Target Assistance for Subtly Balancing Competitive Play

Scott Bateman[1], Regan L. Mandryk[1], Tadeusz Stach[2], Carl Gutwin[1]

[1]University of Saskatchewan
Saskatoon, Saskatchewan, Canada
<firstname.lastname>@usask.ca

[2]Queen's University
Kingston, Ontario, Canada
tstach@cs.queensu.ca

ABSTRACT
In games where skills such as targeting are critical to winning, it is difficult for players with different skill levels to have a competitive and engaging experience. Although several mechanisms for accommodating different skill levels have been proposed, traditional approaches can be too obvious and can change the nature of the game. For games involving aiming, we propose the use of target assistance techniques (such as area cursors, target gravity, and sticky targets) to accommodate skill imbalances. We compared three techniques in a study, and found that area cursors and target gravity significantly reduced score differential in a shooting-gallery game. Further, less skilled players reported having more fun when the techniques helped them be more competitive, and even after they learned assistance was given, felt that this form of balancing was good for group gameplay. Our results show that target assistance techniques can make target-based games more competitive for shared play.

Author Keywords
Game design, game balance, target assistance, competition.

ACM Classification Keywords
H.5.3 Group and Organization Interfaces: CSCW.

General Terms
Algorithms, Design, Experimentation, Human Factors.

INTRODUCTION
Many kinds of video games, such as shooting or driving games, involve competition between two or more players. In these games, success is based on motor-control skills such as aiming or steering – and for the game to be evenly matched, players' skill levels must be roughly equal. If one player is far more skilled than the other, the game will not be competitive and will be less fun to play. Because many games rely on competition as means to provide fun, finding ways to provide an evenly-matched contest is desirable.

Several approaches have been used to address imbalance between players of different skill levels. Techniques in real-world games include handicaps in golf, extra points given to a player at the start of a score-based game, or explicit disadvantages such as playing with one's wrong hand in tennis. Methods in video games include automatic 'catch-ups' in racing games to bring slower players back into the action, differential capabilities (e.g., restricting a player to weaker weapons), or score-based changes (e.g., "fatboy mods" make successful players easier to shoot).

The fact that skill-accommodation mechanisms have been invented in so many different contexts shows that balancing for differing skill levels is an important concern for game designers. However, there are still problems with traditional mechanisms. First, they are often too obvious to players – it is often clear that one player has been given an advantage to make up for a lack of skill. This can lead to feelings of artificiality [8]; and many players do not want to win a game only because of an advantage. Second, traditional accommodation mechanisms can change gameplay – such as when players have different weapons, or must achieve different objectives; in these cases, two people can end up playing different individual games rather than one shared game. Third, traditional methods are usually static – they are defined and set before the game. If the accommodation is incorrectly calculated (e.g., the weaker player receives too few initial points), the game can remain unbalanced.

Overcoming these problems could improve group play for users with different skill levels. In this paper we present the idea of using *target assistance* to accomplish these goals for games that involve aiming and shooting. Target assistance techniques – such as area cursors [20], gravity [5], or sticky targets [7] – have been shown to improve targeting performance (e.g., [5,7,20,22,33]). Target assistance has been seen in games before – to reduce variance in game controllers, or as a way to cheat in online shooter games. The techniques have not previously been used to balance different skill levels, but since they can be applied differentially to different players, they hold considerable promise for these situations. Target assistance can also solve the problems of traditional balancing mechanisms: they are often minimally perceptible [22]; they need not change the basic interactions with a game; and they can be changed adaptively, scaling assistance as needed.

To test the usefulness of target assistance as a balancing technique, we carried out a controlled experiment with pairs of differently-skilled players in a simple shooting-gallery game. The experiment compared the effectiveness of three techniques (area cursors, target gravity, and sticky targets) for balancing player performance, and also examined the perceptibility of the techniques, the ways in which they altered game mechanics, and the degree to which they affected players' sense of fun. Techniques were tested in both a static mode – providing a fixed amount of assistance to the weaker player – and an adaptive mode – where assistance is scaled based on the current score differential.

Our study showed that games were more balanced with area cursors and target gravity than when no technique was used. These techniques reduced the average score differential from 47 points with no technique to less than 10 points, and the static versions resulted in the weaker player winning about half the games. The study also showed that the target assistance techniques were not perceived by the players, did not affect player ratings of fairness, and improved the sense of fun for assisted players while not affecting the fun had by non-assisted players. After we revealed that one player had been assisted, 22 of 24 participants agreed that balancing the game benefits group gameplay.

Our study is the first to show that target assistance can be an effective tool for improving competition in situations where players have different skill levels. Our results provide game designers with new tools that can extend the enjoyment of shared gaming to a much wider audience.

RELATED WORK: TARGET ASSISTANCE

The goal of target assistance is to help users select on-screen targets quickly and accurately. The underlying principles of aimed movement are well understood. In particular, Fitts's Law [13] states that targeting difficulty is predicted by the *index of difficulty (ID)*, which is calculated based on the size of the target (its width) and its distance from the starting location (its amplitude) [21]. Selecting a target involves three phases [24]: a ballistic motion, a corrective phase, and a final acquisition phase where the pointer is moved into the target and the selection action is performed. Assistance techniques have been directed at all of these phases and can be organized into three groups: manipulation of amplitude, width, or both (see review in [4]). We briefly review this work to provide the basis for our investigation into target assistance in video games.

Amplitude Manipulation

One class of techniques reduces targeting time by reducing the distance between the pointer and target. Some techniques move targets closer to the cursor [6]; the decision about which targets to move is determined by a command gesture or by the initial movement of the cursor. Other techniques warp the cursor to the target by predicting the end location of the targeting motion [3]. Both methods work well for large displays, but prediction of the final position of a moving cursor remains a problem [4].

Width Manipulation

A second class of techniques increases the target width in visual space or motor space. Visual expansion makes the target appear bigger (e.g., using a fisheye lens), which can help users determine whether their pointer is on the target; however, visual-space expansion can make users think that targets are larger in motor space than in reality, leading to errors. Increasing the target size in both visual and motor space still leads to problems, such as occlusion [23].

