The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Control Design and Experimental Evaluation of the 2D *CyberWalk* Platform

Alessandro De Luca Raffaella Mattone
Dipartimento di Informatica e Sistemistica
Università di Roma "La Sapienza"
Via Ariosto 25, 00185 Roma, Italy
{deluca,mattone}@dis.uniroma1.it

Paolo Robuffo Giordano Heinrich H. Bülthoff
Max Planck Institute
for Biological Cybernetics
Spemannstraße 38, 72076 Tübingen, Germany
{paolo.robuffo-giordano,heinrich.buelthoff}@tuebingen.mpg.de

***Abstract*— The *CyberWalk* is a large size 2D omni-directional platform that allows unconstrained locomotion possibilities to a walking user for VR exploration. In this paper we present the motion control design for the platform, which has been developed within the homonymous European research project. The objective is to compensate the intentional motion of the user, so as to keep her/him always close to the platform center while limiting the perceptual effects due to actuation commands. The controller acts at the acceleration level, using suitable observers to estimate the unmeasurable intentional walker's velocity and acceleration. A moving reference position is used to limit the accelerations felt by the user in critical transients, e.g., when the walker suddenly stops motion. Experimental results are reported that show the benefit of designing separate control gains in the two orthogonal directions (lateral and sagittal) of a frame attached to the walker.**

I. Introduction

Exploration of virtual reality worlds by allowing omni-directional unconstrained locomotion possibilities for a walking user is the main objective of the European research project *CyberWalk* [1]. Different locomotion interfaces already exist that allow walking in virtual environments (see, e.g., the surveys in [2] and [3]). However, they usually constrain the feet, the body, or the legs of the user. Moreover, locomotion is either restricted to a 1D motion on a linear treadmill, like in the Treadport platform [4] with possible slope effects [5], or is limited in space and speed, like in the case of the moving tiles of the CirculaFloor [6].

Within the *CyberWalk* project, two different 2D motion concepts have been considered for unconstrained walking on a plane: the "ball-array" *CyberCarpet* and the more recent "omni-directional" *CyberWalk* platform. Both concepts have been analyzed in terms of mechanical feasibility, perceptual effects, and motion control point of view. The ball-array principle is similar to that in [7]: a conveyor belt and a turntable are used to transmit translational and angular motion to the walker through a ball-array board. The kinematic model in this case is nonlinear and the corresponding control problem has been tackled in [8], [9], [10].

The second concept of the omnidirectional *CyberWalk* platform, which is the one considered in this paper, is sketched in Fig. 1. The platform consists of an array of synchronous linear belts $C_1, C_2, \ldots, C_N$ that can be displaced with a common velocity v_{cx} in the (blue) direction,

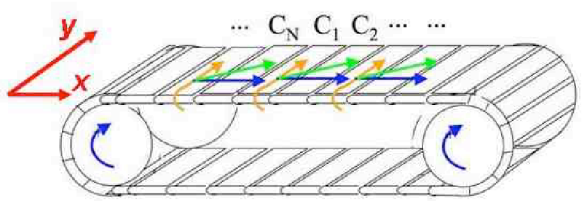

Fig. 1. The kinematic concept of the 2D *CyberWalk* platform (German Patent filed in 2006 by TUM, Munich)

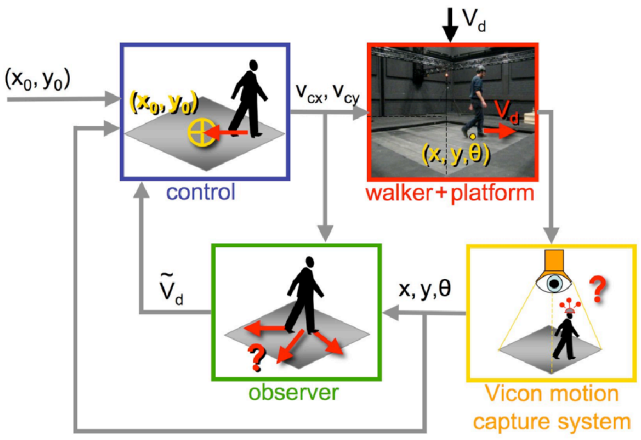

Fig. 2. Control system architecture of the *CyberWalk*

which is orthogonal to the (yellow) velocity v_{cy} of each independent belt. Thus, a velocity with any (green) direction in the plane can be obtained as a combination of the two linear motions. In this case, the system kinematics is linear. This concept is similar, although with different implementations of the mechanical, actuation, and control parts, to the Omnidirectional Treadmill proposed in [11], which uses two perpendicular belts and a large number of rollers, and to the Torus Treadmill [12], having a torus-shaped belt arrangement but of quite small dimensions.

The overall control system architecture of the *CyberWalk* platform is illustrated in Fig. 2. The walker is allowed to execute slow or fast locomotion in a natural way and in any

Fig. 3. The realized 2D *CyberWalk* platform before it was moved to its final location in Tübingen (picture courtesy of TUM, Munich)

planar (possibly infinite) direction. The platform controller counteracts her/his motion and pulls the walker toward the center of the platform, taking into account physiologically acceptable velocity/accelerations bounds. The head pose on the platform is tracked by a Vicon motion capture system and is used for estimating the walker voluntary motion (through suitably designed observers) and for driving the actuators that move the platform in the two directions[1]. The combined walker-platform displacement is also needed to update the scene of the virtual environment shown to the user through a head mounted display. A 3D virtual representation of the ancient city of Pompeii, generated with CityEngine [13], has been used in the actual experiments. A full-scale prototype of the platform has been built by the Technical University of Munich within our European project (see Fig. 3), and is currently located at the Max Planck Institute for Biological Cybernetics in Tübingen. With a side of 5 m, *CyberWalk* is the largest VR platform in the world.

