Understanding, Modeling and Simulating Unintended Positional Drift during Repetitive Steering Navigation Tasks in Virtual Reality

Hugo Brument* Univ. Rennes, Inria, IRISA, France
Gerd Bruder† University of Central Florida
Maud Marchal* Univ. Rennes, INSA, IRISA, Inria, CNRS – France and IUF
Anne Hélène Olivier* Univ Rennes, Inria, CNRS, IRISA, M2S, Rennes, France
Ferran Argelaguet* Inria, Univ. Rennes, CNRS, IRISA, France

Abstract—Virtual steering techniques enable users to navigate in larger Virtual Environments (VEs) than the physical workspace available. Even though these techniques do not require physical movement of the users (e.g. using a joystick and the head orientation to steer towards a virtual direction), recent work observed that users might unintentionally move in the physical workspace while navigating, resulting in Unintended Positional Drift (UPD). This phenomenon can be a safety issue since users may unintentionally reach the physical boundaries of the workspace while using a steering technique. In this context, as a necessary first step to improve the design of navigation techniques minimizing the UPD, this paper aims at analyzing and modeling the UPD during a virtual navigation task. In particular, we characterize and analyze the UPD for a dataset containing the positions and orientations of eighteen users performing a virtual slalom task using virtual steering techniques. Participants wore a head-mounted display and had to follow three different sinusoidal-like trajectories (with low, medium and high curvature) using a torso-steering navigation technique. We analyzed the performed motions and proposed two UPD models: the first based on a linear regression analysis and the second based on a Gaussian Mixture Model (GMM) analysis. Then, we assessed both models through a simulation-based evaluation where we reproduced the same navigation task using virtual agents. Our results indicate the feasibility of using simulation-based evaluations to study UPD. The paper concludes with a discussion of potential applications of the results in order to gain a better understanding of UPD during steering and therefore improve the design of navigation techniques by compensating for UPD.

1 Introduction

The size of the physical workspace in Virtual Reality (VR) setups limits the user's range of motion in the real world while immersed in a virtual environment (VE). A survey made on Reddit four years ago showed that 90% of VR users had a workspace smaller than 3×3 meters.[1] This constraint on the workspace size therefore influences the applications and user interfaces designed by the VR industry. In particular, when users need to navigate in VEs, the use of natural walking is often not possible as it would require to have a workspace that matches the size of the VE. Hence, alternative navigation techniques have been proposed to enable users to navigate in larger VEs regardless of the size of the workspace in the real environment (RE), such as redirected walking, walking-in-place or virtual steering techniques [22].

However, several studies have shown that even navigation techniques that do not require physical movements when used in combination with a Head-Mounted Display (HMD) yield unintended physical movement of the user (i.e. moving physically forward, leftward, rightward, or backward in the workspace while virtually navigating in the VE) [12]. This phenomenon, referred to as *Unintended Positional Drift* [31] (UPD), can be problematic since a user may come near the physical boundaries or obstacles in the workspace without noticing it. However, while previous works have assessed UPD while using Walking-In-Place (WIP) techniques [29,30,32], little is known about UPD during steering navigation in VEs. Yet, UPD could have a negative impact on the user's experience as it can increase the risk of collisions with real obstacles and decrease presence if the application needs to overtly reposition the user in the workspace. Besides, having a better understanding of UPD behaviours has the potential to improve the design of navigation techniques to minimize UPD and avoid or minimize such adverse effects.

In this paper, we propose for the first time an analysis of UPD during a repetitive navigation task using a virtual steering technique. The paper aims at, on one hand, presenting a characterization and models of a user's UPD while navigating in a VE, and on the other hand, an assessment of these models for a given task. Based on experimental data of users performing a slalom task in a VE using a torso-steering technique, we introduce two models (linear regression and Gaussian mixture) encoding the UPD based on the movement of the users. We designed a simulation framework that supports simulations of users navigating and drifting with a steering technique. The purpose of the simulator is to approximate realistic conditions for comparing ground truth data with simulated data. Simulations have been frequently used for assessing new navigation techniques [5,19,43]. These evaluations often rely on simulated user paths to test new redirection techniques. Yet, they often use unrealistic navigation paths such as random walks with perfectly straight path segments and in-place turns, which do not necessarily correspond to ecological navigation performed in VR setups. In this paper, we then reproduced the same experimental task in a simulation framework to assess our UPD models. We showed that our simulator can produce reliable UPD behavior in line with that observed with real users for the given repetitive steering navigation task. The results contribute to the understanding of UPD while navigating in VEs and our methodology could open new perspectives to new designs and assessments of navigation techniques that could reduce UPD.

In summary the contributions of this paper are: (1) Analyses of the UPD during a repetitive virtual navigation task when using a virtual steering navigation technique. (2) UPD models characterizing users' motion in the workspace based on their movements. (3) The design and validation of a virtual simulation system able to reproduce the characterized UPD.

The remainder of this paper is structured as follows. Section 2 provides an overview of related work. In Section 3 we describe our analysis of UPD during virtual steering, based on which we present a basic and a refined model in Section 4. Section 5 presents our simulations based on the UPD models. We discuss our findings, their practical importance, and future works in Section 6. Section 7 concludes the paper.

*e-mail: firstname.lastname@inria.fr
†e-mail: bruder@ucf.edu

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

[1] https://www.reddit.com/r/Vive/comments/4fqq4a/vr_roomscale_room_size_survey_answers_analysis/

2 Related Work

2.1 Virtual Steering Navigation in VEs

Virtual steering techniques are generally decomposed into three components: input, direction and speed. The input mechanism refers to the conditions of input required by the application to determine the navigation state (initiate, continue and stop) [9]. Typically, navigation techniques may require continuous (e.g. a joystick) or binary inputs (e.g. a button). For instance, the motion may be automatic, therefore no input may be required.

The direction is mostly defined by a body segment (e.g. head, hand or torso) or a derivative (e.g. the projection of the head position). There exist some implementations in which the direction of motion is defined by other interaction devices (e.g. a joystick). We refer to [47], which reports most of the existing implementations of virtual steering techniques for navigating in VEs.

The speed component is adjusted through a control law which updates user translation and rotation viewpoints in the VE considering the state of the system. The control law is a key component of steering techniques as it should provide the user the ability to achieve a comfortable navigation speed. Poorly designed control laws can lead to usability issues such as too low speeds resulting in boredom or too high speeds decreasing the control and even potentially generating cybersickness [38]. The control law, in addition to providing a smooth control of the navigation speed, must handle two particular states, namely the beginning and the end of the motion [10]. Control laws generally differ in degree of control, for further information we refer to [2, 12].

In general, steering techniques enable users to easily navigate in VEs while remaining stationary in the workspace. Yet, it is possible that users may experience conscious or unconscious displacements in the workspace while navigating, generating some drift from their initial position. This drift phenomenon remains substantially unexplored.

2.2 Unintended Positional Drift

First introduced by Nilsson et al. [31], UPD, is a phenomenon where users move unintentionally in the RE while using a different technique than real walking (e.g walking-in-place or steering). Although research works explicitly addressing the problem of UPD are rare, the UPD could have a high impact during prolonged VR sessions due to the limited workspace in common VR setups. Based on their observations, Nilsson et al. [31] split the potential approaches to reduce UPD into two categories, either by focusing on the input modalities of the navigation control law or on the physical constraints of the workspace.

First, in [29] authors investigated the impact of input gestures during WIP locomotion on UPD. They compared three different gestures: Marching, *"Wiping"* and Tapping. They differ in terms of how the lower-body parts are moving in order to trigger virtual motion. Their results showed that the gesture had an impact on users' UPD. In particular, tapping (alternately lifting each heel of the ground while keeping the toes in contact with the ground) led to less UPD than classical leg gestures such as marching or wiping. The authors argued that keeping constant contact of the feet with the ground could reduce UPD while using WIP techniques. Nilsson et al. obtained similar results in another study, in which the difference in UPD was probably caused again by the leg movements defined by common WIP gestures [30].

Regarding the physical constraints, a first example is the work by Williams et al. [44], who conducted a study comparing gaze and torso directed WIP. Even though the objective of this study was not to assess UPD, the authors tried to minimize it by placing a 1×1 meter cardboard pad taped on the ground. The authors suggested that the users would be able to remain within the area by relying on the passive haptic feedback provided by the pad. Note that, some steering approaches based on leaning exist to mitigate UPD by constraining user motion [27]. Nilson et al. assessed different modalities for minimizing UPD including different sensory feedback (auditory, visual, audiovisual and passive haptic) [32]. The results showed that both passive haptic feedback and feedback types with gradual onset were the most efficient at reducing UPD, where the passive haptic feedback tended to be more helpful and less distracting than some feedback with a gradual onset.

More recently, other works have shown that the UPD could be influenced by other parameters and other locomotion techniques. For example, in the context of virtual steering, Brument et al. [12] suggested that UPD may exist without demonstrating its presence. Their analyses only considered the movement of physical rotation performed. Even though they noticed an effect of the curvature type of the amount of physical movement done, their results could not conclude on the nature of the UPD for their given experiment.

Finally, Montano et al. demonstrated that the UPD could also occur using scale adaptive techniques (that dynamically adapt a user's displacements) during walking [26]. They noticed that due to the mismatch between the computed position in VR and the position in the workspace, the result of these accumulated differences between the user's physical and (scaled) movements in VR resulted in UPD. They quantified the UPD using scale techniques and proposed a UPD correction model to minimize UPD while walking in VEs. They demonstrated that UPD can be reduced and therefore increase the distance traveled in the VE without being overtly redirected.

