Dynamic Game Balancing: an Evaluation of User Satisfaction

Gustavo Andrade[1], Geber Ramalho[1,2], Alex Sandro Gomes[1], Vincent Corruble[2]

[1]Centro de Informática
Universidade Federal de Pernambuco
Caixa Postal 7851, CEP 50732-970, Recife, Brazil
{gda,glr,asg}@cin.ufpe.br

[2] Laboratoire d'informatique de Paris 6
Université Paris 6
4 Place Jussieu 75252 Paris CEDEX 05, France
{Geber.Ramalho,Vincent.Corruble}@lip6.fr

Abstract

User satisfaction in computer games seems to be influenced by game balance, the level of challenge faced by the user. This work presents an evaluation, performed by human players, of dynamic game balancing approaches. The results indicate that adaptive approaches are more effective. This paper also enumerates some issues encountered in evaluating users' satisfaction, in the context of games, and depicts some learned lessons.

Introduction

Usability is widely recognized as critical to the success of interactive systems (Maguire 2001). One of the attributes associated with a usable system is the satisfaction that the user feels when using it (Nielsen 1993). In computer games, one of the most interactive domains nowadays, the satisfaction attribute is the most important component of the overall usability, as the main goal of a game user is to be entertained (Pagulayan et al. 2003).

A game player's satisfaction is influenced by different variables, like the graphical interface, the background story, the input devices, and, in particular, game balancing. Game balancing aims at providing a good level of challenge for the user, and is recognized by the game development community as a key characteristic for a successful game (Falstein 2004). Balancing a game consists in changing parameters, scenarios and behaviors in order to avoid the extremes of getting the player frustrated because the game is too hard or becoming bored because the game is too easy (Koster 2004). The idea is too keep the user interested in playing the game from the beginning to the end.

The traditional approach to providing game balancing is to fix some pre-defined and static difficulty levels (e.g., beginner, intermediate and advanced) and let the user choose the best one for him or her. However, this approach fails to deal with the great diversity of players in terms of skills and/or domain knowledge, as well as their capacity to learn and adapt over time. Moreover, as players may improve their performance at different rates and use different learning strategies, an alternative approach is to provide user adaptation mechanisms (Langley 1997) to ensure a dynamic game balancing. Dynamic game balancing allows not only the classification of users' skill levels to be fine-grained, but the game difficulty can also follow the players' personal evolution, as they make progress through learning, or as they regress (for instance, after a long period without playing the game).

In order to deal with the dynamic game balancing problem, different approaches have been proposed, based on genetic algorithms (Demasi & Cruz 2002), behavior rules (Spronck, Sprinkhuizen-Kuyper, & Postma 2004), reinforcement learning (Andrade et al. 2005) or environment variables manipulation (Hunicke & Chapman 2004). These approaches have been generally validated empirically with artificial agents simulating the diversity of humans' strategies. However, as user satisfaction within a game is hard to infer using only such agents, it is necessary to involve human players to effectively validate which game balancing strategy provides the highest satisfaction level to game users.

Analyzing users' satisfaction within a game raises some issues. Simply asking the users whether they liked a game provides only superficial information about its overall usability. In order to focus the evaluation on game balancing, it is necessary to choose carefully the variables to be measured, as well as the correct methods to collect them. This paper extends our previous work (Andrade et al. 2005) by including an evaluation by human players of some of the current game balancing approaches, and by validating the idea that dynamic game balancing is an effective method to increase user satisfaction in games.

In the next section, we introduce the dynamic game balancing task and some approaches to address the problem. Section 3 briefly describes previous work. Section 4 presents elements that should be considered when evaluating user satisfaction in games. Then, in Section 5 we apply the concepts from the previous sections into a real-time fighting game. Finally, we present some conclusions and ongoing work.