Previous works on locomotion interfaces have paid little attention to control algorithms, relying mostly on very simple PID laws or heuristic schemes. With the few notable exception of [14] for a 1D treadmill, no stability analysis has been considered until now, while control performance could be predicted in 2D only by restricting motion to slow and piece-wise constant walker velocity, and with few directional changes. In particular, control design issues are fully overlooked in [11], [12], the two other papers dealing with platforms of similar construction. In this paper, we will address the complete motion control design for the omni-directional 2D *CyberWalk* platform, including its experimental evaluation. Related information on the mechanical design of the platform and on the VR representation and integration in the system can be found in [15] and [16], respectively.

After recalling the second-order kinematic model of the platform (Sect. II), we present the control strategy in Sect. III: this includes the estimation of walker's acceleration and velocity (see Sects. III-A, III-B), the tuning of the reference position (Sect. III-C), whose effects on the closed-loop behavior is evaluated in Sect. III-D, and the rotation of the control frame (Sect. III-E). The control objective, in fact, is keeping the position of the walker near to the platform center, without violating the physiological constraints on comfortable accelerations and jerks, that are particularly strict in the side direction w.r.t. that of motion. Performance has been evaluated through a series of experiments reported in Sect. IV, which can be further appreciated in the video clip accompanying this paper. Further movies are available at http://www.dis.uniroma1.it/∼labrob/research/CW.html.

II. Kinematic model

Each direction (i.e., x and y) of the *CyberWalk* platform is independently actuated by a combined electrical/hydraulical system, with low-level servo-controllers providing torques and accepting high-level velocity reference commands[2]. However, we have chosen to design the control law at the acceleration level, with the velocity control inputs computed then by numerical integration. In fact, acceleration control is more suitable to take into account the limitations imposed to the platform motion by the actuation system and/or by the compliance with the physiology of a human walker.

To this purpose, the interaction between the walker and the platform can be modeled by the following second order system

$$\begin{array}{rcll} \dot{x}_i & = & v_i, & \\ \dot{v}_i & = & a_{ci} + a_{wi}, & i = 1, 2 \end{array} \qquad (1)$$

where x_i is the (measurable) position of the walker, v_i the walker's absolute velocity, a_{ci} is the platform acceleration, and a_{wi} is the walker's own (intentional) acceleration. The indices $i = 1, 2$ represents the two planar directions $x_1 = x$ and $x_2 = y$ on the platform surface, and $x_1 = x_2 = 0$ (or $x = y = 0$) is the platform center. Since the system model and the corresponding control problem is exactly the same in the two orthogonal directions of platform motion, we will drop in the following the index i whenever it is not strictly necessary, and consider the control problem in the first place as one-dimensional (1D). We note that the orientation θ_w (around a vertical axis) of the walker cannot be modified by the platform motion and is thus not included in the kinematic model (1). Still, this orientation can be measured through the Vicon system by assuming that the user looks in the direction of his planar motion, and will be used in Sect. III-E to improve the control behavior in terms of perceptual effects on the walker.

[1]In general, non-intrusive tracking systems, such as overlooking cameras, could also be used to obtain the planar walker position. However, an accurate Vicon tracking of the head position/orientation would still be needed for correctly displaying the virtual scene viewpoint to the user. Therefore, we chose to use the head pose tracked by the Vicon system as an estimation of the walker's planar position.

[2]The platform dynamics, i.e., the relationship between commanded and actual velocity in each of the two main directions, has been identified through experiments. The delay/attenuation is negligible up to 50 Hz in the y-direction of the N belts, and up to 3.5 Hz in the common x-direction. Inclusion of this dynamic model in the control design was found irrelevant, see [17] for details.

III. CONTROL DESIGN

The simplest way for stabilizing the linear system (1) to a desired position x_{ref} would be using, in each of the two controlled directions of the plane, a control law of the form

$$a_c = -a_w - k_v v + k_p(x_{ref} - x), \tag{2}$$

with $k_v > 0$ and $k_p > 0$. However, this would require the availability of the unmeasurable quantities v and a_w, which are describing the a priori unknown intentional walker motion. This problem can be solved by replacing v and a_w in eq. (2) with proper estimates $\hat{v}$ and $\hat{a}_w$, i.e., by taking

$$a_c = -\hat{a}_w - k_v \hat{v} + k_p(x_{ref} - x). \tag{3}$$

The problem of obtaining the estimates $\hat{v}$ and $\hat{a}_w$ is considered in the next two subsections.

A. Estimation of the walker voluntary acceleration

The walker voluntary acceleration a_w (an external disturbance for the control system) is estimated by the linear observer

$$\begin{aligned} \dot{\xi}_1 &= \xi_2 + k_1(x - \xi_1) \\ \dot{\xi}_2 &= a_c + k_2(x - \xi_1) \\ \hat{a}_w &= k_2(x - \xi_1), \end{aligned} \tag{4}$$

where $k_1 > 0$, $k_2 > 0$, and (ξ_1, ξ_2) is the observer state. Transforming these equations in the Laplace domain, we obtain

$$\hat{A}_w(s) = \frac{k_2}{s^2 + k_1 s + k_2} A_w(s) = F_1(s) A_w(s), \tag{5}$$

showing that the estimation $\hat{a}_w$ is a stable, low-pass filtered version of the unknown quantity a_w. Moreover, by exploiting the model (1), eq. (5) can be rewritten in terms of the measured walker's position and the platform acceleration command as

$$\hat{A}_w(s) = \frac{k_2}{s^2 + k_1 s + k_2} \left(s^2 X(s) - A_c(s) \right). \tag{6}$$

This equivalent expression will be useful in the analysis of the closed-loop system behavior.