When the space between targets is not used for other purposes, the effective area of targets can be increased by enlarging the size of the target or the activation size of the cursor. The area cursor technique [20] increases the size of the cursor, from a single point to a larger square area. This effectively increases target size – a target may be selected when any part of the cursor's square intersects any region of the target. A major drawback of area cursor is the problem of the cursor intersecting multiple targets; it cannot determine which is the intended target. DynaSpot addresses the major drawbacks of area cursor by only increasing the area of the cursor at higher movement speeds, and reducing it to a single point for slower, more precise targeting [9].

Another approach is to allow the cursor to make selections even when it is not within a target. Bubble Cursor maps every point in the display to the closest target, allowing selection regardless of the actual cursor location [14].

A third approach for increasing width, called 'sticky targets' or 'pseudo-haptics,' changes the control-to-display ratio of the input device when the cursor is over a selectable target [7]. The user must move the pointing device farther to achieve the same cursor movement, effectively increasing the motor space of the target. Sticky targets can improve aiming time in both 1D [7, 22] and 2D [33] tasks. User perception of CD manipulation is not strong – low to moderate levels of the effect are rarely noticed [22].

Manipulating both Amplitude and Width

A third class of techniques changes both the size of the targets and their distance. Some CD-manipulation techniques alter the cursor position throughout the targeting motion [18,33], rather than just over the target.

'Force fields' [2] and 'target gravity' [5] are two such approaches. The techniques warp the cursor by increasing motions toward the target, and motions away are decreased. A similar technique leaves 'magnetic dust' around windows and widgets as they are used. Dust accumulates over time and frequently used interface elements become more attractive [18]. The TractorBeam technique for distant pointing on tabletop displays compared increasing target width and decreasing target distance (by snapping to the target). Users performed equally with both techniques [28].

RELATED WORK: GAME BALANCING TECHNIQUES

Flow is a balance between anxiety and boredom [10], and is an important part of game design. To achieve optimal flow, games must balance challenge and skill [12]. If a game is too easy, players may become bored. Conversely, if a game is too hard, players may become frustrated. In this section, we explore game play balancing and player balancing.

Game Play Balancing

Competition is a contest between two or more players, and has been shown to be a motivating factor for participating in multiplayer video games [32]. To maintain flow, competition between players must be balanced – that is, players of equal skill and experience have an equal chance of winning [1]. If a player or group has an advantage, opposing players may have to perform at a higher level than the advantaged players, and may be incapable of winning the game. If unbalanced game elements are exploited, the advantaged player may no longer be challenged, and the disadvantaged player may quickly become frustrated [26]. Therefore, balancing game elements is an important aspect of video game design that has been recognized in industry, where simulations and extensive play testing are often used to uncover potentially unbalanced game elements [15].

Player Balancing

Although games often provide optimal game play balance, it can be difficult to balance the differences between players [1]. In video games, players have varying skill, experience, tactics, and reaction time [30], and because many games reward skill, novices can have difficulty playing against experienced players. We review three common approaches for addressing the differences between players below.

Matchmaking

Grouping players by their abilities is one technique used to adjust competition. For example, ranking and ladders are often used in sports and traditional games (e.g., squash and chess). Similarly, many multiplayer online games track performance to match players and balance teams [26]. TrueSkill assesses player skill based on a number of game outcomes using Bayesian networks and the Elo chess scoring system, and has been widely used in games such as *Halo 2* [16]. A limiting factor of matchmaking is that it requires a measure of player skill, which can be difficult to determine when players initially join a game [35].

Asymmetric Roles

Providing unique tasks is another approach for addressing skill differences. Many team sports include several roles requiring different skills. For example, positions in American football differ between roles; the quarterback's job is to handle the ball on every offensive play, while an offensive lineman may play many games without ever touching the ball. Asymmetric roles can also be found in video games. Some multiplayer games (e.g., *Alien Swarm)* require players to fill several roles to succeed (e.g., Officer, Special Weapons, Medic, and Tech). Asymmetric roles allow players with different skill sets to play together; however, a team may be limited by the performance of the weakest member. Further, the use of asymmetric roles does not address the situation of competitive play.

Difficulty Adjustment

Difficulty adjustment is a popular way to support different skill levels in video games – many games allow a choice of difficulty setting. Adaptive difficulty adjustments alter game elements in real-time based on performance [17].

Adaptive difficulty adjustment has been applied in various game genres for single and multiplayer games. A version of the classic *Pong* game was implemented to support play among children and parents of different skill [19]. Players' racquet lengths, movement speed, and incoming ball speed were adjusted based on performance. Generating game content (e.g., in-game obstacles and enemies) based on the ability of the player is another approach [29]; for example, *Infinite Mario* provides endless new game levels that are suitable for a player's ability (www.marioai.org).

Many arcade-style racing games include "catch-up" or "rubber band" systems that provide advantages to a losing player. For example, in *Mario Kart* trailing players receive items that give them a greater chance of capturing the lead [26]. The "fatboy" modification of *Unreal Tournament* adjusts the size of a player's avatar, increasing its width based on the number of kills scored. This makes successful players easier targets than losing players.

Adaptive difficulty adjustments have several potential limiting factors. For example, the "catch-up" effect that occurs in *Mario Kart* can lead to a sense of frustration as more experienced players begin to feel cheated [26], and scaling game difficulty based on a player's progress can result in the player losing their sense of achievement [8].

RELATED WORK: TARGET ASSISTANCE IN GAMES

Games involving aiming tasks often include some form of computer-supported target assistance. Many FPS console-games include target assistance in order to address the lack of precision when aiming with a thumb-stick on a gamepad [25]. Alternatively, third-party software called "aimbots" have been developed to provide target assistance for many FPS games [34]. These systems are often considered as "cheats" by serious gamers [31], as they may provide an unfair advantage. Common approaches for target assistance include: reticule magnetism and target lock-on.