One of the main challenges related to the analysis and reduction of UPD, is that for some situations, the UPD appears after a long exposure, which makes the evaluation of UPD correction methods difficult. Yet, recently, a number of works have started to explore the validity of the simulation of user behaviour for the analysis and design of redirected walking techniques.

2.3 Redirected Walking and Simulation Based Evaluations

Redirected walking was firstly proposed by Razzaque et al. [33] and aimed to achieve infinite walking in the VE in a limited physical workspace. By adding imperceptible yaw rotational gains to a user's viewpoint, they could reorient the user to ensure that it remained inside the workspace. This research work was the beginning of a lot of additional research on redirection techniques [21, 28].

However, one challenge of redirected walking is to find the right parameters, such as the gain thresholds, the alignment functions, or redirection strategies among numerous possibilities. While the gold standard assessment remains the conduction of user studies [40], these studies are time-consuming, prone to cybersickness, and are only able to address a reduced number of the dimensions of the control space. These challenges have invigorated the research on simulation-based evaluations. In the context of the COVID-19 pandemic, they are even more relevant.

For instance, simulations have been performed to compare different redirection strategies [19], design new redirection methods [5, 43], optimize obstacle avoidance in constrained workspaces [42, 48], study the size and shape of the workspace [3, 24], or minimize collisions between several redirected users in the same workspace [4, 5, 23]. Where most implementations of redirection techniques rely on human-engineered logic, it is worth noting that recent work showed that machine-learning with simulation approaches could outperform the state of the art steering algorithms used for redirected walking [23, 36, 41].

Although some works have addressed the question of UPD during WIP, little is known about this phenomena when using other virtual locomotion techniques, and in particular virtual steering techniques. In order to limit the potential negative effects of the UPD, first, there is the need to understand the factors that generate such UPD. The following sections describe: (1) how we characterize and model the UPD for a slalom navigation task performed with a virtual steering technique and (2) how we envision the use of such a model for simulation-based evaluations, with the ultimate goal of reducing the UPD.

3 UPD analysis during steering navigation

In this section, we describe how we analysed UPD while using a virtual steering technique to navigate in VEs.

3.1 Navigation Task

3.1.1 Dataset

We obtained and analyzed the dataset from a previously published experiment [12]. Users had to perform a virtual slalom task while wearing a HMD. They were following three different sinusoidal-like trajectories

(with low, medium and high curvature) using a torso-steering navigation technique with three different control laws (constant, linear and adaptive). The virtual movement was provided by pressing a controller's trigger (allowing virtual translation) and the heading direction was provided by the user's torso orientation. The control laws differ by their speed update and the implementations are described in the following research paper [12]. The curvature type of the trajectory and the control law were randomized within-subjects variables. 18 users participated to the user study. We considered the whole dataset for analyzing UPD because we wanted to assess if the UPD could be influenced by the type of trajectory or the control law to update virtual motion in the VE.

During the experiment, users were immersed in the VE with an HTC Vive HMD and two HTC Vive trackers were used to track users' shoulders. The HTC Vive tracking system was used as the reference coordinate system. In the rest of the paper, we define the X axis of the HTC Vive tracking system as the *mediolateral* (ML) axis and the Z axis as the *anteroposterior* (AP) axis. The virtual slalom consisted of a sinusoidal trajectory composed of twelve turns, either starting with a left turn or a right turn. Three different curvature types were used for the slalom, with a sine amplitude of 2 meters. Curvature type was defined by different frequencies f to alter the task difficulty: Small Curvature (SC), $f = 1$; Medium Curvature (MC), $f = 1.5$ and High Curvature (HC), $f = 2$. They were based on curvature chosen in previous work done in REs [7]. The path to follow was indicated through virtual gates ($1 \times 2.3 \times 1$ meters) located at the peaks of the sinusoidal trajectory. Users had to go through the gates to perform the navigation task. There was one block per control law, that consisted of twelve randomized trials (four per curvature type). Users performed six left and six right turns in each trial. In total, each user performed 36 trials (3 control laws $\times$ 12 trials), resulting in 432 turns (36 trials $\times$ 12 turns). The dataset consists of data recorded for each frame of every trial. It contains the frame identifier, the execution time, head and shoulders position and orientation in both VE and workspace.

3.1.2 Pre-processing

In our paper, the UPD is defined as the physical displacement of the user in the workspace while navigating in the VE with a virtual steering technique. To analyse only the UPD behaviour during the continuous trajectory, we removed the first and last turns from each trial as they were respectively the beginning and the end of the slalom in which users were mainly performing rather straight-line trajectories than curvilinear. We considered a turn as a piece of trajectory between two inflection points of the slalom (Fig. 1). To evaluate the UPD over time, we first resampled the dataset, then filtered data with a Butterworth low-pass filter (1Hz cutoff frequency) to remove natural body oscillations when turning [18] and finally temporally normalized them. Users' movements in the workspace were calculated by considering the barycenter of the shoulders' trajectories. We computed the amplitude of users' shoulders as the unwrap arc-tangent function of the ratio between the coordinates of the shoulders' orientation vector in the ML/AP plane (Fig. 1). To evaluate the effect of our independent variables on average kinematics of the trajectories, we performed a two-way analysis of variance (ANOVA) with repeated measures when the distribution of the dependent variables was normal or an Aligned Rank Transformation (ART) ANOVA test if not [46]. To avoid any violation of the sphericity assumption, we used Greenhouse-Geisser adjustments to the degrees of freedom. T-tests with Bonferroni corrections were used as post-hoc analyses. To evaluate the temporal evolution of UPD, we used the Statistical Parametric Mapping (SPM) method [15], which allows to compare time-series data considering the variability over time.

We performed two different analyses to gain a better understanding of how the users drift in the workspace: (1) Global Analysis: We quantified the UPD as the difference between the position of the user at the beginning and the end of the virtual turn. We wanted to confirm first that the users drift after performing one turn. (2) Continuous Analysis: (i) Whole turn: We evaluated the temporal evolution of the UPD during the whole virtual turn by computing the cumulative sum of the UPD both on ML and AP axes. We aimed to assess when UPD occurs within the turn. (ii) Sliding window analysis: We aimed at explaining the UPD by relating the cumulative sum of the UPD on a specific period of time with other variables such as amplitude of the rotation.

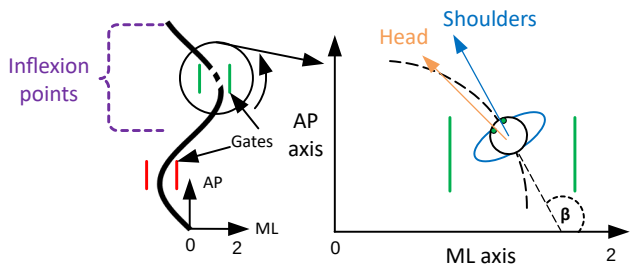

Fig. 1. Left: users performed a virtual slalom by going through gates. A turn is defined as the trajectory between two inflection points. Right: Orientation of body segments in the horizontal plane. Horizontal angles for each body segment are defined as the unwrap tangent function of the ratio ML/AP, where ML and AP represent orientations of the body segment.

3.2 Global Analysis of UPD

For each turn performed, we gathered its first and last point and computed the displacement of users in the workspace on both ML and AP axes. Fig. 2 shows the density map of users' positions at the end of a left turn and a right turn. We can notice that after a left turn, the UPD on the ML axis is negative whereas it is positive for right turns. It means that the users drifted towards the direction of the turn (a virtual left turn provokes displacements towards the left in the workspace and a virtual right turn towards the right in the workspace). The static analysis showed that, on average, users drifted on the ML axis -0.22m ($SD = 0.13$) for left turns and 0.22m ($SD = 0.14$) for right turns (Fig. 3). Regarding the AP axis, it is hard to determine whether users tend to drift rather forwards or backwards. On average, users showed no drift on this axis, even though we can notice in Fig. 2 that there are some points spread along the AP axis.

Statistical analyses confirmed our first observations. First, we observed that the turn number (from the 2nd to the 11th completed in the slalom) did not influence the UPD that remained consistent across the turns. Therefore we aggregated all the turns together. Then we performed a three way ANOVA (control law $\times$ curvature type $\times$ turn side) and we noticed an effect of the turn's direction on the UPD on the ML axis ($F_{1,17} = 73.74$, $\boldsymbol{p < 0.001}, \eta_p^2 = 0.80$), where post-hoc analysis showed that users tend to drift more leftwards (UPD x < 0) during a left turn and more rightwards (UPD x > 0) during a right turn ($\boldsymbol{p < 0.05}$). However, we did not notice an effect of the control law ($F_{1.73,29.34} = 0.28, p = 0.72$) nor the curvature type ($F_{1.55,26.41} = 1.18, p = 0.31$) on the UPD on the ML axis. Regarding the AP axis (Fig. 3), we found an effect of the control law ($F_{1.66,28.20} = 8.48$, $\boldsymbol{p < 0.05}$), where post-hoc analysis showed that users tend to slightly drift more backwards with the adaptive control law than the constant or the linear ones ($\boldsymbol{p < 0.05}$). However, we did not find an effect of the curvature type ($F_{1.24,21.11} = 3.28, p = 0.07$) nor the turn direction ($F_{1,17} = 0.78, p = 0.38$) on UPD on AP axis.