Dynamic Game Balancing

Dynamic game balancing is a process which must satisfy at least three basic requirements. First, the game must, as quickly as possible, identify and adapt itself to the human player's initial level, which can vary widely from novices to experts. Second, the game must track as closely and as fast as possible the evolutions and regressions in the player's performance. Third, in adapting itself, the behavior of the game must remain believable, since the user is not meant to perceive that the computer is at times playing with a virtual hand tied behind its back (e.g., by executing clearly self-defeating actions).

There are many different approaches to address dynamic game balancing. In all cases, it is necessary to measure, implicitly or explicitly, the difficulty the user is facing. This measure can be performed with a heuristic function, which some authors (Demasi & Cruz 2002) call a "challenge function". This function is supposed to map a given game state into a value that specifies how easy or difficult the game feels to the user at that specific moment. Examples of heuristics used are: the rate of successful shots or hits, the numbers of pieces which have been won and lost, life point's evolution, time to complete a task, or any metrics used to calculate the game score.

Hunicke and Chapman's approach (Hunicke & Chapman 2004) controls the game environment settings in order to make challenges easier or harder. For example, if the game is too hard, the player gets more weapons, recovers life points faster or faces fewer opponents. Although this approach is effective, its application is constrained to game genres where such particular manipulations are possible. This approach could not be used, for instance, in board games, where the players share the same features.

Another approach to dynamic game balancing is to modify the behavior of the Non-Player Characters (NPCs), characters controlled by the computer and usually modeled as intelligent agents. A traditional implementation of such an agent's intelligence is to use behavior rules, defined during game development using domain-specific knowledge. A typical rule in a fighting game would state "punch opponent if he is reachable; chase him otherwise". Besides the fact that it is time-consuming and error-prone to manually write rule bases, adaptive behavior can hardly be obtained with this approach. Extending such an approach to include opponent modeling can be made through dynamic scripting (Spronck, Sprinkhuizen-Kuyper, & Postma 2004), which assigns to each rule a probability of being picked. Rule weights are dynamically updated throughout the game, reflecting the success or failure rate of each rule. This technique can be adapted for game balancing by not selecting the best rule, but the one deemed closest to the user level. However, as game complexity increases, this technique requires a lot of rules, which are hard to build and maintain. Moreover, the performance of the agent becomes limited by the best rule available, which can be too weak for very skilled users.

A natural approach to address the dynamic game balancing problem is to use machine learning. Demasi and Cruz (Demasi & Cruz 2003) built intelligent agents employing genetic algorithms techniques to keep alive agents that best fit the user level. Online coevolution (Wiegand, Liles & Jong 2002) is used in order to speed up the learning process. Online coevolution uses pre-defined models (agents with good genetic features) as parents in the genetic operations, so that the evolution is biased by them. These models are constructed by offline training or by hand, when the agent's genetic encoding is simple enough. This is an innovative approach. However, it shows some limitations when considering the requirements stated before. Because it uses pre-defined models, the agent's learning is heavily restricted, jeopardizing the application of the technique for very skilled users or users with uncommon behavior. As these users do not have a model to speed up learning, it takes a long time until the agents reaches the user level. Furthermore, this approach works only to increase the agent's performance level. If the player's skill regresses, the agent cannot regress also. This limitation compels the agent to always start the evolution from the easiest level. While this can be a good strategy when the player is a beginner, it can be bothering for skilled players, since they will need to wait significantly for the agent to evolve to the appropriate level.

Challenge-Sensitive Game Balancing

Our approach to the dynamic game balancing problem is to use Reinforcement Learning (RL) (Sutton & Barto 1998) to build intelligent adaptive agents capable of providing challenge-sensitive game balancing. The idea is to couple learning with an action selection mechanism which depends on the evaluation of the current user's skills. This way, the dynamic game balancing task is divided into two dimensions: competence (learn as well as possible) and performance (act just as well as necessary). This dichotomy between competence and performance is well known and studied in linguistics, as proposed by Chomsky (Chomsky 1965).

