Difficulty in Videogames : An Experimental Validation of a Formal Definition

Maria-Virginia Aponte
C.E.D.R.I.C. / C.N.A.M.
292 Rue Saint Martin
75003 Paris, France
Maria-Virginia.Aponte@cnam.fr

Guillaume Levieux
C.E.D.R.I.C. / C.N.A.M.
292 Rue Saint Martin
75003 Paris, France
guillaume.levieux@cnam.fr

Stéphane Natkin
C.E.D.R.I.C. / C.N.A.M.
292 Rue Saint Martin
75003 Paris, France
stephane.natkin@cnam.fr

Résumé

This paper synthetically presents a reliable and generic way to evaluate the difficulty of video games, and an experiment testing its accuracy and concordance with subjective assessments of difficulty. We propose a way to split the gameplay into measurable items, and to take into account the player's apprenticeship to statistically evaluate the game's difficulty. We then present the experiment, based on a standard FPS gameplay. First, we verify that our constructive approach can be applied to this gameplay. Then, we test the accuracy of our method. Finally, we compare subjective assessments of the game's difficulty, both from the designers and the players, to the values predicted by our model. Results show that a very simple version of our model can predict the probability to the player has to lose with enough accuracy to be useful as a game design tool. However, the study points out that the subjective feeling of difficulty seems to be complex, and not only based on a short term estimate of the chances of success.

0. More about the authors :
http ://cedric.cnam.fr/index.php/labo/membre/list
http ://guillaumelevieux.com
ACE 2011 Full Paper
ACM version : doi.acm.org/10.1145/2071423.2071484

1 Difficulty is part of the fun

Difficulty scaling is one of the most fundamental issues of game design [4] [1]. As Bernard Suits puts it, "*playing a game is the voluntary attempt to overcome unnecessary obstacles*" [19]. Difficulty scaling is indeed about properly setting up these obstacles. Scaling the difficulty of the player's goals, and precisely setting the pacing of difficulty all along the game, is thus a crucial part of game design. A good game design provides the right *difficulty slope* (Fig. 1) [17],[5].

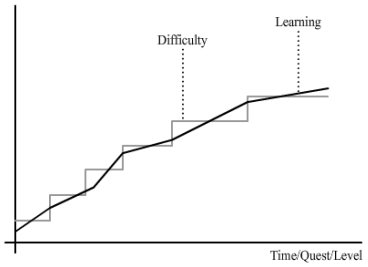

FIGURE 1 – Difficulty and learning curves.

Numerous works in cognitive psychology and game design theory try to explain the relation between the player's enjoyment and diverse characteristics of the game he is playing. Thomas Malone describes video games enjoyment as stemming from the levels of *challenge*, *curiosity* and *fantasy* [15] [16]. The level of challenge is directly related to the game's difficulty. Yanakakis et al have reported many experiments confirming Malone's model [22], [23]. Nicole Lazzaro associates fun with, among others, the *Hard Fun* dimension, also relating to overcoming difficult tasks [12]. Ryan et al apply their self-determination theory to video games and show how enjoyment is related to the feeling of *competence*, and thus, to the game's difficulty [18]. Voerderer et al show that *competition* is linked to video game enjoyment [21]. Sweetser et al see challenge as one of the most important part of their Game Flow framework [20]. All these works clearly point difficulty as one of the core components of fun in video games.

Psychology also helps us to understand the way difficulty is linked with player's enjoyment. The most widely known work in this domain, Mihaly Csikszentmihalyi's theory of *flow*, clearly states that the difficulty of a task has to be just at the right spot, so that the player feels totally engaged [6]. The player may feel bored if the task is too easy, or anxious if it's too hard. But the link between difficulty and pleasure might be more complex, especially when dealing with the variations of difficulty during the whole player's experience. Klimmt et al's work shows that at the beginning, players tend to like a lower level of difficulty [10]. Loftus et al made a judicious comparison between reinforcement schedules and video games, leading to think that variations in the difficulty level would make the game more enjoyable [14]. Indeed, if the player has to deal with both success and failure and is not able to predict whether he will succeed or not, then he's on

a partial reinforcement schedule, which induces the strongest motivation. They also explain that to maximize the player's motivation, we should maximize his regret. Regret is the highest when the player has almost succeeded but did not, and thus when the challenge difficulty is slightly higher than the player's current level. We discuss and further examine these relationships between difficulty and cognitive models in [13], but from this short summary, we can see that there exists many complex relationships between the game's difficulty slope and the player's enjoyment and motivation.

These works clearly show that difficulty scaling is important, but moreover, that much can be done to motivate the player when precisely crafting the game's difficulty slope and not just coarsely adjusting it. However, there is still lack of a precise definition of difficulty as a measurable parameter. Difficulty scaling is mainly a heuristic and iterative process that game designers handle intuitively. We want to help them in this task by proposing a theoretical, out-of-a-specific-context model to measure a video game's difficulty.