This first analysis showed the existence of UPD when performing a turn in VR while using steering techniques. Next step is then to understand the temporal aspects of the UPD by taking into account the temporal variability.

3.3 Temporal Evolution of UPD

3.3.1 Whole Turn Analysis

Fig. 4 shows the average and standard deviation of the temporal evolution of accumulated UPD on ML and AP axes for both right and left turns, depending on the curvature type and the control law. SPM analysis showed a consistent UPD behaviour with no effect of the control law nor the curvature type on these time-series during the turn. For

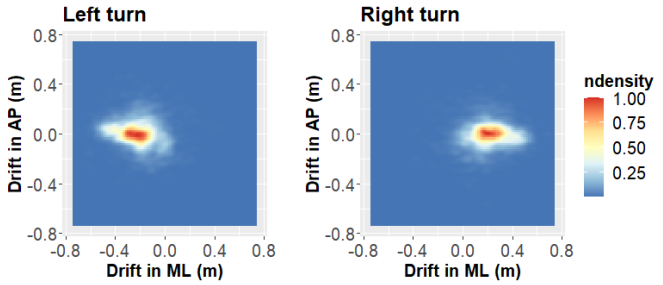

Fig. 2. Density map of users' UPD in the workspace at the end of a left or right turn.

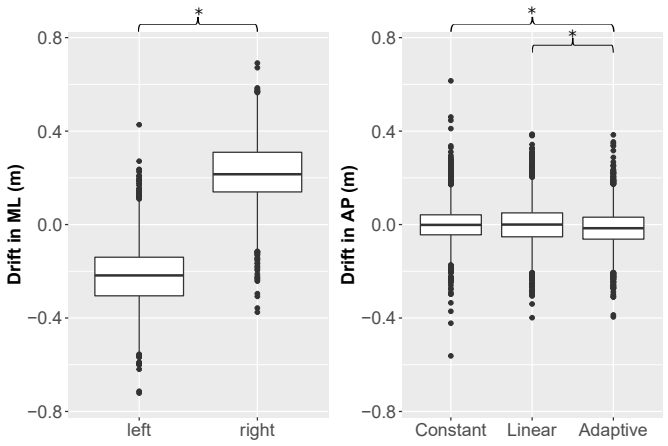

Fig. 3. Boxplots of UPD on the ML axis per turn direction (left, right) and on the AP axis per control law (constant, linear, adaptive). The whiskers indicate pairwise comparisons (* p<0.05).

each sample of the turn trajectory (each turn consisted in 375 samples), the critical threshold of the paired samples t-test statistic was never exceeded, indicating no significantly difference of accumulative UPD across conditions.

Also, it is interesting to notice that the temporal evolution of the accumulated UPD on the ML axis follows a similar pattern for both left and right turns. During the first 40% of the trajectory, the UPD is opposite to the turn direction (i.e. it decreases during the left turn and increases during the right turn) before evolving in the opposite direction from the 40% point until the end of the trajectory (i.e. it increases during the left turn and decreases during the right turn). Regarding the AP axis, we can notice that the average accumulated UPD remains close to 0, but a negative UPD (backwards movement) seems to appear during the turn (40%–60% of the normalized time).

The fact that there are no effects between the experimental conditions would support us in aggregating the data together in order to create our UPD models. In the following subsection, we deepen the analysis in order to get the final results and generate our UPD models.

3.3.2 Sliding Window Analysis

In the previous analyses, we noticed the existence of UPD, and we confirmed that UPD differs according to the turn direction. Considering the UPD as a continuous variable allows us to understand how UPD occurs over time while considering the trajectory variability. For instance, users' amplitude during a turn can vary based on the turn strategy (e.g. regarding the foot placement). Since this information is not available with the global analysis (considering only the first and last data point of a turn), the use of a sliding window is required. In this section, we want to analyze the possible relation between UPD and the difference in the shoulders' amplitude during the turn (i.e. if the amount of turn can explain how users drift). For each turn, instead of computing the difference between the first and the last point of the turn, we computed the difference of amplitude and the UPD between points with different sliding windows. Since the average time of a turn is around 4 seconds, we decided to test windows ranging from 0.5 to 2 with a step of 0.5 seconds. We selected the best dataset for the analysis based on the following criteria: (1) The histogram of the UPD on X follows a normal distribution; (2) The distribution is separated into two clusters based on the amplitude (left and right turns since we noticed a difference between them); (3) The difference of amplitude covers more high values than close to zero values. These criteria are dependent on the task and trajectories we had. It is important to keep in mind that the choice of the sliding window is dependent on the trajectory. In our example, increasing the sliding window helped us discretize the distribution of points. We finally decided to choose the window of 2 seconds as it fulfilled at best the criteria mentioned above.

Then, we plotted the UPD on the ML axis based on the difference of amplitude (all conditions grouped) to determine which type of regression we would use in our UPD models (Fig. 5). We were able to identify different UPD patterns. First, we confirmed that, like users 1 and 2 in Fig. 5, users drifted in general in the turn direction (i.e. UPD < 0 when the difference of amplitude is < 0 and UPD > 0 when the difference of amplitude is > 0). Yet we also had three users (like user 11) in which the patterns were impossible to determine, since they were sometimes drifting either in the turn direction or its opposite. Moreover, one participant (user 18) almost showed no UPD at all (UPD close to 0). Regarding the AP axis (Fig. 6), for both turn directions, the UPD seems either forward or backwards. We checked visually the distribution of the UPD on AP axis, and we noticed that it was following a normal distribution centered on 0.

Based on these observation, we decided to consider the difference of amplitude as our main metric to model UPD and we split the data between the 9 conditions (3 control laws $\times$ 3 curvature type) to see if the patterns would differ across conditions.

4 Generating the UPD Models

Based on visual exploratory analysis, we determined that linear and Gaussian Mixture Model (GMM) were two alternatives that seemed to be well adapted to our data. Our objective was to compare a simple approach (linear model) with a more sophisticated one (GMM) that could both suit to our data. In this section, we describe these UPD models that encompass how users drift in the physical workspace.

4.1 Linear Model

Even though at first sight, the relation between the difference of amplitude Δamp_t and UPD on the ML axis does not seem to be linear, we wanted to investigate whether a first simple approach with a linear regression could be a reasonable approach to model UPD. For each user, condition and turn direction, we computed a linear regression of the UPD for each axis based on the difference of amplitude (Equation 1 for ML axis, Equation 2 for AP axis).

Regarding the ML axis, a 3-way ANOVA (control law $\times$ curvature type $\times$ turn direction) showed that neither the control law ($F_{1.95,33.19} = 1.17$, $p = 0.32$), the curvature type ($F_{1.27,21.56} = 1.95$, $p = 0.17$), nor the turn direction ($F_{1,17} = 0.04$, $p = 0.83$) had a significant impact on the slope. We found similar observations for the intercept. Therefore we compute the parameters of our linear model as respectively the mean of the slope and the mean of the intercept between conditions and users. We used the standard deviation of the slope to model noise in the final UPD output (0.01 for ML and 0.002 for AP). Since the data we gathered is based on the difference of amplitude, information for the [-100,100] range is missing. We hypothesized that the UPD will follow the same behaviour in this range and extrapolated the model. Fig. 7 shows the linear fit for each user. We found $R^2 = 0.25$ for ML fit and $R^2 = 0.1$ for AP fit.