Reticule magnetism, like the sticky target technique, works by adjusting the control-to-display ratio during targeting; as the reticule (cursor) passes over a target, it moves more slowly. This technique is common in FPS console games (e.g., the *Halo* series), as it can be difficult to make fine aiming adjustments using a gamepad or thumbstick [27]. Reticule magnetism can, however, lead to unexpected acceleration of the cursor. As a player moves off of a target, friction is no longer applied, causing the cursor to swing rapidly in the direction they are being moved [25].

Locking a player's cursor onto potential targets is another form of target assistance in games (similar to the Tractor Beam technique [28]). For example, pressing a button on the gamepad will lock the cursor onto nearby enemies in games such as *The Legend of Zelda: Twilight Princess.* A potential downside of target lock-on is that the cursor may lock onto an unintended target as they attempt to aim.

STUDY

To understand how target assistance can balance play, we studied three target assistance techniques applied statically or dynamically, in a two-person shooting gallery game.

Figure 1. The Shooting Gallery game in a level with moving targets. Player 1 has missed and Player 2 has hit.

Apparatus

We designed a simple shooting-gallery game where the goal is to score points by shooting on-screen targets with a Nintendo Wii Remote (Wiimote). A cursor (crosshairs) is displayed for each player (colored yellow or red) at the absolute position reported by the Wiimote. Players score a point for shooting a target and lose a point for each miss. Targets are shot by placing the centre of the cursor (a 1px point) over any region of the target and pressing the trigger. Sound feedback identifies hits and misses. Visual feedback shows hits and misses with a '+1' or '-1' at the shot point for one second, in the same colour as the player's cursor (see Figure 1). We created five game levels that varied the size, movement, speed, and time of appearance of the targets. The goal was not to fully enumerate the space of target behaviours, but to create several interesting levels of increasing difficulty. The game was built using C# and XNA 3.1 with a screen refresh rate of more than 60 fps. The system ran on a Windows 7, Intel Core 2 Duo machine and a Dell 42-inch, 1024x768 plasma screen. Participants sat on a couch or stood approximately 250cm from the screen.

Techniques Chosen for the Comparison Study

We chose three techniques that have been considered in prior target assistance research: sticky targets that change effective target width when the cursor is on the target [7, 22,33]; target gravity that attracts the cursor towards targets [5]; and an area cursor that increases the size of the affected area when a shot is taken [20]. These techniques were selected because we felt they would improve performance, yet offer low perceptibility. Further, because these techniques do not fundamentally change the on-screen appearance of targets or the cursor (e.g., visual expansion) or the way that targets are selected (e.g., bubble cursor) they may be widely applicable in real-world video games.

We tested these techniques using both static and adaptive target assistance. The static approach provided a constant level of assistance, while the adaptive approach adjusted the assistance depending on the difference in score.

The Sticky Targets Technique

Target stickiness was applied when the cursor was over a target by changing the CD ratio of the input: the lower the ratio, the stickier the target. For example, a ratio of 0.4 meant that a movement over a target would result in only 40% of the normal on-screen cursor movement.

A previous study of sticky targets with the Wiimote found that the accumulated difference between cursor position and Wiimote position from passing through sticky targets could lead to increasing 'cursor drift' [5]. While not a problem for player performance, the infrared light source could become out of range of the Wiimote camera after accumulated cursor drift. To side-step the issue, our implementation returned the cursor to the position it would have otherwise reached had it not passed through a sticky target. This design choice resulted in the cursor rapidly accelerating off targets when exited or shot, and has also been described in observations of cursor magnetism in practice [25].

The Target Gravity Technique

Target gravity [5] is similar to the 'force field' technique [2]; but instead of restricting a target's attraction to a limited range, all targets provide an attractive force. Because target gravity is inversely proportional to the square of the distance between the cursor and target, the influence of a target decreases rapidly at greater distances.

Our gravity effect is calculated as follows. For n targets, let $p_1, p_2,..p_n$ be the positions of the targets with radii $r_1, r_2,..r_n$. Let p_0 represent the true position of the cursor (i.e., without any gravity effect applied), and let p_w be the warped position. Let G be the 'gravitational constant' (i.e., weight multiplier). Then, for each target $i=1..n$, compute the target weight with Equation 1. Finally, compute the warped position of the cursor using Equation 2.

The warped position is a weighted average of the true cursor position and the positions of each target. The weight for the cursor position is fixed at 1.0. Manipulating G changes the strength of the gravity effect, and bigger targets provide more gravity (this formulation is similar to previous work in virtual world interactions [11]). Target gravity has been shown to improve performance using a Wiimote [5].

$$w_i = \frac{G r_i^2}{|p_0 - p_i|^2 + 1}, \text{ with } w_0 = 1 \quad (1) \qquad p_w = \frac{\sum_{i=0}^{n} w_i p_i}{\sum_{i=0}^{n} w_i} \quad (2)$$

The Area Cursor Technique

Our area cursor implementation is closely related to the original implementation [20], but in order to minimize perceptibility the physical size of the cursor remains consistent. Further, our technique does change how targets are selected or how the appearance of intersected targets, like newer related techniques [9,14]. However, like the newer techniques we use a circular activation area. The effect was imperceptible to the users before shooting. After shooting it would be possible for them to perceive the increased activation area.

Choosing Levels of Assistance Techniques

We conducted a pilot study to determine the levels of static assistance and upper limits for adaptive assistance. Eight participants tested 10 levels of each technique and rated how noticeable the assistance was on a 10-point scale.

We chose technique levels that combined performance improvement with low perceptibility. For the static levels we chose: area cursor, an area 52px in diameter; sticky targets, a CD-ratio of 0.2; gravity, a gravitational constant of 2.0. We also chose the upper limits of the adaptive assistance techniques. For sticky targets, a CD-ratio of less than 0.1 could make fast moving targets unselectable; the CD manipulation would slow the cursor too much to keep pace. A gravity constant greater than 6 could make it difficult to reach all targets, because the force exerted by closely-spaced targets was too strong to overcome. No upper limit was observed for the area cursor technique, although large area cursors could allow multiple targets to be hit with a single shot.

