



Lower body control of a semi-autonomous avatar in Virtual Reality: Balance and Locomotion of a 3D Bipedal Model

Vincent Thomasset, Stéphane Caron, Vincent Weistroffer

► To cite this version:

Vincent Thomasset, Stéphane Caron, Vincent Weistroffer. Lower body control of a semi-autonomous avatar in Virtual Reality: Balance and Locomotion of a 3D Bipedal Model. VRST 2019 - 25th ACM Symposium on Virtual Reality Software and Technology, Nov 2019, Sidney, Australia. cea-02385996

HAL Id: cea-02385996

<https://cea.hal.science/cea-02385996>

Submitted on 29 Nov 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Lower body control of a semi-autonomous avatar in Virtual Reality: Balance and Locomotion of a 3D Bipedal Model

Vincent THOMASSET¹, Stéphane CARON², Vincent WEISTROFFER¹

¹ CEA, LIST, Interactive Simulation Laboratory, Palaiseau, France

² CNRS-UM LIRMM, Montpellier, France

³ Department of Mechanical Engineering, McGill University, Montreal, Canada

ABSTRACT Animated virtual humans may rely on full-body tracking system to reproduce user motions. In this paper, we reduce tracking to the upper-body and reconstruct the lower body to follow autonomously its upper counterpart. Doing so reduces the number of sensors required, making the application of virtual humans simpler and cheaper. It also enable deployment in cluttered scenes where the lower body is often hidden. The contribution here is the inversion of the well-known capture problem for bipedal walking. It determines footsteps rather than center-of-mass motions and yet can be solved with an off-the-shelf capture problem solver. The quality of our method is assessed in real-time tracking experiments on a wide variety of movements.

1. INTRODUCTION Virtual Humans (VH), also referred to as avatars in this work, are used in the manufacturing field as a tool for design, maintenance studies, operators training or ergonomics studies [5, 21, 29]. Avatars are often animated by replicating the movements of an operator using optical motion capture systems. This method of animation replicate the user's movements but is subject to the following issues. First, the virtual environment may differ from the user's environment (stairs, uneven floor, objects to lift, obstacles to avoid, ...). This can lead to physical inconsistency when the avatar is an exact mirror of the user. Second, legs and especially feet sensors can be occluded by static objects (tool bench, table...) or other persons moving in the environment. This issue may be tackled by reconstructing the lower-body of the avatar. In this paper, we propose a semi-autonomous pipeline where the upper-body (head, hands and waist) is still tracked with high accuracy but legs are reconstructed by an autonomous balancing controller. The objective is to realize a real time application in Virtual Reality (VR) where an operator evolves in a real complex environment.

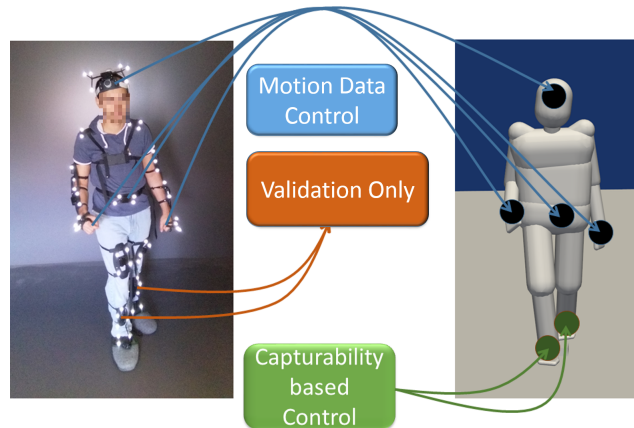


Figure 1: The semi-autonomous avatar follows the user's upper-body trajectory and uses it to reconstruct the lower body trajectory

Our framework represented in Figure 1 features:

- The user, equipped with tracking sensors
- The avatar, representation of the user
- The control algorithm, ensuring avatar motion with user’s upper-body movements as inputs and lower-body positions as outputs.

In this work we focus on Use cases that involve standing and (dynamic) walking. We force upper-body movements to match those measured on the user. The reason for this is that many VR application can be realized with only Headset and Hands tracking [18, 24], which is known to cause nausea sensation if the tracked positions are of low accuracy, low frame rate or latency [28].

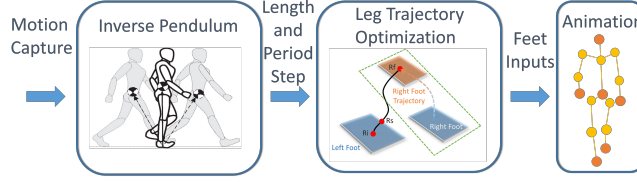


Figure 2: Overall locomotion strategy, solved at every time step

Lower-body motions of the avatar are reconstructed in accordance with the upper body. This reconstruction is based on a capturability-based locomotion control [4] that relies on the Inverted Pendulum Model (Section 2). The animation of the avatar is described in Section 3. The main contribution lies in the optimization of future footsteps (Section 4). The physical consistency of our method is evaluated in Section 5 and compared with the state-of-the-art. Practical uses cases are also detailed.

2. Inverted Pendulum Model

This section recalls background knowledge on the linear and nonlinear inverted pendulum model for locomotion.

The inverted pendulum is a reduced model for dynamic bipedal locomotion. Such models are used in the animation community, however their parameters require offline optimization [1, 2, 15] and are therefore not well suited for real time applications. Reduced models have been applied in balance and locomotion robotics [6, 12, 22] to derive critical properties such as *capturability*. Capturability is the ability of the model to keep balance from a given state and steer towards stable equilibrium configurations [4, 7, 16, 26, 30]. Capturability cannot be computed with full avatar body dynamics in the present state of art, as it would require nonlinear receding-horizon computations running in real time, but it is tractable within reduced models. See *e.g.* [23] for an overview. The most common reduced models for locomotion are the linear [13] (LIPM) and nonlinear inverted pendulum models [25] (IPM), as pictured in Figure 3.

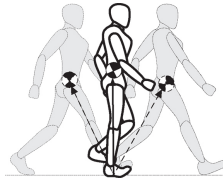


Figure 3: Picture of pendulum and Humanoid resemblance from Ahmed Elhasairi and Alexandre Pechev (from [8])

The LIPM, well suited for horizontal locomotion [9, 10], cannot account for strong dynamical center of mass (CoM) variations in the vertical directions, targeted in the present work (*e.g.* for stair climbing, squatting or obstacle avoidance). While the CoM height of the IPM can change, the angular momentum variations around the CoM are negligible, as in [17, 27].

The equation of motion of the IPM is:

$$\ddot{c}(t) = \lambda(t)(c(t) - r(t)) + g \quad (2.1)$$

where t denotes time, $c(t)$ is the CoM of the avatar, g is the gravity vector, also written $g = -ge_z$ with g the gravitational constant and e_z the upright vertical of the inertial frame. The two control inputs of the system are r and λ . The former, r , is the Center of Pressure (CoP) under the contact surface, called Zero-tilting Moment Point (ZMP) when there are multiple contacts. Intuitively it represents the point where the biped is “putting its weight” and where the contact reaction force is applied, we can use it to measure the stability of the system. The latter, λ , represents the normalized stiffness of the massless leg between CoM and ZMP. To maintain avatar balance, we want our system to remain *capturable* at any time. Capturability quantifies the ability of the avatar to come to a stop at a given location. In the LIPM, it is quantified by the well-known *capture point* [26] pictured in Figure 4.

