An Evaluation of Strategies for Two-User Redirected Walking in Shared Physical Spaces

Mahdi Azmandian* Timofey Grechkin* Evan Suma Rosenberg*

USC Institute for Creative Technologies, Los Angeles CA, USA

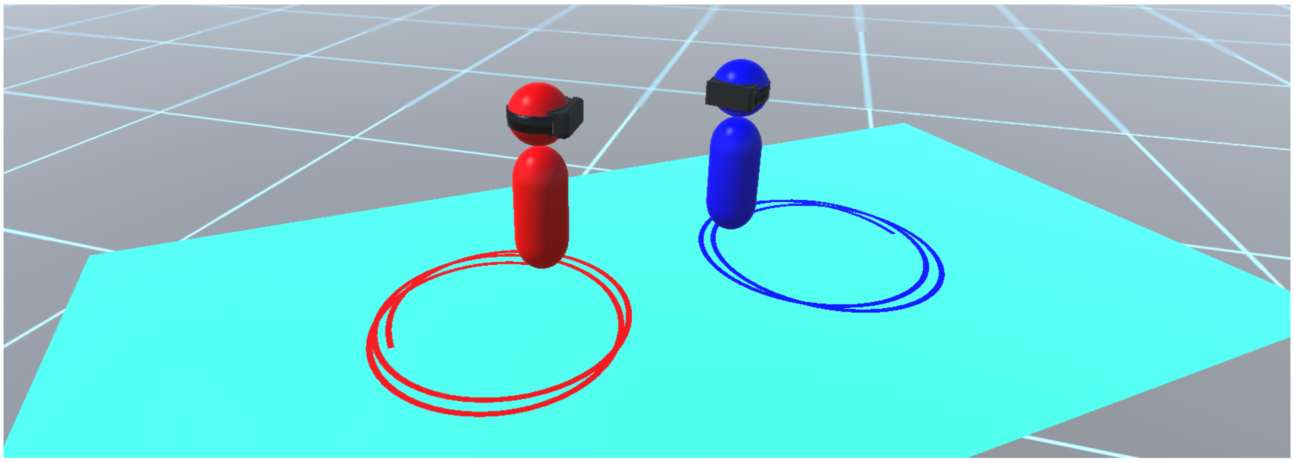


Figure 1: Conceptual representation of two simulated users exploring an abstract virtual environment. In the presence of redirected walking, the users' path in the real world is curved unbeknownst to the users.

ABSTRACT

As the focus of virtual reality technology is shifting from single-person experiences to multi-user interactions, it becomes increasingly important to accommodate multiple co-located users within a shared real-world space. For locomotion and navigation, the introduction of multiple users moving both virtually and physically creates additional challenges related to potential user-on-user collisions. In this work, we focus on defining the extent of these challenges, in order to apply redirected walking to two users immersed in virtual reality experiences within a shared physical tracked space. Using a computer simulation framework, we explore the costs and benefits of splitting available physical space between users versus attempting to algorithmically prevent user-to-user collisions. We also explore fundamental components of collision prevention such as steering the users away from each other, forced stopping, and user re-orientation. Each component was analyzed for the number of potential disruptions to the flow of the virtual experience. We also develop a novel collision prevention algorithm that reduces overall interruptions by 17.6% and collision prevention events by 58.3%. Our results show that sharing space using our collision prevention method is superior to subdividing the tracked space.

Keywords: Virtual Reality, Locomotion, Redirected Walking.

Index Terms: H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—Artificial, augmented, and virtual realities; I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual reality

1 INTRODUCTION

Researchers, social media, and entertainment companies are increasingly turning their attention toward supporting shared social, collaborative, or even competitive virtual reality (VR) experiences. The transition of VR experience from a single-user system to a collaborative system inevitably requires interaction techniques to be adopted for multiple users [4]. Like many other VR interaction techniques, Redirected Walking [13, 16] was originally designed for a single user. Its purpose is to enable the user to explore a large virtual environment by physically walking in a finite tracked space. Increasingly however, there is demand for using navigation techniques that can accommodate more than one user. Emerging VR arcade experiences such as The VOID are already actively experimenting with applying elements of redirected walking techniques to multiple users [14]. This trend will surely extend to at-home room-scaled VR settings as well.

Imagine a group of friends enjoying a VR game in a living room. Ideally, all of them have to be simultaneously immersed into a virtual environment and be able to navigate through it. To accommodate multiple physically co-located users, the available physical space must be shared, even if these users are not directly interacting in the virtual space. Which strategy should it use to accomplish this? One approach is to split the tracked space, dedicating a separate partition for each user to be redirected in. Alternatively, we can allow all users to share the entire tracked space, which would also require a method for avoiding user collisions.

In this paper we will explore strategies for enabling redirected walking for two physically co-located users and preventing collisions between users sharing physical space. Our contributions include introducing novel collision prevention heuristics, a systematic

*{mazmandian, grechkin, suma}@ict.usc.edu

2017 IEEE Virtual Reality (VR)
March 18-22, 2017, Los Angeles, CA, USA
978-1-5090-6647-6/17/$31.00 © 2017 IEEE

performance evaluation of two-user redirected walking, and analysis of the salient performance factors to provide guidelines for an optimal two-user redirected walking strategy.

2 BACKGROUND

2.1 Redirected Walking for a Single User

Redirected Walking was initially proposed by Razzaque [13] as a potential solution to tracked space size limitations when exploring large virtual environments. This interaction technique is made possible by a perceptual illusion exploiting the fact that the human visual system dominates over vestibular cues. In fully immersive VR systems, it is possible to introduce subtle discrepancies between physical and visually perceived movements to "steer" unsuspecting users away from the boundaries of the physical tracked space.

