

# Effects of Tracking Area Shape and Size on Artificial Potential Field Redirected Walking

Justin Messinger\* Eric Hodgson† and Eric R. Bachmann‡  
Miami University

## ABSTRACT

Immersive Virtual Environment systems that utilize Head Mounted Displays and a large tracking area have the advantage of being able to use natural walking as a locomotion interface. In such systems, difficulties arise when the virtual world is larger than the tracking area and users approach area boundaries. Redirected walking (RDW) is a technique that distorts the correspondence between physical and virtual world motion to steer users away from boundaries and obstacles, including other co-immersed users. Recently, a RDW algorithm was proposed based on the use of artificial potential fields (APF), in which walls and obstacles repel the user. APF-RDW effectively supports multiple simultaneous users and, unlike other RDW algorithms, can easily account for tracking area dimensions and room shape when generating steering instructions. This work investigates the performance of a refined APF-RDW algorithm in different sized tracking areas and in irregularly shaped rooms, as compared to a Steer-to-Center (STC) algorithm and an un-steered control condition. Data was generated in simulation using logged paths of prior live users, and is presented for both single-user and multi-user scenarios. Results show the ability of APF-RDW to steer effectively in irregular concave shaped tracking areas such as L-shaped rooms or crosses, along with scalable multi-user support, and better performance than STC algorithms in almost all conditions.

**Index Terms:** Redirected walking, virtual environments, navigation, simulation

## 1 INTRODUCTION

With the introduction of large area position tracking systems and head-mounted displays (HMDs), navigation through immersive virtual environments (VEs) can be achieved through natural walking. This locomotion interface allows for users to experience appropriate proprioceptive, inertial, and somatosensory cues while navigating through virtual worlds. This rich spatial-sensory feedback can result in a greater sense of immersion for the user compared to other locomotion techniques such as walking in place or flying [31]. Natural walking also has advantages over other techniques in allowing for more effective completion of navigational search tasks [24], and in achieving a lower self reported rate of simulator sickness [8].

The main obstacle to utilizing natural walking as a locomotion interface arises in large-scale VEs, where the VE may be larger than the available physical tracking space. Redirected walking (RDW) addresses this issue by taking advantage of the user's inability to detect subtle manipulations of the correspondence between the virtual world in which they are performing a task and the physical world that they are walking in. For instance, small rotations may be injected or movement may be stretched or compressed so that the user is subtly guided towards the center of the tracking area or

away from tracking area boundaries. Ideally, these manipulations are below known perceptual limits [27], [28], as otherwise the users may perceive the (potentially distracting) redirection. Limiting the redirection results in a minimum walking radius on to which users can be steered in the physical world. This radius determines how much physical space is required to support RDW without excessive interruptions and breaks in presence when a user approaches the boundaries of the physical area. These unsafe situations that can result in a physical collision are typically handled using resetting techniques. Resetting reorients the user away from a boundary or obstacle so that safe navigation can continue [20], [33]. It is desirable to minimize resets, since any task the user is attempting to complete is interrupted during the process, which may result in a reduced sense of presence and a loss of immersion.

As tracking technology and HMDs become more affordable, it will likely be desirable to use other locations such as large rooms, fields or parking areas as tracking spaces. These types of environments may be irregularly shaped, and may contain both static and dynamic obstacles; all of which need to be taken into account in order to avoid collisions and minimize resets. Most generalized redirected walking algorithms do not make use of the tracking area architecture when making decisions on where to redirect the user. Tracking areas without obvious centers may be problematic for techniques such as Steer-to-Center (STC) or Steer-to-Orbit (STO) [22]. This especially includes tracking areas with concave shapes. In addition, RDW typically requires a large tracking area to function effectively [1], so it would be desirable to support multiple simultaneous users to make the best possible use of the available space.

Artificial Potential Field Redirected Walking (APF-RDW) and Resetting (APF-RDW) was introduced in [3]. APF-RDW is a generalized redirected walking algorithm that is designed to handle multiple users and make use of the tracking area shape in generating steering instructions. APF-RDW is based on the use of an artificial potential field consisting of forces associated with boundaries and obstacles (such as other users) that effectively "push" the user away. Live user and simulation studies presented in [3], demonstrate that APF-RDW can outperform STC in a 25m x 44m rectangular tracking area in single user applications and significantly reduce the number of required resets when compared to control (no redirection) for multiple users.

This work presents a revised version of the APF-RDW algorithm that was first presented in [3], and serves as a follow-up study. The revision simplifies individual force vector calculations and enables the algorithm to function effectively in irregular shaped tracking areas. The experimental results include a systematic evaluation of the performance of APF-RDW with respect to tracking area size and shape and number of users, which is compared against STC and no steering control conditions. This systematic evaluation would not have been feasible to perform in a live user study. Instead, simulations were performed utilizing data logs from four previous live user studies [3], [11], [12], [13]. The data presented support conclusions related to expectations of how many users can occupy a tracking area simultaneously while a reasonable user experience is still supported. Specific contributions include:

- Presentation of refined version of a generalized APF-RDW algorithm that enables it to support multiple users in irregular

\*e-mail: messinjf@miamioh.edu

†e-mail: Eric.Hodgson@miamioh.edu

‡e-mail: Eric.Bachmann@miamioh.edu

concave tracking areas.

- Experimental examination of scaling steering rates based on the proximity of obstacles that indicates the technique increases RDW performance without significantly increasing steering rates.
- Experimental examination of the impact of tracking area size that indicate performance gains for APF-RDW relative to control and STC in a wide selection of tracking area sizes.
- Experimental results indicating that APF-RDW can support significantly larger numbers of users than control and STC in multi-user applications.
- Experimental results indicating that APF-RDW can function effectively in irregular concave tracking areas.

The remainder of this document is organized as follows: Section two presents background on redirection techniques in redirected walking, detection thresholds for the manipulation of gains, multi-user redirected walking research, and previous investigations into room shape. Section three describes in detail a refined APF-RDW algorithm and the changes made to better support different shaped tracking areas. Section four provides a description of the experiments used to evaluate the APF-RDW algorithm. This is followed by a presentation of the experimental results and a discussion that includes guidelines that the data supports.

## 2 BACKGROUND

Suma et al 2012 gave a taxonomy of different redirection techniques used in the literature [29]. Techniques in the taxonomy are organized based on their perceptibility to the user, whether changes are applied discretely or continuously, and whether they manipulate position or orientation. For instance, approaches that utilize overlapping rooms [30] or modify virtual architecture [32] take advantage of subtle discrete reorientation. Since overt techniques can contribute to more breaks in presence than subtle techniques [29], subtle redirection techniques are typically preferred. However, occasionally subtle techniques are not sufficient to keep users within the boundaries of the tracking space, at which point an overt technique may be necessary. Examples of overt techniques include methods that reposition users through the use of portals [7], [26], and procedures that reorients users through resets [3], [20], [33].