Adaptive assistance used the following algorithm. If the assisted player's score was greater than or equal to the unassisted player, then provide no assistance. Otherwise, give 10% of the static level for each point the assisted player trails the non-assisted player. For example, if the assisted player trailed by two points using the gravity technique, the resulting gravity coefficient would be .4 (i.e., 10% x static level x score difference = 10% x 2 x 2).

Task

Participants were tested in pairs and were told that they would be competing in a shooting gallery game where we were testing new "controller algorithms" for the Wiimote. Participants were *not* told that the algorithms gave assistance. We highlighted that we wanted their experience to be like a game with a friend, and they were free to speak to one another. They were asked to try and win each game, to have fun, but not to worry if they didn't win.

Player skill level was determined by an individual game. While one participant played the individual game, the other completed a demographics questionnaire in a separate room. The experimenter used a combination of score and completion time in the individual games to choose which player received assistance during competitive play.

The competitive play portion consisted of a series of games. Game 1 was a control condition; no assistance was given and players were told that this was the "normal algorithm". Games 2-4 presented the static assistance techniques (using a Latin square ordering across groups). Game 5 was another control game. Games 6-8 presented the adaptive techniques (using a Latin square ordering). Game 9 was a final control.

Each game started with a practice level, where the players could get accustomed to any differences in the controller algorithms. The game then randomly presented the five levels of play. A transition screen was presented between each level so that players could take a break, and see their scores. After each game the experimenter would announce the final score on the final transition screen to ensure participants were aware of their performance.

After each of Games 2-9, players answered a survey where they reported on their experience with the "controller algorithm." After the experiment, players were given another questionnaire that asked for their impressions of the game, their performance and the performance of their partner. The experimenter then revealed that the controller algorithms gave assistance to just one of them, and that the goal was to keep the game competitive and to make it more fun. They were told which player received assistance, and that this decision was based on their individual games. A final survey solicited their opinions on game balancing.

Participants

We recruited 24 players (14 male) who were undergraduate students (15), graduate students (6), or university staff (3). Participants were paired with a participant whom they did not know (9 groups), or with a friend (3 groups). Players selected for the assistance condition scored significantly lower (avg: 273.5) scores than non-assisted players (avg: 315.7) in the individual game ($t = -3.87$, $p<.001$).

Of the 12 assisted players (4 males, mean age 26.7), half reported not typically playing video games, with the rest spending under 3 hours per week. They rated their expertise with the Wiimote an average of 1.75 (sd=0.75) on a 7-point scale (1-novice, 7-expert), and five reported having played a shooting-gallery type game before. Of the 12 non-assisted players (10 males, mean age 24.3), all but one played video games regularly (three spent under 3 hours/week, and eight spent 4 or more hours). Non-assisted players rated their Wiimote expertise an average of 4.7 (sd=1.5) on a 7-point scale and all had previously played shooting-gallery games.

Data Analyses

All data were analyzed separately for assisted and non-assisted players. Performance data were gathered from computer logs and analyzed with one-way ANOVAs. Post-hoc tests were conducted using Tukey's HSD. Survey results were analyzed using Friedman's ANOVA for related samples; pairwise comparisons used Wilcoxon Signed Ranks Tests for 2-related samples; correlations used Spearman's rho. For all tests, α was set at 0.05. We conducted a series of planned comparison tests.

RESULTS

Targeting Performance

As players competed in groups, we examined performance at the individual level and relative to their partner.

Did the hit ratio change when one player was assisted? Hit ratio was calculated as the number of successful target hits divided by the total number of shots for an entire game. Hit ratio did not change for non-assisted players when their partner was provided with assistance ($F_{6,77}$=.225, p=.967). For assisted players, there was a significant effect of technique on hit ratio ($F_{6,77}$=26.3, p<.001). Post-hoc tests showed that area cursors (both static and adaptive) yielded the best hit ratio (all p<.04), and that static gravity was better than sticky targets and no assistance (all p<.02), and adaptive gravity was better than no assistance (p=.038). There were no differences between the static and adaptive versions of any of the techniques (all p>.30). See Figure 2.

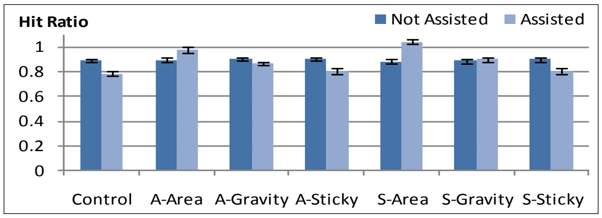

Figure 2: Mean hit ratio (±SE) by technique for assisted and non-assisted players. A-Adaptive, S-Static.

Was game score closer between the players when one was assisted? Game score was a combination of target hits (+1) and misses (-1). Although a sign of individual performance, it is linked to the performance of a player's partner, as a partner hitting more targets would result in fewer targets available for a player to hit. Game score is the measure that was shown to the participants as they played the levels.

To answer our question, we calculated the difference in score between competitors after each game, and conducted a one-way ANOVA with assistance type as the factor. The difference in game score was significantly affected by assistance type ($F_{6,77}$=11.2, p<.001). Post-hoc tests showed that area cursors resulted in a smaller game score difference than sticky targets or no assistance (all p<.002). Static gravity yielded less difference than sticky targets or no assistance (all p<.006), and adaptive gravity produced less difference than no assistance (p=.039). See Figure 3.

Were there learning or ordering effects? We used the final scores in the three competitive control games to test for learning effects. No significant differences were found between control game scores for assisted ($F_{2,22}$=.797, p=.463) or unassisted ($F_{2,22}$=1.66, p=.213) players, suggesting that there was no important learning effect as the study progressed. Therefore, comparisons between static and dynamic techniques should not be biased by ordering.