$$ML_t = 0.0016 \cdot \Delta amp_t + 0.002 \quad (1)$$

$$AP_t = 1.80 \cdot 10^{-5} \cdot \Delta amp_t + 0.03 \quad (2)$$

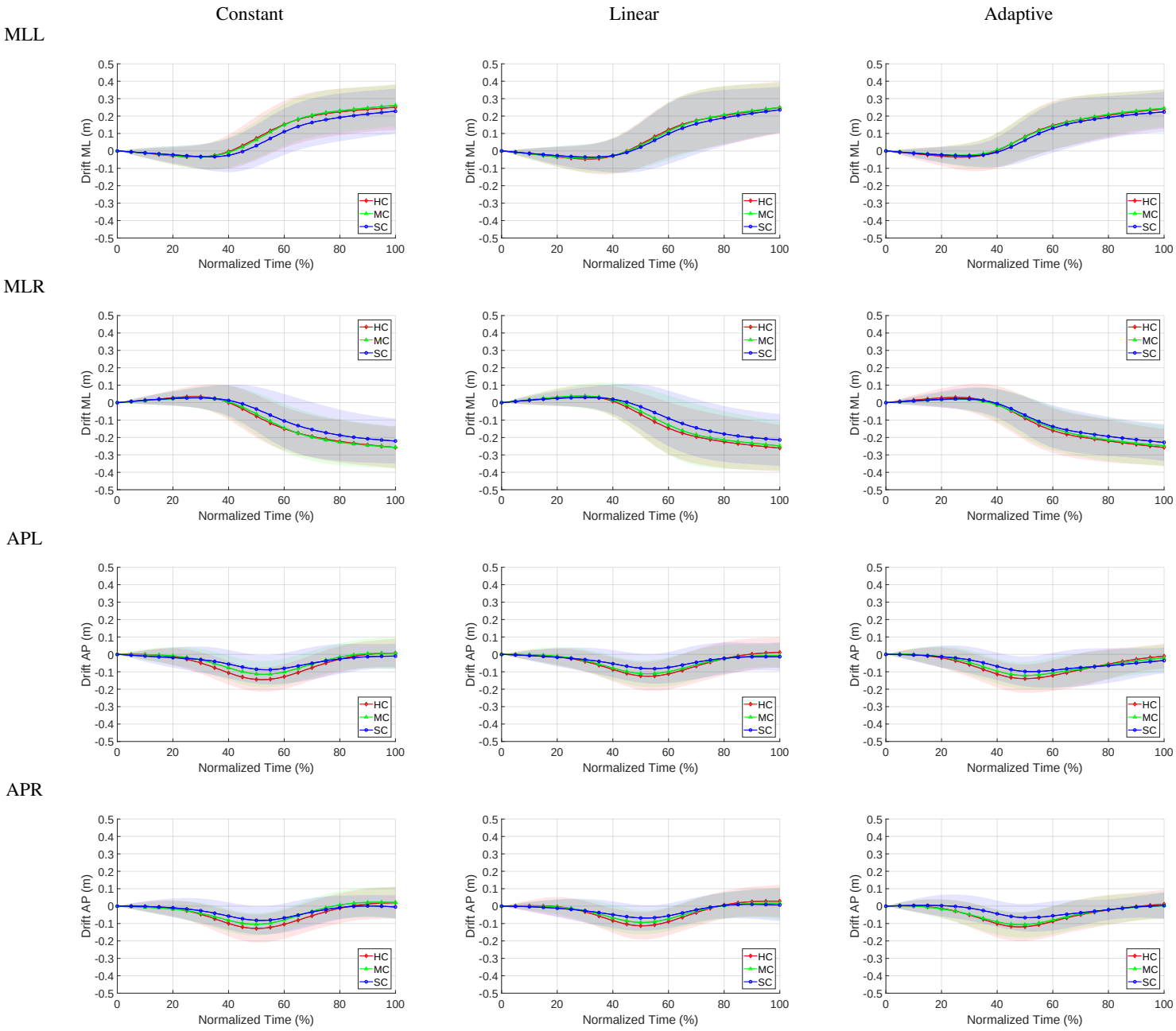

Fig. 4. This figure shows the averaged temporal evolution of means and standard deviations of the accumulated drift on the ML axis for left (MLL) and right (MLR) turns (top rows) as well as on the AP axis for left (APL) and right (APR) turns (bottom rows) for all users. The colors indicate the curvature types: high (HC) in red, medium (MC) in green, and small (SC) in blue. The columns indicate the control laws: Constant in first column, Linear in second, and Adaptive in third column. Each sample of the temporal sequence is a dependent variable (each turn had 375 samples). We found no effect of the control law on the variables during a turn, i.e., the critical threshold of the paired samples t-test statistic was never exceeded.

4.2 Gaussian Mixture Model

Fig. 5 and Fig. 6 show that UPD on both ML and AP axes is divided into two clusters depending on if the difference of amplitude Δamp is either negative or positive. We noticed that the distribution of Δamp, UPD on ML and AP axes followed a normal distribution, therefore we learned a probabilistic model from these data with a Gaussian Mixture Model (GMM) [34]. In particular, we estimated the parameters of the mixtures using the Expectation-Maximization (EM) implementation from the *mixtools* package in R to compute the two mixtures of our multivariate normal distributions [6]. We were able to generate a GMM model giving the probability of UPD on ML and AP axes for each user and condition. Each mixtures provides for the following variables (Δamp, UPD on ML and AP axes) their mean (μ), standard deviation (σ) and covariance matrix (Σ).

Regarding the μ parameter, a 2-way ANOVA (control law $\times$ curvature type) showed that the curvature type had a significant effect on μ_A for both mixtures (left turn: $F_{1.15,6.92} = 1333.58$, $\boldsymbol{p < 0.001}, \eta_p^2 = 0.99$; right turn: $F_{1.16,6.96} = 588.96$, $\boldsymbol{p < 0.001}, \eta_p^2 = 0.99$), where post hoc analyses showed the higher the curvature, the higher μ_a for left turns and the lower for right turns ($\boldsymbol{p < 0.05}$). We noticed the same result for μ_{ML} (left turn: $F_{1.19,7.14} = 5.58$, $\boldsymbol{p < 0.05}, \eta_p^2 = 0.24$; right turn: $F_{1.17,7.03} = 2.95$, $\boldsymbol{p < 0.05}, \eta_p^2 = 0.33$), where the higher the curvature, the higher μ_{ML} for left turns and the lower for right turns ($\boldsymbol{p < 0.05}$). Yet, we did not notice an effect of μ_{AP} for left

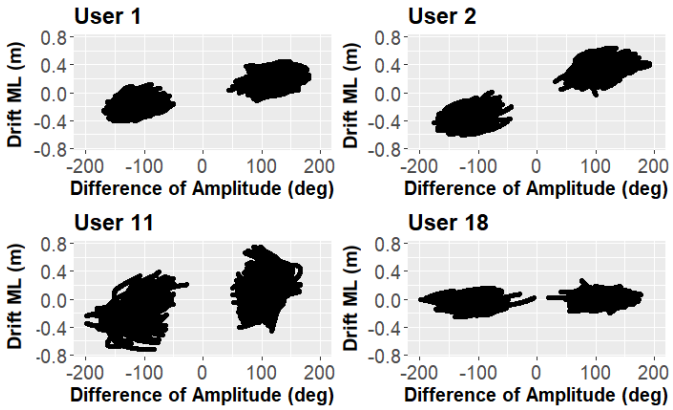

Fig. 5. Selected representative examples of mediolateral UPD patterns based on the difference of amplitude for all conditions. We can identify three different patterns: (1) Users 1 and 2 drift towards the direction of the turn, where user 2 drifted more than user 1; (2) User 11 shows a random drift pattern, sometimes drifting to the opposite of the turn direction; (3) User 18 shows almost no drift.

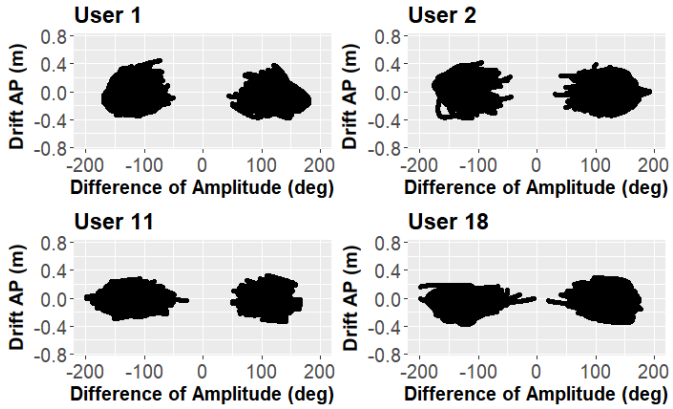

Fig. 6. Examples of anteroposterior UPD patterns based on the difference of amplitude for all conditions.

turns ($F_{1.27,7.64} = 0.19$, $p = 0.73$) nor right turns ($F_{1.57,9.41} = 2.38$, $p = 0.15$).

Regarding the σ parameter, a 2-way ANOVA (control law $\times$ curvature type) showed that the curvature type had a significant effect on σ_A for both mixtures (left turn: $F_{1.41,8.46} = 25.36$, $\boldsymbol{p < 0.01}, \eta_p^2 = 0.80$; right turn: $F_{1.25,7.49} = 22.36$, $\boldsymbol{p < 0.01}, \eta_p^2 = 0.78$), where post hoc analyses showed the higher the curvature, the higher σ_a for left turns and the lower for right turns ($\boldsymbol{p < 0.05}$). We did not notice any effect of the control law or the curvature type on σ_{ML} for both turns. Yet, we did notice an effect of the curvature type for σ_{AP} (left turns: $F_{1.05,6.28} = 8.85$, $\boldsymbol{p < 0.05}, \eta_p^2 = 0.59$; right turns: $F_{1.09,6.54} = 15.19$, $\boldsymbol{p < 0.01}, \eta_p^2 = 0.71$), where post hoc analyses showed the higher the curvature, the higher σ_a for left turns and the lower for right turns ($\boldsymbol{p < 0.05}$).

Since we noticed an effect of the curvature type on the GMM parameters but not the control law, we decided to have 3 GMMs (one for each curvature type) where μ, σ and Σ are computed as the mean of all GMMs generated per user and curvature type. Fig. 8 shows the ellipses computed in our analysis, Table 1 and Table 2 show the GMM estimated parameters for each variable and curvature type for both ML an AP axes. We used the following procedure to compute UPD based on this model. First, we select the best mixtures based on the input Δamp: this can be easily determined by the curvature type of the slalom and the estimated parameters μ_a, where we choose the

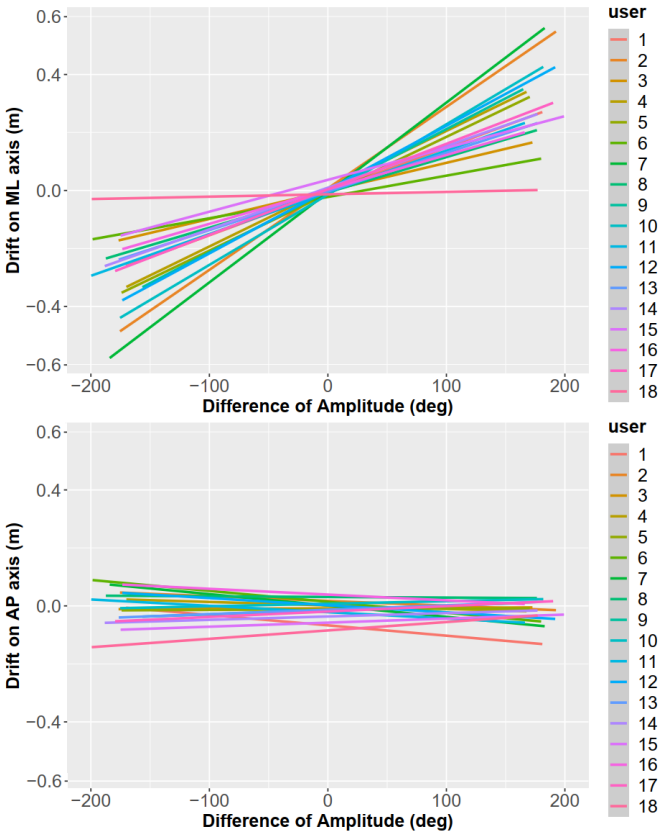

Fig. 7. Linear regression modeling the mediolateral and antero-posterior UPD based on the difference of amplitude. Each regression line corresponds to one user.