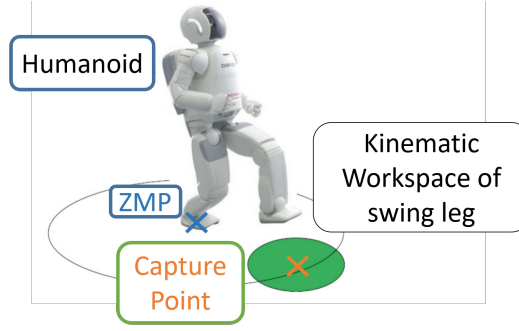


Figure 4: Picture of Capture Area and Capture point of a Humanoid from D. E. Orin, A. Goswami, and S.-H. Lee [23]

The capture point is the point on the ground where the robot should step instantly to absorb its linear momentum and come to a stop. In the classical LIPM where $\lambda(t) = \omega^2$ is maintained at a constant value, it is given by:

$$r_{cp} = c + \frac{\dot{c}}{\omega} \quad (2.2)$$

Moving the ZMP to $r = r_{cp}$, the equation of motion (2.1) yields exponential convergence of the CoM to a static equilibrium above the capture point. Hence the system is “captured” and its linear momentum absorbed towards a resting position.

Footsteps, defined as capture point waypoints, are commonly used in robotic locomotion to generate CoM trajectories. In the more general IPM that allows height variations, capturability is characterized by three properties of its two inputs r and λ : feasibility, asymptotic convergence, and the boundedness condition. Let us recall those properties and cast them into an optimization problem, the *capture problem* [4].

2.1. Feasibility Condition of Contact Stability

In order to make sure contacts do not break, both IPM inputs must satisfy feasibility conditions.

To be feasible, the ZMP must stay at all times inside the contact area under the supporting foot (edges not included). We assume that all contacts are planar and polygonal. In that case, the ZMP r

remains in the plane of contact if and only if $(\mathbf{r} - \mathbf{o}) \cdot \mathbf{n} = 0$, where \mathbf{o} denotes a point of the contact area and \mathbf{n} is the contact normal.

To be feasible, the normalized leg stiffness λ must be non-negative. Furthermore, we impose $\lambda \in [\lambda_{\min}, \lambda_{\max}]$ where $\lambda_{\min} = g/h_{\max}$ and $\lambda_{\max} = g/h_{\min}$ are defined from h_{\min} and h_{\max} , respectively the minimum and maximum height of the CoM.

An input function $t \mapsto (\lambda(t), \mathbf{r}(t))$ is *feasible* when both λ and \mathbf{r} are feasible at all times. The capturability problem is then to find feasible inputs λ and \mathbf{r} such that the output trajectory \mathbf{c} converges to a desired location. We will use this property in Section 4 to generate the desired target location depending on the initial state of the system.

2.2. Static equilibrium Static equilibrium states of the inverted pendulum, also known as *capture states*, are stable positions with zero CoM velocity. Those states are characterized by their CoM position \mathbf{c}_f and their contact area. The only final control input λ_f, \mathbf{r}_f that maintains the system in equilibrium with $\ddot{\mathbf{c}}_f = 0$, is such that $\mathbf{c}_f = \mathbf{r}_f - \frac{g}{\lambda_f}$ with:

$$\lambda_f(\mathbf{c}_f) = \frac{g}{\bar{z}_f} \quad r_f(\mathbf{c}_f) = \mathbf{c}_f - \bar{z}_f \mathbf{e}_z \quad (2.3)$$

where \bar{z}_f is the height of the CoM \mathbf{c}_f above contact:

$$\bar{z}_f = \frac{(\mathbf{c} - \mathbf{o}) \cdot \mathbf{n}}{\mathbf{e}_z \cdot \mathbf{n}} \quad (2.4)$$

Note that \mathbf{o} and \mathbf{n} are relative to the contact area upon which the biped is in static equilibrium.

2.3. Dichotomy of the components of motion A natural choice of the IPM state consists of its CoM position \mathbf{c} and velocity $\dot{\mathbf{c}}$. The IPM equation (2.1) can be written as a first order linear time-variant state function of the form $\dot{\mathbf{x}} = \mathbf{A}(t)\mathbf{x} + \mathbf{b}(t)$:

$$\begin{bmatrix} \dot{\mathbf{c}} \\ \ddot{\mathbf{c}} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \lambda \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{c} \\ \dot{\mathbf{c}} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{g} - \lambda \mathbf{r} \end{bmatrix} \quad (2.5)$$

This state can then be decoupled into its convergent and divergent components. In the case of the IPM, this can be done [11] by defining two positive functions $\omega(t)$ and $\gamma(t)$ and rewriting the state vector:

$$\mathbf{z} = \begin{bmatrix} \zeta \\ \xi \end{bmatrix} = \begin{bmatrix} \gamma \mathbf{c} - \dot{\mathbf{c}} \\ \omega \mathbf{c} + \dot{\mathbf{c}} \end{bmatrix} \quad (2.6)$$

This leads to the new linear time-variant state function:

$$\dot{\mathbf{z}} = \begin{bmatrix} \dot{\zeta} \\ \dot{\xi} \end{bmatrix} = \begin{bmatrix} -\gamma \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \omega \mathbf{I} \end{bmatrix} \begin{bmatrix} \zeta \\ \xi \end{bmatrix} + \begin{bmatrix} \lambda \mathbf{r} - \mathbf{g} \\ \mathbf{g} - \lambda \mathbf{r} \end{bmatrix} \quad (2.7)$$

where we chose ω and γ solutions of the Riccati equations:

$$\dot{\gamma} = \lambda - \gamma^2 \quad (2.8)$$

$$\dot{\omega} = \omega^2 - \lambda \quad (2.9)$$

A *divergent component of motion* ξ of the IPM is then:

$$\dot{\xi} = \omega \xi + (\mathbf{g} - \lambda \mathbf{r}) \quad (2.10)$$

Among all possible leg feedback controllers to maintain the biped's balance, those based on the divergent component of motion maximize the basin of attraction around stable capture states [30].

