\xi,\varphi}(t)} \dot{\mathbf{v}}_{T,\varphi}^T. \end{aligned} \quad (28)$$

and the CoM solution (14) turns into

$$\begin{aligned} \mathbf{x}_\varphi^T(t) &= \gamma_{x,\varphi}(t) \xi_{T,\varphi}^T + \delta_{x,\varphi}(t) \mathbf{x}_{0,\varphi}^T + \mathbf{c}_{P_x \rightarrow x(t),\varphi}^T \mathbf{P}_{v,\varphi} \\ &= \gamma_{x,\varphi}(t) \xi_{T,\varphi}^T + \delta_{x,\varphi}(t) \mathbf{x}_{0,\varphi}^T \\ &+ \underbrace{\mathbf{c}_{P_x \rightarrow x(t),\varphi}^T \mathbf{p}_{v_{A,0}}}_{\alpha_{x,\varphi}(t)} \mathbf{v}_{0,\varphi}^T + \underbrace{\mathbf{c}_{P_x \rightarrow x(t),\varphi}^T \mathbf{p}_{v_{B,0}}}_{\beta_{x,\varphi}(t)} \mathbf{v}_{T,\varphi}^T \\ &+ \underbrace{\mathbf{c}_{P_x \rightarrow x(t),\varphi}^T \mathbf{p}_{\dot{v}_A}}_{\bar{\alpha}_{x,\varphi}(t)} \dot{\mathbf{v}}_{0,\varphi}^T + \underbrace{\mathbf{c}_{P_x \rightarrow x(t),\varphi}^T (\mathbf{p}_{\dot{v}_B} - T \mathbf{p}_{v_{B,0}})}_{\bar{\beta}_{x,\varphi}(t)} \dot{\mathbf{v}}_{T,\varphi}^T. \end{aligned} \quad (29)$$

In the next section, we will use the specific single transition phase trajectories (27)-(29) to derive consistent multi-phase reference trajectories for VRP, DCM and CoM.

#### IV. COMPUTATION OF MULTI-PHASE TRAJECTORIES

The derivations provided in this section are rather compact. A more detailed derivation (for the non-moving ground case) is provided in [14]. The main idea followed here is to concatenate the presented single-phase trajectories to obtain closed-form multi-phase trajectories in matrix-vector form. As mentioned above, we interpolate between  $n_{wp}$  VRP, DCM and CoM waypoints, respectively. The complete trajectories are then composed of  $n_\varphi = n_{wp} - 1$  single transition phases. For a compact formulation, we collect all waypoints in matrices, i.e., all VRP waypoints in  $\mathbf{V}_{wp} = [\mathbf{v}_{wp,1}, \dots, \mathbf{v}_{wp,n_{wp}}]^T$ , all DCM waypoints in  $\Xi_{wp} = [\xi_{wp,1}, \dots, \xi_{wp,n_{wp}}]^T$  and all CoM waypoints in  $\mathbf{X}_{wp} = [\mathbf{x}_{wp,1}, \dots, \mathbf{x}_{wp,n_{wp}}]^T$ , respectively. Note: in [14] we introduced selection matrices  $\mathbf{S}_0$  and  $\mathbf{S}_T$  such that the collection of all initial VRPs  $\mathbf{V}_0$  and final VRPs  $\mathbf{V}_T$  of all  $n_\varphi$  previewed transition phases are obtained from the VRP waypoints via  $\mathbf{V}_0 = \mathbf{S}_0 \mathbf{V}_{wp}$  and  $\mathbf{V}_T = \mathbf{S}_T \mathbf{V}_{wp}$ , respectively. The same matrices are also used in this work to select the corresponding elements of  $\dot{\mathbf{V}}_{wp}$ ,  $\Xi_{wp}$  and  $\mathbf{X}_{wp}$ .

