Deep Player Behavior Models: Evaluating a Novel Take on Dynamic Difficulty Adjustment

Johannes Pfau
Digital Media Lab, TZI,
University of Bremen
28359, Bremen, Germany
jpfau@tzi.de

Jan David Smeddinck
Open Lab, School of Computing,
Newcastle University
Newcastle upon Tyne, NE1 7RU, UK
Jan.Smeddinck@newcastle.ac.uk

Rainer Malaka
Digital Media Lab, TZI,
University of Bremen
28359, Bremen, Germany
malaka@tzi.de

ABSTRACT

Finding and maintaining the right level of challenge with respect to the individual abilities of players has long been in the focus of game user research (GUR) and game development (GD). The right difficulty balance is usually considered a prerequisite for motivation and a good player experience. Dynamic difficulty adjustment (DDA) aims to tailor difficulty balance to individual players, but most deployments are limited to heuristically adjusting a small number of high-level difficulty parameters and require manual tuning over iterative development steps. Informing both GUR and GD, we compare

CCS CONCEPTS
Human-centered computing → User models
Computing methodologies → Neural networks
Applied computing → Computer Games

KEYWORDS
Dynamic difficulty adjustment;
Player Modeling;
Neural Networks;
Deep Learning;
Games

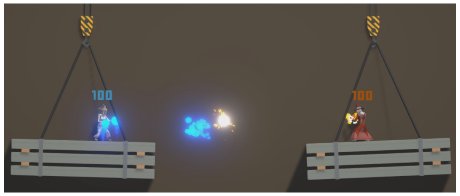

Figure 1: Screenshot of *Korona:Nemesis*. The player (on the left) utilizes Water to counter a Fire projectile.

Fire	Cancels **Restoration** Critically hits **Restoration/Steel** Destroys **Steel** projectiles Applies **burning** damage over time
Water	Immunity against **burning** Critically hits **Fire/Steel** Destroys **Fire** projectiles
Lightning	Immunity against **suffering** Critically hits **Water/Death** Destroys **Water** projectiles
Restoration	Restores 10LP Converts **Water** projectiles into 10LP Immunity against **Pain**
Steel	Reflects **Lightning** projectiles Reflects **Pain** projectiles Critically hits **Lightning/Pain**
Death	Inverts **Restoration** Critically hits **Restoration/Pain** Applies **suffering** damage over time
Pain	Self-ignites **Fire** Critically hits **Fire/Lightning** Applies 0.4 seconds stun

Table 1: Element interactions in the game.

an approach based on deep player behavior models which are trained automatically to match a given player and can encode complex behaviors to more traditional strategies for determining non-player character actions. Our findings indicate that deep learning has great potential in DDA.

INTRODUCTION

Dynamic difficulty adjustment (DDA) addresses potential mismatch between player proficiency and level of challenge in video games by balancing game parameters that increase or decrease the latter. Traditional approaches that manipulate core game variables (such as speed, damage or hit ratio), have been successfully evaluated and integrated in scientific [3] and industrial (e.g. Resident Evil 4)[1] usage. For practical reasons, DDA is usually hidden, since it yields incentives to perform badly on purpose [8]. Current DDA systems are typically limited to a small number of high-level parameters and require careful tuning of threshold-heuristics [10]. Here, we utilize *Deep Player Behavior Models* (DPBM) [6] to introduce a distinct adaptation module that incorporates player proficiency implicitly instead of explicitly and represents and generates game proficiency on a multi-dimensional level, allowing for complex emergent dynamics. In order to investigate the player experience with DPBM for DDA, we designed *Korona:Nemesis*, a platform fighter focused on prediction, learning and decision making. In an exploratory study, we compared player experience when playing against opponents with different decision making strategies including *basic* heuristics, *random* actions, near-*optimal* heuristics, and DPBM. Based on *self-determination theory* [7], we hypothesize that opponents deploying DPBM-guided strategies yield high results in *interest-enjoyment*, due to displaying convincing, but not rigidly perfect strategies, while *tension-pressure* might be increased and *perceived-competence* might be decreased when facing near-optimal opponents. Both are expected to lead to higher motivation and better player experience than traditional, trivial or unadjusted opponents. Our results provide first evidence that DPBM for generating opponent behavior confirms our hypotheses and offers a valuable subject to study within the field of DDA. We contribute to game user research in the form of a novel take on DDA and promote the applicability and value of machine learning techniques in video games.