2.4. Boundedness condition The *boundedness condition* [19] characterizes the fact that state trajectories do not diverge. It can be expressed as:

$$\xi_i = \int_0^\infty e^{-\Omega(t)} (\lambda(t)\mathbf{r}(t) - \mathbf{g}) dt \quad (2.11)$$

where ξ_i denotes the initial value of the divergent component and Ω is the antiderivative of ω . This equation gives new constraints on the initial value ω_i in order to maintain $\xi(t)$ bounded. Based on Equation (2.11), we have:

$$\omega_i \mathbf{c}_i + \dot{\mathbf{c}}_i = \int_0^\infty e^{-\Omega(t)} (\lambda(t)\mathbf{r}(t) - \mathbf{g}) dt \quad (2.12)$$

2.5. Timeless Parametrization Let us define the new adimensional quantity $s(t) = e^{-\Omega(t)}$. This new variable ranges from $s = 1$ when $t = 0$ to $s = 0$ when $t \rightarrow \infty$. Its time derivatives are:

$$\dot{s}(t) = -\omega(t)s(t) \quad \ddot{s}(t) = \lambda(t)s(t) \quad (2.13)$$

Have by x' the derivation with respect to s as opposed to \dot{x} the derivation to t . The Riccati equation (2.9) of ω becomes:

$$\lambda = \omega^2 - \dot{\omega} = \omega^2 - \dot{s}\omega' = \omega(\omega + s\omega') = \omega(s\omega)' \quad (2.14)$$

Injecting this expression into (2.12), the boundedness condition becomes:

$$\int_0^\infty (\lambda(t)\mathbf{r}(t) - \mathbf{g}) e^{-\Omega(t)} dt = \int_0^1 (\omega(s\omega)' \mathbf{r}(s) - \mathbf{g}) \frac{ds}{\omega} \quad (2.15)$$

Let $\mathbf{x}_i = [\mathbf{c}_i \ \dot{\mathbf{c}}_i]$ denote a capturable state and $\mathbf{x}_f = [\mathbf{c}_f \ \dot{\mathbf{c}}_f]$ a static equilibrium. Then $s \mapsto \lambda(s), \mathbf{r}(s)$ drives the biped from \mathbf{x}_i to \mathbf{x}_f if and only if it satisfies the three conditions [4]:

1. Feasibility: $\lambda(s)$ and $\mathbf{r}(s)$ are feasible for all $s \in [0, 1]$
2. Convergence: $\lim_{s \rightarrow 0} \lambda(s) = \lambda_f, \lim_{s \rightarrow 0} \mathbf{r}(s) = \mathbf{r}_f$
3. Boundedness condition:

$$\int_0^1 \mathbf{r}(s)(s\omega)' ds - \mathbf{g} \int_0^1 \frac{ds}{\omega} = \omega_i \mathbf{c}_i + \dot{\mathbf{c}}_i \quad (2.16)$$

2.6. ZMP interpolation strategy The last step in the capturability analysis of the IPM is to define a ZMP trajectory. We choose a line segment going from the initial ZMP to the final ZMP \mathbf{r}_f of the capture state:

$$\mathbf{r}(s) = \mathbf{r}_f + (\mathbf{r}_i - \mathbf{r}_f)f(s\omega) \quad (2.17)$$

where f is chosen as a power law parameterized by a new variable $\alpha \in (0, 1)$:

$$f(s\omega) = \left(\frac{s\omega}{\omega_i} \right)^{\frac{\alpha}{1-\alpha}} \quad (2.18)$$

Doing so ensures the convergence condition (2) above [4], while condition (1) is ensured via a suitable choice of \mathbf{r}_i and \mathbf{r}_f . We will explore in Section 4 the parameterization of the new variable α introduced by this choice.

To enforce the boundedness condition (3), the final ZMP \mathbf{r}_f must satisfy equation (2.16). This equation can be separated in two components, the vertical axis:

$$\int_0^1 \frac{ds}{\omega} = \frac{\omega_i \bar{z}_i + \bar{c}_i}{g} \quad (2.19)$$

and the horizontal plane:

$$\int_0^1 \mathbf{r}(s)(s\omega)'ds = \omega_i \mathbf{c}_i^{xy} + \dot{\mathbf{c}}_i^{xy} \quad (2.20)$$

This second equation is the one we are interested in as it yields a constraint between \mathbf{r}_i and \mathbf{r}_f (2.16):

$$\mathbf{r}_i^{xy} = \mathbf{r}_f^{xy} + \frac{\omega_i(\mathbf{c}_i^{xy} - \mathbf{r}_f^{xy}) + \dot{\mathbf{c}}_i^{xy}}{(1 - \alpha)\omega_i} \quad (2.21)$$

We rewrite it equivalently as:

$$\mathbf{r}_f^{xy} = \mathbf{r}_i \left(1 - \frac{1}{\alpha}\right) + \frac{\mathbf{c}_i}{\alpha} + \frac{\dot{\mathbf{c}}_i}{\alpha\omega_i} \quad (2.22)$$

We can see that for $\alpha = 1$ and $\omega_i = \sqrt{\frac{g}{z_f}}$ we obtain the expression of the well-known capture point [26]. We will see in Section 4 how to optimize ω and α to have variation of length and frequency steps and better match the behavior of the avatar.

3. Animation of the full-body avatar The animation inputs can be separated into two parts. First, the upper-body is tracked by sensors. We need to track the waist user's which behavior is often close to the full body CoM. We also track the hands and head of the user in order to perfectly match their position between the user and his avatar. Those four limbs are the most important to be tracked, the other ones being optional for our application. Secondly, the lower body follows ZMP targets provided by the locomotion pipeline (Section 4). The first foot moving is the farthest from the ZMP. After that, each foot is moving alternatively, following a Hermit polynomial function between the initial and final foot states. The initial position and velocity of the foot can be directly re-used from the last state step. The final position is set to the computed target ZMP \mathbf{r}_f and the final velocity is set to zero. We add an intermediate mid-swing position 25 cm above ground and an offset of 15 cm perpendicular to the standing foot in order to avoid leg crossing. Together, lower and upper-body form the full-body inputs to the avatar animation.

Our animated avatar has 53 degrees of freedom with joints limits. Its geometry and inertial properties are automatically tuned depending on the size and mass of the user using the ART human calibration libraries¹. The waist is the root of the body and is a floating base. The mains limbs positions, waist, hands, feet and head, define the rest of the body positions through a kinematic tree. These positions are the input of a Proportional Derivative (PD) Cartesian controller. This controller moves the avatar like a puppet attached to strings. The whole operation is summarized in Figure 5.

By working with a kinematic tree and Cartesian PD control, we do not have access to joint torques. Dynamics are taken care of in the reduced model, for which we will enforce full physical consistency for user immersion, while maintaining close to perfect upper-body tracking at all times.

4. Optimization of Inverted Pendulum Inputs

4.1. Inversion of the Capture Problem We have seen in Section 2 how to express the final ZMP depending on the initial state \mathbf{c}_i , \mathbf{r}_i and the variables ω and α . We now define the initial parameters and constraints in order to choose a correct ω and α for our problem. First of all, let us assume that the avatar is starting in a physically-consistent state with its initial ZMP \mathbf{r}_i already in the contact area, edges not included, such as in Figure 6. The final ZMP \mathbf{r}_f must also be in the final

¹ <https://ar-tracking.com/>

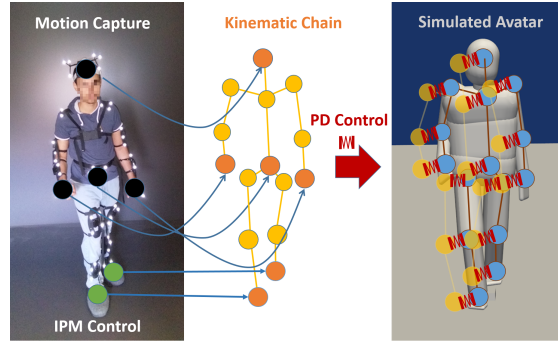


Figure 5: Full-body Animation strategy with Kinematic tree and PD control

contact area \mathcal{R} , in order to ensure the feasibility condition, \mathcal{R} being the *reachable area* of the swing foot. This constraint can be described in half-space representation:

$$\mathbf{F}\mathbf{r}_f^{xy} \leq \mathbf{p} \quad (4.1)$$

where \mathbf{F} is the $m \times 2$ matrix composed of the outward normals to the edges of \mathcal{R} and \mathbf{p} is an m -dimensional vector. \mathcal{R} is not fixed and can rather be modified to define the space we allow the avatar to cross with its swing leg in one step.