B. Estimation of the walker absolute velocity

Similarly, an estimation of the walker absolute velocity v (an unmeasured state of the system) is provided by the linear observer

$$\begin{aligned} \dot{\xi}_3 &= k_3(x - \xi_3) \\ \hat{v} &= k_3(x - \xi_3), \end{aligned} \tag{7}$$

with state ξ_3 and $k_3 > 0$, yielding

$$\hat{V}(s) = \frac{k_3}{s + k_3} V(s) = \frac{k_3 s}{s + k_3} X(s). \tag{8}$$

Note that the dynamics of the two estimations $\hat{v}$ and $\hat{a}_w$ are completely independent. Furthermore, having available a good estimation $\hat{v}$ allows to estimate also the walker voluntary velocity v_w as

$$\hat{v}_w = \hat{v} - v_c, \tag{9}$$

where v_c is the velocity of the controlled platform. This can be either numerically integrated from (3), or measured from the low-level platform platform controller. The value $\hat{v}_w$ is part of the information passed to the VR visualizer for updating online the virtual scene.

C. Tuning the reference position

The most critical situation for platform control with respect to VR immersiveness is when the user abruptly stops from walking at normal speed. Indeed, since at steady state the treadmill velocity v_c exactly matches (with opposite sign) the user velocity v_w, when the user stops the control must accelerate from $v_c = -v_w$ to zero, while keeping her/him within the platform boundaries. This may require a treadmill acceleration that is incompatible with perceptual constraints. A possible way to overcome this problem is to "virtually" increase the size of the treadmill by changing the reference position x_{ref} (nominally set to zero, i.e., at the platform center) according to the user velocity v_w: the faster the user moves, the more x_{ref} is shifted towards the platform border in the direction of the user motion. Such a behavior can be implemented by defining

$$x_{ref} = k_{ref} \arctan \hat{v}_w, \tag{10}$$

where the arctan function is introduced as a saturation block, and the gain k_{ref} tunes the limits of the saturation action.

D. Closed-loop analysis

The closed-loop behavior of the system can be analyzed in the Laplace domain. To this purpose we linearize eq. (10) around $\hat{v}_w = 0$, without loss of generality. Using (9), it follows

$$X_{ref}(s) = \frac{k_{ref} k_3 s}{s + k_3} X(s) - \frac{k_{ref}}{s} A_c(s). \tag{11}$$

Plugging the control law (3) into the system model (1), and using the Laplace relations (6), (8) and (11), the resulting closed-loop transfer function is obtained as

$$X(s) = \frac{(s + k_3) N(s)}{s\, D(s)} A_w(s) = P(s) A_w(s), \tag{12}$$

with

$$N(s) = \left(s^3 + (k_1 + k_p k_{ref}) s^2 + k_p k_{ref} k_1 s + k_p k_{ref} k_2 \right)$$

and

$$\begin{aligned} D(s) = {} & s^5 + (k_3 + k_1 + k_p k_{ref}) s^4 \\ & + (k_p k_{ref} k_1 + k_v k_3 + k_3 k_1 + k_p + k_2) s^3 \\ & + (k_v k_3 k_1 + k_p k_{ref} k_2 + k_p k_1 + k_p k_3 + k_2 k_3) s^2 \\ & + (k_p k_1 k_3 + k_v k_2 k_3 + k_p k_2) s + k_p k_2 k_3. \end{aligned}$$

The six gains k_i, $i = 1, 2, 3$, k_p, k_v, and k_{ref} can be always chosen so as to guarantee that $D(s)$ has (arbitrary) prescribed roots in the left-hand side of the complex plane. The stability intervals for the gains can be studied with the Routh criterion. It is interesting to note that a pole in $s = 0$ structurally appears in $P(s)$, thus implying that the closed-loop system may not be stable when excited with an arbitrary input signal.

In particular, if the system step response is considered, the system output $x(t)$ may become unbounded. Such a behavior is a direct consequence of having designed x_{ref} as in (10), i.e., as a monotonic function of the walker's own velocity. Indeed, the linearized version of (10) used for deriving $P(s)$ in (12) reduces to a proportional gain between x_{ref} and $\hat{v}_w$, i.e., $x_{ref} = k_{ref}\,\hat{v}_w$. Hence, a constantly accelerating user would result in an unbounded $\hat{v}_w$ signal (the user's own velocity keeps increasing) and x_{ref} would be shifted towards infinity. However, an analysis based on the step response, or any equivalent persistent input, does not reflect the physical constraints of the platform/walker system. In fact:

1) Assuming a persistently accelerating walker is, of course, not realistic. A sounder approach would consider the walker's acceleration as a rectangular signal whose area is bounded by the largest achievable velocity. With this respect, the system closed-loop behavior is more suitably represented by the system impulse response. This yields a bounded behavior for $x(t)$, with $\lim_{t\to\infty} x(t) = x_{ref,\infty} = k_{ref}\bar{v}_w$, being $\bar{v}_w$ the magnitude of the input impulse (i.e., the walker's final velocity after the acceleration phase is over).
2) The saturation effect of (10) is not taken into account by a linear analysis. In particular, the actual arctan function guarantees that, for any value $\hat{v}_w$, $|x_{ref}|$ never exceeds $k_{ref}\pi/2$ so that, even in the case of a hypothetical persistently accelerating walker, $x(t)$ would still be kept bounded within the platform boundaries by appropriately choosing k_{ref}.

