Dynamic Walking on Compliant and Uneven Terrain using DCM and Passivity-based Whole-body Control

Mesesan,_2019_Dynamic_Walking_on_Compliant_and_Uneven_Terrain.pdf

Passivity-based Whole-body Control (PB WBD) using trajectory generated by DCM approach

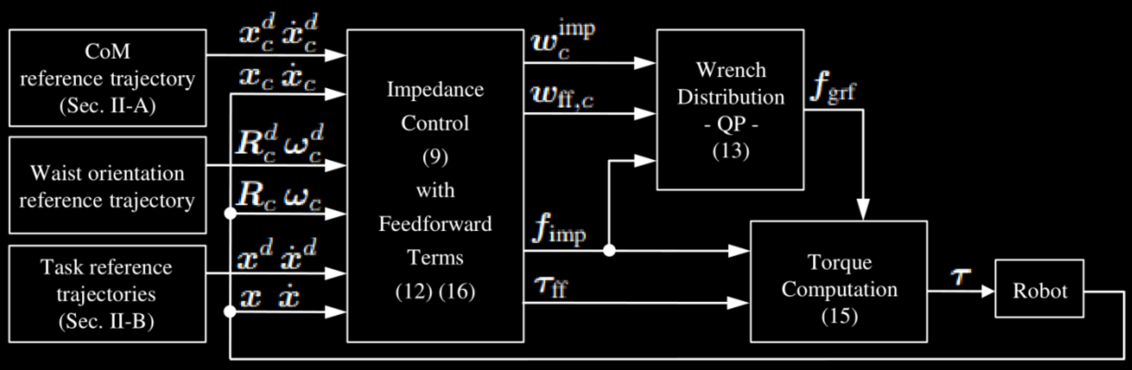

Fig. 2: Overview of the whole-body torque controller

I/ DCM approach with elimination of commanded torques discontinuities:

DCM stands for Divergent Component of Motion defined as:

$$\xi = x_c + \sqrt{\frac{\Delta z}{g}}\dot{x}_c \qquad (1)$$

Virtual Repellent Point (VRP):

$$v = x_c - \frac{\Delta z}{g}\ddot{x}_c$$

I.1/ CoM reference trajectory:

Relation between these 2 points:

$$v = \xi - \sqrt{\frac{\Delta z}{g}}\dot{\xi} \qquad (2)$$

Procedure to generate CoM trj x_c

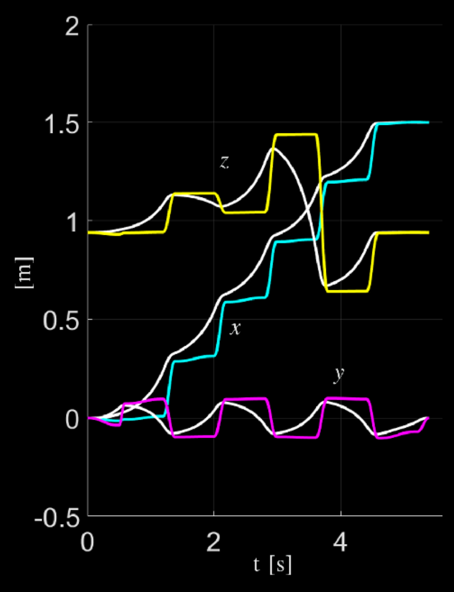

Fig. 1. Planning of smooth and
a) black: DCM, colored: VRP,

1/ VRP way points $v_k \in \mathbb{R}^3$ are placed under the foot center at foot step k, the height of $v_k = \Delta z$

2/ Trajectory of VRP is created by keeping its stationary points under the stance foot in the single support phase, then using linear interpolate for the transition from one foot to the other during the double support phase.

