Entertainment Computing 5 (2014) 43–54

Contents lists available at ScienceDirect

Entertainment Computing

journal homepage: ees.elsevier.com/entcom

The effect of temporal adaptation granularity and game genre on the time-balancing abilities of adaptive time-varying minigames ☆

Amin Tavassolian *, Kevin G. Stanley, Carl Gutwin

Department of Computer Science, University of Saskatchewan, 110 Science Place, Saskatoon, SK S7N 5C9, Canada

ARTICLE INFO

Article history:
Received 2 November 2012
Revised 4 June 2013
Accepted 30 June 2013
Available online 25 July 2013

Keywords:
Game balance
Time balancing
Minigames
Adaptation

ABSTRACT

Game designers spend a great deal of time developing well-balanced game experiences. However, differences in player ability, hardware capacity (e.g. network connections) or game mechanic constraints make it difficult to balance games for all players in all conditions. Adaptive balancing systems have been employed in an attempt to automatically compensate for these differences in real time as the game is being played. However, due to the complex non-linear mechanics underlying modern games, automated balancing systems can be highly unstable for all but the simplest mechanics, restricting the design space. In prior work we advanced the concept of using adaptive minigames deployed from within a larger game to decouple the adaptive mechanics from the main game mechanics. In particular, we looked at time-adaptive minigames (ATMs) which attempt to control the time to completion of a minigame. In this paper, we extend the ATM framework with additional time-adaptation algorithms and analyze the interaction between adaptive algorithm, game mechanic, and game difficulty in a controlled experiment. We find significant effects and interactions for all three factors, confirming our intuition that these processes are important and linked. We further find that finer temporal granularity leads to less-perceptible adaptation and smaller deviations in game completion times. This work provides an empirically-grounded algorithmic foundation for the design and practical deployment of ATMs in larger games, a foundation that can improve the balance and experience in these games.

© 2014 Published by Elsevier B.V.

1. Introduction

Video games attract players with many different skill levels – from casual gamers to tournament champions. When differently-skilled players try to play the same game, the experience is often problematic for both people. Games should be balanced in terms of *fairness*, in that players with greater skill should usually prevail, but also in terms of *competitive flow*, in that the game should provide an engaging and competitive experience for all players even if they have different skill levels. Several aspects of games have been investigated as potential means for accomplishing balance (such as the available strategies for different character types, or the allocation of initial resources), but our interests lie in the use of *time* – that is, the amount of time needed for players to complete certain activities in the game (such as obtaining resources, building units, or moving to different locations). For game mechanics with a significant temporal component, the time taken for different activities is the most obvious way that more-skilled players differentiate themselves from less-skilled players.

When players with different skill levels play time-based games, the game can lose its flow, becoming either dull or frustrating for the players. Therefore, it is important for game designers to be able to adjust the time balance of multi-player games, without making the game seem unfair. As Rollins and Adams state in *On Game Design*, "you need to keep the players in the balance sweet spot for as long as is practical in order to keep the game fun and let the underdogs have a chance to catch up. [However,] the major factor that determines winners [should be] player skill" [1]. In essence, we want the best players to finish first, but by a smaller margin.

Time-based activities can be seen in many current games: in race-based games such as MarioKart, in games requiring synchronized motion between heterogeneous agents [2], in games employing rates of production such as StarCraft, and in games with 'cooldown' mechanics such as World of Warcraft. The time-based mechanisms and actions in these games could be manipulated to balance players of different skill levels; however, directly manipulating the time or timing parameters of these main game activities can be disruptive for the player and complex mechanics could be rendered unstable by the feedback loop created by the adaptation algorithm. An alternative approach is to manipulate time through

☆ This paper has been recommended for acceptance by Junia Anacleto, PhD on Physics.

* Corresponding author. Tel.: +1 306 966 4886.
E-mail addresses: amin.tavassolian@usask.ca (A. Tavassolian), kevin.stanley@usask.ca (K.G. Stanley), carl.gutwin@usask.ca (C. Gutwin).

1875-9521/$ - see front matter © 2014 Published by Elsevier B.V.
http://dx.doi.org/10.1016/j.entcom.2013.06.003

activities that are outside the main game – such as through *minigames* that appear at various points within the game environment, but whose (usually simple) mechanics are different from the main game activities.

In our previous work [3], we introduced a novel way of carrying out time balancing through the use of *adaptive time-variant minigames* (ATMs). ATMs are simple activities contained within a larger game that balance temporal flow by adding varying amounts of time to a player's main-game task or mission. For example, a player might have to complete a lock-picking minigame to break into a building – and the amount of time taken can be controlled by appropriate parameterization of the lock picking activities. Adaptive time-variant minigames provide designers with considerable flexibility: in an ATM, the minigame is parameterized over a range of completion times, based on the game state and player skill. Minigames can be started as part of traditional game mechanics, such as when a character casts a spell in World of Warcraft or when a production order is issued to a building in StarCraft. The minigame would then spawn as part of the main-game mechanics. In order for the primary task to be completed, the minigame must be completed successfully.