The steering strategy can combine three types of perceptual manipulations (known as gains): translation, rotation, and curvature gains. Translation gain scales virtual translations relative to the real world movement, resulting in faster or slower displacement in the virtual world. Rotation gain applies scaling to rotations, effectively increasing or decreasing the amount of virtual rotation relative to user's real-world movement. Finally, curvature gain induces virtual rotations when the user is walking (i.e. primarily translating) in the real environment resulting in a curved real-world trajectory. To ensure that the user remains unaware of the manipulation, the gains need to be within the limits of corresponding perceptual detection thresholds, which are determined experimentally [17, 5]. Over the years a variety of steering algorithms have been proposed [13, 7, 9, 20], however one of the original redirection algorithms — Steer-To-Center (S2C) — remains most widely used. It is based on the simple heuristic that in order to avoid boundaries one can constantly attempt to steer the user toward the center of the tracked space.

In addition to the steering strategy, a complete redirected walking algorithm also requires a fail-safe reorientation mechanism (also known as a *reset*) that can be used at the boundary to prevent the user from leaving the tracked space [19, 11]. An example of this is the 2:1-Turn which is the most commonly used form of reset. This method instructs the user to perform a 360 rotation in place while scaling the virtual rotation by a factor of 2, resulting in a 180 degree rotation in the real world. Thus by the time the reset task is complete, the user will be facing back toward the tracked space.

The performance of a redirected walking algorithm depends on a variety of interacting factors including user behavior, tracked space dimensions, the structure of the virtual environment and the type of virtual path, as well as the internal parameters of the algorithm such as perceptual thresholds and the reset mechanism, which makes comparative evaluation of such algorithms a non-trivial problem. Azmandian et al [1] introduced a systematic method of evaluation that controlled for the most salient factors impacting performance and proposed using a simulated user to enable experiments with large numbers of lengthy trials, both of which cannot be feasibly accomplished via user studies. A key characteristic of this system was accounting for the effect of the tracked space and boundary collisions on the simulated user, requiring a reset action instead of permitting simulated users to exceed these limits. Furthermore, the frequency of these interruptions were used as the primary measure of performance instead of the previously common procedure of finding the minimum tracked space dimensions that could fully contain a redirected user without ever reaching a boundary.

2.2 Redirection for Multiple Users

The introduction of two or more users into the same tracked space for redirected walking creates a possibility of user-to-user collisions. As one might intuitively expect, naive walking users immersed into a virtual environment using a head-mounted display are generally unaware of their surroundings in the real world and cannot detect the physical proximity of another walking user [12]. This creates a new set of requirements for redirected walking algorithms to try to avoid user collisions, detect when a collision is imminent, and implement an effective technique to resolve imminent collisions. To date, there have been only a few attempts to address the issue of sharing a tracked space among multiple users.

One area where this issue has been examined is the domain of remotely-controlling mobile robots, where multiple operators might share the same tracked space. The related literature describes a technique called *motion compression* [10], which addresses a problem similar to that of redirected walking. Motion compression maps the predicted trajectory of a remote-controlled robot to a curved path with the greatest possible radius within the operator's tracked space limits. Due to the lack of expectation that the operator is walking naturally, the algorithm attempts to minimize overall gains, but does not impose rigid constraints on instantaneous gains. The resulting gains can be very significant. For example, in lieu of resets, motion compression relies on large added rotations injected when the operator approaches a physical boundary to turn her back into the physical space [18]. Rossler [15] considered the case of multiple operators sharing the same physical space and proposed to address the issue of potential collisions by enclosing each operator in a convex shaped security area and represented it as an obstacle. They were then able to avoid these obstacles using their motion compression algorithm for a non-convex environment.

To the best of our knowledge, the only work to date addressing the problem of sharing a tracked area with multiple users in the redirected walking domain is that by Holm [8, 3]. She proposed to augment S2C algorithm by introducing separate redirection "centers" for each user, which could be offset within the tracked space, promoting physical separation of the two users. In addition, this work proposes mechanisms for collision avoidance and also rules for stopping to prevent collisions. Both of these mechanisms are based on *collision avoidance* that takes into account a minimum safety buffer, assumed reaction time, and current speed. When collision avoidance is activated, the algorithm temporarily overrides the default steering rules and instead attempts to steer one or both users to a temporary target to reduce the likelihood of collision. Translation gains are also used to slow down users and prevent collisions. If a collision is imminent, one or both users will be prompted to stop.

Unfortunately, Holm's work has a number of limitations that restrict the applicability of the proposed approach as an actual two-person redirection algorithm. First, it does not provide a resolution mechanism for the situation where both users have stopped to prevent collision, which can create a starvation condition, stalling the experience for both users. Second, she used pre-recorded user trajectories for a single-user redirection experience to evaluate the performance for the two-person redirection scenario. It is unclear if this type of data is predictive of the actual behavior for the two-person scenario because it does not contain reactions to collisions-avoidance events that are introduced. Finally, the analysis of the proposed algorithm seems to account only for a special situation where the tracked space is effectively infinite. However, in a more realistic scenario, the actual real-world trajectory of the user will be shaped by both boundary and collision avoidance constraints. It is, therefore, critical to examine how the effects of these two types of constraint, on the redirection algorithm's performance interact with each other.

In this work, our goal is to address these limitations by evaluating two-user redirected walking using a simulation framework that controls user behavior and accounts for tracked space limitations. We build upon Holm's algorithm by introducing novel mechanics to improve its applicability and efficacy in preventing boundary and user collisions. Each modification is investigated alongside Holm's original components, and compared with a baseline (no collision

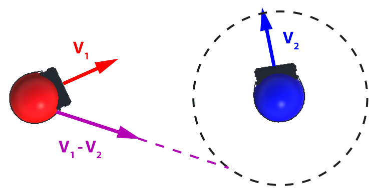


Figure 2: The Relative Velocity Heuristic. The relative velocity is attributed to the red user and intersected with a 0.5 meter radius around the other user to determine the future time of collision.

prevention condition) to tease out their contributions and trade-offs. The most effective of these solutions is then selected to examine how sharing a tracked space using collision prevention fairs against the alternative of partitioning the tracked space. This analysis is used to provide guidelines to an optimal strategy for utilizing a tracked space in a two user redirected walking experience.