3/ Terminal DCM coincides with final VRP

Compute VRP → DCM → CoM

Detailed in the cited paper [10] and [11]

(We may or may not adopt this scheme or we can use the available CoM trj generated in the given code)

I.2 / Foot trajectory:

I think this one was given in the code

II/ Whole-body torque controller:

Dynamic of humanoid as a floating base manipulator. The components of the state vector are:

- The pose of the CoM: $x_c \in \mathbb{R}^3, R_c \in SO(3)$: (rotation matrix of the waist (torso, etc...))
- Velocity of the CoM: $\dot{x}_c, \omega_c \in \mathbb{R}^3$. Stack them together yields: $\nu_c = [\dot{x}_c, \omega_c]^T \in \mathbb{R}^6$
- Joint angle and joint velocities $q, \dot{q} \in \mathbb{R}^n$

Then the dynamics is:

$$M \begin{bmatrix} \dot{\nu}_c \\ \ddot{q} \end{bmatrix} + C \begin{bmatrix} \nu_c \\ \dot{q} \end{bmatrix} + \begin{bmatrix} -w_g \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \tau \end{bmatrix} + \bar{\tau}_{ext} \tag{3}$$

$w_g = [0\,0\,mg\,0\,0\,0]^T, g = 9.8m/s^2$, $\bar{\tau}_{ext} \in \mathbb{R}^{\bar{n}=n+6}$ is the generlized external forces from the environment, $\tau \in \mathbb{R}^n$ is the joint torque.

—

II.1/ Formulate the controller:

For walking robot context, consider the set of velocity levels of a task $\dot{x}$ as:

- CoM location and waist orientation $\nu_c = [\dot{x}_c, \omega_c]^T \in \mathbb{R}^6$
- Taks that generate contact forces w.r.t contact constraint $\dot{x}_{\text{grf}} \in \mathbb{R}^{n_{\text{grf}}}$. $n_{\text{grf}} = 6$ in the case of single contact, $= 12$ in the case of double contact.
- Remaining tasks to build a square Jacobian denoted as $\dot{x}_{imp} \in \mathbb{R}^{n+6-n_{\text{grf}}}$. This can contain Cartesian tracking tasks for the free end-effectors (swing foot, etc....), joint space subtasks, visual servoing tasks, etc

$$\dot{x} = \begin{bmatrix} \nu_c \\ \dot{x}_{\text{grf}} \\ \dot{x}_{imp} \end{bmatrix} = \begin{bmatrix} I_{6\times 6} & 0_{6\times n} \\ J_{\text{grf},c} & J_{\text{grf},q} \\ J_{imp,c} & J_{imp,q} \end{bmatrix} \begin{bmatrix} \nu_c \\ \dot{q} \end{bmatrix} = J \begin{bmatrix} \nu_c \\ \dot{q} \end{bmatrix}, J \in \mathbb{R}^{\bar{n}\times\bar{n}}$$

Assume J is invertible → DAMN IT.

Given the desired velocity reference task (taken from the given CoM and foot trj generator) $\dot{x}_d$. The corresponding reference CoM and joint velocities is computed as:

$$\begin{bmatrix} \nu_c^d \\ \dot{q}^d \end{bmatrix} = J^{-1}(x_c, q)\dot{x}_d$$

Now, we want to create a torque controller τ that realizes the desired close-loop dynamic formulated as:

$$M\begin{bmatrix}\dot{\nu}_c-\dot{\nu}_c^d\\ \ddot{q}-\ddot{q}_d\end{bmatrix}+C\begin{bmatrix}\nu_c-\nu_c^d\\ \dot{q}-\dot{q}_d\end{bmatrix}=\bar{\tau}_{ext}-J^T\begin{bmatrix}w_c^{imp}\\ f_{\text{grf}}\\ f_{\text{imp}}\end{bmatrix} \tag{4}$$

$$M\begin{bmatrix}\Delta\dot{\nu}_c\\ \Delta\ddot{q}\end{bmatrix}+C\begin{bmatrix}\Delta\nu_c\\ \Delta\dot{q}\end{bmatrix}=\bar{\tau}_{ext}-J^T\begin{bmatrix}w_c^{imp}\\ f_{\text{grf}}\\ f_{\text{imp}}\end{bmatrix} \tag{5}$$

In which the CoM impedance wrench w_c^{imp} is defined as:

$$w_c^{imp}=\begin{bmatrix}K_c(x_c-x_c^d)+D_c(\dot{x}_c-\dot{x}_c^d)\\ \\ \tau_r(\sum_c, R_c^T R_c^d)+B_c(\omega_c-\omega^d)\end{bmatrix}$$

τ_r is something so-called rotational spring. ω_c^{imp} is a mass-spring-damper w.r.t the error

Similarly, f_{imp} can be computed as the same form with the error in Cartesian space or even in the joint space for some defined joint task.