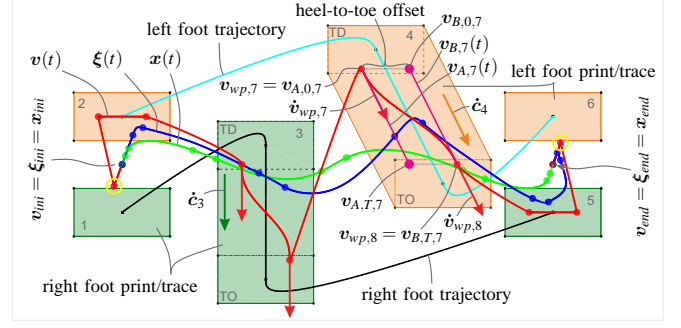


Fig. 5. Top view of VRP  $\mathbf{v}(t)$ , DCM  $\xi(t)$ , CoM  $\mathbf{x}(t)$  and foot trajectories, plus foot positions / traces (left: orange, right: green). Only contacts 3 and 4 have a non-zero velocity ( $\dot{\mathbf{c}}_3 = [0, -0.4, 0]^T \frac{m}{s}$ ,  $\dot{\mathbf{c}}_4 = [0.2, -0.4, 0]^T \frac{m}{s}$ ).

To obtain  $\Xi_{wp}$ , we first evaluate (28) for  $t_\varphi = 0$ :

$$\begin{aligned} \xi_{0,\varphi}^T &= \alpha_{\xi,0,\varphi} \mathbf{v}_{0,\varphi}^T + \beta_{\xi,0,\varphi} \mathbf{v}_{T,\varphi}^T \\ &+ \bar{\alpha}_{\xi,0,\varphi} \dot{\mathbf{v}}_{0,\varphi}^T + \bar{\beta}_{\xi,0,\varphi} \dot{\mathbf{v}}_{T,\varphi}^T + \gamma_{\xi,0,\varphi} \xi_{T,\varphi}^T. \end{aligned} \quad (30)$$

Here  $\alpha_{\xi,0,\varphi} := \alpha_{\xi,\varphi}(0)$ ,  $\beta_{\xi,0,\varphi} := \beta_{\xi,\varphi}(0)$ ,  $\bar{\alpha}_{\xi,0,\varphi} := \bar{\alpha}_{\xi,\varphi}(0)$ ,  $\bar{\beta}_{\xi,0,\varphi} := \bar{\beta}_{\xi,\varphi}(0)$  and  $\gamma_{\xi,0,\varphi} := \gamma_{\xi,\varphi}(0)$ , respectively. These single-phase parameters are collected in diagonal matrices  $\mathbf{A}_\xi = \text{diag}(\alpha_{\xi,0,1}, \dots, \alpha_{\xi,0,n_\varphi})$ ,  $\mathbf{B}_\xi = \text{diag}(\beta_{\xi,0,1}, \dots, \beta_{\xi,0,n_\varphi})$ ,  $\bar{\mathbf{A}}_\xi = \text{diag}(\bar{\alpha}_{\xi,0,1}, \dots, \bar{\alpha}_{\xi,0,n_\varphi})$ ,  $\bar{\mathbf{B}}_\xi = \text{diag}(\bar{\beta}_{\xi,0,1}, \dots, \bar{\beta}_{\xi,0,n_\varphi})$  and  $\mathbf{\Gamma}_\xi = \text{diag}(\gamma_{\xi,0,1}, \dots, \gamma_{\xi,0,n_\varphi})$ , that allow us to write (31) in matrix form (for all  $n_\varphi$  transition phases) as

$$\begin{aligned} \Xi_0 &= \mathbf{A}_\xi \mathbf{V}_0 + \mathbf{B}_\xi \mathbf{V}_T \\ &+ \bar{\mathbf{A}}_\xi \dot{\mathbf{V}}_0 + \bar{\mathbf{B}}_\xi \dot{\mathbf{V}}_T + \mathbf{\Gamma}_\xi \Xi_T. \end{aligned} \quad (31)$$