RELATED WORK

DDA has developed from flow maximization [3] over multi-player balancing [11] up to a tool for proficiency estimation [2]. In order to estimate the discrepancy between challenge and skill, various assessment techniques have been researched, such as success probability estimation [3] or biofeedback [4]. For the adjustment however, most approaches focus on adjusting game difficulty parameters. In the meantime, machine learning approaches in video game playing that harness continual improvement through simulated play [9] have become popular. Bringing these developments together, we assess the experience of players that provide behavior samples feeding a continuous learning process, facing opponents driven by DPBM on the same proficiency level.

When facing an incoming **Fire** projectile, there are multiple viable choices. The player might react with a **Water** attack, since **Water** projectiles destroy **Fire** projectiles (cf. Figure 1). A more offensive choice would be to counter this attack with a **Pain** attack, which will not stop the incoming projectile (and thus cost 10LP), but critically hit and self-ignite the opponent. At the same time, the opponent has the opportunity to re-counter this counter-attack, depending on making good predictions (e.g. if (s)he predicts the counter-attack to be a **Water** attack and wants to counter it with **Lightning**, but in fact it is a **Pain** attack, it will incur a critical hit).

Sidebar 1: Decision making example.

variable	value
timestamp	12/27/2018 5:16:29
mapID	Map_Steel6
playerCurrentEnergy	WATER
playerChosenAction	ATTACK_WATER
playerHPpercentage	100
playerIsBurning	0
playerIsSuffering	0
targetCurrentEnergy	FIRE
targetHPpercentage	100
targetIsBurning	0
targetIsSuffering	0
absoluteXdistance	12.194
absoluteYdistance	0.211
fireProjectileAhead	1
waterProjectileAhead	0
lightningProjectileAhead	0
steelProjectileAhead	0
deathProjectileAhead	0
painProjectileAhead	0

Table 2: Sample player model entry for the situation given in Figure 1.

GAME DESIGN

In order to construct a setting for studying crucial decision making in real-time, we designed a fast-paced physic-based platform fighter called *Korona:Nemesis* that extends the classic rock-paper-scissors scheme to 7 types of element projectiles (cf. Table 1). In each level, players are placed in a 2D environment, start with 100 life points (LP) and have the objective to eliminate their opponents LP (last player standing wins). Players can *move* (left or right), *jump*, *attack* or *switch* actions. *Switching* changes the current stance to one of the 7 elements. *Attack* will launch an elemental projectile depending on the current stance. Getting hit by a hostile projectile deals 10 damage. Since damage is doubled on a critical hit and projectiles can be destroyed, reflected or influenced by other projectiles (cf. Table 1), players constantly have to be aware of present projectiles, their own and enemies' stances and adapt quickly to the situation. As in rock-paper-scissors, predicting the opponent is key to success and since players adapt and react constantly, there is no single dominant strategy (e.g., cf. Sidebar 1). Players need to learn not only the in-game element-interactions, but also their preferred way to counter attacks and maximize their chances, depending on the current situation. The presence of multiple viable choices, preferences and dislikes makes for a fertile ground for player modeling and decision making studies.

ENEMY TYPES

To focus on the players' experience of the opponents' decision making, enemies differed only in terms of their action selection behavior (and appearance), so possible action and movement choices, damage calculation, elemental interactions etc. were equal between all opponent types. The following categories of opponents were pseudonymized in-game to prevent revealing their strategies.

Basic. The basic opponent choses (and stays with) a single elemental stance per level. It is designed to be the easiest to counter since all actions are trivially predictable and serves as a baseline.

Random. The most balanced enemy in rock-paper-scissors is a completely random one. We decided to include this strategy as it is impossible to predict and thus hard to counter, since every action is independent from the preceding behavior or the current situation. Due to the symmetrical setup of the game, it will make both advantageous and disadvantageous decisions and should therefore not be (near-)impossible to beat.