Our approach faces the first dimension (competence) with reinforcement learning. Due to the requirement of being immediately able to play at the human player level, including expert ones, at the beginning of the game, offline training is needed to bootstrap the learning process. This can be done by letting the agent play against itself (self-learning) (Kaelbling, Littman & Moore 1996), or other pre-programmed agents (Spronck, Sprinkhuizen-Kuyper, & Postma 2004). Then, online learning is used to adapt continually this initially built-in intelligence to the specific human opponent, in order to discover the most suitable strategy to play against him or her.

Concerning the second dimension (performance) the idea is to find an adequate policy for choosing actions that provide a good game balance, i.e., actions that keep both agent and human player at approximately the same performance level. In our approach, according to the difficulty the player is facing, the agent chooses actions with high or low expected performance. For a given situation, if the game level is too hard, the agent does not choose the optimal action (provided by the RL framework), but chooses progressively less and less sub-optimal actions until its performance is as good as the player's. This entails choosing the second best action, the third one, and so on, until it reaches the player's level.

Similarly, if the game level becomes too easy, it will choose actions whose values are higher, possibly until it reaches the optimal performance. In this sense, our idea of adaptation shares the same principles with the one proposed by Spronck et al. (Spronck, Sprinkhuizen-Kuyper, & Postma 2004), although their work does not state explicitly the division of competence and performance, the techniques used are different, and the works have been developed in parallel.

It is not in the scope of this paper to detail any of the dynamic game balancing approaches, since the focus here is the evaluation of user's satisfaction with respect to the these approaches. More details can be found in the cited literature.

Evaluating User Satisfaction

Game balancing is a property related to the challenge faced by the user. It can be inferred from different variables. A natural approach is to use the evolution of the player's score. Scores are based on objective measures, such as, number of won and lost pieces, life points' evolution, or rate of successful shots, and can be automatically computed during a game. However, as the overall goal of game balancing is to increase user satisfaction, it worth to check if a fair score is really entertaining for the player.

Some authors already addressed the task of relating objective variables to users' satisfaction (Yannakakis & Hallam 2005), creating a generic measure for the user interest in a game. However, the lack of validation of this measure with human players jeopardizes the application of the method as a substitute for tests with human players. Other authors developed a model for evaluating player satisfaction in games (Sweetser & Wyeth 2005), integrating different heuristics found in the literature. The resulting model includes eight elements that impact on player satisfaction within a game: concentration, challenge, player skills, controls, goals, feedback, immersion and social interaction. As game balancing strongly influences variables like challenge and player skills, it seems to impact strongly players satisfaction. Unfortunately, the proposed model is validated only through expert reviews, which do not represent adequately the broad spectrum of game players.

Our approach of the task of associating game balancing to user satisfaction makes use of usability tests, and combines the measurement of concrete variables, the opinions of the players collected through structured questionnaires, and user open feedback about the game.

Before starting such tests, we must define: the goals, the users (e.g., the testers), the usability methods used, the tasks that each tester must execute, and the variables used to measure the performance (Nielsen 1993). Our goal is to check the best strategy to balancing a game, as well as if it provides a good level of user satisfaction.

In computer games, there is a great diversity among users in terms of skills and/or domain knowledge. As usability tests should reflect the real range of users of a system (Nielsen 1993), a game should be evaluated with all its user's categories. The only restrictions for the testers are the same as the ones of the game, which usually relates to age requirements.

In order to consider all the range of aspects that influence game balance and user satisfaction, our usability test includes controlled user testing, satisfaction questionnaires, and post-experience interviews (Maguire 2001). The controlled tests, intended to collect data when a user performs a pre-defined set of tasks, are used to measure the variables directly related to game balance, isolating it from other game aspects. Data can be collected by automatically logging user actions and performance or by observing his/her actions, comments and expressions during the test. Satisfaction questionnaires are applied to collect subjective data, through options lists and evaluations scales, like Likert scales (Nielsen 1993). Finally, post-experience interviews are used to collect data not covered by the last methods, like user opinions and suggestions about improvements on the game. In these interviews, it is worthwhile to make users feel comfortable to expose his/her perceptions and opinions. So, although a semi-structured questionnaire is useful to guide the interview, the interviewer is free to change the script inserting or removing questions.