The first instance of redirected walking was introduced as a subtle continuous reorientation technique, whereas subjects were steered away from the boundaries of the tracking area to complete a task in a VE that was twice as large [23]. This work was expanded upon in a seminal work, where Razzaque proposed three steering strategies: steer the user to the center of the tracking area (STC), steer the user in an orbit around the tracking area center (STO), and steer the user to multiple targets in the tracking area (Steer-to-Alternating Targets) [22]. Other redirection methods may use information about the VE to predict possible paths and avoid potential collisions [9], [34]. These methods have achieved some success in reducing the frequency that resets need to be performed, with regards to the mean amount of redirection applied [11].

In order for users to not notice the injected rotation, they must be convinced that their motion is the result of themselves moving (self-motion), rather than the result of objects around them moving (external motion) [22]. The goal for redirected walking is to manipulate visual and audio cues subtly enough that they remain consistent with the user's vestibular and proprioceptive cues, and appear below perceptible thresholds. Steinicke et al identified separate redirection thresholds for translating, rotating, and curvature gains [27], [28]. These thresholds have also been measured in several other studies [5], [10], [16], [25], with curvature threshold in particular ranging from 6.4m to 22m. These disparities may be a result of numerous factors, including walking speed [21], optical

flow [6], and even estimation methodology [10]. Higher rates of redirection can impose cognitive demands on the user [5], which could justify using lower redirection rates depending on the task.

Overall, these human threshold studies suggest that considerable space is needed to utilize RDW effectively. Because of the size needed, there has been recent interest in extending RDW techniques to support multiple users. The first study of multi-user redirected walking was conducted by Holm et al. [15]. This study used a technique called Steer-to-Offset Centers, which gives each user a different redirection target that is offset from the center of the tracking area. This method was paired with a technique to predict potential collisions between users, and steer users away if a collision is imminent. In this study, tracking area boundaries were not considered when steering users away from potential collisions and the approach was found to be intractable with more than two users. In [2], Azmandian et al. expanded on Holm's work and added two modifications to overcome previous limitations: a more tractable relative velocity heuristic, and the ability to resolve situations where both users are required to stop through the use of resets. To our knowledge, only one study on redirected walking has investigated the effects of tracking area size and shape [1]. Through simulations, it was found that 6m x 6m is the minimum area in which redirection techniques are effective. The study also determined that STC and STO perform best in square tracking areas as opposed to elongated rectangles. All shape testing in the simulation experiments was performed using standardized 400 sq meter rectangular tracking areas having one of three different aspect ratios (1.0, 1.5 or 2.0). Both [1] and [2] utilized a computer simulated model of a walking user and computer generated virtual paths. Neither study took into account tracking equipment noise, user walking patterns, and gazing behaviors that can introduce differences between the simulations and live user studies.

## 3 REFINED ARTIFICIAL POTENTIAL FIELD REDIRECTED WALKING

APF-RDW is designed to be adaptive to the shape of the tracking area and to effectively support multiple users. In APF-RDW, each obstacle or boundary exerts a repulsive force on the user. These forces are summed to determine a safe direction in which the user is steered. Hoffbauer first documented APF-RDW and resetting (AFP-R) algorithms in a MS thesis [14]. The algorithms are also described in [3] along with additional experimental results. Like the APF-RDW algorithm described in [3], the algorithm that is the subject of this work uses an artificial potential field of forces with magnitudes that are inversely proportional to the distance of the user from the obstacle. The rotation calculation used to steer users in a safe direction is represented by the sum of the force vectors. However, the AFP-RDW algorithm has been refined since the original work was completed. The changes relate to how the individual force vectors are calculated and applied. They have resulted in simplified calculations and have made the algorithm capable of dealing effectively with irregularly shaped concave tracking areas. The most significant difference is the sub-division of large obstacles into numerous small obstacles that are each handled separately. In the following sub sections, the revisions to APF-RDW are described in detail.

The resetting algorithm used in this study, Artificial Potential Field Resetting (APF-R), is unchanged from the that presented in [3] and for purposes of brevity is not described. The resetting algorithm utilizes the direction of the total force vector at the start of the reset as the desired final orientation after the reset is completed. During a reset, the user is turned in the direction of the largest angle between the starting orientation and the total force vector, and the virtual rotation gain is adjusted so that the user completes a turn of 360° in the virtual world. For complete details consult [3].

### 3.1 Total Force Vector

The total force vector,  $\mathbf{t}$ , is the sum of individual repulsive force vectors associated with fixed obstacles and tracking area boundaries,  $\mathbf{w}_i$  and other users within the tracking area,  $\mathbf{u}_j$ . It is calculated as

$$\mathbf{t} = \sum_{i=0}^n \mathbf{w}_i + \sum_{j=0}^{m-1} \mathbf{u}_j \quad (1)$$

where  $n$  is the number obstacle segments and  $m$  is the number of users in the tracking area. While individual force vectors are calculated differently depending on whether they are associated with fixed obstacles or another user, in both cases the magnitude of the force is inversely proportional to the vector difference,  $\mathbf{d}_i$ , between the the position of the user,  $\mathbf{p}$ , and the center of an obstacle segment or the position of another user,  $\mathbf{c}_j$ .

$$\mathbf{d}_i = \mathbf{p} - \mathbf{c}_i \quad (2)$$

Thus, the length of the vector sum,  $\mathbf{t}$ , will increase as the user approaches obstacles and decrease as they move away from obstacles into a relatively open area.

### 3.2 Fixed Obstacles Repulsive Force Vectors

In this work, fixed obstacles such as walls that define tracking area boundaries are sub-divided into smaller obstacle segments as shown in Figure 1. For each obstacle segment, a perpendicular unit length vector,  $\mathbf{n}$ , indicates the side of the segment that faces into the tracking area. The individual force vector associated with each obstacle segment is given by

$$\mathbf{w}_i = \begin{cases} CL_i \frac{\mathbf{d}_i}{\|\mathbf{d}_i\|} \frac{1}{\|\mathbf{d}_i\|^\lambda}, & \text{if } \mathbf{n} \cdot \frac{\mathbf{d}_i}{\|\mathbf{d}_i\|} > 0. \\ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, & \text{Otherwise.} \end{cases} \quad (3)$$

where the constant  $C$  is a tunable scaling factor for adjusting the relative strength of obstacle and other user repulsive forces,  $L_i$  is the length of obstacle segment  $i$ , and  $\lambda$  is the obstacle fall-off factor.

The inclusion of the segment length,  $L_i$  in equation (3) remedies a limitation of the previous APF-RDW algorithm where all walls exert the same force, regardless of the length of the wall. In that implementation, if one wall is segmented into three separate walls due to a small step-out, then those three walls would exert three times as much force as an unsegmented wall of the same length. Instead, in this implementation, all walls are divided into short segments. The fall-off exponent  $\lambda$  in equation (3) enables tuning to control how fast the force of walls falls off with distance. In the equation the second case will be satisfied in concave tracking areas when an obstacle segment is facing away from the user. This criteria ensures that such segments are not included in the force calculation.