Optimal. In order to provide an upper bound of performance, the optimal opponent reacts to each player action with one of the optimal counter-attacks and tries to maximize the damage applied to the player.

Player model. Utilizing every player action (together with game state context) executed in the preceding levels, the DPBM opponent will learn from the player's behavior and calculate weights for each possible action, whenever it chooses an action. Depending on the situation, it will make

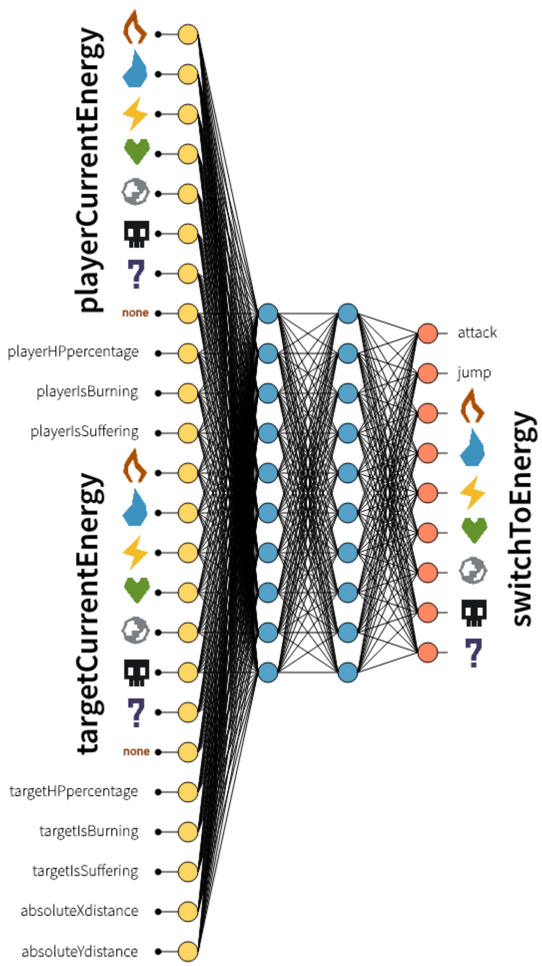

Figure 2: Network used for a single player. Real valued variables are mapped to the range from 0 to 1, energies and conditions are binary.

decisions similar to the player from whom the behavior originated. In this novel DDA approach, the opponent will continually develop while the player learns and advances in proficiency and it will make similar mistakes to the player. In contrast to traditional approaches, players simultaneously have to overcome and exploit their own flaws to win, potentially leading to an upward spiral of learning in both the player and the opponent. The ideal win/lose outcome would be an even split, demonstrating the closeness to the player's skill.

PLAYER MODELING

Based on insights about expressive data and suitable modeling techniques from our earlier work [6], we recorded all crucial player action decisions (*attacking* or *switching* with or to the respective element and *jumping*) together with contextual data from the current situation (cf. Table 2). After each level, this data was fed at run-time into a 24x10x10x9 multilayer perceptron with backpropagation and a logistic sigmoid activation function (cf. Figure 2). The network was initialized randomly and its architecture was determined beforehand, selecting for an efficient trade-off between training time (< 1 second on tested machines) and prediction accuracy (70-90% on testing set).

PILOT STUDY

Over the course of 2 weeks, we conducted a within-subjects study online. Subsequently to the tutorial, the experiment manipulated one independent variable (opponent behavior) with four conditions in randomized order: *basic*, *random*, *optimal* and *player model*. Data was gathered through game protocols and a post-study questionnaire.

Measures. In-game, we logged winning scores of all enemies, all of the players' actions and the resulting deep learning accuracies. Through the questionnaire, demographics and experience in video games were recorded. With respect to each specific enemy type, we asked for subjective assessments how strong and how balanced the particular opponent appeared, captured the player experience using the *Intrinsic Motivation Inventory* [5] (all 7-point Likert scales) and asked players to explain the opponent behaviors in their own words. Conclusive comments, questions and registering an email address for further studies were optional.

