Imaginary Reality Gaming: Ball Games Without a Ball

Patrick Baudisch, Henning Pohl[*], Stefanie Reinicke, Emilia Wittmers,
Patrick Lühne, Marius Knaust, Sven Köhler, Patrick Schmidt, and Christian Holz
Hasso Plattner Institute, Potsdam, Germany
{firstname.lastname}@hpi.uni-potsdam.de
[*]University of Hannover, Germany
Henning.Pohl@hci.uni-hannover.de

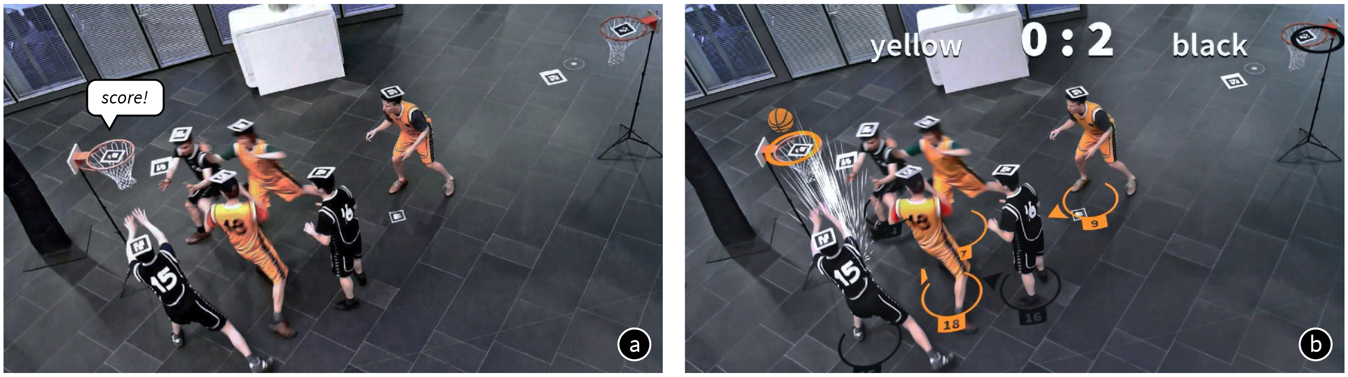

Figure 1: (a) Six players in a game of *Imaginary Reality Basketball*. Player 15 on the Black team has thrown the *imaginary ball* at the basket and scored. There is no visible ball; players get all information from watching each other act and a small amount of auditory feedback. (b) Under the hood & invisible to the players, the system represents the imaginary ball as a large number of *ball particles,* each of which represents one plausible ball trajectory. Players are tracked using accelerometers and an overhead camera.

ABSTRACT

We present *imaginary reality games,* i.e., games that mimic the respective real world sport, such as basketball or soccer, except that there is *no visible ball*. The ball is virtual and players learn about its position only from watching each other act and a small amount of occasional auditory feedback, e.g., when a person is receiving the ball.

Imaginary reality games maintain many of the properties of physical sports, such as unencumbered play, physical exertion, and immediate social interaction between players. At the same time, they allow introducing game elements from video games, such as power-ups, non-realistic physics, and player balancing. Most importantly, they create a new game dynamic around the notion of the invisible ball.

To allow players to successfully interact with the invisible ball, we have created a physics engine that evaluates all plausible ball trajectories in parallel, allowing the game engine to select the trajectory that leads to the most enjoyable game play while still favoring skillful play.

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces.

Keywords: Physical gaming; augmented reality gaming; probabilistic; imaginary interfaces; motion capture.

INTRODUCTION

Sports, such as soccer or basketball, offer many wonderful qualities. They are highly immersive, lead to physical exertion, and create immediate social interaction between players. Unfortunately, physical games are limited by the constraints of the real world, restricting their game mechanics to what is physically possible.

Researchers have tried to merge physical and virtual play in display-based augmented reality games such as Human Pacman [5] or AR Quake [16]. These games overlay a virtual world onto the physical world using hand-held or head-mounted see-through displays. This allows these games to introduce virtual game elements, such as power-ups (e.g. [22]) or creating virtual game elements that are not limited by the rules of physics (e.g. [20]).

Unfortunately, the use of displays takes away many of the qualities of physical sports, as players now perceive the world only indirectly through displays, which are of limited resolution, limited field of view, and exhibit inevitable lag. This causes many subtleties to be lost, such as details of body language, facial expressions, and co-player actions taking place in the user's visual periphery.