### 3.3 Other User Repulsive Force Vectors

Similar to [3], individual repulsive force vectors are calculated for each of the other users in the tracking area. The calculation takes into account the distance between the users as well as the relative movement directions of the users. Unlike equation (3), all users are assumed to be the same size and thus a segment length is not included. The revised user force equation is

$$\mathbf{u}_j = \kappa \frac{\mathbf{d}_i}{\|\mathbf{d}_i\|} \frac{1}{\|\mathbf{d}_i\|^\gamma} \quad (4)$$

where  $\gamma$  is a tuning factor that allows the falloff rate of the force with distance to be adjusted in a manner similar to  $\lambda$  in equation (3).  $\kappa$  scales the force and is based on the average cosines of the

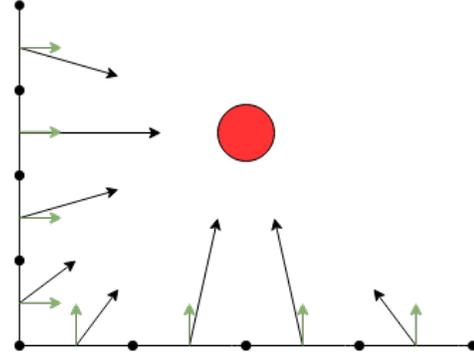


Figure 1: Force (black) and normal (green) vectors for obstacle segments for a user position (red dot).

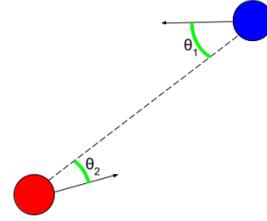


Figure 2: Relative heading angles,  $\theta_1$  and  $\theta_2$  (green arcs), for two users (blue and red dots). User movement directions are represented by black arrows. Reproduced from [3]

angle between each user's movements and the line segment between players in same way as described in [3] and is given by

$$\kappa = \text{clamp}\left(\frac{\cos \theta_1 + \cos \theta_2}{2}, 0, 1\right) \quad (5)$$

The angles used in the calculation are depicted in Figure 2.

### 3.4 Proximity-Based Steering Rate Scaling

APF-RDW is capable of taking into account the geometry of the tracking area as well as the positions of other users. Increases in the length of  $t$  indicate that obstacles or other users are in close proximity. Additionally, the algorithm can easily track the distance to the nearest obstacle. The walking threshold steering rate is based on the linear velocity of the user,  $v$ , and the radius of the arc onto which the user is being guided,  $r$ , is given by

$$h = 360 \times \frac{v}{2 * \pi * r} \quad (6)$$

In [3], this steering rate is scaled based on the ratio of the current length of the total force vector and the observed average length of the total force vector. In this work the steering rates are linearly increased when it is determined that the nearest obstacle cannot be avoided using the arc based steering rate. When steering rate scaling is enabled, the distance to the nearest obstacle is continuously monitored. If the distance to the nearest obstacle,  $m$ , is closer than the radius of the walking arc,  $r$ , then  $h$  is increased through linear interpolation up to a max value,  $M$ . The applied steering rate becomes

$$\text{appliedSteeringRate} = (1 - t)(h) + t(M) \quad (7)$$

where the parameter,  $t$ , used in the linear interpolation is

$$t = 1 - \frac{m}{r} \quad (8)$$

Table 1: Constant Parameter Values

Constant	Value	Constant	Value
$C$	0.00897	$L_i$	1 m
$\lambda$	2.656	$\gamma$	3.091
$r$	7.5 m	$M$	15°/sec.

The application of equation (7) may increase RDW rotation rates above imperceptible levels. While this can be seen as undesirable, possible noticeability of the increased rates must be weighted against the even more obtrusive break in presence that occurs when a reset must be performed to avoid a collision.

### 3.5 Parameters used in this Study

Numerous constants appear in equations (3), (4), (6), (7), and (8). The values of the constants used in this study are presented in the table 1. The values associated with imperceptibility were drawn from previous work by other researchers in this area [27], [28]. The values of constants that are specific to the APF-RDW algorithm were determined by conducting 100 simulation multi-user trials using random values. The set of random values that resulted in the best performance in terms of number of resets and distance between resets are those that appear in the table. Optimization of these values is a topic of future work.

## 4 SIMULATION EXPERIMENTS

### 4.1 Source Paths for Simulations

A total of 288 logged paths were taken from four previous user experiments [3], [11], [12], [13] which served as navigational data for the simulations. All previous experiments used a backpack rendering system, (Alienware Aurora M9700 [11], [13]; Dell M4500 [12]; Alienware 13 R3 [3]), an aluminum pack frame, and an HMD (Nvis SX60 [11], [13]; NVIS SX111 [12]; Oculus Rift CV1 [3]). Head position was tracked using the Worldviz PPT for all experiments. Head orientation was measured via an InterSense InertiaCube 2+ in [11], [12], [13], and via the Oculus Rift CV1 in [3]. These logs recorded the paths of live users performing a variety of search tasks in both open VEs (i.e., a forest) and constrained VEs (e.g., a grocery store with fixed aisles). In all cases, the user's virtual position and virtual orientation were recorded at an rate of 60 Hz (with some slower sampling depending on the specific experiment, momentary rendering demands, and the equipment used). These virtual paths represent each user's desired movement through the VE, and include a variety of individual differences in walking speed, decisiveness, stopping and starting, and spontaneous changes in direction. In each case, different types of redirection can be applied to derive the corresponding physical paths needed to travel this virtual path. To construct the physical paths, each simulated user was given a physical starting position, and their subsequent linear and angular velocity was used to calculate the type and amount of redirection that could be applied during each update. For example, a user who is standing still and looking around would have low linear velocity but high angular velocity, and could be redirected with rotation gains during that period. The direction and magnitude of redirection was determined by the algorithm being simulated. The need for boundary resets was determined by the parameters of the room being simulated, as described below.

### 4.2 General Simulation Methods

Four different RDW algorithms were considered: Steer-to-Center (STC), Artificial Potential Field without Scaling (APF-U), Artificial Potential Field with Scaling (APF-SC), and no steering (Control). The difference between APF-U and APF-SC is that the latter takes into account ones proximity to walls and other users, and scales the current rate of redirection up to the maximum imperceptible

levels as users draw nearer to an obstacle. It should be noted that APF-SC is the canonical version of APF-RDW, as its entire purpose is for walls and obstacles to repel the user, thus allowing it to take advantage of odd shapes and sizes. The un-scaled version, APF-U, is an intentionally simplified version of APF that was included for the sake of comparison to STC, and does not include the proximity-based scaling described in Section 3.4. Specifically, APF-U and STC both use the base steering rate  $h$  in Equation (6). In the case of a square or round room, the APF steering vector should always point to the center of the tracking space in a square or round room with a single user, making it functionally identical to STC. Thus, these two algorithms should perform indistinguishably under these conditions, but differ with multiple users or oddly-shaped rooms. The full implementation of APF (APF-SC), on the other hand, should show an advantage over STC in all conditions, as it inherently accounts for room shape, additional users, and obstacle proximity.