Using the selection matrices  $\mathbf{S}_0$  and  $\mathbf{S}_T$ , and introducing a DCM terminal constraint  $\xi_f$  (see [14] for details), the DCM waypoint matrix is found to be

$$\Xi_{wp} = \begin{bmatrix} \Xi \mathbf{C}_V & \Xi \bar{\mathbf{C}}_V & \Xi \mathbf{c}_\xi \end{bmatrix} \begin{bmatrix} \mathbf{V}_{wp} \\ \dot{\mathbf{V}}_{wp} \\ \xi_f^T \end{bmatrix}, \quad (32)$$

where

$$\begin{aligned} \Xi \mathbf{C}_V &= \Xi \mathbf{H}^{-1} \mathbf{S}_0^T (\mathbf{A}_\xi \mathbf{S}_0 + \mathbf{B}_\xi \mathbf{S}_T) \\ \Xi \bar{\mathbf{C}}_V &= \Xi \mathbf{H}^{-1} \mathbf{S}_0^T (\bar{\mathbf{A}}_\xi \mathbf{S}_0 + \bar{\mathbf{B}}_\xi \mathbf{S}_T) \\ \Xi \mathbf{c}_\xi &= \Xi \mathbf{H}^{-1} [\mathbf{0}_{1 \times n_\varphi} \quad \mathbf{1}]^T \\ \Xi \mathbf{H} &= \mathbf{I} - \mathbf{S}_0^T \mathbf{\Gamma}_\xi \mathbf{S}_T. \end{aligned}$$

Now, to obtain  $\mathbf{X}_{wp}$ , we evaluate (29) for  $t_\varphi = T_\varphi$ :

$$\begin{aligned} \mathbf{x}_{T,\varphi}^T &= \alpha_{x,T,\varphi} \mathbf{v}_{0,\varphi}^T + \beta_{x,T,\varphi} \mathbf{v}_{T,\varphi}^T \\ &+ \bar{\alpha}_{x,T,\varphi} \dot{\mathbf{v}}_{0,\varphi}^T + \bar{\beta}_{x,T,\varphi} \dot{\mathbf{v}}_{T,\varphi}^T + \gamma_{x,T,\varphi} \xi_{T,\varphi}^T + \delta_{x,T,\varphi} \mathbf{x}_{0,\varphi}^T, \end{aligned} \quad (33)$$

where  $\alpha_{x,T,\varphi} := \alpha_{x,\varphi}(T)$ ,  $\beta_{x,T,\varphi} := \beta_{x,\varphi}(T)$ ,  $\bar{\alpha}_{x,T,\varphi} := \bar{\alpha}_{x,\varphi}(T)$ ,  $\bar{\beta}_{x,T,\varphi} := \bar{\beta}_{x,\varphi}(T)$ ,  $\gamma_{x,T,\varphi} := \gamma_{x,\varphi}(T)$  and  $\delta_{x,T,\varphi} := \delta_{x,\varphi}(T)$ , respectively. In the same way as for the DCM waypoints, these parameters are collected in diagonal matrices  $\mathbf{A}_x$ ,  $\mathbf{B}_x$ ,  $\bar{\mathbf{A}}_x$ ,  $\bar{\mathbf{B}}_x$ ,  $\mathbf{\Gamma}_x$  and  $\mathbf{\Delta}_x$ . These allow us to write (33) in matrix form:

$$\begin{aligned} \mathbf{X}_T &= \mathbf{A}_x \mathbf{V}_0 + \mathbf{B}_x \mathbf{V}_T \\ &+ \bar{\mathbf{A}}_x \dot{\mathbf{V}}_0 + \bar{\mathbf{B}}_x \dot{\mathbf{V}}_T + \mathbf{\Gamma}_x \Xi_T + \mathbf{\Delta}_x \mathbf{X}_0. \end{aligned} \quad (34)$$

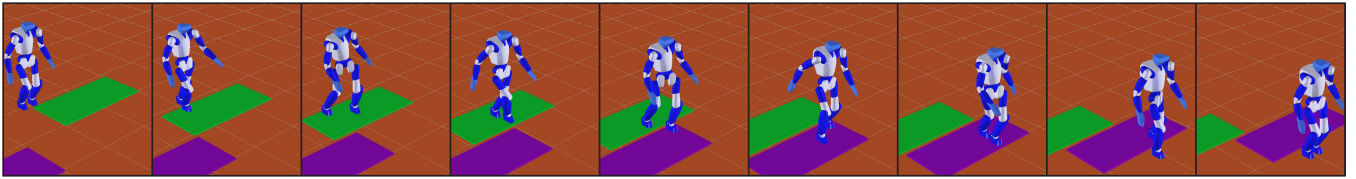


Fig. 6. Toro traversing two moving plates in OpenHRP [20].