In this paper, we propose a different approach: We drop the displays, thus creating a sort of "screenless AR". Instead, players obtain most of the information they need to play from watching co-players' positions, movements, and gestures. In addition, but only where necessary, we provide a small amount of auditory feedback to disambiguate. So rather than trying to give players a high-fidelity representation of the virtual game world, we embrace the uncertainty and develop it into a new game mechanic. The result is a new class of interactive games. While these games maintain the ability to have virtual game elements, such as power-ups and non-physical objects, they bring back some of the qualities of physical sports games, as players again interact *directly* with each other.

UIST'13, October 8–11, 2013, St. Andrews, United Kingdom.

IMAGINARY REALITY GAMING

Figure 1a shows a scene from imaginary reality basketball, one of the games we have implemented. Like every imaginary reality game, it mimics the respective real world sport—except *there is no visible ball*. Players learn about the position of the virtual ball by watching each other act. Occasionally, players *hear* the ball make contact with a physical object or being played by a person, which disambiguates the ball's location. In between, the ball position is *uncertain*. At the shown moment, Player *15* on the Black team is throwing the imaginary ball at the basket, scoring.

Figure 2 shows a slightly longer example. (a) Player *9* on the Orange team (top left) chest passes the ball towards his teammate *18*. (b) *18* receives it (Audio: "eighteen") and (c) tries to pass it back to *9*. (d) Unexpectedly, *16* from the Black team intercepts the pass (Audio: "sixteen"). He starts running towards the basket and (e) dunks the ball. (Audio: "Score! Two-zero for Black").

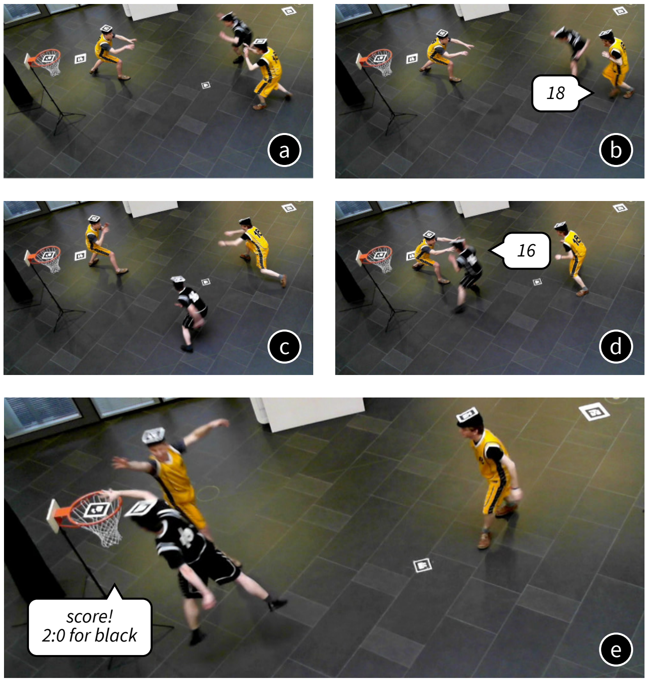

Figure 2: A game sequence leading up to a slam-dunk.

Players obtain most of the information about the ball from observing each other interact with the ball. Consequently, auditory feedback can be *and is* sparse and non-spatial. Besides game status updates, such as scores, auditory feedback is essentially limited to communicating when the ball makes contact with a person or object, such as the commentator voice announcing who just received the ball.

As mentioned earlier, imaginary reality games cannot only emulate real-world sports, but they also allow us to introduce power-ups and non-physical behavior, thereby introducing some of the richness of video games into physical sports. Figure 3 and Figure 4 illustrate this at the example of the "get-ball" and the "safe" power-ups, which appear periodically with an audio announcement over the marked center location of the playfield. Both figures use the "debug" view of our system, which reveals the ball etc; the players do not see any of this additional information.