Once we have defined the usability methods to be used in the test, the next step is to define the tasks to be executed. When a user begins to interact with a system, it is possible to distinguish two phases, as shown in Figure 1. At the beginning, the user usually improves his or her performance rapidly, as consequence of learning to use the system. As time progresses, learning tends to slow down and user skill becomes stable.

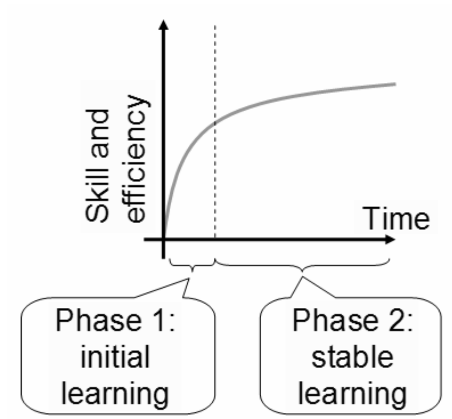

Figure 1: User learning curve

To define the tasks to be executed in the controlled test, it is important to divide them in two distinct sets, according to the learning curve. The first phase, which we will name "learning phase", is used to check *learnability* (Nielsen 1993), which means how easy it is to start using a system. In fact, this is a key issue in game development, as a user can give up playing if he or she feels its beginning too easy or too hard.

The second phase, which we will name “evaluation phase”, is used to perform the main measurements and comparisons among different game balancing strategies, as learning does not have a strong influence on the player performance between subsequent tasks.

A key issue when dividing the user learning curve in two phases is setting the point in which learning becomes stable. As each user has different skills and experiences, this point can indeed vary. After this point, we must ensure that all users have, approximately, the same skill level. Therefore, while beginners must take more time to reach this point, experienced players can reach it faster. A straightforward approach to this task is to use checkpoints: when the user reaches it, he/she goes to the next phase. Examples of possible checkpoints are winning a percentage of opponent pieces in a board game, defeating an intermediate boss in a fighting game, or exploring the full map in a first-person-shooter game.

While the user executes the pre-defined tasks, each method collects different data. The controlled test collects data related, for instance, to game score, time to complete tasks, and user efficiency. Satisfaction questionnaires collect user perceptions in a structured way, possibly been applied at different times. Finally, post-experience interview collect qualitative data about the test, such as information not covered by the other methods.

Case Study

Game Description

As a case study, we evaluated different game balancing approaches with human players into Knock’em (Andrade et al. 2004), a real-time fighting game where two players face each other inside a bullring and whose functionalities are similar to those of successful commercial games, such as Capcom Street Fighter and Midway Mortal Kombat. The main objective of the game is to beat the opponent. A fight ends when the life points of one player (initially, 100 points) reach zero, or after 1min30secs of fighting, whatever comes first. The winner is the fighter which has the highest remaining life at the end. The environment is a bidimensional arena in which horizontal moves are free and vertical moves are possible through jumps. The possible attack actions are to punch (strong or fast), to kick (strong or fast), and to launch fireballs.

Four types of agents were implemented in the game: a state-machine (SM, static behavior), a trained genetic learning agent (GL, intelligent and with genetic learning skills) (Demasi & Cruz 2002), a trained traditional RL agent (TRL, intelligent and with reinforcement learning skills) (Sutton & Barto 1998), and a trained Challenge-Sensitive RL agent (CSRL, our RL-based model for dynamic game balancing) (Andrade & al. 2005). All agents’ initial strategy is built through offline learning against a random agent.