In conclusion, stability of the closed-loop system represented by (12) can be guaranteed by restricting the class of input signals to the sole *impulsive* ones, since no persistent walker's acceleration actually reflects the physics of the system under consideration.

E. Rotating the control frame

The control law (3) is most naturally designed in the fixed frame (X, Y) attached to the platform (see Fig. 1). In this frame, the same control gains should be used in the two directions of the plane in order to avoid a bias in the platform motion while re-centering the walker. These gains, however, must be chosen so as to comply with acceleration and jerk constraints set by the limitations of the actuation system, and especially by the physiological comfort of the human walker. The latter results to be particularly critical in view of its dependence on the direction of the walker motion: in fact, physiological studies have clearly shown that human walkers are much more sensitive to accelerations/jerks along the lateral direction (Y_w axis in Fig. 4) than they are along the sagittal direction of motion (X_w axis). As a result of this sensitivity, the experimental validation of control law (3) on the *CyberWalk* platform was characterized by the fact that most users often needed to laterally cross the legs in order to keep balance because of the excessive lateral accelerations/jerks that sometimes annoyed the walking gait (see also the first part of the accompanying video clip).

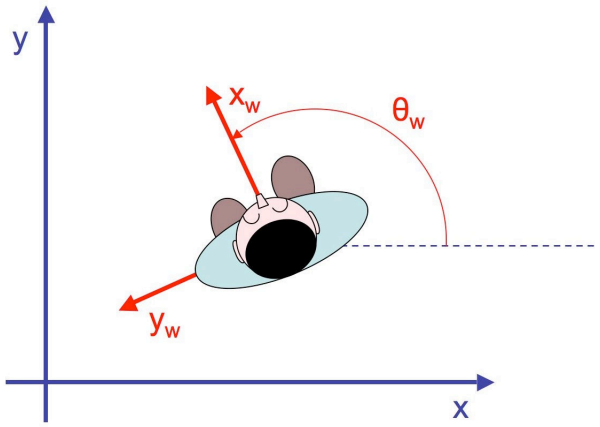

Fig. 4. Definition of a control frame attached to the walker

This problem can be overcome by designing the control law (3) in the frame (X_w, Y_w) attached to the walker. In particular, this allows to choose different control gains along the two current X_w and Y_w axes, namely, larger in the X_w direction of motion and smaller in the side Y_w direction, so as to comply with the different desired bounds on acceleration and jerk. The resulting feedback part of the acceleration commands must be rotated back to the frame (X, Y) attached to the platform, before being provided to the actuation system. From an implementation point of view, this is equivalent to computing (3) with variable (i.e., θ_w-dependent) position and velocity control gains. For example, we will use the position gain matrix

$$K_p(\theta_w) = R(\theta_w) K_{p_w} R^T(\theta_w),$$

with

$$R(\theta_w) = \begin{bmatrix} \cos\theta_w & -\sin\theta_w \\ \sin\theta_w & \cos\theta_w \end{bmatrix},$$

where $K_{p_w} = \mathrm{diag}\{k_{p_{x_w}}, k_{p_{y_w}}\}$ is the chosen diagonal and constant position gain matrix in the frame (X_w, Y_w) attached to the walker. This control frame rotation requires indeed the availability of the walker's body orientation angle θ_w[3].

IV. Experimental results

The controller described in Sect. III has been extensively tested on the *CyberWalk* platform with several different walking users. While the tuning of the reference position (see Sect. III-C) was always included, we present results both without and with rotation of the control frame.

The physical limits of the platform motion are as follows: maximum constant velocity of 1.4 m/s in each direction, with admissible peaks up to 2 m/s; maximum acceleration in the y direction (of the belts) equal to 1.3 m/s^2 and in the x directions slightly larger than 0.25 m/s^2. The control loop

[3]In the reported experiments, we have used instead the information on the walker's head orientation provided by the Vicon motion capture system, under the simplifying assumption that the user is mainly looking in the direction of motion.

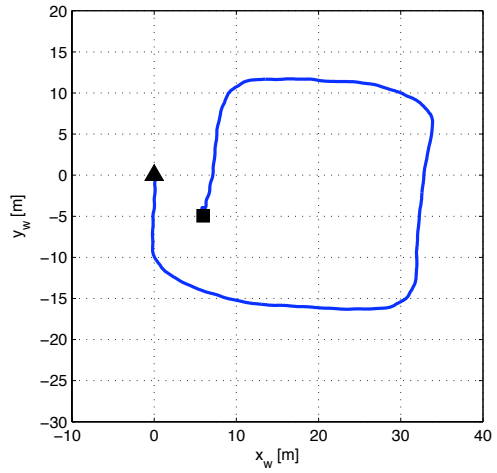

Fig. 5. Experiment 1 (without control rotation): Approximately square path executed by the walker in the virtual world, with starting point at the triangle mark and end point at the square mark

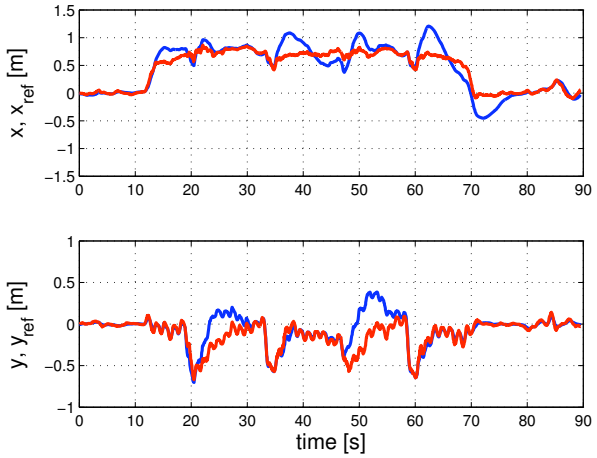

Fig. 6. Experiment 1 (without control rotation): Walker position along the axes of the frame (X_w, Y_w) attached to the walker (in blue); for each coordinate, the corresponding modified reference position is shown (in red)

works at command rate of 20 Hz, while the Vicon motion capture system has a data acquisition rate of 120 Hz.