Procedure. Following informed consent and a quick tutorial that explained the controls and interactions of the game, participants encountered all four enemy types in permuted order. Each enemy was faced in the first 10 levels of the game, which were kept simple in order to focus on the opponent. Pausing the game was possible at all times and happened whenever the enemy type changed. After facing all of the opponents (in M = 12.5 minutes), the subject was redirected to the web questionnaire and unlocked the multiplayer mode (not part of the study).

	basic	random	optimal	player model
Score	4.8 ± 1.9	4.4 ± 2	7.8 ± 1.2	3.3 ± 1.9
strength	3.7 ± 1.9	4.5 ± 1.4	6.1 ± 1.1	4.3 ± 1.7
balance	2.8 ± 1.6	3.5 ± 1.4	3.7 ± 1.8	4.2 ± 1.9
IMI:				
INT	3.1 ± 1.6	3.7 ± 1.3	3.5 ± 1.8	5 ± 1.6
COMP	3.9 ± 1.9	3.6 ± 1.2	2.8 ± 1.6	4.9 ± 1.9
EFF	4.6 ± 2.2	5.5 ± 1.3	6.2 ± 1	5.6 ± 1.9
TEN	3.7 ± 2	4.8 ± 1.7	5.8 ± 1.1	4.6 ± 1.9

Table 3: Mean statistics ± standard deviations for the four enemy types. Score depicts the number of wins of the opponent. Strength and Balance were subjectively reported. INT: interest-enjoyment, COM: perceived-competence, EFF: effort-importance, TEN: tension-pressure of the IMI.

basic	*"repetitive"* *"some kind of predictable"*
random	*"changing his strategy/weapon very often"* *"unpredictable"*
optimal	*"very strong and fast"* *"always one upping me"* *"i had no chance and i hate him"* *"too OP"* (overpowered)
player model	*"kinda like the* [optimal opponent], *but not as OP"* *"a mixture of the other opponents"* *"my favorite so far, he was smart and fast but not too powerful"*

Table 4: Qualitative statements by players about the different opponents.

Participants. ($n = 98$) participants submitted behavioral data and 16 completed the optional questionnaire (75% male, 18% female, aged 18-37 ($M = 26.6, SD = 4.86$)). 75% described themselves as active, 19% as casual or occasional gamers and 6% said that they do not really play video games.

RESULTS

Using a one-way RM ANOVA, we found significant effects for the IMI scores *interest-enjoyment* ($F = 3.88, p < .05$), *perceived competence* ($F = 3.74, p < .05$), *tension-pressure* ($F = 3.47, p < .05$), as well as perceived *strength* ($F = 5.66, p < .01$), between opponents. These outcomes were further evaluated using two-tailed paired t-tests. Regarding the perceived strength, the optimal opponent significantly outmatched all other types ($p < .01, d_{basic} = .96; d_{random} = 0.76; d_{dpbm} = 1.01$). In terms of *perceived competence*, the *player model* resulted in higher values than the random ($p < .05, d = .72$) or optimal ($p < .05, d = .93$) opponent. For *interest-enjoyment*, DPBM significantly outperformed all of the other approaches ($p < .01, d = .75$ for basic, $p < .05, d = .57$ for random and $p < .05, d = .54$ for optimal). Means and deviations are depicted in table 3. When asked to explain the enemies' behavior in their own words, participant statements reflected these sentiments (cf. Table 4). Split into 80/20 training/test sets individually, neural network accuracy scored 49.1% to 100% ($M = 70.3\%, SD = 13.5\%$).

DISCUSSION AND FUTURE WORK

As hypothesized, the mean subjective strength with DPBM lies between basic and random/optimal, though no significant difference was found. The same holds for the mean subjective balance, exceeding all other enemy types. It did not lead to significantly increased *tension-pressure* and *effort-importance* compared to the other strategies, which might be due to the already high temporal pressure of the general gameplay and the short session duration. Nevertheless, the significantly higher score for *interest-enjoyment* indicates a distinct advantage of playing against the player model. It also apparently avoids frustrating players by displaying overly strong (and rigid) behavior, which is reflected in the significantly decreased *perceived competence* when facing the near-optimal opponent. These interpretations are supported by the qualitative statements, in which all positive comments relate to the DPBM approach and all negative ones to the remaining enemy types. This exploratory study is limited by a small sample size and the low conversion rate from participants who played the game to actually submitting the questionnaire. In further ongoing work, we will lay more emphasis on the questionnaire to consolidate the findings concerning player experience. We are also planning to extend the insights of this short-term study to a prolonged period of time to evaluate the long-term consistency of the approach. With this early study, we provided an experimental comparison between DPBM opponents and heuristic ones, yielding evidence for potential to improve DDA capabilities in general. A comparison to alternative traditional DDA approaches remains future work. We plan to investigate the difference in player experience between player modeling and threshold-based parameter adjustments.