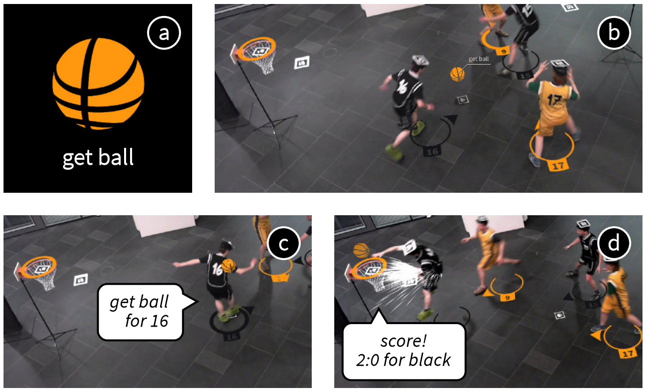

Figure 3: Imaginary reality games can offer non-physical elements in the form of power-ups: here the "get-ball" power-up.

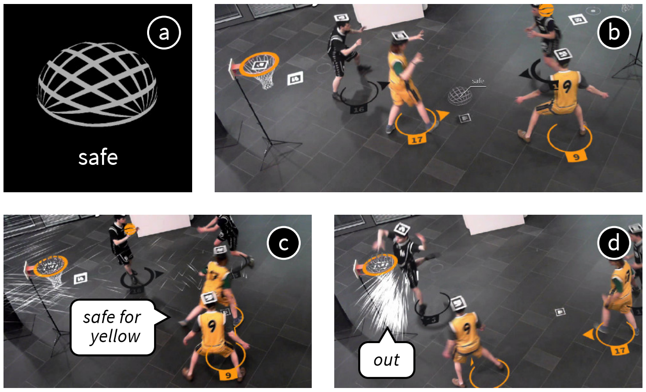

Figure 4: The "safe" power-up protects a team's basket. This prevents the opposing team from scoring.

SYSTEM ARCHITECTURE

As illustrated by Figure 5, the imaginary reality games system consists of three main components. The *tracking system* tracks players and their gestures (as well as playfield boundaries, baskets, and power-up locations). It reports player moves to our custom physics engine, which we call *quantum engine*. The quantum engine simulates the imaginary ball, computes the probabilities of outcomes, and reports these to the *game engine*. The game engine refines probabilities by applying handicaps and power-ups, etc. It then determines the outcome of the game move randomly based on these probabilities. Finally, the game engine conveys its decision to the players using auditory feedback.

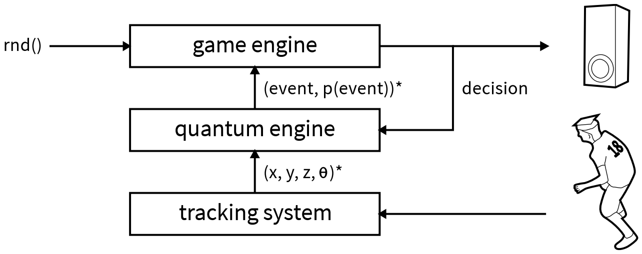

Figure 5: Imaginary reality games system

Figure 6 shows the tracking solution we have created for imaginary basketball. An overhead camera (Microsoft LifeCam 1080p) monitors the 6×4m playfield. It tracks players' head position and orientation based on head-worn ALVAR markers [1]. To allow the system to sense gestures, players wear tiny accelerometers with radio senders on their hands and belts (*Axivity* [2]). A simple peak detection algorithm extracts players' game gestures, such as jumping, throwing, or catching a ball.

Note that the tracking is incomplete. For example, it does not allow sensing in what direction a player passes a ball—it merely approximates this direction based on the player's head orientation. This conscious design decision allows us to run the game on moderate tracking hardware accessible to a broader audience (unlike, say, a commercial motion capture system). The quantum engine compensates for the lack of tracking information, which is one of the key engineering contributions of our system, as we explain in the following section.

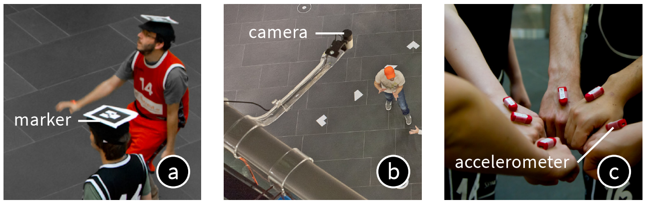

Figure 6: Our Tracking is based on (a) *ALVAR* markers [1], (b) a webcam, and (c) hand-worn accelerometer (*Axivity* [2]).