3 Experiment 1: What Is the Optimal Collision Prevention Strategy?

The fundamental challenge when multiple users are introduced into the same tracked area is avoiding user collisions. The *collision avoidance* strategy has potentially competing goals with general redirection because steering the user away from another person might lead to a boundary collision event. Therefore, such a strategy can potentially have an associated cost that can be expressed in terms of interruptions that occur to prevent either user or boundary collisions.

In this experiment, we attempt to evaluate the relative costs associated with each component of Holm's collision prevention method as well as propose two modifications designed to overcome the limitations of the original approach:

1. A modified condition for activating the collision avoidance and stopping component of Holm's method based on using a *relative velocity heuristic*,

2. A newly designed *starvation resolution* heuristic, which alleviates the need for physical intervention in deadlocks where both users are stopping.

3.1 Proposed Modifications

3.1.1 Relative Velocity Heuristic

Holm's algorithm relies on a *safety condition* based on user proximity, reaction time and speed to determine when and if collision avoidance or stopping is required. This condition tends to be conservative, resulting in false positives where the algorithm uses collision prevention when it is not necessary [8]. We propose a safety condition that accounts for the relative velocity of the two users. The key idea is to predict the users' trajectory in the tracked space to determine the time of a future collision; if this duration is less than the user's reaction time, the situation is considered unsafe.

We first assume both users will maintain a constant linear motion in the tracked space. To simplify calculations, we calculate the relative velocity of the users, then attribute this to one of the them, allowing us to assume the other user is static. Then the question becomes when will the moving user get too close to the static user. This is calculated assuming a 0.5 meter radius circle around the static user and performing an intersection test between this circle and the relative velocity ray from the moving user (Figure 2). Once the location of the nearest collision (if an intersection exists) is found, the relative speed is used to determine the time of collision.

3.1.2 Starvation Resolution

In situations where stopping one user is not sufficient to preventing a collision between users, both users are required to stop. This results in a case of starvation where both users are waiting and further progression is not possible without physical intervention (Figure 3a). To provide a complete working solution we resolve starvations using resets. This is achieved by showing both users a secondary prompt after they have come to a full stop, requiring them to rotate in place, similar to the case of a reset triggered near a boundary. Using the 2:1-Turn mechanics, once a perceived 360 degree rotation in place is complete, each user will continue moving in the opposite direction, allowing them to continue their progression (Figure 3b).

3.2 Simulated User Framework

To perform simulated experiments, we used a modified version of the simulated user functionality included in The Redirected Walking Toolkit by Azmandian et. al [2]. A walking user was simulated by an autonomous agent programmed to traverse the virtual path by walking toward the next waypoint with a constant linear velocity of 1 m/s while maintaining its heading toward the waypoint (i.e. attempting to walk on a straight line in the virtual environment). Upon reaching a waypoint, the simulated user stopped and turned in place with angular velocity of 90 deg/s to face the next waypoint.

We simulated user reaction time modelled as a 0.5 second delay in stopping/reset response and a constant 0.5 second linear deceleration when stopping. Note that for this study no noise was introduced to the simulated user's translation and rotation. This guaranteed the simulated user would walk along the virtual path defined by the series of waypoints.

To create a different virtual path for each trial, we used a procedural random path generator used in [1]. We opted for using only one variation of random path generators since the performance of general redirection algorithms were shown not to be significantly influenced by virtual path type [1]. The virtual paths created in our system had randomized 90 degree turns, aiming to mimic walking in an office setting.

Boundary collision prevention was accomplished by a variation of the 2:1-Turn Reset[19] that used rotation gains to steer users towards the center of the tracked space. Resets were triggered when a user would come within a 0.5 meter distance from a tracked space boundary.

To simulate Holm's algorithm without requiring physical intervention, in cases where both users were stopped, the simulation simply allowed users to continue walking again. Note that collisions were temporarily discounted until the condition became safe again.

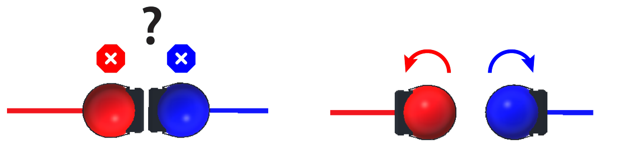
The simulation framework operates using *simulated time* to emulate frame rate during a real interactive experience. In our simulation we set the time elapsed between frames to $\frac{1}{60}s$ effectively simulating a 60Hz system.

3.3 Procedure

To provide a reference for comparison, we defined a baseline condition designed to count how many boundary events and user collision would have occurred if no collision prevention was used. Under this condition, simulated users were simply allowed to pass through each other, however a collision was still counted.

Six variations of Holm's algorithm were compared to the baseline: *Holm*, *RelHolm*, *Holm+SR*, *RelHolm+SR*, *Holm-CA+SR*, *RelHolm-CA+SR*; with "Rel" indicating the relative velocity heuristic, "-CA" meaning collision avoidance was disabled, and "SR" signifying the use of starvation resolution.

The dimensions of the tracked space was set to 10×5 meters. This layout was imagined as two commercial tracked spaces placed side by side representing a foreseeable common case for having two



(a) Starvation resulted from stopping (b) Starvation Resolution with resets

Figure 3: Resolving Starvation. Imminent collisions lead to starvation (a) where both users are stopping on each other. This is resolved by triggering 2:1-Turn resets (b) causing users to turn away and walk in the opposite direction.

users share a tracked space. Furthermore, a tracked space of this scale is more likely to cause boundary and user collisions, which can better accentuate the differences between the algorithms.

Each condition had 10 repetitions, and at the start of each trial, one user was positioned at (-2.5,0) and the other at (2.5, 0) with both users facing the positive y direction. These points also served as their corresponding *centers* which were used by S2C and the boundary reset. Each user would then traverse a 1000 meter randomly generated virtual path.

For each condition, all interruptions were tallied which included resets, stops and *collisions*. Collisions were defined as situations where the distance between users dropped below 0.5 meters, and were counted as 2 interruptions, one for each user. Note that starvation resolution resets were not regarded as additional interruptions since they were always preceded by a stop interruption.