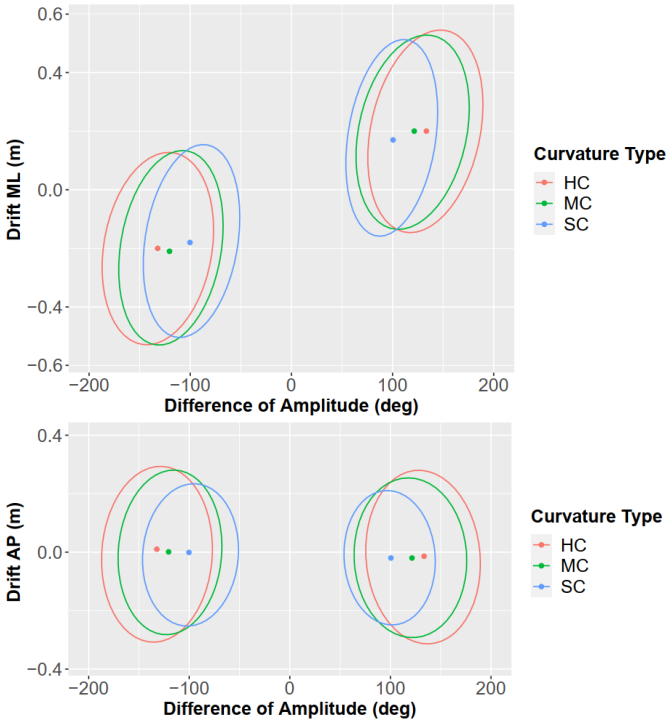

Fig. 8. Ellipses of the mixtures plots fit with GMMs for modeling UPD on the ML and AP axes given the difference of amplitude for each curvature type: high (HC) in red, medium (MC) in green, small (SC) in blue.

Table 1. Mean, standard deviations and covariance matrix of estimated parameters (amplitude, UPD ML axis) from GMMs for both turn directions and each curvature type: high curvature (HC), medium curvature (MC), small curvature (SC).

	Left Turn	Right Turn
HC	$\begin{pmatrix}-131.57\\-0.20\end{pmatrix}$ $\begin{pmatrix}501.03 & 0.70\\0.70 & 0.01\end{pmatrix}$	$\begin{pmatrix}133.03\\0.20\end{pmatrix}$ $\begin{pmatrix}426.83 & 0.61\\0.61 & 0.01\end{pmatrix}$
MC	$\begin{pmatrix}-118.93\\-0.19\end{pmatrix}$ $\begin{pmatrix}390.41 & 0.57\\0.57 & 0.01\end{pmatrix}$	$\begin{pmatrix}120.26\\0.19\end{pmatrix}$ $\begin{pmatrix}413.55 & 0.56\\0.56 & 0.01\end{pmatrix}$
SC	$\begin{pmatrix}-98.38\\-0.17\end{pmatrix}$ $\begin{pmatrix}325.05 & 0.54\\0.54 & 0.01\end{pmatrix}$	$\begin{pmatrix}99.83\\0.17\end{pmatrix}$ $\begin{pmatrix}245.66 & 0.45\\0.45 & 0.01\end{pmatrix}$

Table 2. Mean, standard deviations and covariance matrix of estimated parameters (amplitude, UPD AP axis) from GMMs for both turn directions and each curvature type: high curvature (HC), medium curvature (MC), small curvature (SC).

	Left Turn	Right Turn
HC	$\begin{pmatrix}-132.17\\0.01\end{pmatrix}$ $\begin{pmatrix}78.29 & 0.18\\0.18 & 0.01\end{pmatrix}$	$\begin{pmatrix}133.48\\-0.014\end{pmatrix}$ $\begin{pmatrix}65.56 & -0.2\\-0.2 & 0.01\end{pmatrix}$
MC	$\begin{pmatrix}-120.49\\0.001\end{pmatrix}$ $\begin{pmatrix}56.86 & 0.09\\0.09 & 0.008\end{pmatrix}$	$\begin{pmatrix}121.45\\-0.02\end{pmatrix}$ $\begin{pmatrix}71.35 & -0.08\\-0.08 & 0.006\end{pmatrix}$
SC	$\begin{pmatrix}-100.23\\-0.001\end{pmatrix}$ $\begin{pmatrix}57.06 & 0.07\\0.07 & -0.001\end{pmatrix}$	$\begin{pmatrix}100.53\\-0.02\end{pmatrix}$ $\begin{pmatrix}62.24 & -0.05\\-0.05 & 0.006\end{pmatrix}$

mixture that minimizes $|\Delta amp - \mu_a|$. In case no mixtures correspond, we can extrapolate our model by creating a new multivariate normal distribution by transforming the closest mixtures to Δamp into a new normal distribution with an affine transformation. Yet, this was not required in our model since the implementation of the UPD in our simulator will never face cases like this one. Then, we compute the conditional distribution of UPD (on both ML and AP axes) given Δamp. In a bivariate case where UPD is partitioned into two random variables (in our case A for the amplitude, X for UPD on ML axis and Z for UPD on AP axis), the conditional distribution of UPD X (or Z) given A is computed through Equation 3 and Equation 4. Last, we sample a value from this new normal distribution and we apply the UPD on both ML and AP axes.

$$X|A = a \sim \mathcal{N}(\mu_{ML} + \frac{\sigma_{ML}}{\sigma_A}\Sigma_{A,ML}(a-\mu_A), (1-\Sigma^2_{A,ML})\sigma^2_{ML})) \quad (3)$$

$$Z|A = a \sim \mathcal{N}(\mu_{AP} + \frac{\sigma_{AP}}{\sigma_A}\Sigma_{AP,Z}(a-\mu_A), (1-\Sigma^2_{A,AP})\sigma^2_{AP})) \quad (4)$$

where μ_A, μ_{ML}, μ_{AP}, σ_A, σ_{ML}, σ_{AP} are the mean and standard deviation of the mixtures and $\Sigma_{A,ML}$, $\Sigma_{A,AP}$ respectively the correlation coefficient between A and X or A and Z.

5 Simulating Drift During Steering Navigation

The previous section aimed at characterizing the user's UPD when virtually navigating. In this section, we describe a simulation framework which aims at using such characterization to simulate new navigation data which could be used to assess virtual navigation tasks and serve as a testbed to propose UPD compensation methods. First, we describe the simulation framework, then we analyze how the simulation framework allows to reproduce the navigation task considered in Sect. 3.

5.1 Simulation Framework

Our framework enables simulation of virtual agents navigating through VEs using a virtual steering technique (Sect. 2) for a given navigation task. It is divided into two different components: the navigation task and the agents (virtual and real) behavior. The framework aims at simulating the actual user behavior (real agent) given a constrained virtual navigation task, where control mechanisms enable updating position and orientation of both agents during the simulation.

The navigation task considers the virtual area where the virtual agent will navigate, as well as the actions required to perform it. It can represent for example a user wearing a HMD with a limited tracking space and even physical obstacles inside this area. The framework simulates two agents, one virtual representing its position and its orientation in the VE and a second one representing its position and its orientation in the RE. Both agent behaviors can be independent, can consider the physical drift model and can also consider redirection heuristics. The simulation can be performed at any fixed framerate to be equivalent to VR setups. Position and orientation of the agents should be updated at each frame of the simulation, regarding the navigation task. Any kind of navigation tasks can be embedded (e.g. performing a slalom, finding the exit in a maze, traveling in a virtual forest and pickup objects, etc.). For more comprehensive analysis, the simulator can record several metrics at each frame (e.g. time, distance achieved in VE and RE, position and orientation of agents, linear and angular speed, etc.). The simulator enables to visualize virtual and real trajectories of an agent in real time, but it also allows to run the simulation in test mode, where the rendering pipeline is not including in order to speed up the simulation calculations.