How did assistance type affect the number of games that were won by the assisted player? We counted the number of games won by the assisted player. Chi-squared tests showed no difference between the number of games won and lost by the assisted player using static area cursor (8/12 won, χ^2_1=1.3, p=.248) or static gravity (5/12 won, χ^2_1=0.3, p=.564). All others had significant differences between the number of won and lost games (adaptive area cursor: 1/12 wins, no assistance: 2/36, all others: 0/12 wins, all p<.004).

Taken together the performance results show that assisting one player boosts their hit ratio, while not affecting the hit ratio of the non-assisted player. This boost in hit ratio evens the score between the players when using the area cursor and gravity techniques. Although the scores were closer when using area cursor and gravity (for both adaptive and static) than with no assistance, this only resulted in assisted players winning games with static techniques.

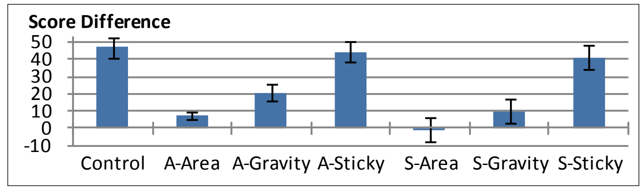

Figure 3. Mean score difference in points (unassisted–assisted) by technique. A-Adaptive, S-Static.

Perceptibility of Assistance Techniques

After each game, we asked users whether they noticed a difference in the controller algorithms compared to the normal case (4-point scale; lower-less noticeable). We also asked players about the difficulty of aiming, their perceived performance and the perceived performance of their partner after each game on a 5-point scale (lower-worse).

Did participants notice differences? There were no differences in the noticeability ratings between controller algorithms, for either assisted players (χ^2_7=6.6, p=.469) or non-assisted players (χ^2_7=6.8, p=.446). For example, there was no difference between static sticky targets and adaptive sticky targets. See Table 1.

	Noticeability (1-4)		Difficulty (1-5)	
	Not Assisted	Assisted	Not Assisted	Assisted
Control	2.6 (0.15)	2.1 (0.18)	3.2 (0.22)	2.4 (0.15)
A-Area	2.1 (0.26)	2.3 (0.26)	2.9 (0.23)	3.8 (0.21)
A-Gravity	2.3 (0.23)	1.9 (0.19)	2.8 (0.22)	3.1 (0.23)
A-Sticky	2.7 (0.31)	2.0 (0.28)	3.0 (0.30)	2.1 (0.26)
S-Area	2.6 (0.26)	2.5 (0.26)	2.8 (0.35)	3.6 (0.19)
S-Gravity	2.4 (0.26)	2.5 (0.34)	2.8 (0.27)	3.7 (0.14)
S-Sticky	2.4 (0.34)	2.4 (0.29)	3.7 (0.28)	2.3 (0.25)

Table 1. Mean noticeability and difficulty ratings ±SE. Lower is less noticeable/more difficult. A - Adaptive, S - Static.

Did players report differences in the difficulty of the techniques? Non-assisted players did not report differences in the difficulty of the controller (χ^2_7=8.8, p=.264), while assisted players did (χ^2_7=48.1, p=.264). Pairwise tests showed that the sticky technique was perceived as more difficult than area cursor or gravity for both the static and adaptive versions (all p< .02), that no assistance was more difficult than area cursor or gravity for the static techniques, and that no assistance and adaptive gravity were more difficult than adaptive area cursor (all p< .02). There were no differences between the static and adaptive versions of any of the techniques. See Table 1.

Did assisted players think they performed better when there was assistance provided? Assisted players noticed the change in their performance compared to the normal base case (χ^2_7=47.2, p<.001). They thought that they performed better with area cursors and gravity than with sticky targets or no assistance for the static techniques (all p<.01), and better with area cursors than sticky targets, gravity, or no assistance for adaptive techniques (all p<.05). In addition, adaptive gravity was perceived as resulting in better performance than adaptive sticky (p=.03). There were no differences between the static and adaptive versions of any of the techniques. See Table 2.

Did players notice the difference in their partner's performance? Non-assisted players did notice the change in their assisted partners' performance (χ^2_7=38.9, p<.001). They felt that their partner played better when provided with both area cursors and gravity over sticky targets and no assistance for both the static and adaptive techniques (all p<.05). There was no perceived difference between the static and adaptive versions of any of the techniques. Assisted participants did not think that their partners performed differently depending on the assistance they received themselves (χ^2_7=7.0, p=.430). See Table 2.

	Performance (1-5)		Partner's Performance (1-5)	
	Not Assisted	Assisted	Not Assisted	Assisted
Control	3.3 (0.19)	2.3 (0.16)	2.3 (0.20)	3.3 (0.14)
A-Area	2.7 (0.19)	3.9 (0.19)	3.6 (0.26)	3.3 (0.19)
A-Gravity	2.8 (0.21)	3.2 (0.24)	3.0 (0.21)	3.3 (0.19)
A-Sticky	3.6 (0.23)	2.2 (0.27)	2.2 (0.24)	3.2 (0.21)
S-Area	2.5 (0.36)	4.1 (0.19)	3.8 (0.25)	3.0 (0.17)
S-Gravity	2.8 (0.25)	3.8 (0.22)	3.5 (0.2)	3.2 (0.17)
S-Sticky	3.9 (0.26)	2.2 (0.35)	2.7 (0.23)	2.9 (0.23)

Table 2. Mean ratings ±SE for player performance and partner performance. Lower is worse. A - Adaptive, S - Static

Did participants without assistance think that their own performance changed when their partners were given assistance? Although the performance of non-assisted players didn't change, their performance relative to their competitor did. Non-assisted players noticed this change (χ^2_7=19.6, p=.007). They thought their own performance was better when their partner was assisted using sticky targets than with area cursor or gravity for both the static and adaptive techniques (all p<.03). There were no differences between the static and adaptive versions of any of the techniques. See Table 2.