3.4 Results

Our first step was to compare the effectiveness of collision prevention methods in terms of the overall number of interruptions (Figure 4). A one-way ANOVA model with interruptions as the response variable and the collision prevention method as the explanatory variable revealed a systematic difference between collision prevention methods ($F(6,6) = 89.2044, p < 0.0001$). The follow-up comparisons using Tukey's HSD revealed a consistent increase in the number of interruptions relative to Baseline: 30% for Holm ($t(63) = 17.89, p < 0.0001$), 19% for Holm+SR ($t(63) = 14.81, p < 0.0001$), 13% for Holm-CA+SR ($t(63) = 10.2, p < 0.0001$), 8.5% for RelHolm ($t(63) = 6.68, p < 0.0001$), and 7.5% for RelHolm+SR ($t(63) = 6.08, p < 0.0001$). The only exception was the RelHolm-CA+SR method, for which no significant difference was found ($t(63) = 0.98, p = 0.956$). We further found that the introduction of the relative heuristic reduced the number of interruptions in Holm vs. RelHolm by 11.6% ($t(63) = -11.21, p < 0.0001$), Holm+SR vs. RelHolm+SR by 9.3% ($t(63) = -8.73, p < 0.0001$) and for Holm-CA+SR vs. RelHolm-CA+SR by 10.4% ($t(63) = -9.22, p < 0.0001$). We also found that enabling CA increased the number of total interruptions in Holm+SR vs. Holm-CA+SR by 4.8% ($t(63) = 4.61, p = 0.0004$) and in RelHolm+SR vs. RelHolm-CA+SR by 6.3% ($t(63) = 5.1, p < 0.0001$). Finally, introducing the starvation resolution heuristic had a small but significant effect (reduction by 3.2%) on the number of interruptions when comparing Holm vs. Holm+SR ($t(63) = -3.08, p = 0.0452$), and no effect when comparing RelHolm vs. RelHolm+SR ($t(63) = 0.60, p = 0.996$).

To better understand the underlying dynamics behind these results we further explored the breakdown of interruptions into boundary and user-related events as a function of collision prevention method as shown in Figure 4. A one-way ANOVA model with boundary events as the response variable and the collision prevention method as the explanatory variable confirmed the existence of significant differences between methods ($F(6,6) = 14.9404, p < 0.0001$). The followup comparisons using Tukey's multiple comparison adjustment revealed that including CA on average increased boundary events by 4% for Holm+SR vs. Holm-CA+SR ($t(63) = -3.43, p = 0.0177$), and by 6.7% for RelHolm+SR vs. RelHolm-CA+SR ($t(63) = -5.83 p < 0.0001$). There were no significant differences when using relative velocity for Holm vs. RelHolm ($t(63) = 0.07, p = 1$), nor Holm+SR vs. RelHolm+SR ($t(63) = 2.47, p = 0.1882$) and neither for Holm-CA+SR vs. RelHolm-CA+SR ($t(63) = -0.07, p = 1$). Lastly, a one-way ANOVA with user-triggered events as response variable and the collision prevention method as the explanatory variable indicated existence of systematic differences ($F(6,6) = 92.8082, p < 0.0001$). The follow-up comparisions using Tukey's HSD did not show significant effects for including CA ($t(63) = 2.04, p = 0.4; t(63) = 0.39, p = 0.997$). Though for the relative velocity heuristic, Holm vs. RelHolm had a reduction 53.8% reduction in user-triggered events ($t(63) = 12.55, p < 0.001$), Holm+SR vs. RelHolm+SR had 54.4% reduction ($t(63) = 12.08, p < 0.0001$), Holm-CA+SR vs. RelHolm-CA+SR had a decrease of 51.7% ($t(63) = 10.43, p < 0.0001$).

3.5 Discussion

Our data shows that the introduction of Holm's heuristics to prevent collisions between users with shared tracked space during redirected walking causes a sizable increase in interruption events relative to the baseline. This suggest that the heuristics are not very efficient in addressing the issue.

One factor contributing to the increase in the overall interruption count is the CA heuristic, which produced a significant increase in boundary events while failing to significantly affect the number of user-related interruptions. This shows that when the available tracked space is limited, CA heuristic interferes with the S2C steering and increases the chance of boundary events. At the same time, it largely fails in its stated purpose of reducing the chance of user collisions. While the performance of the CA heuristic may improve in larger tracked spaces, it seems that a more efficient CA may need to factor in the proximity of the boundary or attempt to augment rather than fully override the redirection algorithm's steering. Such a method can also factor in the virtual environment layout to predict the user's actions similar to MPCRed [9]. In the mean time we conclude that the current CA heuristic should not be included into the repertoire of actions for preventing user collisions.

The data also shows that our relative velocity heuristic for activating collision-prevention mechanisms provided sizable gains in efficiency in terms of overall number of interruptions relative to Holm's original safety distance heuristic. This was achieved by reducing user-related events while avoiding generating extra boundary collisions. At the same time we saw that neither Holm's original heuristic, nor our own method is completely 100% fail-safe and may have difficulties managing cases of a sudden change in speed and direction of movement when users are very close to each other. We suggest that these methods could be complemented by rendering a visual representation of the other user as a "ghost" when it is necessary to alert the users of close proximity to another person.

The results also show that we were successful in introducing the SR heuristic to avoid the algorithm's starvation condition present in Holm's original version. We were able to achieve this without significantly impacting total interruption counts and even achieved reduction for Holm's original algorithm.

Overall, we conclude that RelHolm-CA+SR combination of heuristics seems to be the best performing version of collision prevention methods. It is notable, that the total number of interruptions generated by this combination of heuristics was the closest (and the only one that did not differ significantly) to the baseline condition, suggesting a high degree of efficiency in preventing user collisions.