In the reported experiments, the observer gains were set to $k_1 = 1$, $k_2 = 0.2$, $k_3 = 2.5$, while the reference position parameter was chosen as $k_{ref} = 3/\pi$ in both controlled directions. In the absence of control rotation, the feedback gains were chosen constant and equal along the fixed X and Y axes, i.e., $K_p = k_p I$ and $K_v = k_v I$. The proportional and derivative gains $k_p = 1.4$ and $k_v = 0.5$ in (3) were found acceptable to the users. For the rotated controller, the following gains were chosen along the X_w and Y_w axes attached to the walker: $k_{p_{x_w}} = 0.5$, $k_{v_{x_w}} = 1.4$ for the sagittal motion (the same as before), and $k_{p_{y_w}} = 0.2$, $k_{v_{y_w}} = 1$ for the lateral motion (reduced gains). For a better comparison, in the following all plots of the relevant variables will be represented in the frame (X_w, Y_w) attached to the walker. The three following experiments are shown in the accompanying video.

In Experiment 1, we tested the controller without the rotation strategy. The user starts from the platform center and walks with rather constant speed along the approximately square path in the virtual world shown in Fig. 5. Data are recorded for a total time of 90 s, with the actual walker locomotion starting around $t = 10$ s and ending at $t = 70$ s. Figures 6–8 display the main quantities of interest. In particular, the speed-dependent behavior of the modified reference positions, computed through eq. (10), and their tracking is clearly illustrated in Fig. 6. We note that the commanded platform velocities (bottom of Fig. 8) are almost equal to the opposite of the (filtered) intentional velocity that is being estimated for the walker (top of Fig. 7), resulting in a good motion cancellation at walker speeds up to 1 m/s.

In Experiment 2, a similar walking path was repeated as shown in Fig. 9, but considering now the control rotation strategy, with an actual locomotion time of about 70 s (data recording is extended for 20 s before and after). The

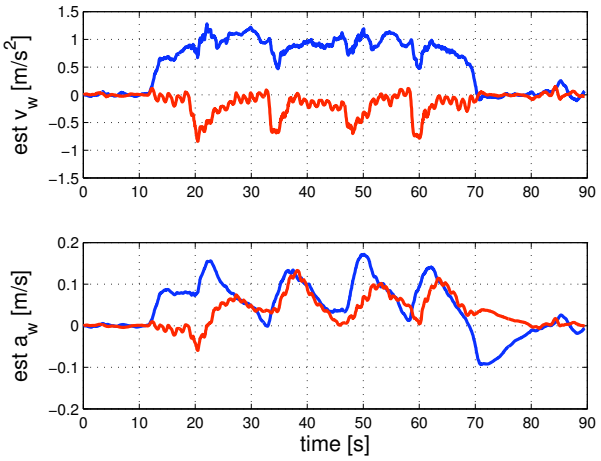

Fig. 7. Experiment 1 (without control rotation): Estimated intentional walker velocities and accelerations in the frame (X_w, Y_w) —x components in blue, y components in red

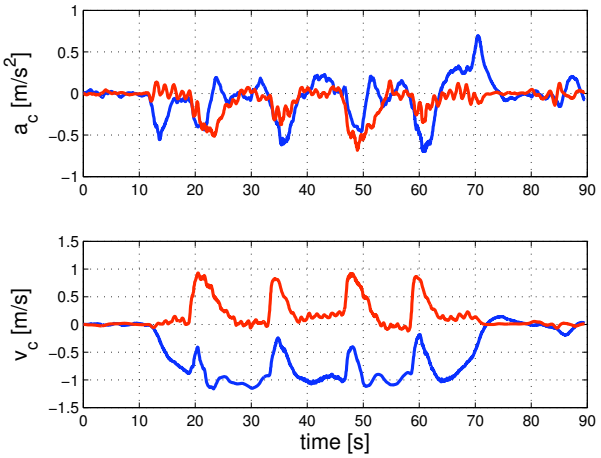

Fig. 8. Experiment 1 (without control rotation): Commanded platform accelerations and velocities in the frame (X_w, Y_w) —x components in blue, y components in red

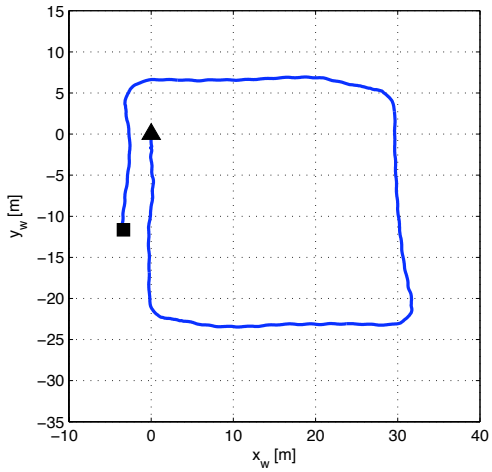

Fig. 9. Experiment 2 (with control rotation): Approximately square path executed by the walker in the virtual world, with starting point at the triangle mark and end point at the square mark