It is important to point out that we do not propose a psycho-cognitive model of video game playing's difficulty. Our approach is bottom-up : we do not start from the player cognitive models but from game design practices. Our goal is to provide a way to compare the difficulty curve forecast by the level designer and the observed one, to compare the progression of difficulty in several games and to analyse scientifically the relationships between the player's engagement and the level of difficulty. We consider that any gameplay can be split up into atoms [11]. We then consider that game and level designers build many successive playing contexts, or challenges, using these atoms. Each challenge proposes a specific goal the player has to attain [5]. In challenges, the designer combines and tunes these atoms to scale the difficulty. The resulting may ask different kind of effort to the player, which we categorize as *sensitive*, *logical* or *motor* [13]. Sensitive difficulty corresponds for example to make useful game objects hard to find by the player. Logical difficulty forces the player to make complex inferences to determine his next move. Motor difficulty stems from the time and space constraints of each player's action. But whichever kind of difficulty the challenge is based on, we consider that the designer manipulates the player chances of success. Of course, game design is more than just preparing success or failure [9]. Nevertheless, the probability of success is the principal and most observable aspect of difficulty. We thus base our evaluation of difficulty on this parameter.

We thus take the designer's constructive approach to build our model. We start by identifying the core mechanics the player will learn to master, and use them to build a player model. We then try to find the link between this player model and the probability the player has to reach the successive goals of the game. We already detailed this approach in a previous work [2] [3], and synthetically present it in the next section, before the case study. In the next sections, we evaluate our model within an experiment : is it accurate enough, and does it correspond to the subjective assessment of difficulty by players ?

2 Challenges and abilities

To measure a video game's difficulty, we first propose to decompose the whole gameplay into a follow-up of challenges. This follow-up might be dynamically generated by an emergent gameplay or statically designed as a network of quests in a more scripted kind of gameplay [8]. Nevertheless, playing always implies being attached to a result, if we rely on the Jesper Juul's synthetic and widely accepted definition of video games [7]. As a consequence, we can always choose a moment when we decide whether the player succeeded or failed to get a specific result. If we state a *challenge* as *the objective to get a specific result*, we can thus always extract a set of challenges that any gameplay will propose to a player.

Measuring the difficulty of a gameplay can thus be stated as measuring the difficulty of the challenges composing it. The currently widely adopted approach is to calculate a score, based on a heuristic function mixing the different achievements and failures of the player during the challenge. But as we can't assert the validity of the heuristic, this measure may be unreliable and give us a wrong or distorted image of the difficulty. We thus propose that a challenge can only be measured, and thus stated, in term of *success* or *failure*. The difficulty of a challenge then becomes *the probability the player has to fail at it*. Talking about success or failure leads to a more precise definition of what a challenge is, and also leads to a reliable and out-of-context way to describe it's difficulty.

Nevertheless, the probability of failing to overcome a challenge must still be evaluated. To statistically evaluate a parameter, we need it to be constant. However, the player's level increases as he plays, and thus, his probability to lose tends to decrease. Difficulty is thus always changing, and we must find a correct approximation to evaluate it anyway. We propose to distinguish two kinds of challenges, the *core* challenges and the *composite* challenges. Core challenges are at the very basis of the gameplay. They are the basic mechanics the whole gameplay is built with. Core challenges tend to be the most repetitive and short ones we give to the player, like shooting a target in any FPS, or a kind of jump in a platformer. The composite challenges are then built from the core ones. They are special combinations of a set of core challenges. They tend to be longer and unique. They often correspond to the objectives we clearly give to the player : they are the visible surface of the gameplay's machinery.

Thus, as core challenges are short and repetitive, we consider that the player's level won't change that much before we get enough samples to make a statistic evaluation of his failure probability. But for composite challenges, the player's level will change too much before we get enough results. To do so, we must take many players, and evaluate the failure probability from the results we got *when they all had the same level*. As we always can measure the core challenges' difficulty, and as they represent the very heart of the gameplay, we propose to evaluate the player's level from his probability to win the core challenges that the composite challenge is made of. We can create a multi dimensional model of the player's level based on his observed performance on core challenges. In this model, each *ability* of the player is measured using one of the gameplay's core challenge. In our experiment, we have chosen to simplify this player model,

and to only take into account the most crucial ability for the current composite challenge. We then evaluate the challenge's difficulty from the result of players showing the same level for this ability. The difficulty of a composite challenge c can thus be defined as a conditional probability, for a given ability a.

$$D(a, c) = Probability\{Lose(c)|Level(a, c)\} \quad (1)$$

For further explanations on the calculus of the difficulty equation, the reader may refer to [13] or [2]. In the reminder of this paper, we refer to composite challenges as challenges, and to core challenges as abilities.