In the experiments below, two sets of physical rooms were simulated. In the first, shape was held constant and the room size was varied to test RDW algorithm performance in spaces of either 10x10m, 15x15m, 20x20m, 25x25m, 30x30m, 40x40m, or 50x50m. While it is possible to implement specially crafted RDW in smaller spaces commonly seen with consumer VR systems [18], generalized RDW is typically not recommended due to the high number of boundary resets. 10x10m VR laboratories are similarly undersized, but are relatively common in the research space and were thus chosen as the low end of the spectrum. In each of these square rooms, the four RDW algorithms were tested in both single user and multi-user scenarios. Single users were always started in the center of the tracking space facing North. For each tracking area and RDW algorithm, simulated redirection of all 288 logged users was performed.

The live user data underlying these simulations were collected in a space that is 45m across in its longest dimension, and is one of the largest facilities of its kind, making 50m a reasonable upper limit. Excessive tracking jitter in the recorded path could cause the simulated user to leave the tracking area causing APF algorithms to push users even further out of the tracking space as the walls repel them. In such cases, the trial was omitted. Despite the excessive jitter present in some of the logged data, the studies that had subjects complete simulator sickness questionnaire (SSQ) suggest that simulator sickness was generally not a factor. For the studies that did not use a SSQ, rates of reported simulator sickness were comparable to other redirected walking studies, except for [12] which reported a 31.9% incidence rate of simulator sickness. In this particular study, it was concluded that this abnormally high simulator sickness rate was due to the high global visual flow and reported feelings of claustrophobia due to closeness of objects in the virtual environment (aisles in a grocery store), which are known to contribute to sickness in users [17], [19].

For multiple users, users were staggered diagonally (NW - SE) 3m apart, and all facing North. The first user started in the center and additional users were added to available starting positions nearest the center. This arrangement placed an upper limit on the number of users that could be started in a given room (e.g., 3 users in a 10x10m room, 5 users in a 15x15m room, etc.), which in turn served as an upper limit on the simulated multi-user scenarios, but this upper limit was not always reached. Users were added until each user averaged more than 1 reset per minute. If this threshold was reached for a single user, then no multi-user trials would be run. The threshold of 1 reset / min is arbitrary, but was chosen to be a reasonable upper threshold for interrupting users. For each room size, algorithm, and number of users, recorded paths were selected randomly to create 500 unique redirected path combinations. If any trial failed due to the tracking jitter issue noted above, the trial was replaced with a new random assortment of users.

The second set of rooms simulated held size constant but varied

the room shape. Four rooms were constructed to be 1000m<sup>2</sup>. Shapes included a 2:1 rectangle, a trapezoid, a cross, or an L-shape. In these odd-shaped rooms, only single-user scenarios were simulated. For the 2:1 rectangle and cross, the center was used as the target for STC. However, since the centers for the L-shape and trapezoid were more disputable, we choose the center target to be a point that maximized the distance from the closest wall. For each room shape and algorithm, all 288 logged paths were redirected. Paths that failed due to tracking jitter were removed.

In all simulated conditions, a boundary reset was triggered if a user traveled within 1m of a wall or 2m of another user. During resets, travel updates from the log files were paused and users were rotated at a constant 15°/s until completion of the reset. Any resets that had originally occurred during the live study, causing the user to stop, turn, and restart their travel, were removed from the log files prior to the simulation to create an uninterrupted virtual path. Under all conditions, simulated resets were performed using the APF-R resetting algorithm described in [3].

### 4.3 Hypotheses and Measures

For each simulated trial, data was logged with the number of wall resets per user, the total distance traveled (in meters), and the average rate of redirection (°/s). For multi-user trials, the number of user resets and total resets was also recorded. Distance travelled was divided by the total number of resets to derive a metric of the average amount of distance users could travel between resets with each method.

The following trends were expected:

- *All conditions:* The un-steered Control condition will provide a worst-case baseline of how often un-corrected users collide with walls or each other.
- *Varied Room Sizes - Single User:* APF-U and STC should perform equivalently. APF-SC should generate fewer resets and longer distances traveled by reacting to nearby walls and temporarily increasing steering rates.
- *Varied Room Sizes - Multi User:* Here, APF-U should begin to outperform STC, since it accounts for other users and STC does not. APF-SC should provide the fewest resets and longest distances travelled.
- *Varied Room Shape - Single User:* STC should perform nearly as well as APF-SC and APF-U in open spaces like the rectangle or trapezoid, but will struggle in concave spaces like the cross or L-shape. APF-SC should perform well across all room shapes. APF-U should permit users to make more use of the available space than STC, but may be less effective than APF-SC in tighter portions of the rooms.

## 5 RESULTS

Results are presented below for ANOVAs to determine the interplay between room size (or shape), steering algorithm, and the number of concurrent users when applicable. Group distributions and residual plots were visually inspected for anomalies to ensure that the assumptions of an ANOVA test were met. Due to the high number of samples in the simulations (e.g., 7301 simulated users in the single-user analyses), any numeric differences tended to attain statistical significance at the  $p < .0001$  level or greater. For this reason, the discussion below focuses largely on descriptive statistics, graphical presentation, and effect sizes to tease out *practical* significance of any differences between conditions in addition to the statistical significance. Effect sizes are listed in  $\eta_p^2$ . As a general rule of thumb,  $\eta_p^2 = .01$  is considered a small effect,  $.09$  is considered a medium effect, and  $.14$  is considered a large effect.

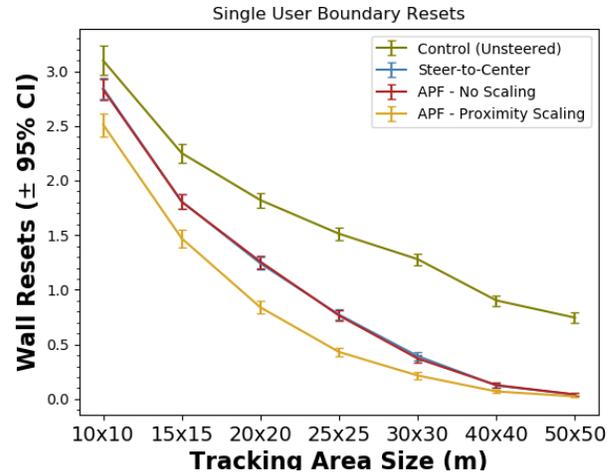


Figure 3: Number of wall resets per min (during a 4 min walking task) for simulated users who are redirected by each algorithm in square rooms of different sizes.

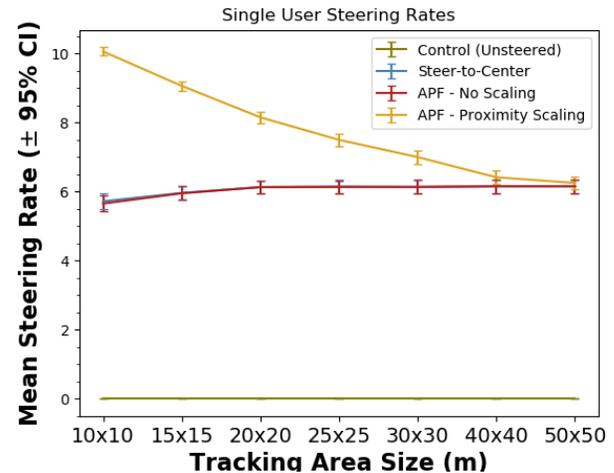


Figure 4: Mean rate of redirection for each RDW algorithm for single users in square rooms of different sizes.