The ATM approach has several strengths: it decouples the balancing activity from primary game play; it allows the creation of specific minigame-based interactions to mask the temporal adaptation; and it provides the designer with two primary points to exert balance in the game: the initial difficulty level (often based on main game state), and dynamic elements of the game adjusted during gameplay (often based on player performance in the minigame).

Our previous work reported two studies of four different ATMs [3]. The first study examined whether the minigames were able to manage time correctly in isolation, and showed that both the minimum-time prediction and the adaptation mechanisms worked well, leading to game times that tracked the desired values. The second study tested the real-world effectiveness of ATMs in a real mixed-reality game called *Stealth Hacker*. Our results showed that the adaptive time-variant minigames were able to provide temporal balance without detracting from the main game. These experiences with ATMs suggested that the underlying principle could be used more generally to assist designers with time balancing in a wide variety of single-player and multi-player games.

Despite these early positive results, however, it was clear from the prior studies that the adaptation mechanism used in [3] was insufficient. The mechanism adapted the minigame at only one point during game play, leading to several problems:

- The size of the adaptations was often too large as the adaptation algorithm reacted to accumulated player error;
- The adaptations were noticeable and disruptive to gameplay; the ATMs achieved better player balance at the cost of a reduced sense of fairness;
- The interaction between the adaptation mechanism, the type of game, and the starting level could not be adequately analyzed.

In this paper we provide a much more in-depth investigation of ATMs to address these issues. We compare three different adaptation algorithms in four different game types. The adaptation algorithms are:

- Discrete balance, which replicates the one-shot algorithm from our earlier work;
- State balance, which adapts game parameters when particular game states change;
- Continuous balance, which adapts on every game update (i.e., every heartbeat).

All of the algorithms employ a calibrated baseline model, an exemplar which describes the expected progress of the player through the minigame. The Discrete method uses the average expected completion time; the State and Continuous methods use moment-by-moment comparisons with the expected progress through the game as encoded by the exemplar.

To investigate their effectiveness, we carried out a study that compared these three approaches in a 24-participant controlled experiment. The study showed that all of the adaptive algorithms worked well, and that the Continuous balancing technique provides the best results in balancing performance, as measured through survey instruments and log file analysis.

Our work provides three contributions. First, we provide additional evidence that adaptive time-varying minigames are effective tools for time balancing. Second, we show the differences between three adaptive approaches with different adaptation granularities, and show that the type and difficulty of the minigame can have a substantial effect on the adaptation. Third, we demonstrate that Continuous balancing performs best both in terms of time manipulation and perceptibility. Overall, our results provide new and valuable information for multiplayer game developers on the design, deployment, and evaluation of minigame-based techniques for time balancing.

Having established the efficacy of ATMs for Mixed Reality Games in [3], we have turned our attention to the construction of the minigames themselves in this work. In particular, we are interested in exploring the impact of adaptation algorithm on balance performance and player experience. Because the focus of this work is on the minigames themselves, we do not provide an analysis of their integration into a larger gaming context.

2. Background: game balance and player balance

Video games are designed to generate interactive, engaging, and entertaining experiences [4,5], and the balance of the game is widely recognized as a design issue that has profound effects on enjoyment [6], mutually influencing both challenge and user satisfaction [7,8].

2.1. Balancing fairness in multiplayer games

A primary issue in competitive games is that the different teams or players should have equal chances to win the game [1]. Balancing fairness can involve manipulations to different game elements – for example, the capabilities and initial resources allocated to player types such as Orcs and Humans in WarCraft [9]. This type of balancing (called 'static balancing') is often carried out through repeated playtesting of the game rules [10], such as tuning the capabilities of individual weapons [1].

2.2. Balancing competition

One aspect of *flow* [4] is the degree to which a game provides an experience for players that has an appropriate level of challenge: if the game's challenges exceed the player's ability, it leads to frustration; if challenges are lower than the player's skill, the player becomes bored [11]. There are three main ways that designers can balance competition in multiplayer games (also called *player balancing* [12]). First, a few methods exist for balancing competition without changing the game itself – for example, ranking systems and ladder tournaments help match players with opponents who have similar skill levels. Second, games can be designed so that a stronger player is given an explicit disadvantage, such as handicapping in golf or a "head-start" in playground games. In computational environments, games can also be designed with

Download English Version:

https://daneshyari.com/en/article/381915

Download Persian Version:

https://daneshyari.com/article/381915

Daneshyari.com