Finding the abilities the player has to develop in order to get better at the game, and thus play with a lower difficulty, can be a tough task. Indeed, it demands a thorough understanding of the studied gameplay. In the following example, we will study basic abilities, but we must keep in mind that one can choose much more complex abilities. One may for example choose to describe a specific play style such as *sniping*, or *close range combat*. Our contribution here is to state what one must always try to respect when defining an ability. First, the player's behavior must always be as short as possible, so that we can observe this behavior as many times as possible in a short period of time, while the player's level did not raise too much. Second, one must state a failure / success condition for this behavior, in order to avoid any complex and unverifiable heuristic. Third, one should always be able to determine whether the player is actually trying to use this ability. If we see the player failing at something he wasn't really trying to do, we won't learn anything valuable.

It is also to note that the way we propose to analyse a gameplay helps a designer to find the best set of abilities. As we will describe in section 3.1, our software use logs of ingame events to study the correlations between abilities and challenges difficulty. When preparing for a playtest, the designer will have in mind a set of abilities. However, the designer should always try to log any event that seems important, even if unrelated to these abilities. That way, if the chosen abilities are shown not to have any link with the game's difficulty, he may always modify them or look for more satisfying ones without running a new playtest. The wider the set of logged event, the more the designer will be able to explore his gameplay offline and find the good set of core challenges.

3 Case study

To test our measure of difficulty, we designed a short video game, using a standard game engine and a basic and widely used kind of gameplay (Fig. 2). Our main objective was to determine if we were able to measure the main player's abilities, and to evaluate the challenges' difficulty.

The game was developed using Unreal Development Kit. As the experiment was supposed to be short, we designed a small level. We changed the AI so that any automatic skill adjustment was removed and enemies only chased the player and did not fight each other. We modified the standard gameplay of Unreal : our gameplay releases waves after waves of enemies, and choose these waves to

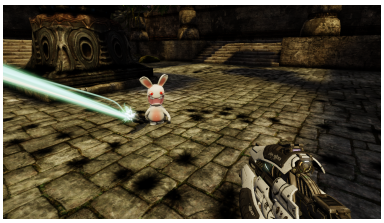

FIGURE 2 – Screen shot of the game.

match our a-priori difficulty curve. We tested the player's abilities to aim right and to keep moving. The core challenge *aiming* was won when a shot hit a target, and the core challenge *moving* was lost when the player stayed at the same place during a certain period of time.

The difficulty adjustment system of the game, responsible of driving the experience along our difficulty curve, was not based on our measuring model. We just played the game again and again, and evaluated the challenges as we played them. Indeed, another way to evaluate our model is to compare our hypothetical, subjectively evaluated difficulty to the evaluated one. After a tutorial session, our difficulty slope was slowly growing and then oscillating between easy challenges, and just-above-your-level challenges. Between challenges and after a minimum time lapse, players were asked, for this last group of enemies, how enjoyable and how hard the game was, both on five points Lickert scales (Fig. 3).

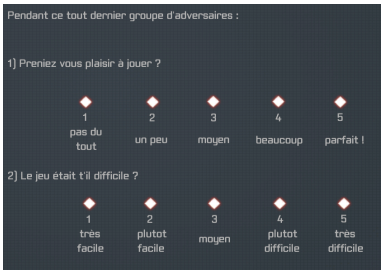

FIGURE 3 – Evaluation screen.
Translation :
During this last wave of enemies :
Did you enjoy playing :
not at all, a little, medium, a lot, perfect !
Was the game easy :
very easy, easy, medium, somewhat hard, very hard.

We designed 10 challenges, using the number of enemies and each enemy's

skills level to adjust the difficulty. The skills level of the enemy changes their ability to aim precisely, to move fast, and to do tricky jumps to dodge the player's attacks. Figure 4 shows, for each challenge, the number of enemies and their skill level, ranging from 0 to 5.

Challenge	Nb of enemies	Skill level
0	1	0
1	1	1
2	1	2
3	2	2
4	2	3
5	3	3
6	3	4
7	4	4
8	4	5
9	5	5
10	6	5

FIGURE 4 – List of challenges and their characteristics.

Difficulty is a relationship between a player and a challenge. It has no sense to just talk about the difficulty of a challenge, but we can tell the difficulty of a challenge with regard to a specific player. The Dynamic Difficulty Adjustment (DDA) algorithm calculates the player's level in two steps. First, we compute a *temporary level* for the player. The player's temporary level only changes in two cases : if he wins a hard challenge or loses an easy one. More precisely, when a player of level n loses an easy challenge, that is a challenge of level $m \leq n$, we give him the temporary level $m - 1$. If the player wins a hard challenge, that is a challenge of level $m \geq n$, we give him the *temporary level* $m + 1$. Second, we compute the player's level as the mean of the 5 last temporary levels. While designing the game, we made hypothesises on what the difficulty would be when assigning a specific challenge to a specific player. For example, we thought that if we assigned a challenge more than two levels under the player's level [1], then his probability to loose wan only 0.05. Figure 5 summarizes our a-priori, theoretic difficulty levels.