### 5.1 Varied Room Sizes - Single Users

Figure 3 shows the mean number of wall resets experienced by single users for each algorithm and room size. The control condition, which utilized resets at tracking boundaries but did not attempt to steer the user in between, provides a good baseline of the number of resets that could be expected if users' navigation is left uncorrected. All three steering algorithms improve on this baseline by redirecting users. As predicted, APF-U and STC perform nearly identically, as each pushes or pulls the user towards the center of the tracking space with equal strength. APF-SC shows a distinct advantage with smaller rooms, reducing the number of resets by an extra 0.35 resets per min in 10m - 25m spaces on average. This represents an extra 11 - 23% reduction in expected resets relative to the control condition, which is enough to be practically significant. Given enough space, however, all three steering algorithms converged and were equally effective. There was little difference in a 40m square, in which STC and APF-U reduced the number of resets by an average of 86.1% relative to the control condition and APF-SC reduced resets

by 92.3%, an inconsequential difference of 0.06 resets per min. By 50m, the difference was negligible with reductions of 94.5% and 96.9%, respectively.

A 4 (RDW algorithm) X 7 (room size) ANOVA was conducted to gauge the performance of the different algorithms in square rooms of various size in terms of wall resets, steering rate, and distance traveled between resets. For wall resets, there was a strong main effect of RDW algorithm ( $F(3, 7273) = 1138.58, \eta_p^2 = .32, p < .0001$ ), as described above. Planned contrasts confirmed that there was no measurable difference between APF-U and STC ( $p = .813$ ). There was also a large main effect of room size ( $F(6, 7273) = 3809.75, \eta_p^2 = .76, p < .0001$ ), with fewer resets in larger spaces for all algorithms, as one might expect. Finally, there was a weak algorithm x room size interaction ( $F(18, 7273) = 16.70, \eta_p^2 = .04, p < .0001$ ), driven by slightly different trends in each algorithm (e.g., a more linear decrease in the control condition compared to tapering and converging trends in the others).

Steering rates are illustrated in Figure 4. The Control condition was excluded from this ANOVA, since its steering rate was fixed to be zero and could not vary during navigation. In the other three conditions, there was a large main effect of algorithm ( $F(2, 5622) = 733.61, \eta_p^2 = .21, p < .0001$ ), reflecting a generally higher rate for APF-SC than APF-U and STC. Planned contrasts indicated that the latter two did not differ ( $p = .803$ ). There was also a weak effect of room size ( $F(6, 5622) = 101.79, \eta_p^2 = .04, p < .0001$ ), as rates changed slightly across different room sizes. More interestingly, the rates changed in different directions with different algorithms, yielding a significant interaction between algorithm and room size ( $F(12, 5622) = 198.55, \eta_p^2 = .14, p < .0001$ ). For APF-U and STC, which were nearly identical, the rates held fairly steady, but tended to be slightly lower in smaller spaces. This is likely due to the higher number of resets experienced in smaller spaces, as simulated users spent an increased amount of time facing the center after a reset - a period in which steering is reduced as no course correction is needed. APF-SC, on the other hand, experienced higher levels of steering in smaller spaces. Because it is designed to increase steering strength based on object proximity, these higher rates are naturally encountered more often with smaller spaces and decrease as more area is available.

Results for the average distance traveled between resets mirrored those of the number of resets. Users were generally able to travel further in larger rooms regardless of algorithm, yielding a large main effect of room size ( $F(6, 7273) = 1235.70, \eta_p^2 = .51, p < .0001$ ). Because APF-SC was more successful at reducing resets, users traveled further than their counterparts in APF-U or STC, yielding a large main effect of algorithm ( $F(3, 7273) = 563.46, \eta_p^2 = .19, p < .0001$ ). An algorithm x room size interaction was also observed, as the differences between algorithms decreased in larger room sizes ( $F(18, 7273) = 72.87, \eta_p^2 = .15, p < .0001$ ).

## 5.2 Varied Room Sizes - Multiple Users

When evaluating the algorithms with multiple users, one point of emphasis was to determine how many users could be supported by each algorithm in a given space. As described in the methods section above, the simulation in each room was re-run with additional users until the total number of resets (wall resets + user resets) exceeded an average of 1 reset per min per user. These results are shown in Figure 5. For small rooms, up to 15x15m, even single users experienced more than 1 reset per minute with every RDW algorithm. APF-SC was the lone exception in a 20m square space. In larger rooms, the two APF algorithms strongly differentiate themselves from STC and the control condition. In a 50x50m room, both were able to redirect as many users as would fit in the available starting positions (i.e., 16 users). APF-SC generated an average of 0.66 resets per min per user under these conditions, while APF-U generated 0.90 resets per min. By contrast, STC and control were unable to support more than 4

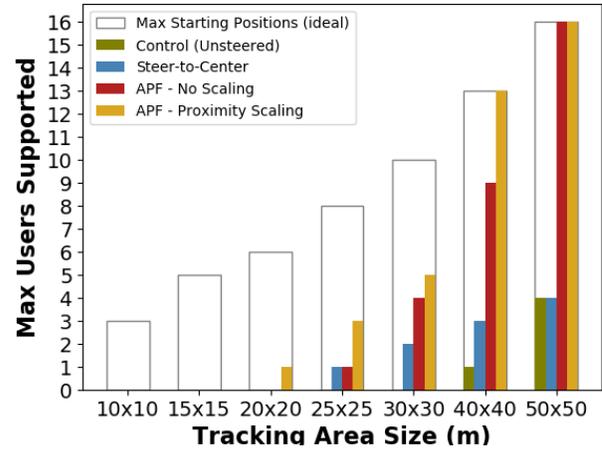


Figure 5: Maximum number of users that could be redirected in a given space without exceeding one reset per minute (including both wall resets and user resets). The empty bars show the total number of potential starting spaces for users to occupy.

users in a 50m space.

Reset data was analyzed in a 4 (algorithm) x 7 (room size) x 16 (number of users) ANOVA with the number of users as a random factor. There was a predictably large main effect of room size ( $F(6, 21.47) = 779.59, \eta_p^2 = .995, p < .0001$ ) and number of users ( $F(15, 23.03) = 13.89, \eta_p^2 = .90, p < .0001$ ). That is, total resets increase when the room size is reduced (extra wall resets) or when there were more users to bump into (extra user resets). There was also a large main effect of algorithm, with a clear rank ordering of the algorithms ( $F(3, 39.75) = 955.08, \eta_p^2 = .99, p < .0001$ ). APF-SC performed best, then APF-U, then STC, and finally Control. All two-way and three-way interactions were also significant ( $p < .0001$ ), with the exception of the algorithm x number of user interaction ( $p = .163$ ). This was driven by the relatively larger increase in resets for STC and Control as room size and numbers of users increased.