Injecting (2.21) into (4.1) leads a new half-space representation:

$$\omega_i(\mathbf{F}\mathbf{r}_i\alpha - \mathbf{F}\mathbf{r}_i + \mathbf{F}\mathbf{c}_i - \alpha\mathbf{p}) \leq -\mathbf{F}\dot{\mathbf{c}}_i \quad (4.2)$$

This equation has the form $\omega_i\mathbf{B} \geq \mathbf{A}$. Each line of this vector inequality provides a lower or upper bound on ω_i depending on the sign of the factor in front of it:

$$\omega_{i, \min} = \max \left(\sqrt{\lambda_{\min}}, A_j/B_j, B_j > 0 \right) \quad (4.3)$$

$$\omega_{i, \max} = \min \left(\sqrt{\lambda_{\max}}, A_j/B_j, B_j < 0 \right) \quad (4.4)$$

We see here that the constraint $\omega \in [\omega_{i, \min}, \omega_{i, \max}]$ depends on both α and on the dimensions of \mathcal{R} . In order to obtain ω , we assume for now that α and \mathcal{R} are known constants and that the interval $[\omega_{i, \min}, \omega_{i, \max}]$ is non-empty.

As in [4], we partition the interval $[0, 1]$ into n fixed segments $0 = s_0 < s_1 < \dots < s_{n-1} < s_n = 1$ where λ is a constant in each interval, i.e., $\forall s \in (s_j, s_{j+1}], \lambda(s) = \lambda_j$. Define:

$$\varphi(s) = s^2 w^2 \quad \delta_j = s_{j+1}^2 - s_j^2 \quad (4.5)$$

The two variables λ and w are then expressed from φ as:

$$\lambda_j = \frac{\varphi_{j+1} - \varphi_j}{\delta_j} \quad \omega_j(s) = \sqrt{\varphi_j + \lambda_j(s^2 - s_j^2)} \quad (4.6)$$

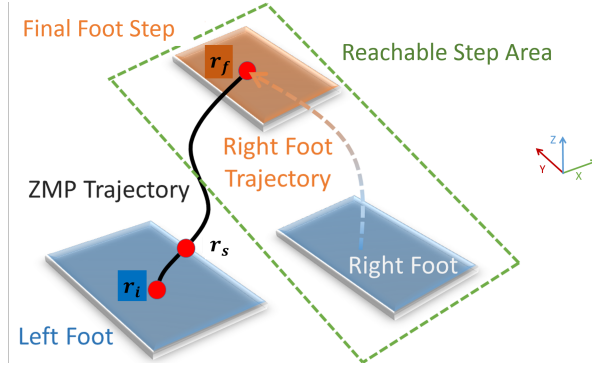


Figure 6: ZMP estimated trajectory for one step strategy

Adding a regularizing cost function over variations of λ yields the following nonlinear problem:

$$\underset{\{\varphi_1, \dots, \varphi_n\} \in \mathbb{R}^n}{\text{minimize}} \sum_{j=1}^{n-1} \left[\frac{\varphi_{j+1} - \varphi_j}{\delta_j} - \frac{\varphi_j - \varphi_{j-1}}{\delta_{j-1}} \right]^2 \quad (4.7a)$$

$$\sum_{j=1}^{n-1} \frac{\delta_j}{\sqrt{\varphi_{j+1}} + \sqrt{\varphi_j}} = 0 \quad (4.7b)$$

$$\omega_{i,\min}^2 \leq \varphi_n \leq \omega_{i,\max}^2 \quad (4.7c)$$

$$\forall j < n, \lambda_{\min} \delta_j \leq \varphi_{j+1} \leq \lambda_{\max} \delta_j \quad (4.7d)$$

$$\varphi_1 = \delta_0 \lambda_f = \delta_0 g / \bar{z}_f \quad (4.7e)$$

With the following parameters:

- Feasibility bounds $(\lambda_{\min}, \lambda_{\max})$ and $(\omega_{\min}, \omega_{\max})$
- Initial height $\dot{\bar{z}}_i$ and target height \bar{z}_f
- Number of steps n and discretization steps $\delta_1, \dots, \delta_n$

We solve this nonlinear problem using the dedicated Capture Problem Solver (CPS) which is an open source software² and solves typical problem instances in less than 100 microseconds using a tailored Sequential Quadratic Programming algorithm. The corresponding solution provides values for all φ_j 's, in particular φ_i , and therefore a new value of ω_i by equation (4.6).

We now have to define properly the parameterization of α and \mathcal{R} that determine the values of ω_{\min} and ω_{\max} .

4.2. Selection of α and \mathcal{R} On the one hand, \mathcal{R} defines the length of the future step. We want this area far enough from the supporting foot so as to avoid leg crossing and small enough to reduce the step length. On the other hand α affects the time-varying behavior of the future estimated ZMP trajectory, and therefore, the stepping frequency. While the strategy commonly found in humanoid robotics is to match predefined distance and/or step timings, we propose a method that enables variations of these parameters.

Our approach works in two steps. First, before applying CPS, we find the smallest area \mathcal{R} allowing a non-empty solution for $[\omega_{\min}, \omega_{\max}]$ and $[\alpha_{\min}, \alpha_{\max}]$. Then, we seek for the optimal solution to the capture problem by browsing the allowed α values.

² <https://github.com/jrl-umi3218/CaptureProblemSolver>

Remember that Equation (4.2) provides new constraints on the limits of ω_i , depending on the reachable swing foot area \mathcal{R} and ZMP interpolation parameter α . This inequality can be written:

$$(\mathbf{u} - \alpha\mathbf{v})\omega_i \geq \mathbf{w} \quad (4.8)$$

To avoid singling out corner cases, \mathbf{u} , \mathbf{v} and \mathbf{w} are extended with two additional lines:

- $u_j = 1, v_j = 0, w_j = w_{i,\min}$
- $u_i = 0, v_i = 1, w_i = -w_{i,\max}$

Next, note the two sets $\mathcal{A}_{\min} = \{i, u_i - \alpha v_i \geq 0\}$ and $\mathcal{A}_{\max} = \{i, u_i - \alpha v_i \leq 0\}$ are such that:

$$\omega_{i,\min}(\alpha) = \max \left(\frac{w_i}{u_i - \alpha v_i}, i \in \mathcal{A}_{\min}(\alpha) \right) \quad (4.9)$$

$$\omega_{i,\max}(\alpha) = \min \left(\frac{w_i}{u_i - \alpha v_i}, i \in \mathcal{A}_{\max}(\alpha) \right) \quad (4.10)$$

A necessary and sufficient condition for $\omega_{i,\min} \leq \omega_{i,\max}$ is then that, for all pairs $(i, j) \in \mathcal{A}_{\min}(\alpha) \times \mathcal{A}_{\max}(\alpha)$,

$$u_i w_j - u_j w_i \leq \alpha(v_i w_j - v_j w_i) \quad (4.11)$$

The resolution of this system of linear inequalities yields an interval $[\alpha_{\min}, \alpha_{\max}]$ given the three vectors \mathbf{u} , \mathbf{v} and \mathbf{w} [4].