After a short tutorial, where the player was not able to lose and thus get frustrated, the game start follows a specific difficulty slope. We designed two difficulty slopes that the game may follow. Theses slopes are shown in figure 6. The experiment was 24 minutes long. During the first 12 minutes, the game randomly chooses to follow one of the two difficulty slopes. It then switches to the other one for the last 12 minutes.

In this paper, we report the results related to the following questions :

1. Is there indeed a link between measured abilities and the probability the player has to lose a challenge : i.e. can we apply our constructive approach

1. for example challenge number 2 to a level 5 player

Level difference	Theoretic difficulty
< -2	0.95
-2	0.8
-1	0.6
0	0.5
1	0.4
2	0.2
> 2	0.05

FIGURE 5 – Theoretic difficulty of challenges.

Level differences corresponds to the challenge's level minus the player's level.

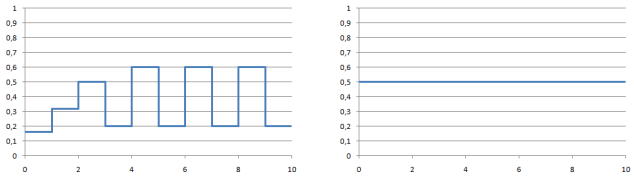

FIGURE 6 – Difficulty slopes.

to this experiment ?

2. Can we easily parametrize the relationship between measured abilities and the outcome of a challenge : i.e. can we create a simple parametric model of our difficulty function ?

3. Is there a strong relationship between our theoretical difficulty and our measured difficulty : i.e. do the designer feel the difficulty the same way we evaluate it ?

4. Is there a strong relationship between the difficulty reported by players and our measured difficulty : i.e. do the players feel difficulty the same way we evaluate it ?

The experiment was run in three sessions, with a total of 72 players, 56 men and 16 women. Average age was 26 ($\sigma = 6.5$). The test was 24 minutes long, but two players quit in the middle, and were removed from the experiment. We thus analysed 70 players, for a total of 2368 attempts to win one of the 10 challenges our game was able to propose.

3.1 Tools

To record ingame events, we created a dynamic linked library in C++. Unreal Development Kit allows to link any dll file, and to use the exported functions with Unreal Script. We then created a *logEvent* function, which allows to add an event to a XML file, with a time stamp. We did not use the standard *'log()* function of UDK, as the log file is erased every time the engine is launched, and as it's also filled with UDK's own messages. Using our own files let us record every test in a separate file, and automatizes the experimentation. Two of the three test sessions were supervised by students in Ergonomic, and they did not have to manually manipulate the data files, just focusing on giving the first basic instructions to the player.

Then, we interpreted our XML files with our custom software, which allows us to define core and composite challenges, to calculate the player's abilities, and to calculate the first basic statistics. This software allows us to measure the difficulty of the challenges, the linear correlations between abilities and challenges difficulty, and to draw each game's difficulty curve. We use specific *Lua* code to calculate the player's abilities from the recorded events. For further analysis, the results were then exported as CVS files, and loaded into Excel with XLStats plugin. Figure 7 shows a screen shot of our custom software.

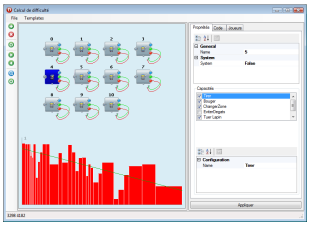

FIGURE 7 – Screen shot of our difficulty calculus software.

4 Results

4.1 Abilities and challenge outcome

For both abilities, *moving* and *aiming*, the samples were not normally distributed. Thus, we may not rely on parametric tests for the link between difficulty and abilities. We thus studied correlations, and used χ^2 independence tests. We did not get enough data to study the very hard and very easy challenges, as they were not played enough. We present the results of the 5 most played challenges.

According to these results, we chose the aiming ability as the most crucial one. Correlations are very low, but χ^2 tests show a strong relationship between the ability to aim right and the challenge outcome. It is to note that the challenge result is a binary variable, which cannot lead to high linear correlation. The moving ability seems also important for 3 challenges on 5, but as stated before,

Challenge	Aiming(ρ)	Moving (ρ)
2	**-0,229****	-0,084
3	**-0.245****	**-0.277****
4	**-0,215****	-0,111
5	**-0,307****	**-0,058****
6	**-0,148***	**0,080****

FIGURE 8 – Correlation between abilities and win/lost result per challenge (Pearson).

Results are bolded when χ^2 test shows that results and abilities are not independent (**p<0.01, *p<0.05).

we chose as a first step to build a very simple model, based on only one ability of the player.