5.2 Simulation of Slalom Task

In order to simulate the slalom task discussed in Sect. 3.1, we considered the following constraints when defining the trajectories of the virtual agent. First, we considered a constant velocity of the virtual agent of 1.3 m/s, which was the mean velocity during the navigation task. Second, we modeled the virtual slalom trajectory with a parametric formulation by using cubic Bézier curves (Equation 5). The virtual agent will follow this trajectory and their orientation will be defined by the tangent at a given position. The trajectories between two gates were computed using the following equation:

$$P(t) = (1-t)^3P_0 + 3(1-t)^2tP_1 + 3(1-t)t^2P_2 + t^3P_3 \quad (5)$$

where P_0 is the previous gate the agent crossed, $P_1 = P_0 + \alpha$, P_2 the next gate the agent should cross, $P_3 = P_2 + \alpha$. To add variability in the trajectories, we used a 2D vector α where $\alpha_x = 0$ and $\alpha_z \sim \mathcal{N}(1, 0.25)$.

Adjustments of real agents' heading direction were based on the pose of the virtual agent. Simulation was performed with a fixed framerate of 90 frames per second (framerate of the HTC Vive used in the user study). We embedded our models (linear, GMM) to simulate UPD of the real agent based on the behaviour of the virtual agent. UPD was simulated with respect to the sliding window we used in our analysis. Therefore, UPD was computed with a two-second sliding window. If the real agent reached the limits of the workspace (2×2 meters, same as the one used in the user study), a reset was performed: we automatically replaced the agent at the center of the workspace. We reproduced the experimental protocol from [12] as described in Sect. 3.1.1 by simulating 18 users performing each condition.

5.3 Analyses of Simulated Data

We used the same methodology as described in Sect. 3 to analyse the simulated data. We estimated linear regression and GMM parameters for each simulated user. Using global analysis (difference between end and beginning of turn), a one way ANOVA with the type of data as dependent variable (simulated linear, simulated GMM or real data) showed no significant effect on UPD on the ML axis after a turn ($F_{1.34,22.82} = 0.15$, $p = 0.77$), but we noticed an effect on the AP axis ($F_{1.13,19.24} = 41.09$, $\boldsymbol{p < 0.001}, \eta_p^2 = 0.51$), where post-hoc analyses showed that the linear model generated on average more backwards drift than the GMM or real ones ($\boldsymbol{p < 0.05}$).

For the linear model, Table 3 shows parameters of estimated linear regressions between simulated and real data. Regarding the UPD on the ML axis, a one way ANOVA with the type of data as dependent variable (simulated or real data) showed no significant effect on the linear regression parameters (slope: $F_{1,17} = 1.73$, $p = 0.20$, intercept: $F_{1,17} = 0.12$, $p = 0.73$). Regarding the UPD on the AP axis, a one way ANOVA with the type of data as dependent variable (simulated or real data) showed no significant effect on the linear regression parameters (slope: $F_{1,17} = 0.001$, $p = 0.97$, intercept: $F_{1.95,33.19} = 2.47$, $p = 0.13$).

For the GMM model, Table 4 shows parameter of estimated mixtures between simulated and real data for both axes. Regarding the μ parameter, a one way ANOVA with the type of data as dependent variable (simulated or real data) showed an effect of type of data on μ_A for both

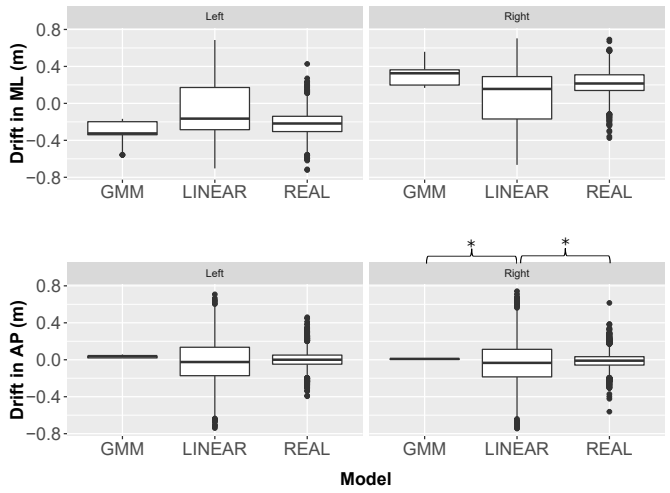

Fig. 9. Boxplots of UPD on the ML and AP axes per turn direction (left,right) for each type of data (simulated, linear, GMM). The whiskers indicate pairwise comparisons (* p<0.05).

Table 3. Average (SD) estimated parameters (slope, intercept) from linear regression for each type of data (real, simulated) on ML and AP drift axes. No significant difference was found.

	X		Z	
	Slope	Int.	Slope	Int.
Real	0.016(0.01)	0.002(0.13)	$1.80 \cdot 10^{-5}$(0.02)	0.03(0.17)
Sim.	0.013(0.01)	-0.002(0.16)	$1.55 \cdot 10^{-5}$(0.002)	0.006(0.05)

Table 4. Averaged estimated parameters ($\mu(\sigma)$) from mixtures (left/right) for each type of data (real and simulated) and drift axes (ML,AP).

	X		Z	
	Real	Simu	Real	Simu
μ_A	-116.33/117.15	-126.63/126.62	-116.34/117.15	-126.63/126.62
μ_X	-0.19/0.19	-0.11/0.11	-	-
μ_Z	-	-	-0.005/-0.02	0.01/0.007
σ_A	292.13/311.03	198.32/198.94	292.15/311.01	198.35/198.99
σ_X	0.008/0.008	0.016/0.017	-	-
σ_Z	-	-	0.009/0.009	0.00003/0.00002

mixtures (left turn: $F_{1,5} = 15.01$, $\boldsymbol{p < 0.01}$; right turn: $F_{1,5} = 12.77$, $\boldsymbol{p < 0.01}, \eta_p^2 = 0.72$), where post hoc analyses showed respectively lower and higher μ_A for left and right turn with simulated data than real data ($\boldsymbol{p < 0.05}$). We noticed the same result for μ_{ML} (left turn: $F_{1,5} = 11.99$, $\boldsymbol{p < 0.05}, \eta_p^2 = 0.70$; right turn: $F_{1,5} = 14.55, \eta_p^2 = 0.74$, $\boldsymbol{p < 0.05}$), where simulated μ_{ML} was lower for left turns and right turns ($\boldsymbol{p < 0.05}$). Yet, we did not notice an effect of μ_{AP} for left turns ($F_{1,5} = 1.74$, $p = 0.23$) nor right turns ($F_{1,5} = 2.05$, $p = 0.20$).

Regarding the σ parameter, a one way ANOVA with the type of data as dependent variable (simulated or real data) showed that the type of data had a significant effect on σ_A of the amplitude for both mixtures (left turn: $F_{1,5} = 37.76$, $\boldsymbol{p < 0.001}, \eta_p^2 = 0.83$; right turn: $F_{1,5} = 55.15$, $\boldsymbol{p < 0.001}, \eta_p^2 = 0.90$), where post hoc analyses showed higher σ_A for real data than simulated ($\boldsymbol{p < 0.05}$). We did notice an effect of type of data on σ_{ML} for both turns (left turn: $F_{1,5} = 165.13$, $\boldsymbol{p < 0.001}, \eta_p^2 = 0.97$; right turn: $F_{1,5} = 297.90$, $\boldsymbol{p < 0.001}, \eta_p^2 = 0.98$), where post hoc analyses lower higher σ_{ML} for real data than simulated ($\boldsymbol{p < 0.05}$). We also noticed an effect of type of data on σ_{AP} (left turns: $F_{1,5} = 9.77$, $\boldsymbol{p < 0.05}, \eta_p^2 = 0.62$; right turns: $F_{1,5} = 14.49$, $\boldsymbol{p < 0.01}, \eta_p^2 = 0.70$), where post hoc analyses showed lower σ_{AP} with simulated data than real ($\boldsymbol{p < 0.05}$).

6 General Discussion

If UPD is accumulated over time, it may become problematic for two main reasons: 1) safety purposes where users could reach the limits of the workspace or collide with obstacles within, 2) repositioning users to the center would break immersion. It is important to be aware of the effects of UPD, to gain a better understanding of it, and to find solutions that could minimize it. In this paper, we aimed to characterize for the first time UPD patterns for a given steering navigation task. We aimed to raise awareness of this yet undisclosed phenomenon and provide first insights into UPD challenges to foster research in this direction among the research community. We presented a methodology to analyse UPD and create UPD models. In addition, we presented first results of simulated UPD with virtual agents. Our results showed the existence of UPD during steering and the possibility to simulate it. Yet, it is important to consider that our approach is a first step towards establishing UPD models for virtual steering navigation, using the virtual slalom task as an example. Therefore, the paper is not providing a generalizable model, but a proof of concept of how to construct and assess models (e.g. the linear and GMM ones we used in our paper).

6.1 Understanding UPD During Steering Navigation

We demonstrated the existence of UPD while using a steering navigation technique. During a navigation task, users may unintentionally move in the workspace without noticing it. In particular, we characterized UPD patterns and noticed that users tended to drift on the ML axis towards the direction of the turn: performing a virtual left turn resulted in leftwards UPD whereas a virtual right turn resulted in rightwards UPD (Fig. 3). Besides, we observed UPD on the AP axis, but we were not able to determine any particular patterns or reasons that could lead to a forwards or backwards UPD. Our initial hypothesis is that UPD during slalom turns may depend on the user's turn strategy.

Performing quick whole body rotations while still maintaining balance and some momentum can be difficult. There exist two main strategies to perform a turn [17]: (1) To turn to the right when the right foot is placed in front, subjects generally altered direction by spinning the body around the right foot (spin turn); (2) To turn left when the right foot is in front, subjects shifted weight to the right leg, externally rotated the left hip, stepped onto the left leg, and continued turning until the right leg stepped in the new direction (step turn). We suggest that the turn strategy influenced UPD on the AP axis: a step turn strategy resulted in backwards UPD and a spin turn in forward UPD. However, it is important to keep in mind that the task was constrained by gates with a given position and orientation.