However, contrary to [4], our vectors \mathbf{u} , \mathbf{v} , \mathbf{w} are not constant as they will change while we vary the reachable area \mathcal{R} . Our strategy is then to look for the smallest area \mathcal{R} where we want the avatar to step, and compute a solution interval $[\alpha_{\min}, \alpha_{\max}]$ using the above algorithm. If the interval is empty, we increase \mathcal{R} , thus increasing \mathbf{u} , \mathbf{v} , \mathbf{w} values, in an iterative way, until a non-empty interval is found or if we reach the maximum reachable area. If the latter case, we step in the direction of the ZMP final as fast as possible, but with a step size limited by \mathcal{R} . We resume our method, as illustrated in Figure 7, with the presentation of the simple and double contact area, delimited by the simulated positions of the feet boundaries along x and y axis, x_l, x_u and y_l, y_u .

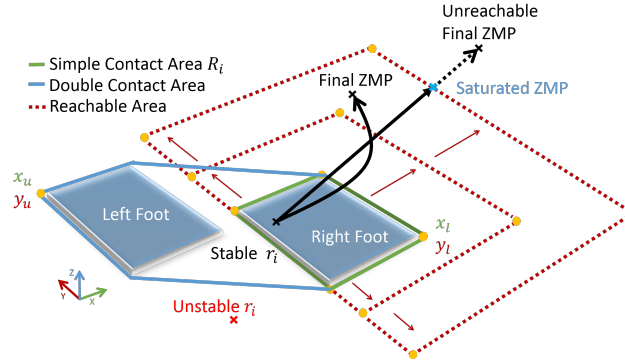


Figure 7: Seeking Iteration of Reachable Area Step

Now that we have α intervals, along with a pair of $[\omega_{\min}, \omega_{\max}]$ for each α by (4.9)–(4.10), we want to use the best α value for the CPS. CPS computation times are not affected by our selection of values for α nor \mathcal{R} , since we only change the values of problem parameters without affecting problem structure. We exploit this feature to search many times with constant α and \mathcal{R} to find an optimal solution of the more general optimization where α and \mathcal{R} are variables. From reachability constraints and initial state conditions, the capture problem solver returns a feasible ZMP trajectory $\mathbf{r}(s)$ over the index s defined in Section 2. Next, we want its equivalent over time $\mathbf{r}(t)$ to obtain

the time period and quantify the quality of the solution. To do so, we compute the particular *contact-switch index* s_c solution to $\mathbf{F}\mathbf{r}_s^{xy} = \mathbf{p}$ with \mathbf{F} and \mathbf{p} function of the supporting foot area. This s_c defines the moment the avatar must have completed its step. Its time equivalent t_c can be computed from Equations (4.6) (2.22) by:

$$t(s_j) = t_{j+1} + \frac{1}{\sqrt{\lambda_j}} \log \left(\frac{\sqrt{\varphi_{j+1}} + \sqrt{\lambda_j s_{j+1}}}{\sqrt{\varphi_j} + \sqrt{\lambda_j s_j}} \right) \quad (4.12)$$

For now, we try to reach a fixed average time period of 0.5 second with this method, but we could apply it to have a pair of r_f and t_f which would match the velocity of the CoM or the capture point derivative. The general strategy is summarized in Algorithm 1:

```

Result:  $r_f, t_f$ 
 $\alpha_{\min}; \alpha_{\max};$ 
 $i = 0; t_f = 0;$ 
while  $i \leq \text{size}(\alpha_{\min})$  do
     $\alpha_j = \alpha_{\min}(i);$ 
    while  $j < \text{size}(\alpha_{\max})$  do
        Computation of  $[\omega_{\min}, \omega_{\max}]$  with  $\alpha_j$ ;
        Computation of  $r_f$  with CPS;
        Computation of  $t_s$  from  $s_f$ ;
        if  $\text{abs}(t_s - 0.5) < \text{abs}(t_f - 0.5)$  then
             $t_f = t_s;$ 
             $\alpha_j = \alpha_j + \frac{(\alpha_{\max} + \alpha_{\min})}{2};$ 
        end
         $i++$ 
    end
end

```

Algorithm 1: Two Stage Optimization

5. Implementation in real time application

Several experiments were set up to test the implementation with real time motion. We evaluated the physical consistency of the system, the difference with other locomotion controls and the robustness of the CPS method. For this purpose, we realized three experiments depicted in Figure 8:

- Ideal double loop walking
- Ideal and Motion Capture double loop walking
- Motion Capture standing balance
- Motion Capture double loop walking with height variations

The locomotion algorithms proposed are the CPS method with α optimization, CPS method without α optimization (to show the impact of variable frequency step) and the classical Capture Point r_{cp} defined in Equation (2.2). The classical capture point is defined for the LIPM with no CoM height variation. It also corresponds to (2.22) with $\alpha = 1$ and $\omega = \bar{z}_f/g$.

Experiments elements are represented in Figure 1. We experimented first with artificial reference CoM trajectories such as sines and cosines, then with motion capture data shown in the accompanying

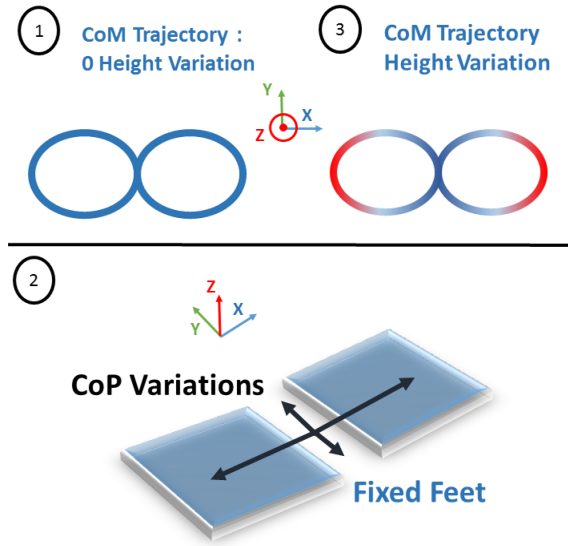


Figure 8: The three different CoM positions over time from the user used to test the semi-interactive control of the avatar

video. The operator is equipped with 17 infra-red passive sensors and filmed by 8 infra-red cameras working at 150 Hz in a 3 square meter room. This information is saved to be replayed off-line, yet at the same speed as the original to keep a real time simulation. The sensors on the legs are used to compare our results between the Capture Point strategy and real user movements. Our simulation runs at 100 Hz on a 4-core Intel(R) Xeon(R) CPU E5630 @ 2.53 GHz.