The correlations presented in the previous figure (fig. 8) are not between abilities and difficulty, but between abilities and the binary result of the challenge. To get the relation between difficulty and abilities, we proceeded as explained in section 2. We grouped players by level, and computed the challenge's difficulty for each of this class of level. Thus, we clustered the results into 10 different classes, using the k-means algorithm with regard to the aiming ability. For each class of the aiming ability, we computed the probability the player had to lose, for each challenge. Figure 9 shows the linear correlation between the aiming ability and difficulty that we obtained. Figure 10 details the linear regression between each level for the aiming ability and the difficulty for the most played challenge.

Challenge	Aiming(ρ)
2	-0.675
3	-0.706
4	-0.757
5	-0.878
6	-0.563

FIGURE 9 – Correlation between Aiming and difficulty per challenge (Pearson).

These first results tend to show that, within the context of our gameplay, it is possible to find core challenges that describe the player's abilities. Moreover, we can measure the player's level for each ability, and these levels influence the outcome of a composite challenge. For the most played challenges, χ^2 test shows that there exists a strong relationship between the aiming ability and the challenge's outcome. For some of them, challenge 3, 5 and 6, there is also a strong relationship between the capacity to move and the challenge's outcome. The linear correlations show that we may try to approximate this relationship using a simple linear function.

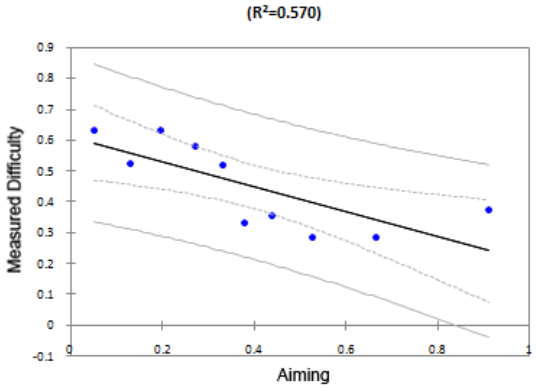

FIGURE 10 – Linear regression between aiming and difficulty for challenge 4 (most played one).

4.2 Linear approximation

The previous sections show that there exist a link between our measured abilities and the difficulty of the game. If we can parametrize this relationship, then we may be able to evaluate the difficulty of a challenge during the game, only by observing the player's abilities.

We chose to approximate our difficulty function using a simple linear model, with only one ability. The previous sections shows that the aiming ability may be a very good start for our model. We thus based our model on the aiming ability, and built it using the least squares method.

To investigate the precision of our linear model, we evaluated it's ability to generalize on new data. We calculated the relationships between abilities and difficulty using a certain amount of data, and tested, on new datas, how far our prediction was from the measured difficulty. We did not run further experiments, but just built the model using 75% of the data and then tested the precision of the predictions on the remaining 25% of the datas.

We classified the test data into 15 classes using the k-means algorithm, based on the aiming ability. We were then able to compute the measured and predicted difficulty for each one of these classes. Figure 11 shows the linear regression between the predicted and measured difficulty, on the 25% of datas we used for testing. The linear correlation is only -0.64, and our model seems to be less precise for low difficulty values.

We computed the model's error as the difference between predicted and measured values, depicted in figure 12. For the absolute error e, $\overline{e} = 0.07$ and $\sigma = 0.064$.

Figure 12 results clearly show that our model is overestimating easy challenges. When the challenge's difficulty is around 0.3, that is, when we actually measured that 30% of the players failed at it, the model predicts difficulties between 0.35 and 0.5. The prediction is more accurate for higher difficulty le-

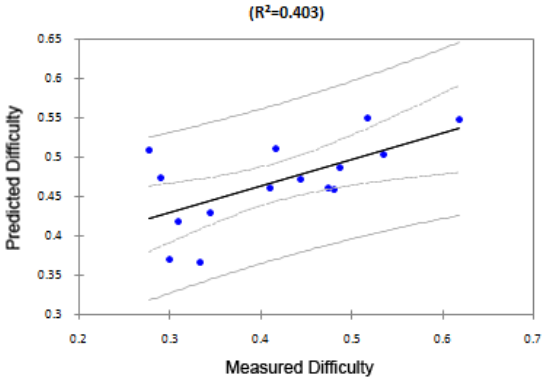

FIGURE 11 – Predicted and measured difficulty.

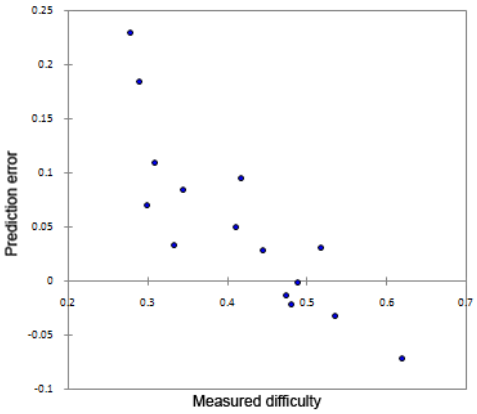

FIGURE 12 – Prediction error for each measured difficulty.

vels. When the challenge's measured difficulty is higher than 0.35, the prediction stays under a 10% error. We will further discuss these results in section 5 .