Test Plan

The tasks executed by the users are divided in two phases, according to Figure 1. In the learning phase, each user faces only one of the four agents being evaluated, as user performance changes a lot between subsequent fights. The agent which each user faces is randomly chosen by the application. The checkpoint used is an evaluator agent, which is the same to all users. Only when the user defeats the opponent chosen by the application can he/she face the evaluator. The evaluator is a traditional reinforcement learning agent, previously trained against a random agent. It is a different character of the game, stronger than the user character, in order to ensure that the testers are really skilled after defeating it. The learning phase ends when the player defeats the evaluator.

In the evaluation phase, all users face all four agents (SM, TLR, GL and CSRL, in this fixed order), sequentially, during 5 fights each. So, while the first phase duration depends on the player skill, the second phase is constant (20 fights).

In all tests, the player is accompanied by an expert, who is responsible for introducing the test, observing him/her while executing the pre-defined tasks, and interviewing him/her in the end. In the introduction, all players are told that the main goal of the test is to evaluate the game, not the players, and so they should act as natural as possible (Nielsen 1993). The testers are also told that the collected data will be used so that the identity of each player is not revealed. The test itself is automatically conducted by the game, with the evaluation tasks integrated to its story. While the player performs the test, the expert just observes him/her, without providing any kind of help. Then, the expert interviews the tester, collecting the impressions about the game.

Beyond the data manually collected by the expert through observations and interviews, some data are automatically registered by the game through logs and questionnaires. Logs are used to register the time each user spends in the learning phase, the fighters’ life points’ difference after each fight, and the efficiency of the agents. The questionnaires are used at two stages: before the test, to determine the group (beginner or expert) to whom the user belongs, and after the evaluation phase, in order to compare the four different game balancing strategies.

Experimental Results

The tests were executed with four players: B1, B2, E1, and E2. Two of them were beginners (B1 and B2, not used to playing games) and two were experienced (E1 and E2, 6 years or more regularly playing games), according to the self-description questionnaire applied before the test.

The length of the learning phase changed according to the user profile. While the beginners took 18 and 13 fights in this phase, the experienced players needed only 3 and 8 fights to learn how to play and defeat the evaluator character. This result indicates that the user skill in games has a strong influence in the learning phase, and so must be considered when designing intelligent adaptive agents. However, as each tester has played with a different agent (SM, GL, TRL, and CSRL), it is not possible to compare, with only these users, the best game balancing strategy with respect to *learnability*.

After the end of the test (after the learning and the evaluation phase), all players answered a satisfaction questionnaire, in which the main variable, user satisfaction, was collected. When asked "Which opponent was most enjoyable", 3 users chose the CSRL agent, while 1 chose the GL. In the feedback interview, the testers highlighted that the CSRL agent was most enjoyable because it wasn't predictable, like the SM and the GL, whose movements could be anticipated by the testers. Moreover, the CSRL, the players argued, was not as difficult to defeat as the TRL agent. However, an interesting note was cited by the user who preferred the GL agent. The GL agent is implemented as a population, in which each individual has a static behavior, and only after each fight this individual is evaluated and enhanced by the genetic operations, resulting in a new behavior. Therefore, this agent created an expectation to be predictable (as in a single fight it is static), but surprised the user when the behavior changed in subsequent fights. This feature was highlighted by some users as a positive feature of the GL agent.

In the same questionnaires, the testers were also asked about some characteristics of the most enjoyable agent that each one chose. In these questions, a Likert scale (Nielsen 1993) was used. In such scales, the users are faced with an affirmative and are asked to agree or not with it. Users can answer according to the following scale: (1) completely agree, (2) agree, (3) indifferent, (4) disagree, (5) completely disagree. The users' answers are in Table 1.

The second column denotes the means of the four users' choice in the Likert scale, which range from 1 (completely agree) to 5 (completely disagree). Table 1 results indicate that the users disagree that the most enjoyable opponent is predictable, but strongly agree that it is intelligent and challenging.