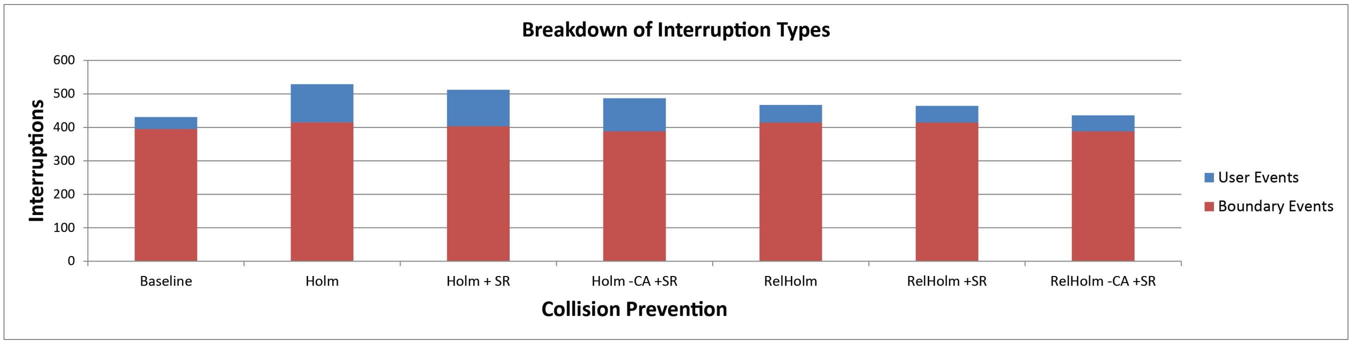


Figure 4: Breakdown of interruptions types (user and boundary events) for various collision prevention methods in experiment 1.

In the remainder of this paper we will use this technique to compare strategies for sharing a tracked space between two users.

4 Experiment 2: Comparing Space-Sharing Strategies for Two Redirected Users

One of the first decisions one has to make when designing an interactive virtual reality system for two physically co-located locomotion users is to pick a strategy for sharing a physical tracked space. Broadly speaking, one could subdivide the available space between the users or share the physical space between two users.

This experiment is designed to explore which strategy would be most beneficial to use for a two-user redirected walking algorithm. To achieve this we compare the performance of the common Steer-to-Center (S2C) algorithm as implemented in [7] under the following three strategies for sharing physical space (illustrated in Figure 5):

1. *Subdivision strategy*: the tracked space is evenly split into dedicated walkable sub-spaces for each user. The boundaries of each partition are treated the same as the outer space boundary of the tracked space. This precludes the possibility of user collisions and effectively reduces the problem of two-person redirection to two largely independent instances of the single person redirection problem. However, the effective size of the usable tracked space for each user is reduced to half of the overall space available, which is likely to increase the overall number of resets at the boundaries.

2. *Space sharing with common redirection center*: the users share a full tracked space and the S2C algorithm uses the center of this common space as the redirection center for both users. This strategy offers larger effective tracking space for each instance of the redirected walking algorithm, but introduces the possibility of user-on-user collisions. Therefore a collision avoidance method is required. In turn, the collision avoidance algorithm is likely to introduce interruption events similar to boundary resets.

3. *Space sharing with offset redirection centers*: the users share a full tracked space, however, the S2C algorithm uses a separate redirection center for each user. As suggested by Holm [8] this strategy may reduce the likelihood of collision between users in a shared tracked space and has the potential of striking a better balance between the number of potential boundary and user collisions compared to the other two strategies.

4.1 Procedure

All 3 strategies (*subdivision*, *sharing+common*, and *sharing+offset*) were tested in this experiment. The offset centers were positioned at (-$\frac{X}{4}$, 0) and ($\frac{X}{4}$, 0) where X is the length of the tracked space along the x axis.

For each strategy, 3 categories of tracked space sizes were considered: small, medium and large. Each size category had two shapes, a 2:1 rectangle and a square. The rectangular sizes were 10×5, 30×15 and 25×50 meters, and the corresponding square-shaped tracked spaces had the same area (7.07×7.07, 21.21×21.21 and 35.35×35.35), for a total of 18 conditions.

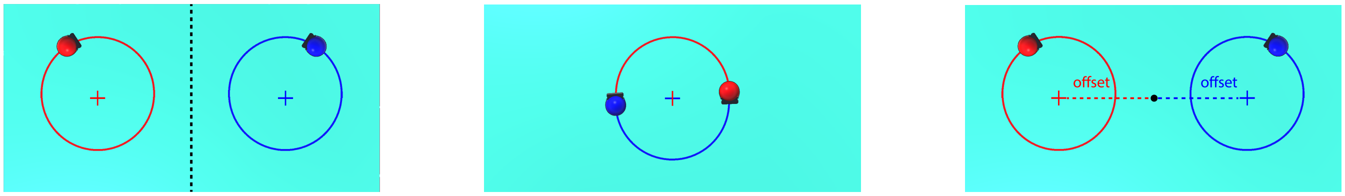
Each condition had 10 repetitions, and at the start of each trial, one user was positioned at (-$\frac{X}{4}$, 0) and the other at ($\frac{X}{4}$, 0), with both facing the positive y direction. Each user would then traverse a 1000 meter randomly generated virtual path. For each condition, all interruptions were tallied which included resets, stops and *collisions*, with collisions counting as two interruptions.

4.2 Results

We first explored how the effects of the space-sharing strategy on the number of interruptions are modulated by the size of the tracking area (Figure 6). We used a two-way ANOVA model with interruptions as the response variable and size and sharing strategy as explanatory variables. To simplify the analysis we considered subsets of rectangular and square-shaped spaces separately.

For rectangular tracked spaces we found a significant interaction between sharing strategy and size ($F(4,4) = 30.6592, p < 0.0001$). To explore this interaction further we performed a series of pair-wise comparisons using Tukey's HSD adjustment. These tests indicate that for small tracked spaces the sharing+offset strategy resulted in 12% fewer interruptions compared to the subdivision strategy ($t(81) = -15.02661, p < 0.0001$) and 10% fewer interruptions compared to the sharing+common strategy ($t(81) = -12.12, p < 0.0001$). There was no significant difference between subdivision and sharing+common strategies ($t(81) = 2.905, p = 0.1026$). In medium-size spaces sharing+offset reduced the number of interruptions by 23% ($t(81) = -4.98, p = 0.0001$) vs. subdivision. However, we found no significant differences between two sharing strategies ($t(81) = 2.46, p = 0.2657$), nor between subdivision and sharing+common($t(81) = 2.58, p = 0.239$). For the large tracked spaces all three comparisons between strategies were no longer significant ($t(81) = 1.3, p = 0.928; t(81) = 0.83, p = 0.995; t(81) = 0.47, p = 0.999$).