Taken together, these results related to perceptibility and performance change suggest that assisted players attributed the difference in score to their own improvement, not to their partners playing poorly or to changes in their control mechanism, while non-assisted players perceived both an improvement in their partner's performance and a decrease in their own performance.

User Response to Game Balancing through Assistance

We asked the players how much fun it was to play the game after each assistance technique was used, at the end of the experiment, and after revealing that we were assisting one of the players, all on a 5-point scale.

Did participants rate their fun differently after playing with different assistance techniques? Players who were provided with assistance rated their fun differently after games, depending on what kind of assistance was provided (χ^2_7=29.7, p<.001). For the static techniques, area cursor and gravity were rated as more fun than sticky targets or no assistance (all p<.03). For the adaptive techniques, area cursor was perceived as more fun than any other technique (all p<.04). There were no differences between the perceived fun of any of the static or adaptive techniques. Players who did not receive assistance did not rate their fun differently after games where their partner was provided with assistance (χ^2_7=6.9, p=.439). Taken together, these results are interesting as play experience was strengthened for those receiving assistance, while not being weakened for those who weren't. See Figure 4.

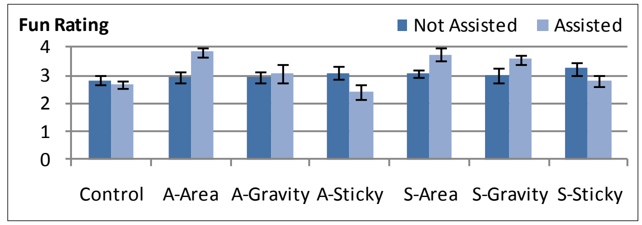

Figure 4. Mean fun ratings (±SE) (lower is less fun) by technique and player. A-Adaptive, S-Static.

Did players' performance affect how much fun they had? Fun ratings for each game were significantly related with game score differential for assisted players (p<.001, ρ=.379), but not for non-assisted players (p=.058, ρ=.194). Assisted players reported having more fun, the closer the score was in a game, but this did not come at the expense of non-assisted players' perception of fun.

Did players' ratings for fun change after finding out that there was assistance provided in the game? Neither assisted nor non-assisted players changed their ratings of fun after finding out that assistance was provided (Z=1.0, p=.257, Z=-1.1, p=.317 respectively).

Did participants think that it is fair to provide assistance to one player? Before revealing that assistance was provided, all assisted players felt that the games were fair, while 10 non-assisted players agreed. Two non-assisted players felt that games were unfair because previous experience could provide an advantage, and that fairness may have depended on how the algorithms were assigned. After finding out that one player was assisted, most non-assisted players felt that it was fair (2-No, 10-Yes). Non-assisted players commented that: it would broaden the demographics of their opponents (e.g., playing against grandmother or kids); it is better for casual gamers; and it would be more fun. Dissenting players felt that it would not be a true competition, but that it would be OK to provide assistance when playing against children. Assisted players were more split in their opinion (7-No, 5-Yes). Assisted players who thought it was fair commented that: it would improve the competition; it would increase the challenge for the better player; and that it would increase the fun. Those who thought it was unfair commented that: it doesn't give you an idea of your actual ability; the win doesn't count if you are being helped by a computer; it may not be as much fun for the better player; and the scores would be meaningless. Some assisted players commented that although there would be benefits to the group experience, it would be through unfair means.

Do players think that balancing benefits the group? Players agreed that balancing the game by providing assistance to one player can improve the experience for the group (11-Yes, 1-No for both assisted and non-assisted players). Comments included that it would: improve the competition; allow people of multiple ages and skill levels to compete together; provide a greater challenge to better players; be less frustrating for less-skilled players; and be more fun.

Would players want to know if one player was assisted? Most players would want to know (8-Yes, 4-No for both assisted and non-assisted). Those who wanted to know commented that: they would want to know if they were beaten by the person, or by the computer and the person; and to evaluate their actual ability. Those who didn't commented that: it would benefit the social aspect of games to not know; and it might ruin the experience if they did.

DISCUSSION

There are six main results from the comparison study.

- Two of the target-assist techniques (area cursors and gravity) significantly improved player performance, and significantly reduced score differences in the game.
- There was no significant difference between static and adaptive versions of the techniques, but the static versions did allow weaker players to win more often.
- Players did not notice differences between the controller algorithms (although they did notice better performance).
- Assisted players reported having more fun as games became more competitive, but this did not come at the expense of perceived fun of the non-assisted players.
- Players reported the game as being 'fair', and did not change their assessment after learning about assistance.
- After learning about the targeting techniques, assisted players' opinions were less favorable than non-assisted players, but both groups felt that assistance could benefit the group play experience.

Explanations for Results

Why did the techniques work (or not work)?

All of the assistance techniques have been effective in prior work, but only area cursors and gravity were beneficial here. One reason is that stickiness may be less useful in situations with moving targets, since the user must be on top of the target for the technique to take effect. Target gravity has an effect wherever the cursor is on the screen (and so avoids this problem), and area cursors allowed players to hit targets simply by being close. A second problem with sticky targets was that when leaving a target, the cursor had to jump to return to its actual location. This means that if the player moved through distractor targets en route to the desired target, the cursor could appear to skip. Area cursors were particularly good at improving performance because players could hit multiple targets at once. Although a success in our game, this could be a limitation in other games where users must pick out targets that are close to other objects that should not be hit.

Why did players not notice the assistance?

Players did not notice differences in the various "controller algorithms", but did notice when they (or their partner) performed better. As in previous work, it appears that people do not easily perceive target assistance [22otice when they (or their partner) performed better. As in previous work, it appears that people do not easily perceive target assistance [22]. There are three reasons. First, the effects of the techniques (if set up correctly) are subtle, and do not dramatically change the visible behavior of the on-screen objects. Second, noise in the input device hides some of the effects of the techniques: there is already some jitter in the way that the cursor tracks the motion of the Wiimote, and it is difficult for users to tell exactly where they are pointing with the device. Third, the player's attention is strongly focused on the game, not on the motion of the cursor, which reduces their ability to notice the effects of target assistance.