We hypothesize that UPD may be task dependent, meaning that UPD patterns we observed in the slalom task may differ for other navigation tasks (e.g. free navigation without constraints). Then, considering UPD as a temporal variable (e.g. with a sliding window analysis) may be necessary to understand it across different navigation tasks, since UPD differed with respect to the slalom curvature type. Future work should explore additional navigation tasks (and in particular trajectories with sharp corners that may require manual labelling instead of automatic detection) to compare whether the UPD patterns remain similar or not, but also improve the robustness of the temporal analysis. In addition, recording foot placement could provide additional insights to analyze the footstep strategy and their potential relation to the characteristics of UPD, for example, foot dominance can alter postural balance [1]. One advantage of our analysis methodology is that it can be applied to any kind of navigation techniques requiring less physical movement (WIP, teleport-based). We could therefore imagine in the future studying UPD for any techniques that require "turning-in-place" movements, as this motion may be one major source of UPD.

6.2 Modeling UPD During Steering Navigation

Based on the UPD patterns observed in the considered dataset and visual exploratory analysis, we determined that linear and Gaussian Mixture Models were 2 alternatives that seemed to be well adapted to our data. We presented two models, linear regression and GMM, to encode the UPD on both ML and AP axes based on users' rotations. We suggested that UPD may be related to the amount of physical rotation

performed during the navigation task. The linear model hypothesized that the higher the user's rotation, the higher the UPD. The GMM model considers the marginal distribution of a user's rotation and given UPD observed in real data, generates UPD accordingly. Overall, while these models provide a first relevant approximate of UPD patterns, we are aware that they could be improved.

For instance, the linear model tends to generate more variability than the GMM since the noise added during the simulation was based on the standard deviation of the estimated parameters of the regression, whereas the GMM parameters had very low standard deviations. This suggests that our models are not entirely encoding the subtleties of the real UPD behavior and that additional features should be added to the model. Linear regression was an interesting option to describe overall the UPD patterns without considering intra-individual variability. The GMM model should rather be considered at the individual level. To improve modeling of UPD, other solutions could be explored. Recent research work showed that deep learning or reinforcement learning could be promising avenues to outperform models that rely on human-engineered logic [25, 37]. In addition, other metrics such as path curvatures, angular speed, or additional tracked body parts should be considered in order to improve UPD models. Finally, it is worth noticing that UPD may also introduce intra-subject variability. We tried to find models that explain UPD for a given task based on a visual exploratory analysis. We determined that linear and Gaussian Mixture Models were 2 alternatives that seemed to be well adapted to our data. However, we are aware that these model are not generic to describe UPD in steering navigation. Other type of models could be assessed and individual UPD models for each user could be learnt on the fly while gathering live data in order to have UPD based on user behaviour.

6.3 Simulating UPD During Steering Navigation

Using simulation-based evaluations is interesting for two major objectives: 1) assessing and validating the UPD models and 2) generating and testing hypotheses about physical UPD. Due to the current COVID-19 pandemic, performing in-person user studies can be a major challenge for VR researchers. Thus, simulation-based evaluations can be an interesting alternative to assess and validate research questions as they do not require researchers to perform experiments with real users. In our paper, we used a simulation to compare generated UPD with ground-truth data from an experiment with real users. Results from simulation showed that it is possible to generate similar UPD behaviour observed with real data. However, one main challenge about simulation-based evaluations is to reproduce virtual trajectories that resemble a real user's movements while navigating. Future work should consider how to add variability from the input (trajectories and movements performed by the agents) to the output (variability in the models).

Moreover, the simulation framework could be extended and generalized in order to perform more simulations. For instance, one solution would be to use existing datasets from other studies, analyzing their UPD, creating new UPD characterizations and assessing them through simulation.. Then, the simulation framework would enable the reproduction of user studies just by reproducing the experimental protocols used in the experiments. Still, in-person user studies would be require in future works to gather new datasets about users behavior when performing different tasks with different navigation techniques.

6.4 Reducing UPD During Steering Navigation

Drift from center of the workspace may increase over time and distance traveled in the VE. Then, compensating its effect during steering navigation could be an interesting option in order to keep the user the closest to the center of the workspace. What could be the navigation methods which could decrease UPD? To answer that question, we may take inspiration from existing redirection techniques used while walking. One approach could be to use existing solutions such as the freeze-and-turn resetting [13, 45] or providing a visual or auditory warning feedback to the user. Although these solutions are easy to implement, they may decrease the sense of presence.

Subtle approaches could also be considered. A lot of research work addressed the potential use of gains to redirect users with navigation technique [14, 35]. So far, few works considered reducing UPD with walking by scaling up translation gains based on the discrepancies between real and virtual movements [26] or gesture based techniques [32] trying to optimize legs movements that could reduce drifting forward while using WIP. Using gains with steering techniques could be possible as well. In the example of the slalom task, we noticed that UPD depends on the direction of the turn and the amplitude of body movement. Then, one approach could be to reduce this amplitude in order to reduce UPD. By applying a constant rotational gains on the virtual camera, we could either decrease (if gains > 1) or even increase (if gain < 1) the user's amplitude, resulting in an eventual different UPD. Other types of gains could be considered such as translation [20], curvature [8, 16] or displacement gains [39] that could be adapted to steering.

These suggestions remain hypothetical since we do not know the effect of rotational gains on UPD. They also require prior knowledge of the navigation task. Still, using gains could be promising to control participant orientation in the workspace and find new heuristics minimizing UPD during steering navigation, and consequently reduce workspace required to navigate, as already studied for redirected walking [3, 24].

6.5 Perspectives

The analysis of UPD in this paper provided interesting effects and insights into how users drift while using virtual steering techniques, with practical implications for different applications and potential vistas for future work to compensate for UPD. However, there are also a few limitations of our current work, which may lead to additional research ideas that may be investigated in future work. To improve knowledge about UPD, further analyses should be done using different navigation tasks. For instance, it could be interesting to study eventual link between UPD and other factors such as the sense of presence or cognitive load [11] as they might influence UPD. In addition, other approaches (e.g. machine-learning based, using other metrics) for modeling UPD should be considered in order to improve the existing models. Simulation-based analysis could enable to have insights about UPD by performing less user studies. We intend to improve the simulation framework, by including a more flexible architecture that can be extended by VR practitioners, so that it could become in the future a testbed for studying UPD in VEs. Regarding the use of gains to compensate UPD, future work is required to study the effects of gains on UPD so that they can be used with steering techniques.

7 Conclusion

The analysis of UPD in the literature remains scarce, yet, the practical implications of UPD are diverse, from the breaks of presence that require reset mechanisms and potential dangerous situations that would result from reaching the boundaries of the workspace. This paper has proposed a first characterization of the user's UPD while navigating in virtual environments. This characterization, in addition to shedding insight on this phenomena, aims to enable the simulation of the user's behavior to assess and potentially propose UPD correction methods. Thus, a simulation framework was presented, first to provide a proof of concept of how such a system should be conceived, but also to provide experimental validation of the proposed characterizations. The simulation results showed that the simulation framework was able to reproduce UPD from real user data. However, this is only a first step towards precise characterization of the UPD, as a number of limitations still remain, such as the generalization to other navigation tasks, the consideration of other drift predictors and the personalization of UPD models to account for user variability. This work opens new perspectives about understanding UPD that could become a key component when designing new navigation techniques, as its aftereffects are still unexplored. We believe that the simulation framework could be an efficient tool in order to explore UPD mitigation strategies paving the way for novel navigation techniques able to efficiently reduce the UPD without impacting the users' experience.

REFERENCES

[1] A. C. Alonso, G. C. Brech, A. M. Bourquin, and J. M. D. Greve. The influence of lower-limb dominance on postural balance. *Sao Paulo Medical Journal*, 129(6):410–413, 2011.

[2] C. Anthes, P. Heinzlreiter, G. Kurka, and J. Volkert. Navigation models for a flexible, multi-mode vr navigation framework. *VRCAI*, 4:476–479, 2004.

[3] M. Azmandian, T. Grechkin, M. T. Bolas, and E. A. Suma. Physical space requirements for redirected walking: How size and shape affect performance. In *Proc. of International Conference on Artificial Reality and Telexistence & Eurographics Symposium on Virtual Environments (ICAT-EGVE)*, pp. 93–100, 2015.

[4] M. Azmandian, T. Grechkin, and E. S. Rosenberg. An evaluation of strategies for two-user redirected walking in shared physical spaces. In *Proc. of IEEE Virtual Reality*, pp. 91–98. IEEE, 2017.

[5] E. R. Bachmann, E. Hodgson, C. Hoffbauer, and J. Messinger. Multi-user redirected walking and resetting using artificial potential fields. *IEEE transactions on visualization and computer graphics*, 25(5):2022–2031, 2019.

[6] T. Benaglia, D. Chauveau, D. Hunter, and D. Young. mixtools: An r package for analyzing finite mixture models. *Journal of Statistical Software*, 32(6):1–29, 2009.

[7] D. Bernardin, H. Kadone, D. Bennequin, T. Sugar, M. Zaoui, and A. Berthoz. Gaze anticipation during human locomotion. *Experimental Brain Research*, 223(1):65–78, 2012.