For square-shaped tracked spaces we also found significant interaction between sharing strategy and size ($F(4,4) = 210.0.240, p < 0.0001$). To explore this interaction further we performed a series of pair-wise comparisons using Tukey's HSD adjustment. These tests indicate that for small tracked spaces the sharing+common strategy resulted in 31% fewer interruptions compared to subdivision ($t(81) = -39.65, p < 0.0001$) and 4.3% fewer interruptions compared to sharing+offset ($t(81) = -3.91, p = 0.0056$). The sharing+offset strategy also reduced interruptions by 28% in compar-



(a) Subdivision Strategy

(b) Sharing Strategy with Common Center

(c) Sharing Strategy with Offset Center

Figure 5: Sharing Strategies. Illustration of three methods for implementing two-user redirected walking based on the sharing strategy.

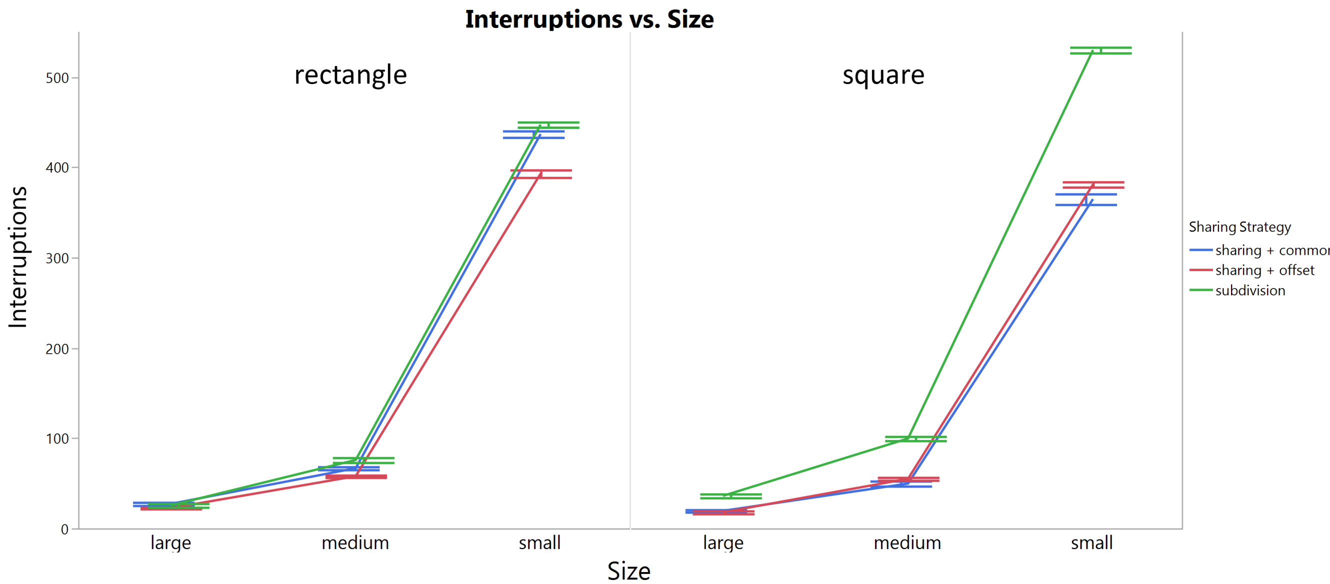


Figure 6: Average interruptions for various choices of sharing strategies and tracked space sizes and shapes in experiment 2.

ison with subdivision ($t(81) = 35.73, p < 0.0001$). In medium-size spaces, in comparison to subdivision, sharing+common reduced the number of interruptions by 50% ($t(81) = 11.98, p = 0.0001$) and sharing+offset also reduced interruptions by 45% ($t(81) = 1.272, p = 0.963$). However, we found no significant differences between two sharing strategies ($t(81) = 2.46, p = 0.2657$) The trend slightly changes for large spaces where in comparison to subdivision, sharing+offset reduced the number of interruptions by 50% ($t(81) = 4.37, p = 0.0012$) and subdivision reduced interruptions by 46% ($t(81) = 3.98, p = 0.0044$), with no significant differences between two sharing strategies ($t(81) = 0.384, p = 1$).

The effects of the space-sharing strategy can also be modulated by the shape of the tracked space, particularly for smaller tracked areas as can be seen in Figure 6. Looking at the subset of data for the small tracked space size we performed a two-way ANOVA with sharing strategy and shape as factors and interruptions as the response variable. The analysis revealed a significant interaction between sharing strategy and shape ($F(4,4) = 30.6592, p < 0.0001$). The follow-up Tukey's HSD comparisons revealed that a change from rectangle to square shape did not have a significant effect on performance for sharing+offset ($t(54) = 2.155, p = 0.275$), but resulted in 16.5% reduction in interruptions for sharing+common strategy ($t(54) = -13.04, p < 0.0001$) and a 18.5% increase for subdivision strategy ($t(54) = 14.95, p < 0.0001$).

We further explored the interplay between the effects of Shape and Sharing Strategy by looking at the breakdown of interruptions into boundary events (restricting users from leaving the boundaries of the tracked space) and user-related interruptions (preventing user-to-user collisions) as shown in Figure 7.