Which is better: static or adaptive assistance?

There was no significant difference in the effect of static or adaptive techniques on game scores or hit ratios, although static techniques allowed the weaker player to win more games than the adaptive techniques. This does not mean that adaptivity is a poor design, but rather highlights the differences between the approaches; pointing to the importance of considering the game outcome and the score differential. The adaptive techniques adjusted with each shot fired, meaning that as the score difference between players approached zero, the technique's strength also approached zero. This resulted in close games where the unassisted player won as opposed to games played with the static techniques where the assisted player also won. For situations where game outcome should be balanced (e.g., parents playing with children), it may be best to use static techniques or to change the adaptive techniques to allow for varied game outcome. For situations where the goal is to have close play, but still reward the stronger player with more wins, adaptive techniques may be more suitable.

Generalizing the Results

Here we consider whether the value of target assistance can be realized with other players, input devices, and games. First, our results suggest that the target assistance techniques will work with a wide variety of players, with a wide variety of skill differentials. In our study, we had some pairs who were closely matched, and others who were very far apart (overall the stronger player scored 15.5% higher than the weaker player in the individual game). The question for players with larger differentials are whether the maximum assistance value will be enough to bridge the gap, and whether an adaptive compensation algorithm will react quickly enough to maintain a reasonable balance. We believe that these issues can be easily addressed (e.g., as discussed below, multiple techniques could be applied in situations where there is a large difference).

Second, there is also reason to believe that these results will transfer to games using other input devices. For some devices (such as game controllers), target assistance is already being used, and our techniques should work for with little need for adjustment. For other devices, particularly those that are more precise (like a mouse), the techniques may be more perceivable (e.g., when a small vertical movement of the mouse results in a diagonal cursor movement). However, the attentional demands of games are likely to hide these effects in most cases.

Third, there is considerable opportunity to apply our results in other types of shooting games, although designers will have to adjust the techniques to suit a particular game. For example, first-person shooters could easily adopt these balancing techniques, but would have to ensure that the semantics of the game are not changed (e.g., a sniper rifle would be a poor match with an area cursor, as sniper rifles do not hit multiple targets). Game semantics, however, could be used as an additional resource for balancing techniques: for example, weapons with a wider spread of damage could be a natural fit for area cursors; spread radius could change based on the amount of assistance desired.

In addition, target assistance (or the basic ideas behind these techniques) could also be used in other game genres. In particular, the gravity technique could possibly be applied to driving games, giving weaker players better 'centre-line gravity' than stronger players. We plan to study the perceptibility of this technique in future work – but it is likely to be much less perceivable than the 'catch-up' adjustments that are currently used in some games.

Design Issues and Lessons for Designers

This study provides several lessons and raises several issues for designers wishing to employ target assistance.

- The main lesson for designers from this work is that target assistance techniques work very well as a balancing mechanism: they are able to compensate for widely different skill levels, without being overly noticeable. The techniques can be used immediately to improve the accessibility and competitiveness of shared games.
- For moving-target games, designers should first consider area cursors and target gravity. Area cursors were best at balancing games, but can only be used effectively in situations where hitting multiple targets is not a problem. Target gravity was also successful, and can work in situations with small targets. Sticky targets were not effective, and although they may be useful in other game genres, they seem to have too many drawbacks to be recommended for use in shooting games.
- It is possible for the techniques to be combined in situations where extra assistance is required. Although we did not test this, it seems clear that gravity and area cursors could both be applied to a player's controller. This would allow either a higher level of assistance – or perhaps more importantly, would allow the levels of each technique to be lowered, reducing overall perceptibility.
- It is extremely important that the values for the techniques (e.g., the maximum amounts of gravity or stickiness) be carefully chosen. The techniques can become highly obvious if these values are too high (e.g., the cursor becomes stuck to a target, or moves in the opposite direction to the player's arm movement).
- The parameters of the adaptation algorithm should also be carefully chosen. These should be matched both to the length of the game (so that the adaptive algorithm can have a reasonable effect before the game ends) and to the desired outcome (e.g., whether a designer wishes to give a boost to the weaker player, or to equalize the game).
- An important design issue is whether to reveal that target assistance is being used. Our participants were divided on whether they wanted to know about the techniques, it may be safest to make this a setting that players can see (although it need not be made obvious in the game itself). In some games that are expressly designed for casual use, or for walk-up play in public settings, designers could choose to add the technique without informing players – but in these cases, it will be important to ensure that the levels chosen for the techniques will keep the assistance below the perceivable level for most players.

CONCLUSIONS AND FUTURE WORK

Video games are an increasingly popular form of entertainment that are played by people of many different skill levels. In this paper we addressed the problem of skill imbalance using target assistance techniques, which have not previously been used for this purpose. Our study of three different techniques showed that target gravity and area cursors can balance competition between players of different abilities. We found that by making the game more competitive, less-skilled players report having more fun, and that this does not adversely affect the experience of the more skilled players. Target assistance techniques are particularly well suited for this application because they can be applied without being noticed by players, and need not change game mechanics.

In future work, we will investigate the application of similar techniques in other game genres, such as first-person shooters and driving games, to test whether our results generalize to other situations. Second, we will refine our adaptive algorithms to see if game outcomes can be more tightly controlled, and to see if technique combinations (e.g. gravity + area cursor) can provide stronger assistance with lower perceptibility. Finally, we will further investigate the real-world requirements of assistance techniques through longer-term testing, and testing with specific player types with larger differences in skill, such as teenagers and grandparents.

ACKNOWLEDGEMENTS

We would like to thank Stephen Damm for his work in implementing the experimental system. This research was supported by NSERC.