In addition, assuming that every player desires a continually learning opponent is a simplification. Further studies that differentiate between player types might yield additional insights.

CONCLUSION

We compared the player experience of facing a continually learning enemy based on *Deep Player Behavior Models* to three classic heuristic game opponent variants. To provide an adequate study environment, we designed the platform fighting game *Korona:Nemesis*. First quantitative and qualitative results indicate significant improvements in player experience when interacting with the DPBM opponent. Thus, this approach successfully demonstrates a novel, implicit take on DDA and corroborates the potential application of DPBM in complex and fast-paced real-time game environments.

ACKNOWLEDGMENTS

We would like to thank all participants. This work was funded by the German Research Foundation (DFG) as part of Collaborative Research Center (SFB) 1320 EASE - Everyday Activity Science and Engineering, University of Bremen (http://www.easecrc.org/), subproject H2.

REFERENCES

[1] Capcom Production Studio 4. 2005. *Resident Evil 4*. Game [Gamecube].

[2] Simon Demediuk, Marco Tamassia, William L. Raffe, Fabio Zambetta, Florian "Floyd" Mueller, and Xiaodong Li. 2018. Measuring Player Skill Using Dynamic Difficulty Adjustment. In *Proceedings of the Australasian Computer Science Week Multiconference (ACSW '18)*. ACM, New York, NY, USA, Article 41, 7 pages. https://doi.org/10.1145/3167918.3167939

[3] Robin Hunicke and Vernell Chapman. 2004. AI for Dynamic Difficulty Adjustment in Games.

[4] Changchun Liu, Pramila Agrawal, Nilanjan Sarkar, and Shuo Chen. 2009. Dynamic difficulty adjustment in computer games through real-time anxiety-based affective feedback. *Int. Jrnl. of Human-Computer Interaction* 25, 6 (2009), 506–529.

[5] E McAuley, T Duncan, and V V Tammen. 1989. Psychometric properties of the Intrinsic Motivation Inventory in a competitive sport setting: a confirmatory factor analysis. *Research quarterly for exercise and sport* 60, 1 (1989), 48–58.

[6] Johannes Pfau, Jan David Smeddinck, and Rainer Malaka. 2018. Towards Deep Player Behavior Models in MMORPGs. In *Annual Symp. on Computer-Human Interaction in Play Ext. Abstracts (CHI PLAY '18)*. ACM, New York, NY, USA, 381–92.

[7] Scott Rigby and Richard M Ryan. 2011. *Glued to games: How video games draw us in and hold us spellbound: How video games draw us in and hold us spellbound*. ABC-CLIO.

[8] Andrew Rollings and Ernest Adams. 2003. *Andrew Rollings and Ernest Adams on game design*. New Riders.

[9] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, and Laurent Sifre et al. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature* 529 (2016), 484–489.

[10] Alexander Streicher and Jan D. Smeddinck. 2016. Personalized and Adaptive Serious Games. In *Entertainment Computing and Serious Games*, Ralf Dörner, Stefan Göbel, Michael Kickmeier-Rust, Maic Masuch, and Katharina Zweig (Eds.). Lecture Notes in Computer Science, Vol. 9970. Springer International Publishing, Cham, 332–377.

[11] Rodrigo Vicencio-Moreira, Regan L. Mandryk, and Carl Gutwin. 2015. Now You Can Compete With Anyone: Balancing Players of Different Skill Levels in a First-Person Shooter Game. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 2255–2264. https://doi.org/10.1145/2702123.2702242