A two-way ANOVA with boundary events as response variable revealed a significant interaction between sharing Strategy and Shape ($F(2,2) = 210.0248, p < 0.0001$). The followup comparisons using Tukey's HSD revealed that for square-shaped spaces sharing+offset strategy reduced the number of boundary events by 44% vs. subdivision ($t(54) = 44.54, p < 0.0001$) and sharing+common strategy achieved a further 10% reduction vs. sharing+offset ($t(54) = 44.54, p < 0.0001$). These differences were similar, but less pronounced in rectangular tracked space with sharing+offset strategy resulting in 23% fewer boundary events vs. subdivision ($t(54) = 19.65, p < 0.0001$) and sharing+common strategy achieving a further 8% reduction vs. Sharing+offset ($t(54) = 5.038, p < 0.0001$). In both shape cases sharing strategies outperformed subdivision and sharing+common is superior at reducing boundary events. Going from rectangle to square reduced boundary events by 16% and 14% respectively for sharing+common ($t(54) = -9.48, p < 0.0001$) and sharing+offset ($t(54) = -9.06, p < 0.0001$) strategies. On the other hand moving to the square shape results in an 18% increase of boundary events for subdivision strategy ($t(54) = 15.82, p < 0.0001$).

Lastly, a two-way ANOVA with user-triggered events as response variable showed a significant interaction between Sharing Strategy and Shape ($F(2,2) = 86.7554, p < 0.0001$). The followup caparisons using Tukey's HSD indicated that sharing+offset strategy reduced user-triggered events vs. sharing+common in both rectangular (by 59% on average, $t(54) = -22.36, p < 0.0001$) and

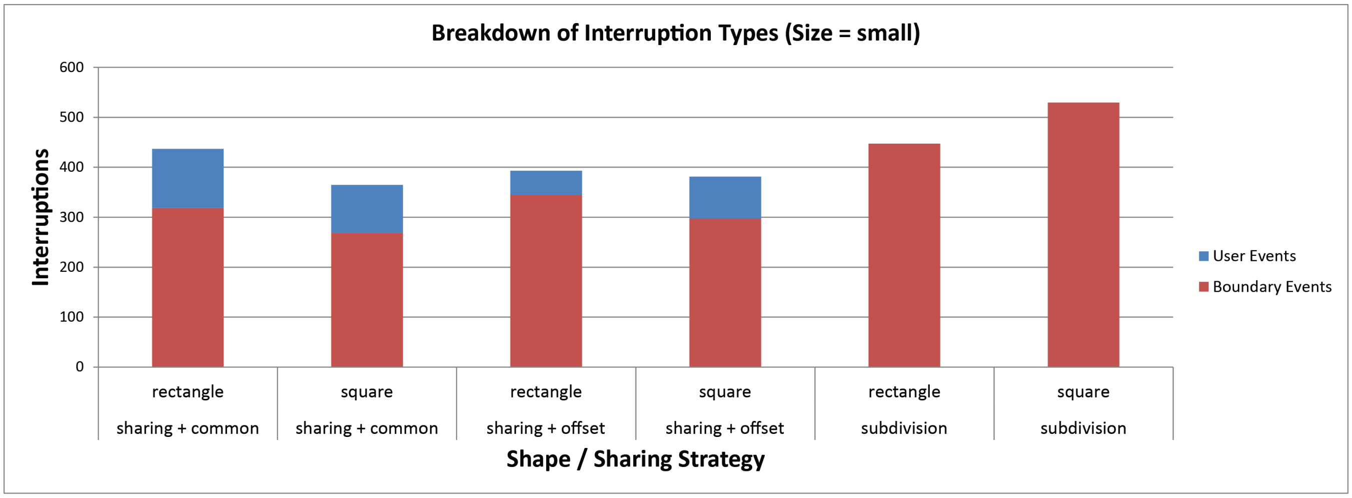


Figure 7: Breakdown of interruptions types by sharing strategy and tracked space shapes for a small-sized space in experiment 2.

square tracked space (by 13% on average, $t(54) = -3.89, p = 0.0036$). Also sharing+common strategy performed better in a square shaped tracked space reducing events by 19% on average ($t(54) = -7.179, p < 0.0001$), whereas sharing+offset strategy had better performance in a rectangular-shaped space reducing events by 42% ($t(54) = -11.269, p < 0.0001$).

4.3 Discussion

These results indicate that the effects of space sharing strategies are modulated by both the size and the shape of the physical tracked space. Generally speaking, the performance of two-person S2C algorithm exhibits patterns similar to those described by Azmandian et al. [1] for the the single-user version. In large tracked spaces the likelihood of interruptions due to either approaching the boundaries or possibility of user-to-user collision is drastically lower compared to small tracked spaces. As a result, the differences between space-sharing strategies are amplified in small spaces and decrease in large and medium sized spaces.

The data also shows that the subdivision strategy benefits from the allocated tracked space of each user being square-shaped. In particular, the subdivision strategy performed better in rectangular tracked spaces where each user was allocated an individual, square-shaped area compared to a square-shaped tracked space that was subdivided into elongated rectangular areas. For sharing+common strategy the pattern reversed. Interestingly, there were no significant differences for sharing+offset strategy suggesting that it might be more robust. One of the key advantages of sharing+ offset strategy is that it consistently reduces the number of user-related events (i.e. is more effective in reducing the number of potential user collisions) versus sharing+common strategy, at least in small tracked spaces.

The overall the pattern of results is relatively complex and suggests that a single strategy cannot universally deliver the best results. For example, in large, rectangular tracked spaces where the three strategies have similar performance, a subdivision strategy that excludes possibility of user collision and therefore reduces the computational overhead may be preferable. However, in large, square-shaped tracked areas, the sharing strategies deliver superior performance compared to subdivision strategy. In medium and especially small tracked spaces, one or both of the sharing strategies consistently outperform subdivision. We believe that for these spaces sharing+offset strategy has an edge due to its robustness. This strategy demonstrates consistently better performance compared to subdivision strategy and is less affected by the differences in the shape of the tracked area versus shared+common strategy.

5 Limitations

The experiments conducted in this work were based on a computer-simulated model of a walking user. An important avenue for improvement is the development of a more realistic representation of the user by introducing elements such as noisy movement, gait oscillations, and random gaze aversion. Such changes can substantially improve the validity of simulated user experiments. Furthermore, it remains an open question what level of user movement fidelity is required for the purpose of studying redirected walking algorithms, and how this fidelity affects the error introduced by using simulation. The computer-simulated user framework is a useful tool to explore large problem spaces, where multiple conditions can be compared using a large number of experimental trials. However, such a framework cannot replace studies with actual users. Ultimately, computer simulations provide a platform for intelligently reducing the number of alternative solutions and laying the groundwork for user studies with actual human participants.