REFERENCES

1. Adams, E. *Fundamentals of Game Design*. New Riders, 2010.
2. Ahlström, D., Hitz, M., and Leitner, G. An evaluation of sticky and force enhanced targets in multi target situations. *Proc. NordicCHI*, (2001), 58-67.
3. Asano, T., Sharlin, E., Kitamura, Y., Takashima, K., and Kishino, F. Predictive interaction using the delphian desktop. *Proc. UIST '05, (2005)*, 133-141.
4. Balakrishnan, R. "Beating" Fitt's law: virtual enhancements for pointing facilitation. *IJHCS*, 61, (2004), 857-874.
5. Bateman, S., Mandryk, R., Gutwin, C., Xiao, R. Targeting-Assistance Techniques for Distant Pointing with Relative Ray Casting. Technical Report 2009-03, Dept. of Comp. Sci., Univ. of Saskatchewan, 2009.
6. Baudisch, P., Cutrell, E., Robbins, D., Czerwinski, M., Tandler, P. Bederson, B., and Zierlinger, A. Drag-and-Pop and Drag-and-Pick: Techniques for Accessing Remote Screen Content on Touch- and Pen-operated Systems. *Proc. Interact 2003*, 57-64.
7. Blanch, R., Guiard, Y., Beaudoin-Lafon, M. Semantic pointing: improving target axquisition with control-display ratio adaptation. *Proc. CHI '04*, 519-525.
8. Bostan, B., and Öğüt, S. Game challenges and difficulty levels: lessons learned From RPGs. In *International Simulation and Gaming Association Conference*, 2009.
9. Chapuis, O., Labrune, J., and Pietriga, E. DynaSpot: Speed-Dependent Area Cursor. *CHI'09*, 1391-1400.
10. Csíkszentmihályi, M. *Flow: The Psychology of Optimal Experience*. Harper and Row, 1990.
11. Elmqvist, N. and Fekete, J-D. Semantic Pointing for Object Picking in Complex 3D Environments. *Proc. GI*, 2008, 243-250.
12. Falstein, N. Understanding fun – the Theory of Natural Funativity. In *Introduction to Game Development*, Rabin, S. (ed.), 71-98. Charles River Media, 2005.
13. Fitt, P. M. The Information Capacity of the human motor control system in controlling the amplitude of movement. *J. Exp. Psychology, 47,* (1954), 381-391.
14. Grossman, T. and Balakrishnan, R. The bubble cursor: enhancing target acquisition by dynamic resizing of the cursor's activation area. *Proc. CHI '05*, 281-290.
15. Hanson, B. Game balance for massively multiplayer games. In *Massively Multiplayer Game Development*, Alexander, T. (ed.), 30-48. Charles Rive Media, 2003.
16. Herbrich, R., Minka, T., and Graepel, T. TrueSkill(TM): A Bayesian skill rating system. In *Advances in Neural Information Processing Systems*, 20, 569--576, 2007.
17. Hunicke, R. The case for dynamic difficulty adjustment in games. In *Proceedings of Advances in Computer Entertainment Technology*, vol. 265, 429-433, 2005.
18. Hurst, A., Mankoff, J., Dey, A. K., and Hudson, S. E. 2007. Dirty desktops: using a patina of magnetic mouse dust to make common interactor targets easier to select. *Proc. UIST '07*, 183-186.
19. Ibáñez-Martínez, J. and Delgado-Mata, C. From Competitive to Social Two-player Videogames. *Proc. Workshop on Child, Computer and Interaction*, 2009.
20. Kabbash, P. and Buxton, W. The "Prince" Technique: Fitts' Law and Selection Using Area Cursors. *Proc. CHI '95*, 273-279.
21. MacKenzie, I. S. Fitts' law as a research and design tool in human-computer interaction. *HCI, 7*, (1992), 91-139.
22. Mandryk, R. L., and Gutwin, C. Perceptibility and Utility of Sticky Targets. *Proc. GI '08*, 65-72.
23. McGuffin, M., and Balakrishnan, R. Acquisition of Expanding Targets. *Proc. CHI 2002*, 57-64.
24. Meyer, D., Smith, J., Kornblum, S., Abrams, R., Wright, C. Optimality in human motor performance: ideal control of rapid aimed movements. *Psych. Rev*. 8, (1988), 340-370.
25. Nelson, D. Fire Control. *Game Developer*, 16(11), 2009.
26. Newheiser, M. Playing Fair: A Look at Competition in Gaming. *Strange Horizons*. www.strangehorizons.com /2009/20090309/newheiser-a.shtml, accessed June 2010.
27. Onyett, C. Shadowrun Interview. *IGN*. http://pc.ign. com/articles/740/740187p1.html, accessed June 2010.
28. Parker, J.K., Mandryk, R., Nunes, M.N., and Inkpen, K. TractorBeam Selection Aids: Improving Target Acquisition for Pointing Input on Tabletop Displays. *Proc. INTERACT '05*, 80-93.
29. Pedersen, C., Togelius, J. and Yannakakis, G. Modeling Player Experience for Content Creation. IEEE Trans. Comp. Intell. and AI in Games, 2 (1), 54-67, 2010.
30. Sweetser, P. and Wyeth, P. GameFlow: a model for evaluating player enjoyment in games. *Computers in Entertainment*, 3, 3-3, 205.
31. Thompson, C. What Type of Game Cheater Are You? Article on *Wired.com*. Retrieved August, 15, 2010.
32. Vorderer, P., Hartmann, T., and Klimmt, C. Explaining the enjoyment of playing video games: the role of competition. *Proc. Entertainment Computing'03*, 1-9.
33. Worden, A., Walker, M., Bharat, K. and Hudson, S. Making computers easier for older adults to use: area cursors and sticky icons. *Proc. CHI '97*, 266-271.
34. Yeung, S., Lui, J., Liu, J., and Yan, J. Detecting cheaters for multiplayer games: theory, design and implementation. *Proc. IEEE CCNS 2006*, 1178-1182.
35. Youssef, A. and Cossell, S. Thoughts on adjusting perceived difficulty in games. *Proc. Australasian Conference on Interactive Entertainment*, 1-4, 2009.