Table 1: Post-test questionnaire

Affirmative	Mean
The most enjoyable opponent is predictable.	4.00
The most enjoyable opponent is intelligent.	1.75
The most enjoyable opponent is challenging.	1.50

The data collected through the questionnaires are confirmed by the measurements on the users' logs. In the evaluation phase, each agent plays 5 fights against each tester, in a total of 20 fights. The first variable analyzed is the agents' efficiency, which is the total of life points taken from the opponent (the human players) divided by the total of hits delivered in a fight. This variable is useful to check whether the agents are acting consistently (high efficiency) or randomly (low efficiency). We noticed that the CSRL is one of the most efficient, in the mean, and also had the lowest variance among users. The SM and TRL, although efficient in the mean, had a high variance, which means that they couldn't successfully deal with users' different profiles. The GL, on the other hand, had low variance, but is the least efficient. The results are in Table 2.

Table 2: Agents' efficiency

Agent	Mean	Std. deviation
SM	4.70	1.00
GL	2.93	0.75
TRL	4.87	1.36
CSRL	4.85	0.65

The life point differences after each fight, which are directly related to the game score, also confirm the previous results, and are shown in Table 3.

Table 3: Life points' differences

Agent	Mean	Std. deviation
SM	32.20	21.42
GL	52.40	36.23
TRL	-5.55	41.27
CSRL	13.30	30.61

Table 4: Life point difference per user

Agent	B1	B2	E1	E2
SM	32.80	18.00	49.20	29.20
GL	69.60	34.00	59.20	46.80
TRL	-46.20	-3.20	39.60	-12.40
CSRL	-2.20	15.20	25.60	14.60

The positive values represent victories of the human player, whereas negative ones represent defeats (and, consequently, victories of the evaluated agent). Values close to zero indicate that both fighters (the player and the agent) performed, approximately, at the same level. Table 3 shows that only the TRL agent could beat, in the mean, all its human opponents; however, its high variance indicates that its performance changes with the user profile. Actually, the results presented in Table 4 show that the TRL agent was too strong against one of the beginners (B1), but couldn't perform so well against one of the experienced players (E1). The SM and GL agents do not have a good overall performance and cannot defeat even the beginners. Finally, the CSRL agent, obtains an average performance (although it should be more challenging), but performed uniformly among the different users, as indicated by its low variance.

Discussion

The different usability methods used in the tests showed that the adaptive approaches to the game balancing problem have the best results. It can successfully deal with the diversity of users skills, providing an adequate challenge to each player. Moreover, the relationship between balance and user satisfaction is also confirmed, as players prefer the agents that act just as well as necessary.

The post-experience interviews also revealed interesting data. All users perceived differences in behavior between the evaluated agents, which mean that different game balancing strategies produce effectively different agents. When asked about the main feature of an entertaining game, all testers highlighted the challenge as a key issue. This result emphasizes the importance of game balancing to increase user satisfaction in games. Finally, users also cited the lack of predictability as an aspect that increase users' overall satisfaction. Indeed, avoiding repetitive behaviors is another dimension that should be addressed by a successful game balancing strategy.

Conclusions

This paper presented the evaluation with human players of different automatic (AI-based) game balancing approaches. We used different usability methods to collect a broad range of variables, including concrete data about the challenge faced by the players and subjective data about the satisfaction that they experienced. The results showed that agents that implement a dynamic game balancing approach performed close to user level, and also provided the highest user satisfaction, validating our hypothesis of mutual influence between game balance and user satisfaction. Specifically, our challenge-based approach was perceived as the best one in terms of satisfaction.

We also provided a detailed explanation of issues and lessons concerning evaluation of user satisfaction in games, showing some variables that should be analyzed and effective methods to collect them.