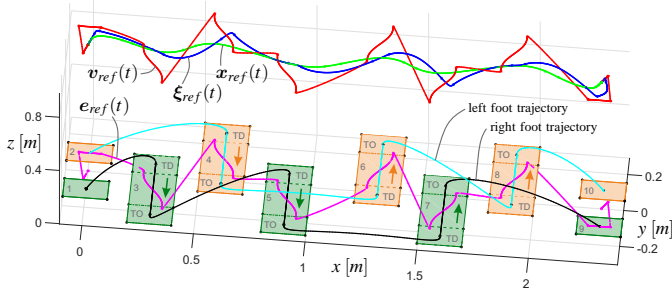


Fig. 7. Three-dimensional eCMP  $e_{ref}(t)$ , VRP  $v_{ref}(t)$ , DCM  $\xi_{ref}(t)$ , CoM  $x_{ref}(t)$  and foot reference trajectories, plus foot positions/traces (left: orange, right: green) for traversing two moving plates. Contact velocities:  $\dot{c}_{\{1,2,9,10\}} = [0, 0, 0]^T \frac{m}{s}$ ,  $\dot{c}_{\{3,4,5\}} = [0, -0.4, 0]^T \frac{m}{s}$ ,  $\dot{c}_{\{6,7,8\}} = [0, 0.4, 0]^T \frac{m}{s}$ .

Again, using the selection matrices  $S_0$  and  $S_T$ , and introducing a CoM initial constraint  $x_s$  (see [14] for details), the CoM waypoint matrix is found to be

$$X_{wp} = \begin{bmatrix} {}^x C_V & {}^x \bar{C}_V & {}^x c_\xi & {}^x c_x \\ \dot{V}_{wp} \\ \xi_f^T \\ x_s^T \end{bmatrix} \quad (35)$$

where

$$\begin{aligned} {}^x C_V &= {}^x H^{-1} S_0^T (A_x S_0 + B_x S_T) \\ {}^x \bar{C}_V &= {}^x H^{-1} S_0^T (\bar{A}_x S_0 + \bar{B}_x S_T) \\ {}^x c_\xi &= {}^x H^{-1} [0_{1 \times n_\varphi} \ 1]^T \\ {}^x c_x &= {}^x H^{-1} [1 \ 0_{1 \times n_\varphi}]^T \\ {}^x H &= I - S_0^T I_X S_T \end{aligned}$$

Equations (32) and (35) provide a tool to compactly compute all DCM waypoints  $\Xi_{wp}$  and CoM waypoints  $X_{wp}$ . Finally, to obtain continuous VRP, DCM and CoM reference trajectories, equations (27), (28) and (29) are evaluated<sup>9</sup>.

## V. IMPLEMENTATION DETAILS

### A. Foot touch-down and take-off locations

In contrast to footprints on stationary ground, on moving ground the feet follow foot-traces (see figures 5 and 7). To account for this support foot motion, we compute appropriate touch-down (TD) and take-off (TO) locations and modify our polynomial-based foot trajectory planner accordingly.

<sup>9</sup>either for multiple transition phases and evaluation times for plotting complete trajectories as shown in figures 5 and 7, or only once for the local time  $t_\varphi$  in the current transition phase  $\varphi$  for the purpose of controls.

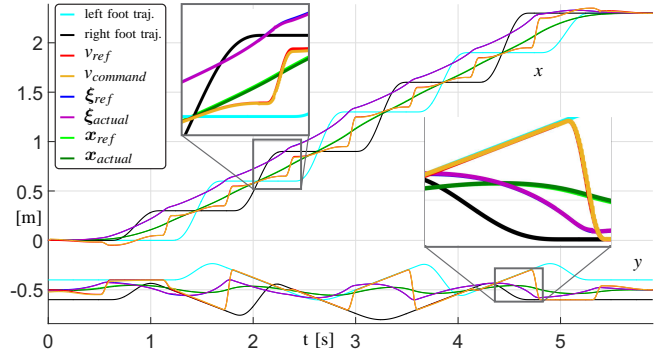


Fig. 8. Trajectory tracking for walking over moving platforms (only  $x$  and  $y$  directions plotted,  $y$  trajectories shifted by  $-0.5m$  to avoid occlusions).