IMAGINARY PHYSICS

The quantum engine is the key component of the imaginary reality system. It allows players to successfully play the ball, despite the uncertain ball location and despite the only approximate tracking.

Traditional approaches of handling imprecise input, such as enlarging targets, area cursors, sticky targets and target gravity techniques [3] do not work for imaginary reality games, as they interfere with gameplay when scaled to this level: as illustrated by Figure 7, enlarged or magnetic players tend to fully "occlude" players behind them, preventing those players from *ever* getting the ball.

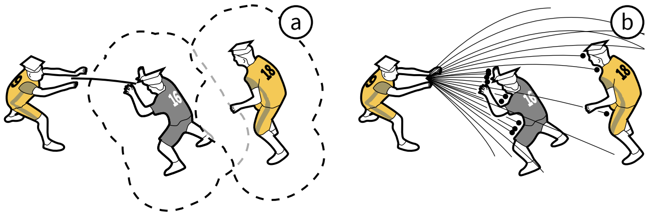

Figure 7: (a) Traditional enlarging of players helps pass the ball but also prevents the player on the left from *ever* passing past the opposing player in the middle. (b) The quantum engine instead samples all plausible ball trajectories, allowing it to pick outcomes that lead to enjoyable game play.

The quantum engine therefore takes a different approach: it first samples *all* plausible outcomes of the current move and their probabilities *and then* makes a well-informed decision, i.e., it chooses in *hindsight* which of the possible outcomes to "make real".

Rather than computing the trajectory of *the* ball, it computes the trajectories of *many versions of* the same ball, here 500 of them. It does so by representing the ball as a collection of *ball particles*, each of which represents a different plausible trajectory of the ball. The system's debug view (e.g., Figure 1b) reveals the trajectories of ball particles as white lines. The quantum engine evaluates each individual particle using regular Newtonian physics and aggregates the probabilities of each outcome. The quantum engine then passes this information to the game engine, which applies modifiers and makes the decision.

As illustrated by Figure 7b, our particle approach allows handling situations with multiple competing targets. As the shown player passes the ball, some ball particles hit the teammate, some the opposing player, and some go out. The quantum engine determines the raw probabilities of the three outcomes by simply counting particles; it then passes on these probabilities to the game engine. In order to favor enjoyable gameplay, the game engine increases the probability of the pass to the teammate and decreases the probability of the ball going out. Finally, it makes the decision randomly according to these probabilities and announces it to the players.

Benefits of the Quantum Engine Approach

While the original objective behind the development of the quantum engine was to help players acquire the invisible ball, it ended up providing a lot of additional value to our games:

1. *Optimize gameflow:* The quantum engine provides the game engine with choice. This choice allows the game engine to favor outcomes that lead to enjoyable gameplay, such as those that keep the ball in the game, enable multiplayer combos, etc.

2. *Hindsight:* Unlike traditional gaming systems that use a visible ball, imaginary reality games do not have to commit to a trajectory until players receive feedback [17]—which is not until the ball has been *received* or until it hits an object. This extra time allows the engine to consider the *entire* move when optimizing for gameflow. This also allows players to increase the chance of being passed to—by performing a great *receiving* gesture. This is another unusual game mechanic enabled by imaginary reality gaming.

3. *Forgiveness:* The quantum engine spreads particles broadly. As a result, even inexperienced players will always catch a few particles, even though they misread the body language of co-player or poorly extrapolate ball trajectories. Combined with a handicap system, this allows the system to help inexperienced players get into the game.

4. *Favor skillful play:* Despite all the modifications, the main contributor to any outcome is always a player's skill: by aiming better, moving into the most likely trajectory of a pass, and by receiving the ball at the perfect moment, skillful play leads to success. This creates an incentive to focus and to practice, adding a long-term incentive to imaginary reality gaming.

IMAGINARY REALITY GAMES ALLOW FOR COMPLEXITY

Despite the low bandwidth of their auditory feedback channel, imaginary reality games allow implementing reasonably complex games, such as basketball or soccer. The reason is that imaginary reality games (1) leverage players' experience with real-world sports and (2) allow them to extract information from mutual observation.

Learning Non-Trivial Rules—by Transfer

Despite the weakness of the auditory feedback channel, imaginary reality games can be learned quickly, because they allow players to transfer skills from the corresponding real-world sports (*transfer learning*, e.g., [7]).