In this work, we investigate a somewhat limited repertoire of collision avoidance methods available in the redirected walking literature. Given the similarities of this field to multi-robot planning, it may be possible to adopt similar approaches to develop avoidance measures that are applicable to two redirected users. Specifically this work can be improved upon by using trajectory planning strategies that explicitly account for static obstacles in addition to moving agents such as the RCAP algorithm [6].

6 Conclusion

In this work, we studied strategies for supporting redirected walking for two physically co-located users sharing the same physical tracked space. First we explored heuristics for avoiding collisions between users. We introduced several improvements to address limitations of the set of heuristics proposed for this purpose by Holm and, using a simulation framework, experimentally determined the most efficient subset, reducing the number of total interruptions by 17.6% and user events (collision prevention events) by 58.3%.

Second, we explored the relative advantage of different strategies for sharing a tracked space between two users. We demonstrated that for small to medium sized tracked space, sharing a tracked space between users is preferable to subdivision. We also found that the sharing with offset redirection centers strategy appears to be more robust to changes in tracked space shape.

In our experience, though simulations may not perfectly replicate realistic conditions and the nuances of real user behavior, they do provide an efficient way to quantify trade-offs between alternative algorithms and strategies, especially when there are many potential variations to consider. This approach allows to significantly narrow-down the number of alternatives to be eventually tested in a user study with human participants. In future work, this methodology can be used to explore advanced collision prevention mechanics and find the optimal center offset for a given tracked space dimension for effective deployment of two-user redirection.

ACKNOWLEDGEMENTS

This work is sponsored by the U.S. Army Research Laboratory (ARL) under contract number W911NF-14-D-0005. Statements and opinions expressed and content included do not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

The authors would like to thank Rhys Yahata for modeling the simulated user avatar that appears in the illustrations of this work.

REFERENCES

[1] M. Azmandian, T. Grechkin, M. Bolas, and E. Suma. Physical Space Requirements for Redirected Walking: How Size and Shape Affect Performance. *In Proc. ICAT-EGVE*, 2015.

[2] M. Azmandian, T. Grechkin, M. Bolas, and E. Suma. The redirected walking toolkit: A unified development and deployment platform for exploring large virtual environments. In *Second Workshop on Everyday Virtual Reality, IEEE VR*, 2016.

[3] 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.

[4] S. Benford, C. Greenhalgh, T. Rodden, and J. Pycock. Collaborative Virtual Environments. *Communications of ACM*, 44(7):79–85, 2001.

[5] 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*, SAP '16, pages 113–120, New York, NY, USA, 2016. ACM.

[6] S. J. Guy, M. C. Lin, and D. Manocha. Modeling collision avoidance behavior for virtual humans. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 2-Volume 2*, pages 575–582. International Foundation for Autonomous Agents and Multiagent Systems, 2010.

[7] E. Hodgson and E. Bachmann. Comparing four approaches to generalized redirected walking: simulation and live user data. *IEEE TVCG*, 19(4):634–43, 2013.

[8] J. E. Holm. Collision prediction and prevention in a simultaneous multi-user immersive virtual environment. Master's thesis, Miami University, 2012.

[9] T. Nescher, Y.-Y. Huang, and A. Kunz. Planning Redirection Techniques for Optimal Free Walking Experience Using Model Predictive Control. *3DUI 2014*, pages 111–118, 2014.

[10] N. Nitzsche, U. D. Hanebeck, and G. Schmidt. Motion Compression for Telepresent Walking in Large Target Environments. *Presence: Teleoperators and Virtual Environments*, 13(1):44–60, 2004.

[11] T. C. Peck, H. Fuchs, and M. C. Whitton. An evaluation of navigational ability comparing Redirected Free Exploration with Distractors to Walking-in-Place and joystick locomotio interfaces. *Proceedings - IEEE Virtual Reality*, pages 55–62, 2011.

[12] I. Podkosova and H. Kaufmann. Mutual Proximity Awareness in Immersive Multi-User Virtual Environments with Real Walking. In M. Imura, P. Figueroa, and B. Mohler, editors, *ICAT-EGVE 2015 - International Conference on Artificial Reality and Telexistence and Eurographics Symposium on Virtual Environments*. The Eurographics Association, 2015.

[13] S. Razzaque. *Redirected Walking*. PhD thesis, University of North Carolina at Chapel Hill, Chapel Hill, NC, USA, 2005.

[14] Road to VR. First hands-on: The VOID, a mixed reality experience that blends real and virtual. `http://www.roadtovr.com/first-hands-on-the-void-a-mixed-reality-experience-that-blends-real-and-virtual/`, May 2015. Accessed: 2016-12-11.

[15] P. Rößler and U. D. Hanebeck. Simultaneous motion compression for multi-user extended range telepresence. *IEEE International Conference on Intelligent Robots and Systems*, pages 5189–5194, 2006.

[16] F. Steinicke, G. Bruder, K. H. Hinrichs, J. Jerald, H. Frenz, and M. Lappe. Real Walking through Virtual Environments by Redirection Techniques. *Journal of Virtual Reality and Broadcasting*, 6(2):999–1004, 2009.

[17] F. Steinicke, G. Bruder, J. Jerald, H. Frenz, and M. Lappe. Estimation of Detection Thresholds for Redirected Walking Thechniques. *IEEE TVCG*, 16(1):17–27, 2010.

[18] J. Su. Motion Compression for Telepresence Locomotion. *Presence: Teleoperators and Virtual Environments*, 16(4):385–398, 2007.

[19] 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. APGV 2007*, 1(212):41, 2007.

[20] M. Zmuda, E. Bachmann, E. Hodgson, and J. Wonser. Improved Resetting in Virtual Environments. *IEEE Virtual Reality*, pages 91–92, 2013.