### B. Design of VRP waypoint positions and velocities

As exemplified in Fig. 5, we use *two VRP waypoints per contact*, which is the minimum number required to implement alternating single support (and thus foot-swing) and double support phases. The VRP waypoints are placed appropriately in the local contact (i.e., foot) frame and may implement a heel to toe offset. The VRP waypoint velocities are equal to the corresponding contact velocities. The second and next to last VRP waypoints (marked by yellow circles in Fig. 5) are not predefined. Instead they are computed (see [18] for details) such that i) the first DCM waypoint coincides with the DCM position that corresponds to a steady stance DCM (start configuration) and ii) the final CoM waypoint coincides with the final VRP waypoint, such that finite-time CoM convergence is achieved. Note: if the walking gait starts or ends on a moving ground surface, the DCM initial constraint as defined here and the DCM terminal constraint from (32) need to comply with the *moving ground boundedness condition* derived in appendix II.

## VI. SIMULATIONS

The proposed algorithm for moving ground walking gait generation was tested in multiple OpenHRP [20] simulations. These simulations include traversing two moving plates (while starting and stopping on stationary ground), starting and stopping on two plates that are moving in different directions, "split-belt like walking" (here: left leg walking on stationary ground, right on moving plate) and walking on moving three-dimensional stepping stones through 3D space. While the complementary video shows all these walking simulations, in the paper we focus on presenting the "traversing two moving plates" simulation (see figures 1 and 6). Toro successfully traverses the two moving plates

(moving at  $\pm 0.4 \frac{m}{s}$  in the y-direction) using a dynamic walking gait (single support time: 0.5s, double support time: 0.1s). Figure 7 shows the corresponding three-dimensional reference trajectories, the foot trajectories and the foot contact traces including touch-down (TD) and take-off (TO) positions. While the explicit CoM reference trajectory is provided by our gait generator, in the presented simulations we use our purely DCM-based walking controller from [16] (i.e., the CoM trajectories can be seen as complementary information here). Figure 8 shows the corresponding tracking performance. Due to the precise tracking (average DCM tracking error: 1.9mm) it has to be mentioned explicitly, that the figure doesn't only show the reference trajectories, but *also the actual ones*. Even in the zoomed-in windows, the tracking errors are hard to discover, the reference signals being overlapped by the actual signals. This provides strong evidence of the high quality of our walking gait generator.

## VII. CONCLUSION AND FUTURE WORK

In this paper, we presented a novel walking gait generator that allows the traversal of moving support surfaces (such as moving plates, conveyor belts, moving stairs etc.), including the case of starting and/or stopping on moving ground (see video attachment). Our method facilitates a multi-step preview (e.g. all steps of the complete gait sequence) via computationally efficient matrix-vector based computations. The excellent tracking performance of our walking gait generator is based on three core features: i) the foot trajectories are designed to match the moving support surface motions, ii) the eCMP/VRP trajectories are designed in such a way that the respective support points are guaranteed to stay within each respective "single foot support polygon" and iii) all trajectories (eCMP, VRP, DCM, CoM) are continuous and obey appropriate initial and terminal constraints.

For our future work, we intend to perform experiments of walking on (or traversing) moving surfaces with our full-size humanoid robot Toro [19]. Also, we intend to generalize our overall DCM-based walking gait generation and control framework to support arbitrary walking gaits and robot configurations (e.g. bipedal, quadrupedal, hexapedal etc.).

## ACKNOWLEDGEMENTS

This work was partially supported by the European Commission (H2020-ICT-645097 COMANOID).