To invite this type of knowledge transfer, we designed imaginary reality games to appear as similar as possible to their real-world counterpart. In particular, we hide the probabilistic inner workings and portray the games as if they were based on a single physical (yet invisible) ball. To do this: (1) we design auditory feedback so as to always refer to a single ball and never mention particles or probabilities. (2) The game engine never breaks the illusion of a single physical ball, by producing only game moves that are plausible under a single-ball interpretation.

Based on a foundation of rules transferred from real-world sports, designers of imaginary reality games are then free to (carefully) challenge players with power-ups and modes.

Following the Invisible Ball—by Observation

Imaginary reality games allow players to participate in *fast* gaming action primarily because what they see of the physical world, such as the actions of their co-players, *implies* the invisible ball position.

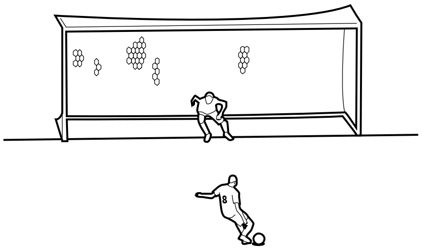

Figure 8: The goalie has to read the strikers body language to decide where to go. (Interestingly, as spectators we do the same. Even before this ball is being kicked, we can already tell that this one will probably go in, as the striker's body posture suggests that he is aiming right, while the goalie has his weight shifted in the opposite direction).

Interestingly, players are typically already trained in the main imaginary reality gaming skills, because these skills are similar to those involved in real-world sports. Consider Figure 8. To maximize his chances of saving the ball, the goalie *anticipates* the shot by (1) reading the opposing player's body posture, (2) imagining possible ball trajectories, and then (3) quickly moving to the expected ball location. A goalie in imaginary reality soccer performs essentially the same process (except that he cannot monitor the flying ball and instead has to solely rely on extrapolation).

CONTRIBUTION

The main contribution of this paper is the concept of imaginary reality gaming—games around an invisible ball.

Imaginary reality games maintain many properties of physical sports, such as physical exertion and *immediate* social interaction between players. At the same time, they allow introducing game elements from video games, such as power-ups, non-realistic physics, and player balancing.

Imaginary reality games produce a new game dynamic. Different not only from most traditional video games, but also from most display-based AR games, players do not look at screens or mobile devices anymore. Instead, our main game element is observation and imagination, as players engage in the shared illusion of an invisible ball.

Our main engineering contribution is the quantum engine, a parallel physics engine that (1) allows optimizing gameplay, (2) can make decisions in hindsight, and (3) helps beginners get into the game, yet that (4) favors skillful play.

IMAGINARY REALITY GAMES—A FAMILY OF GAMES

Imaginary reality gaming is a general gaming concept that applies to a range of games and can be implemented on a range of form factors and tracking systems. In our process towards imaginary reality basketball, we implemented imaginary reality versions of the tabletop games air hockey (tracked using Microsoft Kinect) and foosball (tracked on diffuse illumination) both shown in Figure 9. Both were instrumental in our overall design evolution as they offered easier debugging and allowed for faster design iteration, testing, and tweaking of game mechanics and algorithms, all of which are reflected in imaginary reality basketball.

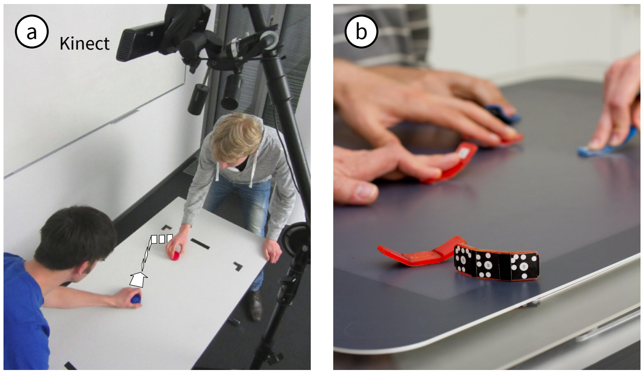

Figure 9: (a) A game of imaginary reality air hockey tracked using a Kinect. Here the blue player scores via the left wall. (b) Imaginary reality foosball tracked on a diffused illumination table. Players kick the ball by performing tilt gesture sequences using our custom *Band-aid* paddles.

RELATED WORK

Imaginary reality games build on physical games, probabilistic simulation, and spatial/non-visual interaction.