To quantify the physical realism of the system, we observe the position of the measured ZMP r_i over the contact area. However, it is not easy to evaluate it directly from plots showing only one axis. The simulated positions of the feet boundaries along x and y axis gives the lower and upper values x_l, x_u and y_l, y_u which represent the contact area of the avatar. However, if you look back at Figure 7, the measured positions give by projection an area in the form of a square, while the real contact area looks more like an hexagon.

That is why we propose to show on each experiment two kind of plots. One kind with steps positions and measured ZMP equal to :

$$r_i^{xy} = c + \frac{c}{\lambda_i} \quad (5.1)$$

And another kind of plot with time percent contact stability representing the fraction of time the following condition is respected:

$$Fr_i^{xy} \leq p \quad (5.2)$$

With F and p defined as in Equations 4.1 by the real contact area.

5.1. Ideal and Motion Capture Double loop walking The first experiment is a trajectory in the form of a double loop. With simple CoM trajectory we can compare the quality of our method with the two other locomotion controls.

We report only here the X axis results which displays the largest CoM variations: similar results were obtained for the Y axis Since the overall movements of the user yield a 360° rotation, X and Y axes play the same role alternatively. The first part of the plot called "Init" show the initialisation time when the avatar does not start at the same position as input tracked motion, this part is highly

unstable until the avatar matches the tracked upper-body position. After that, the avatar enters the second part, called "Walking", and can apply the strategy depending on the chosen locomotion method.

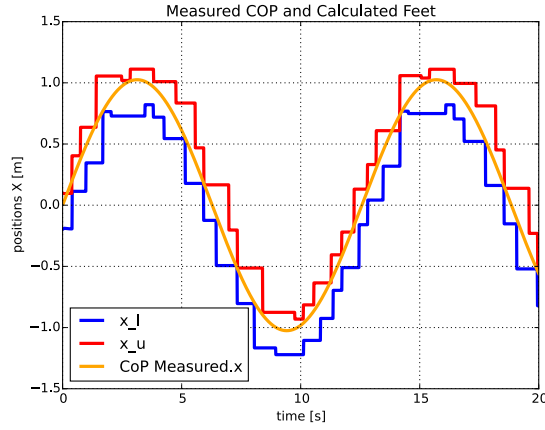


Figure 9: Position of the Contact Area delimited by the feet with x_u and x_l and the Generated ZMP X axis during Ideal Double loop walking

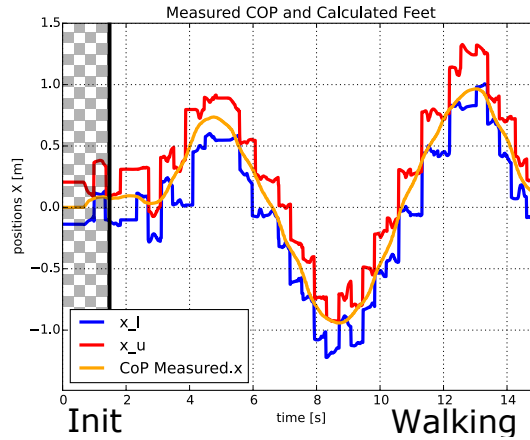


Figure 10: Position of the Contact Area delimited by the feet with x_u and x_l and the Generated ZMP r_i along X axis during Motion Capture Double loop walking

As we can see, the CPS method maintain a higher physical consistency than the other algorithms on both application, with a 80% contact stability over time. The values of the contact area represented by x_l and x_l have small fluctuation on the motion capture data due to the changing avatar orientation over Z axis, thus modifying the orientation of the feet.

5.2. Motion Capture Standing balance The Standing application is here to show the quality of the step intention detection of the algorithm as we know some locomotion methods have tendencies to step continuously like in [14, 22] without being able to bring itself to a stop.

Since the user is not moving a lot, we can observe a high contact stability of 90% over time from our method. The avatar still needs to step 10 times over the 17 seconds of Walking phase while the operator does none. This can be explained by the fact that a person can exploit the flexibility and stiffness existing between ground and feet contact.

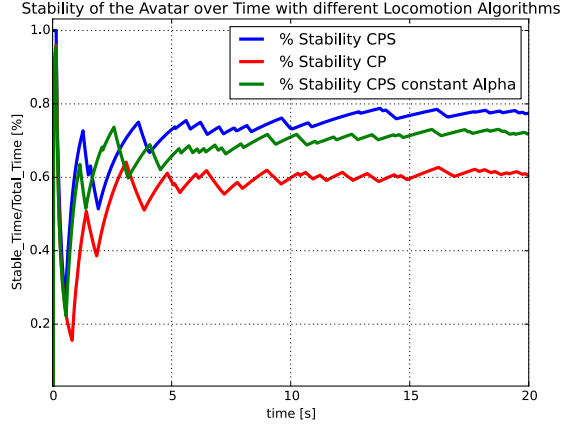


Figure 11: Contact Stability percent overtime with our CPS method top, CPS method with fixed step middle, and classical capture point bottom during Ideal Double loop walking

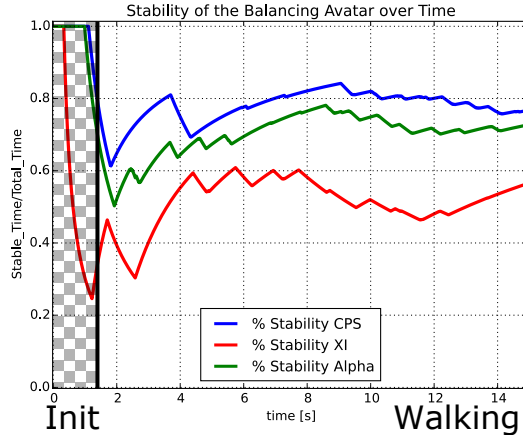


Figure 12: Contact Stability percent overtime with our CPS method on top, CPS method with fixed step on middle, and classical capture point on bottom during Motion Capture Double loop walking

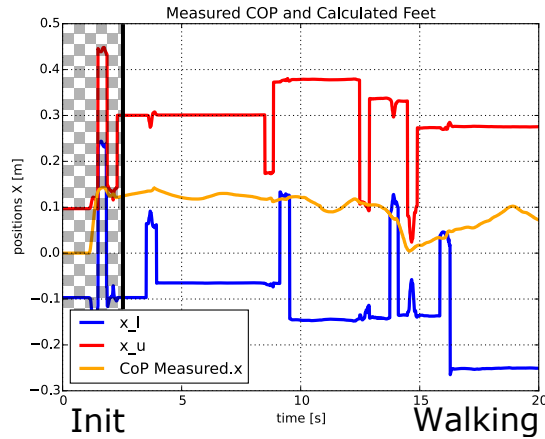


Figure 13: Position of the Contact Area delimited by the feet with x_u and x_l and the Generated ZMP r_i along X axis during Motion Capture Standing balance

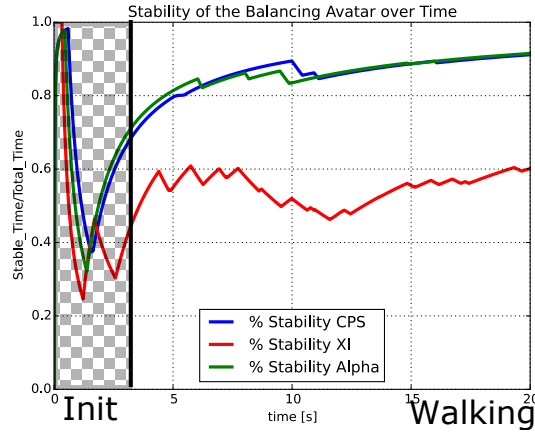


Figure 14: Contact Stability percent overtime with our Contact Stability percent overtime with our CPS method on top, CPS method with fixed step in the middle, and classical capture point on bottom during Motion Capture Standing balance

5.3. Motion Capture Double loop walking with height variation. Finally, we show the results of the avatar walking with a last experiment showing once again a loop of the ∞ trajectory, but this time with more important variation of the CoM height, up to 25 cm.