The same analysis was conducted with average steering rates, excluding the zero-rate Control condition. There was a large main effect of algorithm ( $F(2, 22.86) = 252.36, \eta_p^2 = .96, p < .0001$ ), as well as room size ( $F(6, 41.3) = 111.74, \eta_p^2 = .94, p < .0001$ ). Notably, there was no significant main effect of the number of users ( $F(15, 18.22) = 0.81, p = .653$ ). However, there was a significant interaction between algorithm and the number of users ( $F(20, 22.33) = 9.96, \eta_p^2 = .90, p < .0001$ ), as well as between algorithm and room size ( $F(12, 41.84) = 105.27, \eta_p^2 = .97, p < .0001$ ). These effects both appear to be driven by a small increase in APF-SC steering rates for more users, with increased numbers of users being more likely in larger rooms. No other interactions were significant. Although the effect of increased rates is statistically strong, it is important to keep in mind that these effects are quite small in a practical sense, around  $1^\circ/sec$ , and still below noticeable thresholds.

The average distance that users could travel between resets is shown in Figure 6 for the three largest rooms, along with the associated steering rates. Data for smaller rooms is not illustrated since these generally did not support more than one user. Distance traveled per reset showed a strong main effect of algorithm ( $F(3, 30.15) = 47.81, \eta_p^2 = .83, p < .0001$ ), with APF-SC affording users the most travel, followed by APF-U, STC, and then Control. There was also an expected main effect of room size ( $F(6, 17.42) = 292.25, \eta_p^2 = .99, p < .0001$ ), with longer distances traveled in larger rooms. Similarly, there was a predictable main effect of the number of

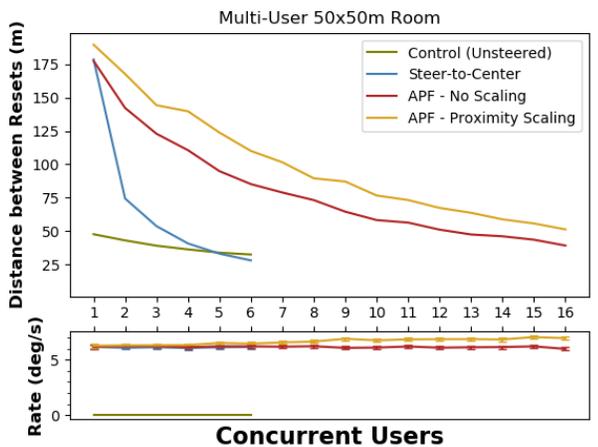
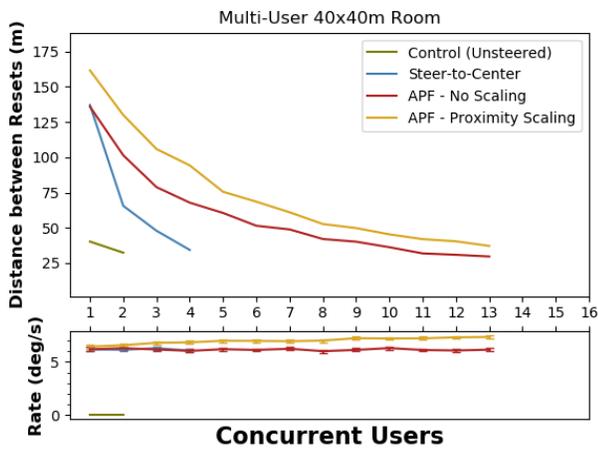
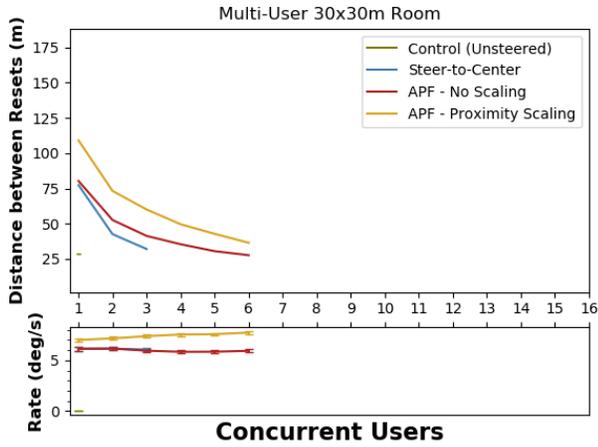


Figure 6: Mean distance traveled between resets with each algorithm for different numbers of users and different room sizes. Note that the rate of redirection is relatively constant as more users are added, and that STC quickly struggles with additional users.

users ( $F(15, 16.6) = 33.83, \eta_p^2 = .97, p < .0001$ ), with less distance traveled per reset as the space became more crowded. There was a significant room size  $\times$  algorithm interaction as the algorithms became more differentiated in larger rooms ( $F(18, 26.47) = 8.45,$

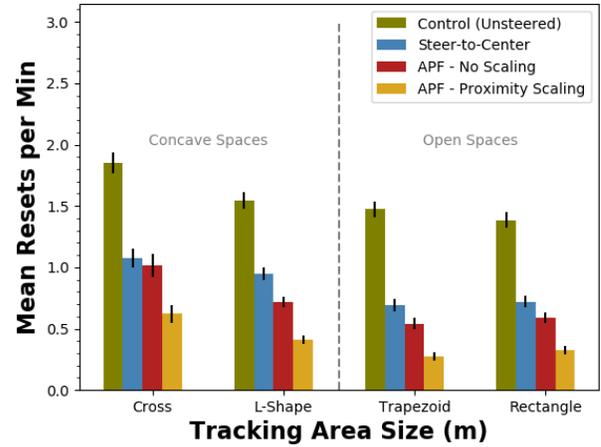


Figure 7: Average number of resets per minute of single users for each algorithm and room shape. APF-SC is designed to account for different wall positions and adapts to each shape, while STC performance suffers in concave spaces.

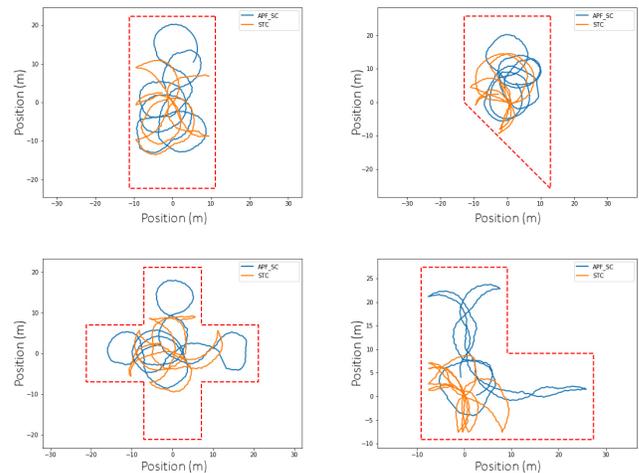


Figure 8: Sample plots for each room shape, illustrating a single path being redirected with either STC (orange) or APF (blue). APF attempts to use any available space, regardless of shape, while STC attempts to fit a series of circular paths into the central region of all rooms.