Physical World vs. Virtual World in HCI

In human-computer interaction, the disconnection between virtual and physical has been investigated in a wide range of contexts. In particular, the field of augmented reality is bridging this gap by merging a visual representation of the virtual world into the physical world users see.

Imaginary reality games draw inspiration from *imaginary interfaces* [6], i.e., interfaces that allow for spatial, non-visual interaction on mobile devices. Gustafson et al. showed that users can learn an imaginary interface based on previous experience with a physical device of identical layout (*transfer learning* [7]). Imaginary reality games might be considered a form of *shared* imaginary interfaces. While imaginary interfaces leave out display to achieve a preferable form factor (*ultra-mobile* devices), imaginary reality games leave out displays to create a new game mechanic—one that cannot be obtained with screens.

Probabilistic Simulation

The quantum engine presented in this paper takes a probabilistic approach to ball trajectories. Probabilistic approaches have been used in a variety of contexts, including processing uncertain input to user interfaces [18]. In graphics, Chenney et al. used Markov chains to generate plausible-looking animations that satisfy a given set of constraints [4]. Twigg and James allow users to create animation sequences by first generating many sequences and then eliminating those that do not meet a given constraint [21].

Games that Combine the Virtual and the Physical

Several researchers explored how to add virtual properties to physical ball games. Ichikawa and Nojima presented a ball than can change its trajectory by ejecting gas [8]. Izuta et al. added challenge to a ball game by occasionally making the ball invisible by turning its illumination off [9].

Virtual game objects in physical games have been explored by mixed reality and pervasive games [12]. *Airhockey Over a Distance* locally re-enacts the shots of an opponent located at a remote site [14]. Players of *AR²Hockey* use physical mallets to shoot a puck that is only visible on head-mounted displays [15]. Jeong et al. use a haptic cave to enable users to pass virtual balls to virtual co-players [11]. Jebara et al. display hints for how to tackle the current shot in pool billiard on a head–mounted display [10].

VI games are games for visually impaired users. Like imaginary reality games, many are inspired by real-world sports. VI games substitute the visual channel, e.g., with vibro-tactile or audio (e.g., [13, 19]). VI Pong is a spatial game that uses vibro-tactile feedback to guide players to the point where the pong ball will hit [19]. When playing VI Tennis, in contrast, players try to perform a swing gesture at the right moment [13]. Paralympics games for the visually disabled are closer renditions of real-world sports—they use audio-enhanced balls. Imaginary Reality games resemble VI games in that the ball is invisible. However, they *rely* on players' ability to see, which allows players to strategize around the locations and current actions of co-players and to anticipate ball trajectories. This comes at the expense of not being playable eyes-free.

ALGORITHM OF THE QUANTUM ENGINE

To allow readers to implement *their* quantum games, we now provide a detailed description of the algorithm. The goal of the algorithm is to place ball particles where players and spectators expect the ball to be located—more particles at more likely positions. For consistency, we use language from imaginary reality basketball, but the mechanism applies to all types of 2D and 3D imaginary reality games we have built. The quantum engine uses the following rules:

Initialization At the beginning of the game a single particle is dropped into the game in a physically marked respawn location, such as the playfield center.

Splitting If the probability *p* of a particle is above a global threshold, say 1/250, the particle splits into two new particles, each of which represent half the original probability. Repeated application of the rule causes the initial particle to spawn the desired number of particles, e.g. 512.

Brownian Motion If a ball drops, players' knowledge of the ball location tends to "blur" over time. By letting particles move slowly but randomly, the ball "spreads" over time, helping players to find the ball; eventually this allows players to pick up the ball anywhere.

Collisions based on Probe lines Figure 10a illustrates the algorithm—here the engine has to decide whether Player 16 intercepts the ball, whether the pass goes to teammate 18, or whether the ball goes out.

To allow for the most informed decision, the game engine delays its decisions as long as possible, i.e., until events that may result in auditory output. This is the case at time t_1, when particles reach Player 16. The game engine now has to make its first decision. It may either implement the intercept or run with "whatever the future might bring".

The quantum engine computes the raw probability of the intercept: if Player 16 has intercepted, say, 200 of 500 particles, (the engine computes this as a collision of the particles with a cylinder surrounding the player) this player has intercepted the actual ball with a raw probability of 40%. However, to optimize gameplay the game engine adjusts raw probabilities based on how desirable the individual outcomes are. The immediate outcome of Player 16 intercepting the ball is well defined—it receives the probability adjustment for "intercepts".