## APPENDIX I

### SOLVING SPECIFIC ORDINARY DIFFERENTIAL EQUATIONS (ODE)

#### A. Solution for ODE with polynomial input

Consider an ordinary differential equation (ODE)

$$\dot{y}(t) = \frac{1}{a} y(t) - \frac{1}{a} u(t) , \quad (36)$$

where  $a$  is the time constant of the system and the input function  $u(t)$  is a temporal polynomial, i.e.,

$$u(t) = \mathbf{t}^T(t) \mathbf{p}_u . \quad (37)$$

Here,  $\mathbf{t}^T(t) = [1, t, t^2, \dots, t^{o_u}]$ ,  $\mathbf{p}_u$  is the polynomial parameter vector and  $o_u$  is the polynomial order of  $u(t)$ . The general solution formula for this kind of ODE is

$$y(t) = \underbrace{e^{\frac{t}{a}} y_0}_{y_{hom}(t)} - \underbrace{\frac{e^{\frac{t}{a}}}{a} \int_{\tau=0}^t \left( e^{-\frac{\tau}{a}} u(\tau) \right) d\tau}_{y_{inhom}(u(t))} . \quad (38)$$

In [18] we used partial integration of the integral term in the inhomogeneous solution  $y_{inhom}$  to obtain

$$y(t) = e^{\frac{t}{a}} y_0 + \mathbf{u}_\Sigma(t) - e^{\frac{t}{a}} \mathbf{u}_\Sigma(0) , \quad (39)$$

$$\text{with } \mathbf{u}_\Sigma(t) = \sum_{j=0}^{o_u} \left( a^j \overset{(j)}{u}(t) \right) = \underbrace{\sum_{j=0}^{o_u} \left( a^j \mathbf{t}^T(t) \right)}_{\mathbf{d}(t)} \mathbf{p}_u .$$

Here,  $\overset{(j)}{u}(t)$  and  $\mathbf{t}^T(t)$  denote the  $j$ -th derivatives of  $u(t)$  and  $\mathbf{t}^T(t)$ , respectively. The sum term  $\mathbf{d}(t)$  can be rewritten as

$$\mathbf{d}(t) = \mathbf{a}^T \mathbf{D}(t) , \quad (40)$$

using the constant vector  $\mathbf{a}^T = [1, a, a^2, \dots, a^{o_u}]$  and time matrix  $\mathbf{D}(t) = \left[ \mathbf{t}(t), \overset{(1)}{\mathbf{t}}(t), \overset{(2)}{\mathbf{t}}(t), \dots, \overset{(o_u)}{\mathbf{t}}(t) \right]^T \in \mathbb{R}^{(o_u+1) \times (o_u+1)}$ . By evaluating the product  $\mathbf{a}^T \mathbf{D}(t)$  and grouping<sup>10</sup> the terms by the time exponentials  $t^i$ ,  $\mathbf{d}(t)$  can be rewritten as

$$\mathbf{d}(t) = \mathbf{t}^T(t) \mathbf{C}(a) . \quad (41)$$

The coefficient matrix  $\mathbf{C}(a)$  is solely dependent on parameter  $a$ . Here, we provide its explicit expression for  $o_u = 6$ :

$$C_a|_{o_u=6} = \begin{bmatrix} 1 & a & 2a^2 & 6a^3 & 24a^4 & 120a^5 & 720a^6 \\ 0 & 1 & 2a & 6a^2 & 24a^3 & 120a^4 & 720a^5 \\ 0 & 0 & 1 & 3a & 12a^2 & 60a^3 & 360a^4 \\ 0 & 0 & 0 & 1 & 4a & 20a^2 & 120a^3 \\ 0 & 0 & 0 & 0 & 1 & 5a & 30a^2 \\ 0 & 0 & 0 & 0 & 0 & 1 & 6a \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} . \quad (42)$$

For any  $o_u$ ,  $\mathbf{C}(a)$  can be constructed by selecting the  $(o_u + 1)$  first rows and columns from a corresponding higher order coefficient matrix  $C_a|_{\delta_u > o_u}$ . To reduce computational cost,  $C_a$  can be precomputed for the highest required order  $\delta_u$ .

Now, with (41), we can rewrite (39) as

$$y(t) = e^{\frac{t}{a}} \underbrace{\left( y_0 - \mathbf{t}^T(0) \mathbf{C}(a) \mathbf{p}_u \right)}_{y_e} + \mathbf{t}^T(t) \underbrace{\mathbf{C}(a) \mathbf{p}_u}_{\mathbf{p}_y} . \quad (43)$$

#### B. Solution for ODE with polynomial and exponential inputs

Consider another ODE

$$\dot{\hat{y}}(t) = \frac{1}{\hat{a}} \hat{y}(t) - \frac{1}{\hat{a}} \hat{u}(t) , \quad (44)$$

where

$$\hat{u}(t) = e^{\frac{t}{\hat{b}}} \hat{u}_e + \mathbf{t}^T(t) \mathbf{p}_{\hat{u}} . \quad (45)$$