4.3 Theoretical and predicted difficulty

We then tested the link between theoretical and predicted difficulty. Before the tests, we roughly estimated the difficulty from our experience of the game, and made specific choices when designing the DDA algorithm. Especially, we chose to evaluate the player by using the specific algorithm described in section 3, which may not be accurate.

The result tends to show that our theoretical difficulty was not that wrong. We computed the prediction error as the difference between theoretical difficulty and predicted difficulty. For the absolute error e, $\overline{e} = 0.17$ and $\sigma = 0.13$. χ^2 tests

show a strong relationship between theoretic and measured difficulty ($p < 10^{-4}$). We found a positive linear correlation of 0.61 between them, but it's important to point out that linear correlation is here biaised by the fact that theoretical evaluation is on an ordinal scale, not a real quantitative variable. We can look closer at the spread of the predicted difficulties, for each theoretical difficulty level. Figure 13 shows the scattergrams of the measured difficulty, for each theoretic difficulty level.

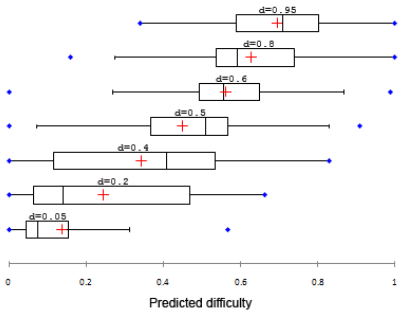

FIGURE 13 – Predicted difficulty scattergrams, for each theoretic difficulty level.

This results are also coherent with results of section 4.2. We previously showed that our model was less accurate, in this experiment, when predicting low difficulty levels. This tends to be confirmed by these results. For low theoretical difficulties, the distributions are much wider than for higher difficulty levels. More precisely, when we hypothesized that the player had a probability of 0.2 or 0.4, the model predictions were the most widely spread.

4.4 Reported and predicted difficulty

During the test, players were asked to evaluate the difficulty of the last challenge they played, on a 5 point Lickert scale. When the player rated the challenge as *very easy*, we valued it 0.1, and added 0.2 for each step of the scale, keeping it linear. Thus, when the player rated a *medium* difficulty, we recorded a 0.5 difficulty value. As another example, we recorded 0.9 for a *very hard* reported difficulty.

We computed the prediction error as the difference between reported difficulty and predicted difficulty. For the absolute error e, $\overline{e} = 0.21$ and $\sigma = 0.17$. We only found a 0.32 linear correlation between measured and reported difficulty, but χ^2 test showed a strong link between them ($p < 0.01$), and linear correlation is biaised by the fact that our evaluation is on an a single ordinal scale, not a real quantitative variable. Figure 14 shows measured difficulty scattergrams, for each reported difficulty level. Again, it seems that the easier the challenge is, the wider the distribution is.

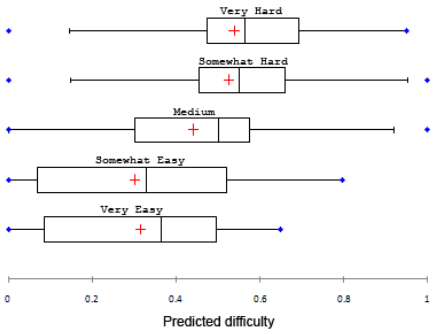

FIGURE 14 – Predicted difficulty scattergrams, for each reported difficulty level.

5 Discussion

First, our results show that for our game, a FPS based on successive waves of enemies implemented on a widely used game engine, one can show the link between the player abilities and his probability to lose. Within this experiment, we were actually able to split the gameplay into core challenges, and to measure some of the player's abilities based on these core challenges. Moreover, we showed that there existed indeed a relationship between abilities and the challenge's difficulty. These first results do not show that what we measure is the game's difficulty, but that our methodology can be applied in a standard gameplay, and that as we supposed, there existed a link between what we call player abilities and the probability the player has to lose.

Then, we tried to parametrize this link, that is, the function that gives a player's probability to lose based on one of his abilities. We have built a simple model of this function, using a linear regression. We tested it on 25% of the data to evaluate it's accuracy. In our case study, this simple linear approximation was found to be relatively accurate. The mean absolute error is $\overline{e} = 0.07$ with $\sigma = 0.064$. When looking closer at it, we found out that it was less precise for the easiest challenges than for the tough ones. These results shows that with a very simple model, and in the context of our experiment, we can get a fair estimate of the player's probability to lose : it may certainly be useful for a game designer to know the probability a certain player will have to lose, plus or minus 0.07. Moreover, we may reach a higher accuracy with a more complex model, taking into account the whole set of player abilities.