In order to estimate how desirable "whatever the future might bring" is, the quantum engine extrapolates particles' trajectories into the future using what we call *probe lines*. Probe lines predict the future under the assumption that players do not move, but this assumption tends to be reasonable on the involved time scales. Here, the probe lines find a possible collision with Player 18 (say 20%) and a possibility for the ball to go out (40%). The game engine adjusts these probabilities. It then normalizes probabilities, and decides randomly based on the resulting probabilities.

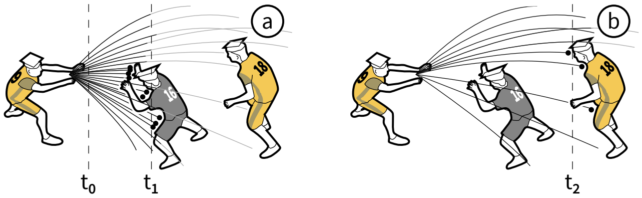

Figure 10: (a) t_0: Player 9 on the left is passing the ball. At t_1, the ball has reached Player 16 and the quantum engine makes its first decision; here it lets be ball through. (b) t_2: The ball has reached Player 18 and the engine decides again.

Let's imagine that the outcome is that Player 16 does not intercept the ball. The particles intercepted by Player 16 now die off and the engine awaits the next collision. At time t_2 the game engine has to decide again: complete the pass to teammate Player 18 or choose the remaining future. All remaining probe lines predict that the ball will go out, which is an undesirable event. The game engine adjusts probabilities accordingly, making the completed pass more likely than the ball going out, i.e., the game engine will most likely complete the pass to teammate 18.

Elastic collisions If the ball collides with an elastic object some, but not all particle move on. The quantum engine picks one particle as the *primary outcome* and this particle determines the auditory feedback. The quantum engine now preserves all particles that would have produced *similar* auditory feedback in *roughly* the same timing. It deletes all other particles, as the auditory feedback has made them *implausible*.

Power-ups In addition to power-ups that modify space, imaginary reality games allow for a range of power-ups that can be implemented by simply increasing or decreasing probabilities. *Penalty:* the probabilities of particles hitting a player are reduced. If set to a very low value, the player is essentially "out", as this player cannot receive the ball anymore. *Attract:* the probabilities of particles hitting a player are increased, helping this player to receive passes and intercept the ball. *Goal seeker:* increases the probabilities of particles that hit the opposing basket, etc. Handicaps work the same way, but persist throughout the game, while most power-ups expire after a few seconds.

EVALUATION

We conducted a brief qualitative study with 30 participants (aged between 18 and 25, 7 female) in the form of five 3 vs. 3 imaginary basketball matches on the 6×4m court. Before each match, the players were introduced to the game rules, but not the underlying mechanisms. They then practiced passing the ball for about 2min. Each participant then played for about 20min.

Results

On a Likert scale (1 = unpleasant, 7 = enjoyable), participants rated game play as 5.9, so clearly enjoyable. They judged that "skillful play" contributed slightly more than "random" (4.2), one player called it "a realistic balance between random and skill." This supports our claim of "enjoyable gameplay while still favoring skillful play."

Five players felt that imaginary basketball was "surprisingly realistic" or "close to actual Basketball." Many players wanted to play longer and one wrote: "We had team spirit and wanted to beat the other team." Four participants considered imaginary basketball as "more tiring than I imagined" and one "faster than real basketball." Asked which game element they liked best, nine participants chose power-ups and eight passing.

Two participants said that "keeping the overview [felt] similar to actual basketball." However, one player mentioned that, compared to actual basketball, "feedback isn't enough." With respect to learning the game, two players mentioned that imaginary basketball was "confusing at first, but actually easy to learn" and "intuitive after a few passes." Two players stated that in imaginary basketball, "height [is] not important", allowing "people of different skill levels to play together."

CONCLUSIONS

In this paper, we presented imaginary reality games, physical ball games without a visible ball. Our main contribution is the game concept itself, which creates a class of games around an invisible ball. We also demonstrate how to implement the concept using a custom probabilistic physics engine. As future work, we plan to explore the concept more broadly, to create more instances of imaginary reality games on a variety of platforms and at different scales.