<sup>10</sup>e.g., by using the "coeffs" command in Matlab

Again, we apply (38) to obtain the solution for (44). Its homogeneous part and the inhomogeneous part corresponding to the polynomial fraction of input function  $\hat{u}(t)$  are equivalent to the solution (43), while the solution for the inhomogeneous part corresponding to the exponential component of  $\hat{u}(t)$  is found to be  $\frac{\beta}{\beta-\hat{a}} (e^{\frac{t}{\beta}} - e^{\frac{t}{\hat{a}}}) \hat{u}_e$ . Combining these findings, the solution of (44) is found as

$$\hat{y}(t) = e^{\frac{t}{\hat{a}}} \hat{y}_0 + \left( t^T(t) - e^{\frac{t}{\hat{a}}} t^T(0) \right) C(\hat{a}) p_{\hat{u}} + \frac{\beta}{\beta-\hat{a}} (e^{\frac{t}{\beta}} - e^{\frac{t}{\hat{a}}}) \hat{u}_e. \quad (46)$$

## APPENDIX II

### MOVING GROUND BOUNDEDNESS CONDITIONS

Here, we consider the boundedness conditions [24] for *stationary standing* on a support surface (e.g. before or after a walking gait) that is moving with a constant velocity  $\dot{c}$ . The corresponding VRP moves with the support surface, i.e.,  $\dot{v}_{stat} = \dot{c}$ . For the DCM to VRP distance to remain bounded, their respective velocities must be equal, i.e.,

$$\dot{\xi}_{stat} = \dot{v}_{stat}. \quad (47)$$

Inserting  $\dot{\xi}$  from (3) and solving for  $\xi$  yields the corresponding boundedness constraint:

$$\xi = v + b \dot{v}_{stat}. \quad (48)$$

Note: previous works [9], [18], [25] had focused on walking on stationary grounds, i.e.,  $\dot{v}_{stat} = 0$ , and thus the condition for boundedness had been  $\xi = v$ .

For a full understanding of the stationary case, we now insert  $\xi$  from (48) into (1) to find that

$$\underbrace{\dot{x}_{stat} - \dot{v}_{stat}}_{\Delta_{x,v}} = -\frac{1}{b} \underbrace{(x - v)}_{\Delta_{x,v}}, \quad (49)$$

i.e., the CoM converges towards the VRP.

## REFERENCES

- [1] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, "The 3d linear inverted pendulum mode: A simple modeling for a biped walking pattern generation," in *IEEE Int. Conf. on Robotics and Automation*, 2001, pp. 239–246.
- [2] 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. on Robotics and Automation*, 2003, pp. 1620–1626.
- [3] T. Sugihara, "Standing stabilizability and stepping maneuver in planar bipedalism based on the best com-zmp regulator," in *IEEE Int. Conf. on Robotics and Automation*, 2009, pp. 1966–1971.
- [4] A. Herdt, N. Perrin, and P.-B. Wieber, "Walking without thinking about it," in *Int. Conf. on Intell. Robots and Systems*, 2010, pp. 190–195.
- [5] J. Urata, K. Nishiwaki, Y. Nakanishi, K. Okada, S. Kagami, and M. Inaba, "Online decision of foot placement using singular LQ preview regulation," in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2011.