[8] L. Bölling, N. Stein, F. Steinicke, and M. Lappe. Shrinking circles: Adaptation to increased curvature gain in redirected walking. *IEEE Transactions on visualization and computer graphics*, 25(5):2032–2039, 2019.

[9] D. A. Bowman, D. Koller, and L. F. Hodges. Travel in immersive virtual environments: an evaluation of viewpoint motion control techniques. In *Proc. of IEEE Annual International Symposium on Virtual Reality*, pp. 45–52, 215, 1997.

[10] D. A. Bowman, D. Koller, and L. F. Hodges. A methodology for the evaluation of travel techniques for immersive virtual environments. *Virtual Reality*, 3(2):120–131, June 1998.

[11] G. Bruder, P. Lubos, and F. Steinicke. Cognitive Resource Demands of Redirected Walking. *IEEE Transactions on Visualization and Computer Graphics*, 21(4):539–544, Apr. 2015. doi: 10.1109/TVCG.2015.2391864

[12] H. Brument, A.-H. Olivier, M. Marchal, and F. Argelaguet. Does the control law matter? characterization and evaluation of control laws for virtual steering navigation. In *Proc. of International Conference on Artificial Reality and Telexistence & Eurographics Symposium on Virtual Environments (ICAT-EGVE)*, pp. 131–140, 2020.

[13] G. Cirio, M. Marchal, T. Regia-Corte, and A. Lécuyer. The magic barrier tape: A novel metaphor for infinite navigation in virtual worlds with a restricted walking workspace. In *Proc. of the ACM Symposium on Virtual Reality Software and Technology*, pp. 155–162, 2009. doi: 10.1145/1643928.1643965

[14] B. J. Congdon and A. Steed. Sensitivity to rate of change in gains applied by redirected walking. In *Proc. of the 25th ACM Symposium on Virtual Reality Software and Technology*, p. 3, 2019.

[15] K. Friston, J. Ashburner, S. Kiebel, T. Nichols, and W. Penny. *Statistical Parametric Mapping*. Academic Press, 2007.

[16] T. Grechkin, J. Thomas, M. Azmandian, M. Bolas, and E. Suma. Revisiting detection thresholds for redirected walking: Combining translation and curvature gains. In *Proceedings of the ACM Symposium on Applied Perception*, pp. 113–120, 2016.

[17] K. Hase and R. Stein. Turning strategies during human walking. *Journal of neurophysiology*, 81(6):2914–2922, 1999.

[18] H. Hicheur and A. Berthoz. How do humans turn? head and body movements for the steering of locomotion halim hicheur and alain berthoz. In *5th IEEE-RAS International Conference on Humanoid Robots, 2005.*, pp. 265–270. IEEE, 2005.

[19] E. Hodgson and E. Bachmann. Comparing four approaches to generalized redirected walking: Simulation and live user data. *IEEE Transactions on Visualization and Computer Graphics*, 19(4):634–643, 2013.

[20] V. Interrante, L. Anderson, and B. Ries. Seven League Boots: A New Metaphor for Augmented Locomotion through Moderately Large Scale Immersive Virtual Environments. In *2007 IEEE Symposium on 3D User Interfaces(3DUI)*, p. null, 2007. doi: 10.1109/3DUI.2007.340791

[21] E. Langbehn and F. Steinicke. *Redirected Walking in Virtual Reality*, pp. 1–11. Springer International Publishing, Cham, 2018.

[22] J. J. LaViola, E. Kruijff, D. A. McMahan, Ryan P. Bowman, and I. Poupyrev. *3D User Interfaces: Theory and Practice*. Addison Wesley Longman Publishing Co., Inc., 2017.

[23] D.-Y. Lee, Y.-H. Cho, D.-H. Min, and I.-K. Lee. Optimal planning for redirected walking based on reinforcement learning in multi-user environment with irregularly shaped physical space. In *Proc. of IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 155–163, 2020.

[24] J. Messinger, E. Hodgson, and E. R. Bachmann. Effects of tracking area shape and size on artificial potential field redirected walking. In *Proc. of IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 72–80, 2019.

[25] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

[26] R. A. Montano-Murillo, P. I. Cornelio-Martinez, S. Subramanian, and D. Martinez-Plasencia. Drift-correction techniques for scale-adaptive vr navigation. In *Proc.of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST)*, pp. 1123–1135, 2019.

[27] T. Nguyen-Vo, B. E. Riecke, W. Stuerzlinger, D.-M. Pham, and E. Kruijff. Naviboard and navichair: Limited translation combined with full rotation for efficient virtual locomotion. *IEEE transactions on visualization and computer graphics*, 27(1):165–177, 2019.

[28] N. C. Nilsson, T. Peck, G. Bruder, E. Hodgson, S. Serafin, M. Whitton, F. Steinicke, and E. S. Rosenberg. 15 years of research on redirected walking in immersive virtual environments. *IEEE Computer Graphics and Applications*, 38(2):44–56, Mar 2018.

[29] N. C. Nilsson, S. Serafin, M. H. Laursen, K. S. Pedersen, E. Sikström, and R. Nordahl. Tapping-in-place: Increasing the naturalness of immersive walking-in-place locomotion through novel gestural input. In *Proc. of IEEE symposium on 3D user interfaces (3DUI)*, pp. 31–38, 2013.

[30] N. C. Nilsson, S. Serafin, and R. Nordahl. The perceived naturalness of virtual locomotion methods devoid of explicit leg movements. In *Proc. of ACM Motion on Games*, pp. 155–164. 2013.

[31] N. C. Nilsson, S. Serafin, and R. Nordahl. Unintended positional drift and its potential solutions. In *2013 IEEE Virtual Reality (VR)*, pp. 121–122. IEEE, 2013.

[32] N. C. Nilsson, S. Serafin, and R. Nordahl. A comparison of different methods for reducing the unintended positional drift accompanying walking-in-place locomotion. In *2014 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 103–110. IEEE, 2014.

[33] S. Razzaque, Z. Kohn, and M. C. Whitton. Redirected walking. Technical report, Department of Computer Science, University of North Carolina, Chapel Hill, North Carolina, USA, 2001.

[34] D. A. Reynolds. Gaussian mixture models. *Encyclopedia of biometrics*, 741:659–663, 2009.

[35] M. Rietzler, J. Gugenheimer, T. Hirzle, M. Deubzer, E. Langbehn, and E. Rukzio. Rethinking Redirected Walking: On the Use of Curvature Gains Beyond Perceptual Limitations and Revisiting Bending Gains. In *2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 115–122, Oct. 2018. doi: 10.1109/ISMAR.2018.00041

[36] W. Shibayama and S. Shirakawa. Reinforcement learning-based redirection controller for efficient redirected walking in virtual maze environment. In *Proc. of Computer Graphics International Conference*, pp. 33–45. Springer, 2020.

[37] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

[38] R. H. So, W. Lo, and A. T. Ho. Effects of navigation speed on motion sickness caused by an immersive virtual environment. *Human factors*, 43(3):452–461, 2001.

[39] F. Steinicke, G. Bruder, K. Hinrichs, J. Jerald, H. Frenz, and M. Lappe. Real walking through virtual environments by redirection techniques. *JVRB-Journal of Virtual Reality and Broadcasting*, 6(2), 2009.

[40] F. Steinicke, G. Bruder, J. Jerald, H. Frenz, and M. Lappe. Estimation of detection thresholds for redirected walking techniques. *IEEE Transactions on Visualization and Computer Graphics*, 16(1):17–27, 2010.

[41] R. R. Strauss, R. Ramanujan, A. Becker, and T. C. Peck. A steering algorithm for redirected walking using reinforcement learning. *IEEE Transactions on Visualization and Computer Graphics*, 26(5):1955–1963, 2020.

[42] J. Thomas, C. Hutton Pospick, and E. Suma Rosenberg. Towards physically interactive virtual environments: Reactive alignment with redirected

walking. In *Proc. of 26th ACM Symposium on Virtual Reality Software and Technology*, pp. 1–10, 2020.

[43] J. Thomas and E. S. Rosenberg. A general reactive algorithm for redirected walking using artificial potential functions. In *Proc. of IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 56–62, 2019.

[44] B. Williams, M. McCaleb, C. Strachan, and Y. Zheng. Torso versus gaze direction to navigate a ve by walking in place. In *Proc. of the ACM Symposium on Applied Perception*, pp. 67–70, 2013.

[45] B. Williams, G. Narasimham, B. Rump, T. P. McNamara, T. H. Carr, J. Rieser, and B. Bodenheimer. Exploring Large Virtual Environments with an HMD when Physical Space is Limited. In *Proc. of the 4th Symposium on Applied Perception in Graphics and Visualization*, pp. 41–48, 2007.

[46] J. O. Wobbrock, L. Findlater, D. Gergle, and J. J. Higgins. The aligned rank transform for nonparametric factorial analyses using only anova procedures. In *Proc. of the ACM SIGCHI conference on human factors in computing systems*, pp. 143–146, 2011.

[47] M. A. Zayer, P. MacNeilage, and E. Folmer. Virtual locomotion: a survey. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2018.

[48] M. A. Zmuda, J. L. Wonser, E. R. Bachmann, and E. Hodgson. Optimizing constrained-environment redirected walking instructions using search techniques. *IEEE Transactions on Visualization and Computer Graphics*, 19(11):1872–1884, 2013.