The only unknown term is f_{grf} which can be expressed by considering $(3)-(4)$ and as:

$$M\begin{bmatrix}\dot{\nu}_c^d\\ \ddot{q}_d\end{bmatrix}+C\begin{bmatrix}\nu_c^d\\ \dot{q}_d\end{bmatrix}+\begin{bmatrix}-w_g\\ 0\end{bmatrix}=\begin{bmatrix}0\\ \tau\end{bmatrix}+J^T\begin{bmatrix}w_c^{imp}\\ f_{\text{grf}}\\ f_{\text{imp}}\end{bmatrix} \tag{6}$$

$$\begin{bmatrix}M_1\\ M_2\end{bmatrix}\begin{bmatrix}\dot{\nu}_c^d\\ \ddot{q}_d\end{bmatrix}+\begin{bmatrix}C_1\\ C_2\end{bmatrix}\begin{bmatrix}\nu_c^d\\ \dot{q}_d\end{bmatrix}=\begin{bmatrix}w_g\\ \tau\end{bmatrix}+\begin{bmatrix}I_{6\times6} & J_{\text{grf},c}^T & J_{imp,c}^T\\ \\ 0_{n\times6} & J_{\text{grf},q}^T & J_{imp,q}^T\end{bmatrix}\begin{bmatrix}w_c^{imp}\\ f_{\text{grf}}\\ f_{\text{imp}}\end{bmatrix} \tag{7}$$

The first 6 rows is independent form τ, so we can write the equality:

$$w_{ff,c}=w_g+w_c^{imp}+J_{\text{grf},c}^T f_{\text{grf}}+J_{\text{imp},c}^T f_{\text{imp}}$$

With the “feed forward”-like CoM wrench

$$w_{ff,c}=M_1\begin{bmatrix}\dot{\nu}_c^d\\ \ddot{q}_d\end{bmatrix}+C_1\begin{bmatrix}\nu_c^d\\ \dot{q}_d\end{bmatrix}$$

With this equality, we can define the QP problem to minimize the cost (keep the error from the above equality as small as possible) $(f_{\text{grf}}\in\mathbb{R}^{n_{\text{grf}}})$

$$J=\frac{1}{2}\delta_c^T Q_c\delta_c+\frac{1}{2}\delta_f^T Q_f\delta_f$$

$$\delta_c=J_{\text{grf},c}^T f_{\text{grf}}+J_{\text{imp},c}^T f_{\text{imp}}+w_g+w_c^{imp}-w_{ff,c}$$
$$\delta_f=f_{\text{grf}}-f_{\text{grf}}^d$$

With respect to the constraints of:

- unilaterality constraints (force always push up)
- Friction cone constraints
- bounded normal force
- Bounded torque on z axis
- center of pressure constraint.

After obtaining f_{grf}, τ can be computed from the n last rows of (7) as:

$$\tau_{ff}=\tau+J_{\text{grf},q}^T f_{\text{grf}}+J_{\text{imp},q}^T f_{\text{imp}}$$

$$\tau_{ff}=M_2\begin{bmatrix}\dot{\nu}_c^d\\ \ddot{q}_d\end{bmatrix}+C_2\begin{bmatrix}\nu_c^d\\ \dot{q}_d\end{bmatrix}$$

In the next section I will note down directly the controller formulation based on walking task.

II.1/ Stacking of the task

Walking task can be stacked as CoM and feet velocities as:

$$\begin{bmatrix} \nu_c \\ \nu_r \\ \nu_l \\ \dot{q}_{pose} \end{bmatrix} = \begin{bmatrix} I_{6\times 6} & 0_{6\times n} \\ J_{r,c} & J_{r,q} \\ J_{l,c} & J_{l,q} \\ 0 & S_{pose} \end{bmatrix} \begin{bmatrix} \nu_c \\ q \end{bmatrix} = J \begin{bmatrix} \nu_c \\ q \end{bmatrix}$$

$\nu_r, \nu_l \in \mathbb{R}^6$ are the right and left feet velocities. (I am confusing about the statement of the paper that these terms are Cartesian velocities?? These terms should contain both linear and angular velocity, 6 dim vector)