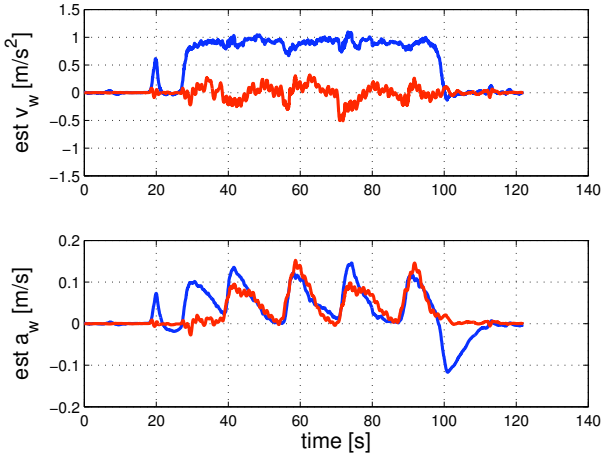

Fig. 11. Experiment 2 (with control rotation): Estimated intentional walker velocities and accelerations in the frame (X_w, Y_w) —x components in blue, y components in red

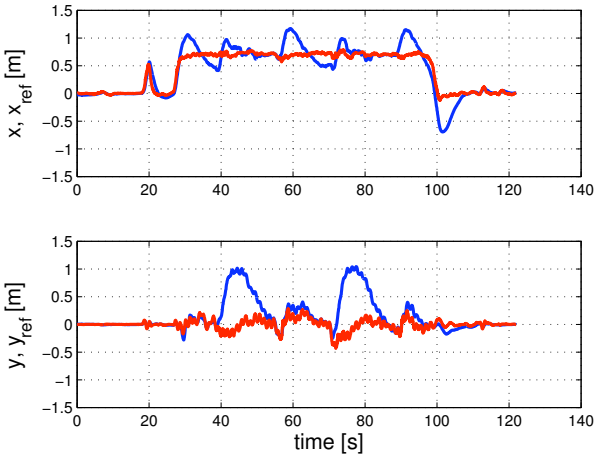

Fig. 10. Experiment 2 (with control rotation): Walker position along the axes of the frame (X_w, Y_w) attached to the walker (in blue); for each coordinate, the corresponding modified reference position is shown (in red)

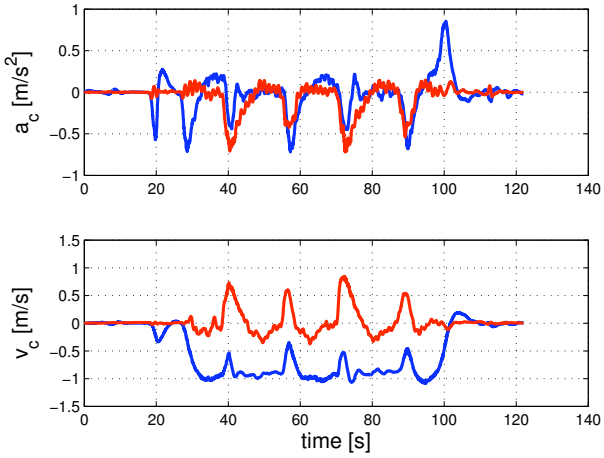

Fig. 12. Experiment 2 (with control rotation): Commanded platform accelerations and velocities in the frame (X_w, Y_w) —x components in blue, y components in red

results are presented in Figs. 10–12, in the same way as for Experiment 1. While the plots appear qualitatively similar in the two cases, a closer look at Fig. 6 in comparison with Fig. 10 reveals longer transients for recovering the moving reference position y_{ref} along the Y_w axis in the second case, because of the smaller proportional gain used in the lateral direction of the walker. Moreover, the estimated walker velocity in the forward motion direction (top of Fig. 11) is closer to being uniformly constant (at 1 m/s) when compared to the unrotated case of Fig. 7. Accordingly, the peak velocities of the controlled platform in the lateral direction of the walker (bottom of Fig. 12) are reduced by more than 25% with respect to those in Fig. 8.

The resulting effects of the control frame rotation on walker locomotion can be even better appreciated from the snapshots in Fig. 13, taken from the accompanying video. Without control rotation the walker tends to cross the legs when turning direction, in order to maintain a more stable dynamic balance. This unpleasant effect vanishes when controlling the platform with selective gains according to the walker orientation. The more comfortable walking experience obtained in this way was confirmed by all tested subjects.

In Experiment 3 we evaluated the performance (including again the control frame rotation) for a more general motion task. The user starts just before $t = 20$ s from the platform center and walks in the virtual world along the erratic path shown in Fig. 14 for about 80 s (recording ends 10 s later). The corresponding variables are reported in Figs. 15–17. Note that, despite the long path traced by the user, the walker position is constantly kept closer than 1 m away from the platform center (Fig. 15). Finally, the commanded accelerations along the X_w and Y_w axes (top of Fig. 17)