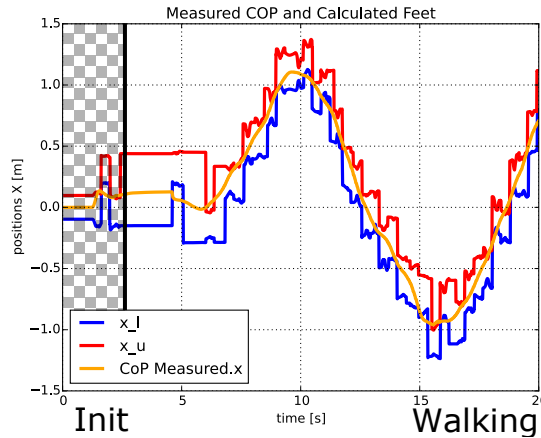


Figure 15: Position of the Contact Area delimited by the feet with x_u and x_l and the Generated ZMP r_i along X axis

This plot shows the impact height variation has on the quality of the methods. Our CPS method with λ variation allow us better predict those variations, and to maintain a better physics consistency. This is not the case of the other methods which gives the worst results on this experiment.

6. Use cases Our method was tested on three Use cases to illustrate the importance of the method and its versatility. The applications involve the tracking of people in a more complex environments than an empty room. The algorithm generating the leg trajectory is running at 30ms while the player screen shows the environment at 60 FPS. The first application, pictured in Figure 17, is a firefighter being captured while applying first aid on a real "mannequin". His movements will be animated in VR for teaching purposes. The room is filled with a stretcher trolley, a mannequin, a first-aid kit, a perfusion bag and other medical tools. The firefighter upper-body is tracked with Xsens inertial sensors on the head and hands and HTC Vive trackers for the arms and the waist. As we can see,

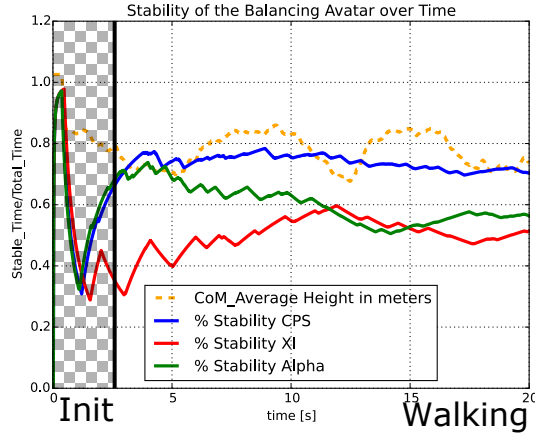


Figure 16: Center of mass height position in meters with Contact Stability percent overtime with our CPS method on top, CPS method with fixed step in the middle, and classical capture point on bottom during Motion Capture double loop walking with height variation

depending on the position of the firefighter, cameras may not be able to track its lower body due to the presence of obstacles.



Figure 17: Firefighter applying first-aid demonstration during Victeams ANR Project

The second application in Figures 18 and 19 involves a worker manipulating around a fake bench tool in VR and Real Time. The operator body is tracked with HTC Vive Trackers. The four main limbs are tracked (waist, head, both hands) with two optional trackers on the elbows. The full manipulation can be seen in the annex video and show cases, like the previous application, of the utility of the legs generated trajectory for complex scenes.

Figure 20 presents a real time VR application with multiple operators in the same simulation. Each user's upper-body is tracked with Vive Trackers technology and equipped with a back-packed laptop. Each laptop computes its own full-body avatar configuration. Those configurations are sent to a master computer which gathers all the user's data in the simulation. This example with multiple moving users shows that it can be complex to track the whole movements with cameras. Our method generated legs trajectory, the most hidden part of the bodies, and assured avatar's movements at each time.

As part of futur works, we would like to study the user's perception as in [3, 20], when looking at another avatar regulated with the CPS method.

7. Conclusions

We proposed to generate an avatar motion semi-autonomously, with perfect upper-body tracking and autonomous, physically-consistent lower body motions. Our approach relies

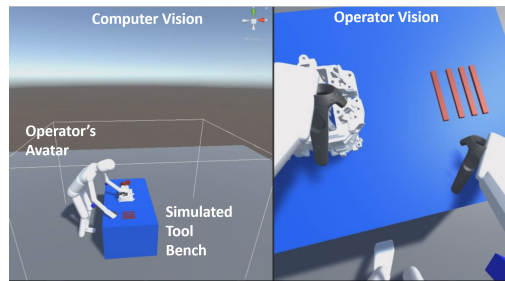


Figure 18: VR environment of an operator working around a bench tool in real time

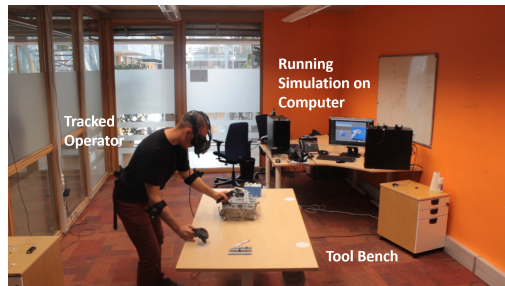


Figure 19: Real environment of an operator working around a bench tool in real time



Figure 20: Multiple operators in same real time VR simulation

on a full-body kinematic tree, proportional-derivative kinematic control, and a reduced dynamic model to ensure physical consistency and thus preserve user immersion as much as possible.

Using a Capture Problem Solver, we produce physically-consistent dynamic walking by bounding the dynamics of the Inverted Pendulum Model to feasible non-diverging solutions. Our main contribution is an inversion of the existing capture problem that produces a footstep sequence from a given CoM trajectory. Our contribution changes inputs and constraints of the capturability problem, but keeps the same solver. This strategy can maintain the contact stability of the avatar while computing leg trajectories that support the upper-body. This method can facilitate the use of VR formation for companies which are not specialized in VR, nor heavily equipped. It allows the tracking of a user into cluttered environments where obstacles may obstruct the lower body from sensors.

Yet, for now the physical consistency of the system is not assured 100% of the time. In future work, we would like to improve the optimization of the external variable α in order to find ZMP trajectories and step timings closer to the user's. To do so, we could take into account first-order variations of the CPS optimum with respect to α . Alternatively, we could get back to a non-reversed robotic control, which would obtain a 100% stability state but not 100% of CoM tracking. In that case, one would have to study the impact it would have on the immersive perception of the operator. The next step for us will be to implement our work in a more complex environment with virtual obstacles, variation of ground height, and evaluate the impact of these experiences on the user's perception.