ACKNOWLEDGEMENTS

We thank Tobias Mohr, Vasily Kirilichev, Dietmar Funck, Konstantin Kaefer, Marvin Killing, Max Plauth, Christoph Sterz, Stefanie Platz, Philipp Tessenow, Katherine Isbister.

REFERENCES

1. Alvar. http://virtual.vtt.fi/virtual/proj2/multimedia/alvar.html
2. Axivity. Wax3 motion sensor. http://www.axivity.com.
3. Bateman, S., Mandryk, R.L., Stach, T., and Gutwin, C. Target assistance for subtly balancing competitive play. *Proc. CHI'11,* 2355–2364.
4. Chenney, S. & Forsyth, D., Sampling plausible solutions to multi-body constraint problems. *Proc. Siggragh'00*, 219–228.
5. Cheok, A., Goh, K., Liu, W., Farbiz, F., Fong, S., Teo, S., Li, Y., & Yang, X. Human Pacman: A mobile wide-area entertainment system based on physical, social, & ubiquitous computing. *Personal & Ubiquitous Computing* 8(2), 2004:71–81.
6. Gustafson, S., Bierwirth, D., and Baudisch, P. Imaginary interfaces: Spatial interaction with empty hands and without visual feedback. *Proc. UIST'10*, 3–12.
7. Gustafson, S., Holz, C., and Baudisch, P. Imaginary phone: Learning imaginary interfaces by transferring spatial memory from a familiar device. *Proc. UIST'11*. 283–292.
8. Ichikawa, T., and Nojima, T. Development of the Motion-Controllable Ball. *Proc UIST'10,* 425-426.
9. Izuta, O., Sato, T., Kodama, S., and Koike, H. Bouncing Star Project: Design and Development of Augmented Sports Applications Using a Ball Including Electronic and Wireless Modules. *Proc. AH'10,* 22:1-22:7.
10. Jebada, T., Eyster, C., Weaver, J., Starner, T., & Pentland, A. Stochasticks: Augmenting the billiards experience with probabilistic vision and wearable computers. *ISWC'97*, 138–145.
11. Jeong, S., Hashimoto, N., and Makoto, S. A Novel Interaction System with Force Feedback Between Real – and Virtual Human. *Proc. ACE'04,* 61-66.
12. Magerkurth, C., Cheok, A.D., Mandryk, R.L., and Nilsen, T. Pervasive games: Bringing computer entertainment back to the real world. *Computers in Entertainment*, 3(3), 2005:1–19.
13. Morelli, T., Foley, J., Columna, L., Lieberman, L., and Folmer, E. VI-Tennis: a vibrotactile/audio exergame for players who are visually impaired. *Proc. FDG '10*, 147-154.
14. Mueller, F., Cole, L., Brien, S., and Walmink, W. Airhockey over a distance. *Proc. ACE'06*, Article number 70.
15. Ohshima, T., Satoh, K., Yamamoto, H., and Tamura, H. AR^{2}Hockey: A case study of collaborative augmented reality. *Proc. VRAIS'98*. 268–275.
16. Piekarski, W., and Thomas, B. ARQuake: The outdoor augmented reality gaming system. *CACM* 45(1), 2002:36–38.
17. Schrödinger, E. Die gegenwärtige Situation in der Quantenmechanik. *Die Naturwissenschaften*, 23(49), 1935:823-828.
18. Schwarz, J., Mankoff, J., and Hudson, S.E. Monte Carlo methods for managing interactive state, action and feedback under uncertainty. *Proc. UIST'11*, 235–244.
19. Strachan, S., Zophoniasson, H., and Anastassova, M. ViPong: Probabilistic Haptic Gaming for the Visually Impaired. *Proc. HAID'12,* 10-12.
20. Sugano, Y., Mochizuki, Y., Usui, T., & Okude, N. Shootball: Ball Sport Using Dynamic Goals. *Proc. ACE'07,* 262-263.
21. Twigg, C.D., and James, D.L. Many-worlds browsing for control of multibody dynamics. *TOG* 26(3), 2007:14:1–14:8.
22. Yamabe, T., Iwate, T., Shichinohe, T., and Nakajima, T. Prototyping Augmented Traditional Games: Concept, Design and Case Studies. *Proc. GPC'11,* 105-116.