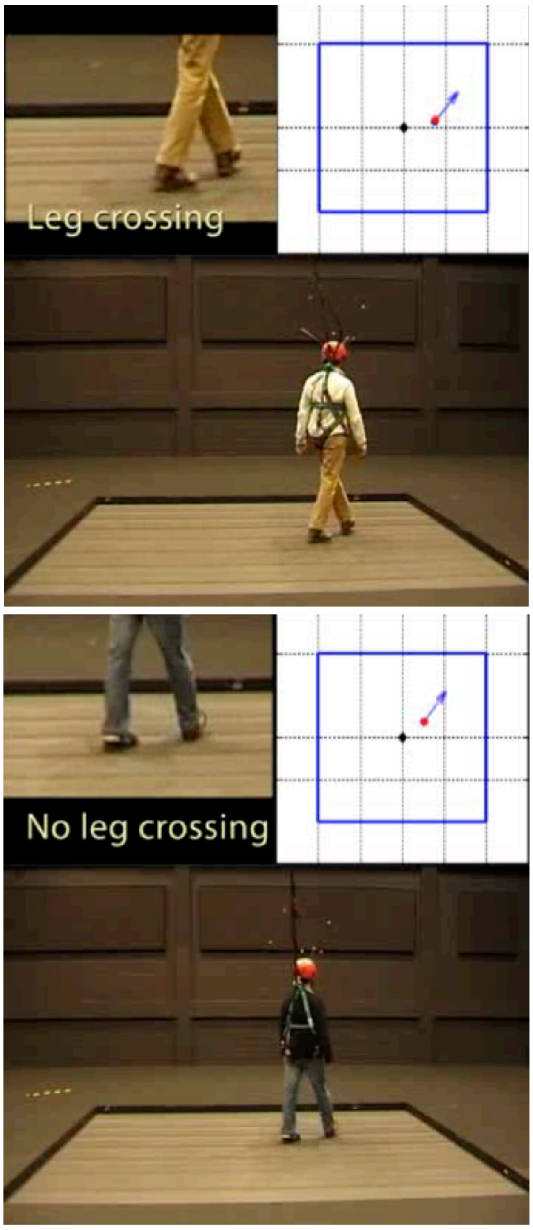

Fig. 13. Comparison of leg crossing (Experiment 1, top) vs. no leg crossing (Experiment 2, bottom) due to the implemented control frame rotation; the small top right window shows the current position and orientation of the walker on the platform

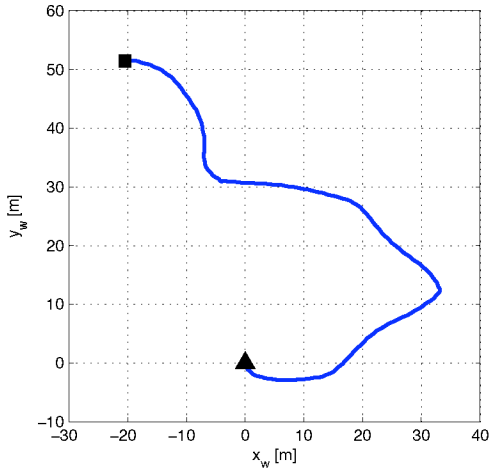

Fig. 14. Experiment 3 (with control rotation): Erratic path executed by the walker in the virtual world, with starting point at the triangle mark and end point at the square mark

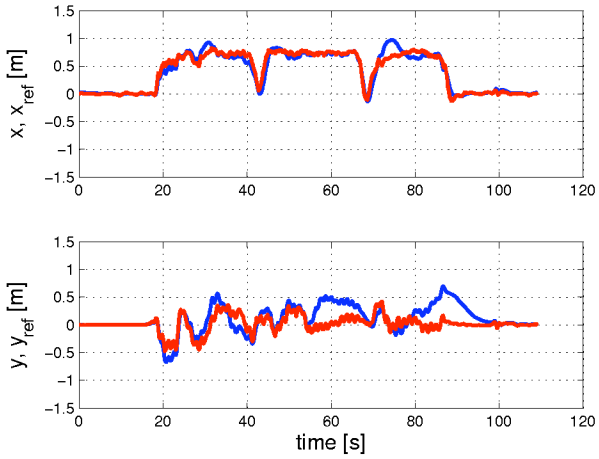

Fig. 15. Experiment 3 (with control rotation): Walker position along the axes of the frame (X_w, Y_w) attached to the walker (in blue); for each coordinate, the corresponding modified reference position is shown (in red)

remain always smaller than 0.5 m/s^2 in absolute value.

V. Conclusions

We have presented the design of a motion control law for a large size 2D platform used for locomotion in virtual worlds. The controller uses two observers to estimate the walker intentional acceleration and velocity, generating acceleration commands along the two actuated directions of the platform with guaranteed stability properties and the capability of smoothly keeping the walker around the platform center. Recovering accelerations due to sudden intentional stops of the user are limited through the definition of a moving reference position that follows in part the walker in motion. In terms of a more natural user's perception, the performance of the controller was improved by tuning the independent scalar control gains in a rotated frame attached to the walker, with axes oriented along the sagittal and lateral body directions.

The platform has been in operation since April 2008, with a successful opening at the final project workshop (see http://cyberwalk.kyb.tuebingen.mpg.de) where many attendees have tested the behavior of the full system under the control laws presented in this paper. Since then, the platform has been used by a hundred of users.

An on-going activity is to combine the current walker localization, obtained using the Vicon system and its targets on the user's head, with that of an overlooking camera system, which would be better able to estimate the orientation of the main body of the walker without additional constraints on the

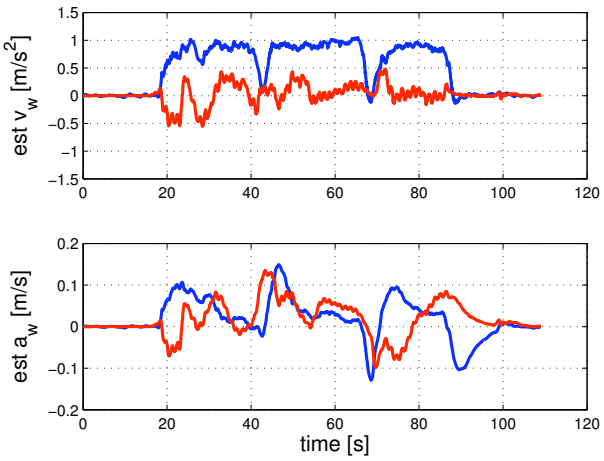

Fig. 16. Experiment 3 (with control rotation): Estimated intentional walker velocities and accelerations in the frame (X_w, Y_w) —x components in blue, y components in red