> In this work, for dynamic walking, we focus on the CoM trajectory and the Cartesian foot trajectories. Let ν_r and ν_l denote the right and left foot Cartesian velocity vectors, respectively, which are defined in an analogous manner to the CoM velocity vector ν_c. The CoM and foot tasks account

Depend on the contact phase, we will have the residual terms in the cost function as

- Double support phase: $\dot{x}_{\text{grf}} = [\nu_r \ \ \nu_l]^T, \dot{x}_{imp} = \dot{q}_{pose}$

$$\delta_c = [J_{r,c}^T \ \ J_{l,c}^T][w_r^T \ \ w_l^T]^T + w_c^{imp} - w_{ff,c}$$
$$\delta_f = [w_r^T \ \ w_l^T]^T - f_{\text{grf}}^d$$

- Single support phase: $\dot{x}_{\text{grf}} = \nu_r, \dot{x}_{imp} = [\nu_l \ \ \dot{q}_{pose}]^T$ for right contact,

  $\dot{x}_{\text{grf}} = \nu_l, \dot{x}_{imp} = [\nu_r \ \ \dot{q}_{pose}]^T$ for left contact

Right contact:

$$\delta_c = J_{r,c}^T w_r + [J_{l,c}^T \ \ 0] f_{imp} + w_g + w_c^{imp} - w_{ff,c}$$
$$\delta_f = w_r - f_{\text{grf}}^d$$

f_{imp} now contains the left wrenchs (defined as impedance model based on the error of the swing foot trajectory) and impedance torque in the joint space

Left contact

$$\delta_c = J_{l,c}^T w_l + [J_{r,c}^T \ \ 0] f_{imp} + w_g + w_c^{imp} - w_{ff,c}$$
$$\delta_f = w_l - f_{\text{grf}}^d$$

f_{imp} now contains the right wrenchs (defined as impedance model based on the error of the swing foot trajectory) and impedance torque in the joint space

The way to compute τ will not change, just the way to define $f_{imp}, J_{\text{imp},q}^T$

> $\dot{x}_{\text{imp}} = (\nu_r^T \ \ \dot{q}_{\text{pose}}^T)^T$, depending on the contact foot. In all cases, however, we can rewrite (15) for dynamic walking as
>
> $$\boxed{\tau = \tau_{\text{ff}} - S_{\text{pose}}^T \tau_{\text{pose}} - [J_{r,q}^T \ \ J_{l,q}^T] \begin{pmatrix} w_r \\ w_l \end{pmatrix},} \quad (23)$$

Noted here in the double support phase, left and right are actually contact wrenches. While in single support with right contact, w_r is a part of f_{imp} and vice versa for the left contact.

II.2/ Continuous foot wrench transition

The bounded value of all the contact force varies from the swing phase → defined transition phase → stance phase. Example in the z axis:

$$f_z^{\text{imp}} = k_z(z - z^d) + d_z(\dot{z} - \dot{z}^d),$$

Finally, during the transition from the swing to the stance phase[5], we use the following contact lower and upper bounds

$$\begin{aligned} \underline{f}_z^a &= \frac{t}{T_a}\underline{f}_z + \left(1 - \frac{t}{T_a}\right) f_z^{\text{imp}} \\ \overline{f}_z^a &= \frac{t}{T_a}\overline{f}_z + \left(1 - \frac{t}{T_a}\right) f_z^{\text{imp}}, \end{aligned} \tag{25}$$

where $t \in [0, T_a]$ is the local time of the transition phase. Since the QP solver is constrained to finding a solution for f_z within the interval $[\underline{f}_z^a, \overline{f}_z^a]$, our approach guarantees that at the beginning of the transition phase $f_z = f_z^{\text{imp}}$, and during the transition phase the constraint interval is gradually widened to reach the default interval $[\underline{f}_z, \overline{f}_z]$ at the end

II.3/ Minor effect on the wrench stabilizer when rigid contact assumption is not valid

as part of the vector $\boldsymbol{f}_{\text{grf}}$. To stabilize the foot, we add a damping wrench $\boldsymbol{w}_r^{\text{stabil}} = -\boldsymbol{D}\boldsymbol{\nu}_r$ to $\boldsymbol{w}_r$ in (23), with $\boldsymbol{D}$