The lack of accuracy may also be explained by our population. In order to have a large amount of testers, we took whoever wanted to play. As a consequence, we had two kinds of easy challenges. It is not the same to give an easy challenge to a good player, than to give a very easy challenge to a bad player. During the tests, we had players who had never played a FPS before, and their behaviour were mostly erratic. The result of the challenge was thus much more random than for skilled players, and as they kept loosing, we kept giving them

easier and easier challenges, down to the point where we could not go easier. We should run another experiment, targeting a more homogeneous population.

We then tried to test if the probability the player had to lose was actually related to the game's difficulty. We tested it in two ways. First, we compared our predicted difficulty with the theoretical difficulty we had hypothesized when coding our DDA algorithm. In this algorithm, we did not evaluate the player's level from his abilities, but from using the result obtained on the previous challenges. We ranked challenges from the easiest one to the most difficult one, and chose hypothetical difficulty values based on the difference between the current challenge's rank and his previous won/lost challenges. χ^2 testing showed a strong link, with a 0.61 linear correlation. The mean absolute error between hypothesized and predicted difficulty was $\overline{e} = 0.17$ with $\sigma = 0.13$. We also found out that when looking closer at scattergrams, our prediction was still less accurate for the theoretically easiest challenges. This is coherent with the previous findings, that shows that our linear model is less accurate with easy challenges.

This results shows that our theoretical and predicted difficulty are linked, and that we may actually be measuring the game's difficulty. Of course, it's hard to tell which one of both variables is the less accurate. We may argue that our prediction model is precise and our theoretical value is just a rough estimate, but also that we as designers know our game's difficulty and thus that our prediction is not accurate enough. But both theoretical and predicted difficulties are still relatively close to each other, especially for hard challenges, which leads to think that we are actually giving an estimate of the game's difficulty.

As a second way to determine whether we were actually predicting something related to the actual difficulty of the game, we compared our predicted difficulty with the difficulty as experienced by players. χ^2 test still show a strong relationship, but correlation is very low (0.32) and the mean absolute error was bigger : $\overline{e} = 0.21$ with $\sigma = 0.17$. We still have wider spread distributions for the easiest challenges.

Reported difficulty is a very important one : what the player is thinking is fundamental, because we try to motivate the player by proposing the best difficulty curve. This is also where we get the less accurate results. Just looking at the mean error, we can see that when we predict a difficulty, the player may tell that it's 0.21 points harder or 0.21 easier.

First, we may explain this lack of accuracy is partly due to the fact that we only assess difficulty on a single 5 points Lickert scale. Indeed, we thought that it was not possible to ask about the game difficulty after the game, because the player may never remember the whole list of challenges he played during 24 minutes. Thus, we had to ask him about difficulty at the end of each challenge. As we did not want to disturb him too much, we only asked one question about difficulty. The first thing to do would be to design another way to get the player's feeling of difficulty, in a more accurate but still non intrusive way.

Nevertheless, these results may be explained by the fact that the players do not only assess difficulty by estimating their short term chances of success. The first thing that stroke us during debriefing sessions was that the players sometimes evaluate difficulty in a very strange way. For example, some players

kept being motivated after many failures, and even reported a low difficulty level. They then explained us that it was not that the game was hard, but that they just were bad players, and had to improve their level. They indeed rated hard games as easy one, based on that perception of the game's difficulty. This leads us to think that the subjective assessment of a challenge difficulty is more complex than just guessing their short term chances of success. Players perceive their own abilities, but also estimate what these abilities will become if they practice. They take this more complex model into account when they assess the game's difficulty and our model does not.

6 Perspectives

In this paper, we propose a method to measure the difficulty of a gameplay, and thus of the challenges composing it. We considered challenges as a predefined game context, where the player tries to reach a certain goal. If the game context changes, then it's a different challenge, which need a new evaluation. For example, if the player has to kill a monster, any modification of this monster speed, life or armor points makes a different challenge.

Instead of having to evaluate again the difficulty of any new challenge, we would like to have a model able to infer this new difficulty. In our current model, we only generalize on new players. That is, players are defined by a set of parameters (i.e. abilities), and by using a big enough sample of players, we make a sufficiently robust model to infer the difficulty of any new player, with a certain accuracy, evaluated in section 4.2. We may also define challenges of the same kind by a set of parameters, like for instance, in the previous example, the monster speed, life and armor points. By testing a sufficient amount of challenges, we may also be able to generalize on new challenges. This would be particularly useful if the game uses a DDA algorithm, and thus add small dynamic variations to the context of a given challenge.

Then, the next step of our research will be to test a more complex approximation of our model and see if we gain in accuracy. Also, we need to investigate the player's subjective evaluation of difficulty and see how we can modify our model to better predict the player's feeling of difficulty. Furthermore, we will need to evaluate many different kinds of gameplay, and explore the link between engagement and difficulty.