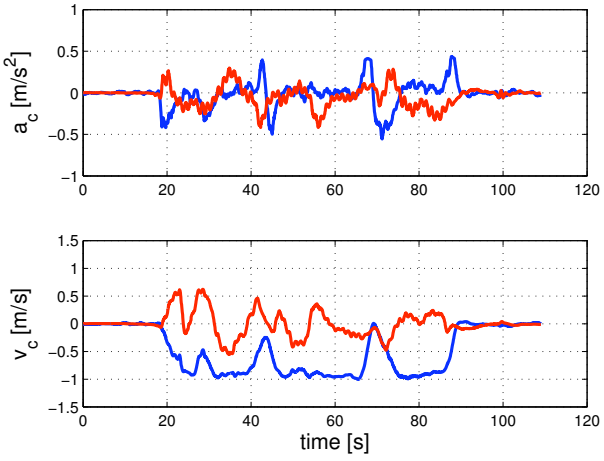

Fig. 17. Experiment 3 (with control rotation): Commanded platform accelerations and velocities in the frame (X_w, Y_w) —x components in blue, y components in red

user. We are also considering to exploit the physiological knowledge of human gait (see, e.g., [18]) to improve the controller performance especially during sharp turns. Also, as a validation of the overall approach, we plan to run specific experiments in which the walker's body will be instrumented with accelerometers. This will allow us to obtain a quantitative measurement of the acceleration values felt by the walker that can be compared to the more qualitative analysis performed in this paper. Finally, perceptual studies are being conducted at MPI on a statistically significant number of users. This may eventually lead to a classification of the best set of control parameters to be used, depending on the walker experience or confidence.

ACKNOWLEDGMENTS

This work was fully funded by the EU STREP project FP6–511092 *CyberWalk*. The authors wish to acknowledge all the colleagues at the MPI for Biological Cybernetics in Tübingen for the support in running the experiments.

REFERENCES

[1] *CyberWalk*, "EU STREP Project FP6–511092, http://www.cyberwalk-project.org", 2005.
[2] H. Iwata, "Locomotion interface for virtual environments", in *Proc. 9th Int. Symp. on Robotics Research*, 2000, pp. 275–282.
[3] J. M. Hollerbach, "Locomotion interfaces", in *Handbook of Virtual Environments Technology*, K. M. Stanney, Ed., pp. 239–254. 2002.
[4] J. M. Hollerbach, Y. Xu, R. R. Christensen, and S. C. Jacobsen, "Design specifications for the second generation Sarcos Treadport locomotion interface", in *Haptic Symposium, Proc. of ASME Dynamic Systems and Control Division*, 2000, pp. 1293–1298.
[5] R. C. Hayward and J. M. Hollerbach, "Implementing virtual stairs on treadmills using torso force feedback", in *Proc. 2002 IEEE Int. Conf. on Robotics and Automation*, Washington, DC, 2002, pp. 586–591.
[6] H. Iwata, H. Yano, H. Fukushima, and H. Noma, "CircularFloor", *IEEE Computer Graphics and Applications*, vol. 25, pp. 64–67, 2005.
[7] A. Nagamori, K. Wakabayashi, and M. Ito, "The Ball Array Treadmill: A locomotion interface for virtual worlds", in *Work. on New Directions in 3D User Interfaces (at VR 2005)*, Bonn, D, 2005.
[8] A. De Luca, R. Mattone, and P. Robuffo Giordano, "The motion control problem for th CyberCarpet", in *Proc. 2006 IEEE Int. Conf. on Robotic and Automation*, Orlando, FL, 2006.
[9] A. De Luca, R. Mattone, and P. Robuffo Giordano, "Feedback/feedforward schemes for motion control of the CyberCarpet", in *Proc. 2006 IEEE Symp. on Robot Control*, Bologna, I, 2006.
[10] A. De Luca, R. Mattone, and P. Robuffo Giordano, "Acceleration-level control of the CyberCarpet", in *Proc. 2007 IEEE Int. Conf. on Robotic and Automation*, Rome, I, 2007, pp. 2330–2335.
[11] R. Darken, W. Cockayne, and D. Carmein, "The Omnidirectional Treadmill: A locomotion device for virtual worlds", in *Proc. Symp. User Interface Software and Technology*, 1997, pp. 213–221.
[12] H. Iwata, "The Torus Treadmill: Realizing locomotion in VEs", *IEEE Computer Graphics and Applications*, vol. 9, pp. 30–35, 1999.
[13] P. Müller, P. Wonka, S. Haegler, A. Ulmer, and L. Van Gool, "Procedural modeling of buildings", *ACM Trans. on Graphics*, vol. 25, pp. 614–623, 2006.
[14] H. Noma and T. Miyasato, "Design for locomotion interface in a large scale virtual environment – ATLAS: ATR locomotion interface for active self motion", in *Proc. 7th Annual Symp. on Haptic Interface for Virtual Environments and Teleoperated System*, 1998, pp. 111–118.
[15] M. Schwaiger, *Konstruktion und Entwicklung omnidirektionaler Laufplattformen* (in German), PhD thesis, Technical University of Munich, January 2008.
[16] S. Hagler, "HMD visualisation of a virtual environment", Tech. Rep., CyberWalk T6.2/D1, April 2006.
[17] A. De Luca, R. Mattone, and P. Robuffo Giordano, "Control implementation", Tech. Rep., CyberWalk T5.3/D1, May 2008.
[18] M. P. Murray, "Gait as a total pattern of movement", *American J. of Physical Medicine & Rehabilitation*, vol. 46, pp. 290–333, 1967.