We are now enhancing our challenge-sensitive approach to incorporate the users' feedback, such as the importance given to surprising behaviors. Then, we will run the experiments with a broader range of users, in order to create more significant statistics about the game balancing task. In this broader evaluation, we plan to include Spronk's approach (Spronck, Sprinkhuizen-Kuyper, & Postma 2004) among the dynamic game balancing approaches that will be evaluated. We also plan to evaluate the applicability of dynamic balance approaches to more complex game categories, which, contrary to fighting games, do not provide instantaneous feedback about the player's performance, and which require from the agent a wide range of actions.

References

Andrade, G.; Santana, H.; Furtado, A.; Leitão, A.; and Ramalho, G. 2004. Online Adaptation of Computer Games Agents: A Reinforcement Learning Approach. Scientia, 15(2), UNISINOS, 140-147. São Leopoldo, RS.

Andrade, G.; Ramalho, G.; Santana, H.; and Corruble, V. 2005. Extending Reinforcement Learning to Provide Dynamic Game Balancing. IJCAI-05 Workshop on Reasoning, Representation, and Learning in Computer Games (eds. David W. Aha, Héctor Muñoz-Avila, and Michael van Lent), Naval Research Laboratory, Navy Center for Applied Research in Artificial Intelligence, 07-12. Washington, DC.

Chomsky, N. 1965. *Aspects of the Theory of Syntax*. Massachusetts: The MIT Press.

Demasi, P.; and Cruz, A. 2002. Online Coevolution for Action Games. In Proceedings of The 3rd International Conference on Intelligent Games And Simulation, 113-120. London, UK.

Falstein, N. 2004. The Flow Channel. *Game Developer Magazine*, 11(5):52a.

Hunicke, R.; and Chapman, V. 2004. AI for Dynamic Difficulty Adjustment in Games. *Challenges in Game Artificial Intelligence AAAI Workshop*, 91-96, San Jose, CA.: AAAI Press.

Kaelbling, L.; Littman, M.; and Moore, A. 1996. Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research*, 4:237-285.

Koster, R. 2004. *Theory of Fun for Game Design*. Phoenix: Paraglyph Press.

Langley, P. 1997. Machine Learning for Adaptive User Interfaces. In *Proceedings of the 21st German Annual Conference on Artificial Intelligence, 53-62. Freiburg, Germany: Springer.*

Madeira, C.; Corruble, V.; Ramalho, G.; and Ratitch, B. 2004. Bootstrapping the Learning Process for the Semi-automated Design of a Challenging Game AI. *Challenges in Game Artificial Intelligence AAAI Workshop*, 72-76, San Jose: AAAI Press.

Maguire, M. 2001. Methods to support human-centred design. *Int. J. Human-Computer Studies,* 55:587-634.

Nielsen, J. 1993. *Usability Engineering*. San Francisco: Morgan Kaufmann.

Pagulayan, R.; Keeker, K.; Wixon, D.; Romero, R.; and Fuller, T. 2003. User-centered design in games. *In Human-Computer Interaction Handbook, J. Jacko and A. Sears*, 883-905. Mahwah, NJ: Lawrence Erlbaum.

Spronck, P.; Sprinkhuizen-Kuyper, I.; and Postma, E. 2004. Difficulty Scaling of Game AI. In Proceedings of the 5th International Conference on Intelligent Games and Simulation, 33-37, Belgium.

Sutton, R.; and Barto, A. 1998. *Reinforcement Learning: An Introduction*. Massachusetts: The MIT Press.

Sweetser, P.; and Wyeth, P. 2005. GameFlow: a Model for Evaluating Player Enjoyment in Games. *ACM Computers in Entertainment*, Vol. 3, No. 3.

Wiegand, R.; Liles, W.; and Jong, G. 2002. Analyzing Cooperative Coevolution with Evolutionary Game Theory. In *Proceedings of the 2002 Congress on Evolutionary Computation*, 1600-1605, Honolulu: IEEE Press.

Yannakakis, G.; and Hallam, J. 2005. A Generic Approach for Generating Interesting Interactive Pac-Man Opponents. In Proceedings of the IEEE Symposium on Computational Intelligence and Games, 94-101, Essex.