- [6] R. Tedrake, S. Kuindersma, R. Deits, and K. Miura, "A closed-form solution for real-time zmp gait generation and feedback stabilization," in *Humanoid Robots (Humanoids)*, 2015 *IEEE-RAS 15th International Conference on*, 2015, pp. 936–940.
- [7] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture point: A step toward humanoid push recovery," in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2006, pp. 200–207.
- [8] A. L. Hof, "The 'extrapolated center of mass' concept suggests a simple control of balance in walking," *Human Movement Science*, vol. 27, pp. 112–125, 2008.
- [9] T. Takenaka, T. Matsumoto, and T. Yoshiike, "Real time motion generation and control for biped robot -1st report: Walking gait pattern generation-," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 1084–1091.
- [10] J. Engelsberger, C. Ott, M. A. Roa, A. Albu-Schäffer, and G. Hirzinger, "Bipedal walking control based on capture point dynamics," in *Int. Conf. on Intell. Robots and Systems*, 2011, pp. 4420–4427.
- [11] 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 Robotics Research*, vol. 31, no. 9, pp. 1094–1113, 2012.
- [12] J. Engelsberger, C. Ott, and A. Albu-Schäffer, "Three-dimensional bipedal walking control using divergent component of motion," in *Int. Conf. on Intell. Robots and Systems*, 2013, pp. 2600–2607.
- [13] M. A. Hopkins, D. W. Hong, and A. Leonessa, "Humanoid locomotion on uneven terrain using the time-varying divergent component of motion," in *2014 IEEE-RAS International Conference on Humanoid Robots*, 2014, pp. 266–272.
- [14] G. Mesesan, J. Engelsberger, C. Ott, and A. Albu-Schäffer, "Convex properties of center-of-mass trajectories for locomotion based on divergent component of motion," *IEEE Robotics and Automation Letters*, 2018.
- [15] V. V. Unhelkar, S. Dorr, A. Bubeck, P. A. Lasota, J. Perez, H. C. Siu, J. C. Boerkoel, Q. Tyroller, J. Bix, S. Bartscher, and J. A. Shah, "Mobile robots for moving-floor assembly lines: Design, evaluation, and deployment," *IEEE Robotics Automation Magazine*, vol. 25, no. 2, pp. 72–81, 2018.
- [16] J. Engelsberger, C. Ott, and A. Albu-Schäffer, "Three-dimensional bipedal walking control based on divergent component of motion," *Robotics, IEEE Transactions on*, vol. 31, no. 2, pp. 355–368, 2015.
- [17] J. Engelsberger, "Combining reduced dynamics models and whole-body control for agile humanoid locomotion," Dissertation, Technische Universität München, München, 2016.
- [18] J. Engelsberger, G. Mesesan, and C. Ott, "Smooth trajectory generation and push-recovery based on divergent component of motion," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 4560–4567.
- [19] J. Engelsberger, A. Werner, C. Ott, B. Henze, M. A. Roa, G. Garofalo, R. Burger, A. Beyer, O. Eiberger, K. Schmid, and A. Albu-Schäffer, "Overview of the torque-controlled humanoid robot toro," in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2014, pp. 916–923.
- [20] F. Kanehiro, K. Fujiwara, S. Kajita, K. Yokoi, K. Kaneko, H. Hirukawa, Y. Nakamura, and K. Yamane, "Open architecture humanoid robotics platform," in *IEEE Int. Conf. on Robotics and Automation*, 2002, pp. 24–30 vol.1.
- [21] M. B. Popovic, A. Goswami, and H. Herr, "Ground reference points in legged locomotion: Definitions, biological trajectories and control implications," *Int. J. of Robotics Research*, vol. 24, no. 12, 2005.
- [22] T. Koolen, M. Posa, and R. Tedrake, "Balance control using center of mass height variation: Limitations imposed by unilateral contact," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, 2016, pp. 8–15.
- [23] S. Caron and B. Mallein, "Balance control using both zmp and com height variations: A convex boundedness approach," in *IEEE Int. Conf. on Robotics and Automation*, 2018.
- [24] L. Lanari, S. Hutchinson, and L. Marchionni, "Boundedness issues in planning of locomotion trajectories for biped robots," in *2014 IEEE-RAS Int. Conf. on Humanoid Robots*, 2014, pp. 951–958.
- [25] T. Koolen, J. Smith, G. Thomas, S. Bertrand, J. Carff, N. Mertins, D. Stephen, P. Abeles, J. Engelsberger, S. McCrory, J. van Egmond, M. Griffioen, M. Floyd, S. Kobus, N. Manor, S. Alsheikh, D. Duran, L. Bunch, E. Morphis, L. Colasanto, K.-L. H. Hoang, B. Layton, P. Neuhaus, M. Johnson, and J. Pratt, "Summary of team ihmc's virtual robotics challenge entry," in *Int. Conf. on Humanoid Robots*, 2013, pp. 307–314.