$\eta_p^2 = .85, p < .0001$ ). There was also an algorithm  $\times$  number of users interaction ( $F(25, 24.8) = 2.60, \eta_p^2 = .72, p < .02$ ) and a very weak three-way interaction between the factors ( $F(25, 50284) = 16.94, \eta_p^2 = .01, p < .0001$ ). In practical terms, the amount of travel generally increased in larger rooms, but decreased as more users were added. Both APF algorithms mitigated this decrease well, but Control and STC users were unable to take advantage of the extra space when too many users were present.

### 5.3 Odd-Shaped Rooms

The purpose of this simulation was to illustrate the ability of APF redirection to adapt to different room shapes and use any available space to steer the user. To this end, single users were simulated in four different rooms of identical surface area: a rectangle, trapezoid,

cross, and L-shape. These are shown in Figure 8, with sample paths of an identical simulated user redirected with either APF-SC or STC. The figure omits APF-U and control condition paths for clarity, but these were also simulated and included in the analyses below. As hypothesized, two trends were clearly visible. First, STC attempts to create circular arcs in the central region of each room regardless of its shape, which works well in open spaces but leads to many resets in concave rooms. Second, APF-SC guided users more broadly throughout the different spaces and was able to avoid corners in concave tracking areas.

The average number of wall resets for each algorithm and room shape is shown in Figure 7. A 4 (algorithm) x 4 (room shape) ANOVA indicated a large main effect of algorithm ( $F(3, 4320) = 1016.89, \eta_p^2 = .41, p < .0001$ ), as well as a medium main effect of room shape ( $F(3, 4320) = 147.66, \eta_p^2 = .09, p < .0001$ ). Planned contrasts supported the hypothesis that there was a difference between the concave spaces (cross, L-shape) and the open rooms (trapezoid, rectangle;  $p < .001$ ), and that wall resets did not differ between the two open rooms ( $p = .653$ ). A weak algorithm x room shape interaction was also observed ( $F(9, 4320) = 2.66, \eta_p^2 = .01, p < .01$ ) due to the relatively worse ability of STC and APF-U to adapt to concave tracking areas.

For average steering rates, the zero-rate control condition was again excluded from the ANOVA. There was a main effect of algorithm ( $F(2, 3259) = 362.62, \eta_p^2 = .18, p < .0001$ ), with APF-SC showing modestly higher steering rates, followed by APF-U and STC. There was also a main effect of room shape ( $F(3, 3259) = 47.57, \eta_p^2 = .02, p < .0001$ ) and an interaction between algorithm and room shape ( $F(6, 3259) = 31.19, \eta_p^2 = .02, p < .0001$ ), as the different room shapes exerted different pressure on APF steering.

For the distance traveled between resets, there was a large main effect of algorithm ( $F(3, 4320) = 762.49, \eta_p^2 = .35, p < .0001$ ) and room shape ( $F(3, 4320) = 113.43, \eta_p^2 = .07, p < .0001$ ). There was also a small but significant interaction ( $F(9, 4320) = 14.04, \eta_p^2 = .03, p < .0001$ ). Given that all of the rooms were of the same area, these results closely mirrored the number of resets. Users who experienced fewer resets were interrupted less and able to travel farther distances, giving an advantage to simulated users with APF-SC.

## 6 CONCLUSION

In general, the data conformed to expected trends and supported the hypotheses listed above. The simplified version of Artificial Potential Field (APF-U), in which proximity-based scaling was removed to make it more comparable to Steer-to-Center (STC), performed nearly identically to STC in square rooms of all sizes. These two algorithms also differed in the areas that were expected; despite having the same rate of redirection, APF-U showed increased performance when multiple users were present and in non-traditional tracking spaces such as L-shaped rooms. The full implementation of APF that included temporary, proximity-based scaling of the steering rates to prevent impending collisions (APF-SC) outperformed the other algorithms in all cases as expected. These simulation results mirror data collected in the original live-user implementation of APF-RDW, in which it markedly outperformed STC for single users, even with comparable steering rates [3].

As noted in the earlier description of the algorithms, the use of rate scaling may be more appropriate for some use cases than others. It was designed to prevent interruptions, but risks becoming perceptible to the user at least temporarily. The experiments above help quantify the trade-off. In medium-to-large spaces and in odd-shaped tracking areas, APF-SC displayed clear advantages over APF-U while having a relatively small effect on the average steering rates. Users would occasionally be exposed to higher steering rates, but with high utility in those cases. In small rooms - particularly the 10x10m space - scaling led to consistently higher steering rates as

users were *always* in close proximity to a wall. It is left to individual developers as to whether scaling makes sense for their application and how much scaling to implement.

When considering the results of the multi-user simulation, the relatively poor performance of the control condition and STC warrant more discussion. One might expect these approaches to support more than a few users in a very large space for no other reason than because the users are dispersed enough to not collide by chance. This is true for the control condition, where users can spread out and are limited mainly by how quickly they reach a wall. With STC, however, the algorithm is self-limiting and constrained almost entirely by an increase in user resets. Because STC is pulling all users towards the center of the space, it makes collisions *more* likely as users are pulled towards each other. With increasing numbers of users, the center of the tracking area becomes increasingly congested. This can be solved by giving each user a distinct “center” to be pulled towards, but these offset “centers” must be increasingly dispersed and thus nearer the walls as more users are added. This is counterproductive to the goal of maximizing the available space for each user. Another solution, proposed by [4], is to predict where a pair of users is heading and when a collision might be imminent, and then to adjust each user’s steering instructions temporarily based on the angle of impact. As Holm noted, however, this solution does not scale to more than two users and is computationally complex. It is difficult to accurately predict paths and collisions for spontaneous human navigators, and any diversion to prevent a collision (e.g., send each user rightward for X sec) may cause a secondary collision with a wall or a third user who was previously predicted to pass by without incident, as discussed by [2]. Thus, each adjustment and its cascading effects must be modeled against all other users, with a moderate degree of uncertainty as to whether each user’s current behavior will continue unchanged. APF alleviates all of this complexity and uncertainty by continually accounting for each user’s current position and deriving the safest direction of travel for each user. Walls and other users naturally repel a given person into open space and automatically adjust to changing conditions. Again, the simulated results here conform to observations in the live user study [3], in which APF-RDW was able to redirect several simultaneously immersed users with a high degree of success, and without notably increasing steering rates while un-steered Control users were limited in how far they could walk before frequently reaching the walls.

The current implementation of APF-SC does have some limitations and could be improved. As described above, position jitter in real-world tracking systems, which is more common in wide-area tracking, can cause a user to appear outside of the tracking space temporarily. This is a current failure case for the algorithm, as it erroneously attempts to push the user further beyond the wall. More sophisticated APF algorithms could also be envisioned that included force vectors or negative force vectors (i.e., an attraction force) to convey information about what areas of the VE are navigable or where the user is attempting to travel.

In sum, these simulations mirror and extend the trends seen with live users under similar conditions, allowing a broader range of conditions to have been tested. These simulations illustrate the ability of APF-RDW to adapt to a wide variety of steering conditions that could not be tested in the physical laboratory, ranging from small rooms to large open spaces, and to tracking areas with concave corners or odd shapes, and to a large number of concurrent users. All of this can be done with low computational complexity, as additional users or wall segments each add a new force vector to be summed and repel the user towards open space. It thus provides a widely generalizable solution for redirected walking. While this technique may not be applicable to single-user consumer VR setups, it may prove effective in large research settings, multi-user VR arcades, outdoor position tracking areas based on GPS, or VR eSports arenas.

## ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under grant no. 1423112.

## REFERENCES

- [1] M. Azmandian, T. Grechkin, M. T. Bolas, and E. A. Suma. Physical space requirements for redirected walking: How size and shape affect performance. In *ICAT-EGVE*, pages 93–100, 2015.
- [2] M. Azmandian, T. Grechkin, and E. S. Rosenberg. An evaluation of strategies for two-user redirected walking in shared physical spaces. In *2017 IEEE Virtual Reality (VR)*, pages 91–98. IEEE, 2017.
- [3] E. Bachmann, E. Hodgson, C. Hoffbauer, and J. Messinger. Multi-user redirected walking and resetting using artificial potential fields. *IEEE Transactions on Visualization & Computer Graphics*, 2018, forthcoming.
- [4] E. R. Bachmann, J. Holm, M. A. Zmuda, and E. Hodgson. Collision prediction and prevention in a simultaneous two-user immersive virtual environment. In *2013 IEEE Virtual Reality (VR)*, pages 89–90, March 2013.
- [5] G. Bruder, P. Lubas, and F. Steinicke. Cognitive resource demands of redirected walking. *IEEE transactions on visualization and computer graphics*, 21(4):539–544, 2015.
- [6] G. Bruder, F. Steinicke, B. Bolte, P. Wieland, H. Frenz, and M. Lappe. Exploiting perceptual limitations and illusions to support walking through virtual environments in confined physical spaces. *Displays*, 34(2):132–141, 2013.
- [7] G. Bruder, F. Steinicke, and K. H. Hinrichs. Arch-explore: A natural user interface for immersive architectural walkthroughs. 2009.
- [8] S. S. Chance, F. Gaunet, A. C. Beall, and J. M. Loomis. Locomotion mode affects the updating of objects encountered during travel: The contribution of vestibular and proprioceptive inputs to path integration. *Presence*, 7(2):168–178, 1998.
- [9] T. Field and P. Vamplew. Generalised algorithms for redirected walking in virtual environments. 2004.
- [10] T. Grechkin, J. Thomas, M. Azmandian, M. Bolas, and E. Suma. Re-visiting detection thresholds for redirected walking: Combining translation and curvature gains. In *Proceedings of the ACM Symposium on Applied Perception*, pages 113–120. ACM, 2016.
- [11] 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, Apr. 2013.
- [12] E. Hodgson, E. Bachmann, and T. Thrash. Performance of redirected walking algorithms in a constrained virtual world. *IEEE Transactions on Visualization & Computer Graphics*, (4):579–587, 2014.
- [13] E. Hodgson, E. Bachmann, and D. Waller. Redirected walking to explore virtual environments: Assessing the potential for spatial interference. *ACM Transactions on Applied Perception (TAP)*, 8(4):22, 2011.
- [14] C. Hoffbauer. Multi-user redirected walking and resetting utilizing artificial potential fields. Master’s thesis, Miami University, 2018.
- [15] J. E. Holm. Collision prediction and prevention in a simultaneous multi-user immersive virtual environment. Master’s thesis, Miami University, 2012.
- [16] C. S. Kallie, P. R. Schrater, and G. E. Legge. Variability in stepping direction explains the veering behavior of blind walkers. *Journal of Experimental Psychology: Human Perception and Performance*, 33(1):183, 2007.
- [17] E. M. Kolasinski. Simulator sickness in virtual environments. Technical report, Army research inst for the behavioral and social sciences Alexandria VA, 1995.
- [18] L. P. B. G. Langbehn, Eike and F. Steinicke. Application of redirected walking in room-scale vr. In *Proceedings of 2017 IEEE VR*. IEEE, 2017.
- [19] B. D. Lawson, D. A. Graeber, A. M. Mead, and E. R. Muth. Signs and symptoms of human syndromes associated with synthetic experiences. In *Handbook of virtual environments*, pages 629–658. CRC Press, 2002.
- [20] E. H. J. W. Michael Zmuda, Eric Bachmann. Improved resetting in virtual environments. *IEEE VR*, 2013.
- [21] C. T. Neth, J. L. Souman, D. Engel, U. Kloos, H. H. Bulthoff, and B. J. Mohler. Velocity-dependent dynamic curvature gain for redirected walking. *IEEE transactions on visualization and computer graphics*, 18(7):1041–1052, 2012.
- [22] S. Razzaque. *Redirected walking*. University of North Carolina at Chapel Hill, 2005.
- [23] S. Razzaque, Z. Kohn, and M. C. Whitton. Redirected walking. In *Proceedings of EUROGRAPHICS*, volume 9, pages 105–106. Citeseer, 2001.
- [24] R. A. Ruddle and S. Lessels. The benefits of using a walking interface to navigate virtual environments. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 16(1):5, 2009.
- [25] J. L. Souman, I. Frissen, M. N. Sreenivasa, and M. O. Ernst. Walking straight into circles. *Current biology*, 19(18):1538–1542, 2009.
- [26] F. Steinicke, G. Bruder, K. Hinrichs, and A. Steed. Gradual transitions and their effects on presence and distance estimation. *Computers & Graphics*, 34(1):26–33, 2010.
- [27] F. Steinicke, G. Bruder, J. Jerald, H. Frenz, and M. Lappe. Analyses of human sensitivity to redirected walking. In *Proceedings of the 2008 ACM Symposium on Virtual Reality Software and Technology, VRST ’08*, pages 149–156, New York, NY, USA, 2008. ACM.
- [28] 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, Jan 2010.
- [29] E. A. Suma, G. Bruder, F. Steinicke, D. M. Krum, and M. Bolas. A taxonomy for deploying redirection techniques in immersive virtual environments. In *Virtual Reality Short Papers and Posters (VRW), 2012 IEEE*, pages 43–46. IEEE, 2012.
- [30] E. A. Suma, Z. Lipps, S. Finkelstein, D. M. Krum, and M. Bolas. Impossible spaces: Maximizing natural walking in virtual environments with self-overlapping architecture. *IEEE Transactions on Visualization and Computer Graphics*, 18(4):555–564, 2012.
- [31] M. Usoh, K. Arthur, M. C. Whitton, R. Bastos, A. Steed, M. Slater, and F. P. Brooks Jr. Walking, walking-in-place, flying, in virtual environments. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 359–364. ACM Press/Addison-Wesley Publishing Co., 1999.
- [32] K. Vasylevska, H. Kaufmann, M. Bolas, and E. A. Suma. Flexible spaces: Dynamic layout generation for infinite walking in virtual environments. In *3D user interfaces (3DUI), 2013 IEEE Symposium on*, pages 39–42. IEEE, 2013.
- [33] B. Williams, S. Bailey, G. Narasimham, M. Li, and B. Bodenheimer. Evaluation of walking in place on a wii balance board to explore a virtual environment. *ACM Transactions on Applied Perception (TAP)*, 8(3):19, 2011.
- [34] 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.