7 Conclusion

Studies in psychology and game design theory show that difficulty is a core issue of game design. We have proposed a method to analyse a gameplay, splitting it into *abilities* and *challenges*, to construct a reliable and generic measure of a game's difficulty. This model does not explain the game's difficulty like a cognitive model would. It takes a game design's constructive approach, and tries to evaluate the player's chances of success. Our measure takes the player's

learning into account, and allows to evaluate the difficulty from a statistical approach. We then tested our model during an experiment based on a generic FPS gameplay using a widely used commercial engine. We built a linear approximation of our model. This simple approximation was found to be relatively accurate, with a mean absolute error $\overline{e} = 0.07$ ($\sigma = 0.064$). The last part of our study shows that the subjective feeling of difficulty is complex and not only linked with the outcome of the challenge.

Références

[1] E. Adams. The designer's notebook : Difficulty modes and dynamic difficulty adjustment. Gamasutra : http ://www.gamasutra.com/ (last access 01/2009), 2008.

[2] M.-V. Aponte, G. Levieux, and S. Natkin. Measuring the level of difficulty in single player video games. *Entertainment Computing*, In Press, Corrected Proof :–, 2011.

[3] V. Aponte, G. Levieux, and S. Natkin. Scaling the difficulty level of single player video games. In *8th International Conference on Entertainment Computing , Paris, France*, 2009.

[4] D. Boutros. Difficulty is difficult : Designing for hard modes in games. Gamasutra : http ://www.gamasutra.com/ (last access 01/2009), 2008.

[5] E. Byrne. *Game Level Design (Game Development Series)*. Charles River Media, December 2004.

[6] M. Csikszentmihalyi. *Flow : The Psychology of Optimal Experience*. Harper Perennial, March 1991.

[7] J. Juul. The game, the player, the world : Looking for a heart of gameness. In M. Copier and J. Raessens, editors, *Level Up : Digital Games Research Conference Proceedings*, pages 30–45, 2003.

[8] J. Juul. *Half-Real : Video Games between Real Rules and Fictional Worlds*. The MIT Press, November 2005.

[9] J. Juul. Fear of failing ? the many meanings of difficulty in video games. In *The Video Game Theory Reader 2*. Wolf & Bernard Perron (eds.), 2009.

[10] C. Klimmt, C. Blake, D. Hefner, P. Vorderer, and C. Roth. Player performance, satisfaction, and video game enjoyment. In *ICEC*, pages 1–12, 2009.

[11] R. Koster. A grammar of gameplay - game atoms : can games be diagrammed ? Game Designer Conference talk : http ://www.theoryoffun.com/grammar/gdc2005.htm (last access 01/2010), 2005.

[12] N. Lazzaro. Why we play games : Four keys to more emotion without story. In *Game Developers Conference*, March 2004.

[13] G. Levieux. *Mesure de la difficulté dans les jeux vidéo*. PhD thesis, Conservatoire National des Arts et Métiers, 2011.

[14] G. R. Loftus and E. F. Loftus. *Mind at Play, The psychology of Video Games*. Basic Books, 1983.

[15] T. W. Malone. What makes things fun to learn ? heuristics for designing instructional computer games. In *SIGSMALL '80 : Proceedings of the 3rd ACM SIGSMALL symposium and the first SIGPC symposium on Small systems*, pages 162–169, New York, NY, USA, 1980. ACM.

[16] T. W. Malone. Heuristics for designing enjoyable user interfaces : Lessons from computer games. In *Proceedings of the 1982 conference on Human factors in computing systems*, pages 63–68, New York, NY, USA, 1982. ACM.

[17] S. Natkin, A.-M. Delocque-Fourcaud, and E. Novak. *Video Games and Interactive Media : A Glimpse at New Digital Entertainment*. AK Peters Ltd, 2006.

[18] R. M. Ryan, C. S. Rigby, and A. Przybylski. The motivational pull of video games : A self-determination theory approach. *Motivation and Emotion*, 30(4) :344–360, decembre 2006.

[19] B. H. Suits. *The grasshopper : Games, life, and Utopia*. University of Toronto Press, 1978.

[20] P. Sweetser and P. Wyeth. Gameflow : a model for evaluating player enjoyment in games. *Comput. Entertain.*, 3(3) :3, July 2005.

[21] P. Vorderer, T. Hartmann, and C. Klimmt. Explaining the enjoyment of playing video games : the role of competition. In *ICEC '03 : Proceedings of the second international conference on Entertainment computing*, pages 1–9, Pittsburgh, PA, USA, 2003. Carnegie Mellon University.

[22] G. N. Yannakakis and J. Hallam. Towards optimizing entertainment in computer games. *Applied Artificial Intelligence*, 21(10) :933–971, 2007.

[23] G. N. Yannakakis and J. Hallam. Real-time game adaptation for optimizing player satisfaction. *IEEE Transactions on Computational Intelligence and AI in Games*, 1(2) :121–133, June 2009.