References

- [1] AGRAWAL, Shailen, Michiel van de PANNE. *Task-based locomotion*. ACM Transactions on Graphics (TOG) 35(4):82, 2016.
- [2] AGRAWAL, Shailen, Shuo SHEN, Michiel van de PANNE. "Diverse motion variations for physics-based character animation". *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. ACM. 2013: 37–44.
- [3] ALLMENDINGER, Katrin. *Social presence in synchronous virtual learning situations: the role of nonverbal signals displayed by avatars*. Educational Psychology Review 22(1):41–56, 2010.
- [4] CARON, Stéphane, Adrien ESCANDE, Leonardo LANARI, Bastien MALLEIN. *Capturability-based pattern generation for walking with variable height*. IEEE Transactions on Robotics, 2019.
- [5] COLLETTE, Cyrille, Alain MICAELLI, Claude ANDRIOT, Pierre LEMERLE. "Robust balance optimization control of humanoid robots with multiple non coplanar grasps and frictional contacts". *IEEE International Conference on Robotics and Automation*. 2008: 3187–3193.
- [6] COROS, Stelian, Philippe BEAUDOIN, Michiel VAN DE PANNE. "Generalized biped walking control". *ACM Transactions on Graphics*. Volume 29. 4. 2010: 130.
- [7] DEL PRETE, Andrea, Steve TONNEAU, Nicolas MANSARD. *Zero step capturability for legged robots in multicontact*. IEEE Transactions on Robotics 34(4):1021–1034, 2018.
- [8] ELHASAIRI, Ahmed, Alexandre PECHEV. *Humanoid robot balance control using the spherical inverted pendulum mode*. Frontiers in Robotics and AI 2:21, 2015.
- [9] ENGLSBERGER, Johannes, Christian OTT, Alin ALBU-SCHÄFFER. *Three-dimensional bipedal walking control based on divergent component of motion*. IEEE Transactions on Robotics 31(2):355–368, 2015.
- [10] GRIFFIN, Robert J, Georg WIEDEBACH, Sylvain BERTRAND, Alexander LEONESSA, Jerry PRATT. "Walking stabilization using step timing and location adjustment on the humanoid robot, atlas". *IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2017: 667–673.
- [11] HAUSER, John, Alessandro SACCON, Ruggero FREZZA. "Achievable motorcycle trajectories". *IEEE Conference on Decision and Control*. Volume 4. 2004: 3944–3949.
- [12] HERDT, Andrei, Nicolas PERRIN, Pierre-Brice WIEBER. "Walking without thinking about it". *IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2010: 190–195.
- [13] KAJITA, Shuuji, Fumio KANEHIRO, Kenji KANEKO, Kazuhito YOKOI, Hirohisa HIRUKAWA. "The 3d linear inverted pendulum mode: a simple modeling for a biped walking pattern generation". *IEEE/RSJ International Conference on Intelligent Robots and Systems*. Volume 1. 2001: 239–246.

- [14] KHADIV, Majid, Alexander HERZOG, S Ali A MOOSAVIAN, Ludovic RIGHETTI. *Walking control based on step timing adaptation*. arXiv preprint arXiv:1704.01271, 2017.
- [15] KIM, Yun-hyeong, Taesoo KWON, Daeun SONG, Young J KIM. *Full-body animation of human locomotion in reduced gravity using physics-based control*. IEEE computer graphics and applications 37(6):28–39, 2017.
- [16] KOOLEN, Twan, Tomas DE BOER, John REBULA, Ambarish GOSWAMI, Jerry PRATT. *Capturability-based analysis and control of legged locomotion, part 1: theory and application to three simple gait models*. The International Journal of Robotics Research 31(9):1094–1113, 2012.
- [17] KOOLEN, Twan, Michael POSA, Russ TEDRAKE. “Balance control using center of mass height variation: limitations imposed by unilateral contact”. *IEEE-RAS International Conf. on Humanoid Robots*. 2016: 8–15.
- [18] KRICHENBAUER, Max, Goshiro YAMAMOTO, Takafumi TAKETOM, Christian SANDOR, Hirokazu KATO. *Augmented reality versus virtual reality for 3d object manipulation*. IEEE transactions on visualization and computer graphics 24(2):1038–1048, 2017.
- [19] LANARI, Leonardo, Seth HUTCHINSON, Luca MARCHIONNI. “Boundedness issues in planning of locomotion trajectories for biped robots”. *IEEE-RAS International Conference on Humanoid Robots*. 2014: 951–958.
- [20] LATOSCHIK, Marc Erich, Florian KERN, Jan-Philipp STAUFFERT, Andrea BARTL, Mario BOTSCH, Jean-Luc LUGRIN. *Not alone here?! scalability and user experience of embodied ambient crowds in distributed social virtual reality*. IEEE transactions on visualization and computer graphics 25(5):2134–2144, 2019.
- [21] LIU, Mingxing. “Personnage virtuel: contrôleur hybride couplant commande dynamique multi-objectifs et mouvements capturés”. PhD thesis. Paris 6, 2012.
- [22] MORDATCH, Igor, Martin DE LASA, Aaron HERTZMANN. *Robust physics-based locomotion using low-dimensional planning*. ACM Transactions on Graphics 29(4):71, 2010.
- [23] ORIN, David E, Ambarish GOSWAMI, Sung-Hee LEE. *Centroidal dynamics of a humanoid robot*. Autonomous Robots 35(2-3):161–176, 2013.
- [24] POULIQUEN, Mamy, Alain BERNARD, Jacques MARSOT, Laurent CHODORGE. *Virtual hands and virtual reality multimodal platform to design safer industrial systems*. Computers in Industry 58(1):46–56, 2007.
- [25] PRATT, Jerry E, Sergey V DRAKUNOV. “Derivation and application of a conserved orbital energy for the inverted pendulum bipedal walking model”. *IEEE International Conference on Robotics and Automation*. 2007: 4653–4660.
- [26] PRATT, Jerry, John CARFF, Sergey DRAKUNOV, Ambarish GOSWAMI. “Capture point: a step toward humanoid push recovery”. *IEEE-RAS International Conference on Humanoid Robots*. 2006: 200–207.
- [27] RAMOS, Oscar E, Kris HAUSER. “Generalizations of the capture point to nonlinear center of mass paths and uneven terrain”. *IEEE-RAS International Conference on Humanoid Robots*. 2015: 851–858.
- [28] REGAN, Clare. *An investigation into nausea and other side-effects of head-coupled immersive virtual reality*. Virtual Reality 1(1):17–31, 1995.
- [29] SALINI, Joseph, Vincent PADOIS, Philippe BIDAUD. “Synthesis of complex humanoid whole-body behavior: a focus on sequencing and tasks transitions”. *IEEE International Conference on Robotics and Automation*. 2011: 1283–1290.
- [30] SUGIHARA, Tomomichi. “Standing stabilizability and stepping maneuver in planar bipedalism based on the best com-zmp regulator”. *IEEE Int. Conf. on Robotics and Automation*. 2009: 1966–1971.