information

Article

Creating Competitive Opponents for Serious Games through Dynamic Difficulty Adjustment

David Kristan *, Pedro Bessa, Ricardo Costa and Carlos Vaz de Carvalho

Games, Interaction, Learning Technologies, Instituto Superior de Engenharia do Porto, 4200-072 Porto, Portugal; 1150046@isep.ipp.pt (P.B.); 1150045@isep.ipp.pt (R.C.); cmc@isep.ipp.pt (C.V.d.C.)
* Correspondence: kristan.david18@gmail.com

Received: 12 February 2020; Accepted: 11 March 2020; Published: 14 March 2020

Abstract: Competition is a basic element of our society. It drives us to rise above previously perceived limitations, increases our engagement and makes the world more interesting. Competition rewards our existing skills and prompts us to identify and improve our weaker skills. In games, player engagement is achieved, at least in part, by providing him/her with competition at the right amount of difficulty. Achieving and maintaining this exact level of challenge is one of the most difficult tasks for a game designer. The use of Dynamic Difficulty Adjustment techniques allows the game to dynamically adjust the challenge according to player performance, therefore keeping him/her always on edge, immersed and fully active. New information can then be more easily acquired, which is especially important in Serious Games. This paper describes how DDA techniques were used to create two strategic, goal-oriented computer-controlled (CC) players in order to deliver a higher level of competitiveness for the user in Transform@, a Serious Game aimed at developing entrepreneurship skills. As a result, the strength of the computer controlled player increased by more than 100%. By developing a good strategy for the AI and using DDA the game includes now a powerful opponent which has increased the engagement level of the player.

Keywords: serious games; artificial intelligence; dynamic difficulty adjustment; flow zone; entrepreneurship

1. Introduction

Serious Games strive to combine the entertainment aspect of a game (so the player is having fun) with the improvement of knowledge, skills or competences [1,2]. Therefore, Serious Games must on the one hand be realistic, as they represent a real phenomenon or a serious purpose, and on the other hand be engaging, fun and end in success or failure [3]. According to Cook's player psychological model, players are driven by their desire to acquire a new skill, so fun is achieved when the player learns something new or masters a new skill [4]. The effect of Serious Games is particularly relevant when players are immersed and focused on the game. Research shows that one of the crucial elements in effective games is an interacting element in the form of competition or cooperation [5,6]. Thus, it is not surprising that video game players perceive a competitive game situation to be more enjoyable [7], independently of the competition form: a competition against oneself, time or against another human player or team [8]. The benefits of competitiveness have been well documented in the literature: potential to draw attention of students [9], excellent motivational tool [10,11] and increase of the learning efficiency [12]. On the other hand, excessive competitiveness can also have negative effects, like reducing performance, increasing anxiety and motivation loss for the losing players [13]. Therefore, when implementing a competitive element into a game, it is crucial to think about all the players, to minimize the negative effects and maximize the positive.

The competitive elements of a game present the player with two possible outcomes, both impacting his/her emotional state. Successful completion of the competitive situation leads to a euphoric experience of enjoyment and thus an increase in motivation [7]. The moment of mastery fills the player with joy and a sense of accomplishment which motivates him/her to look for new challenges. Thus, a constant loop of learning, mastery and reward is created which keeps the player engaged and eager to learn. On the other hand, dissatisfactory outcomes evoke anger and/or frustration, which can still increase motivation (player is eager to defeat a hard opponent), but decreases the entertainment aspect [7]. Competition is motivating only when it provides the appropriate difficulty level: the player must be uncertain about completing or failing to reach the goal [14]. The perceived difficulty of the game should be set just right and unique for each individual player, to maximize positive learning goals. If the challenge is too difficult, the player will probably fail and will feel frustrated; on the other hand, getting a reward for investing little effort is never satisfactory as the player doesn't feel challenged and the game will not achieve the desired amusing effect. Ideally, the difficulty of the game challenge must be adjusted, in each moment, to the individual player, thus providing him/her with the right amount of challenge and maximizing his/her development. The quality of the experience is impacted by the perceived difficulty of the imposed challenge and the perception of one's abilities to overcome it. Difficulty must be adapted to the individual player's skill to provide the motivational impact that we desire. A common approach to ensuring this is Csikszentmihalyi's flow model [15].

The use of Dynamic Difficulty Adjustment techniques allows us to match player ability and the challenge difficulty in real time, while he/she is playing, therefore maximizing the reward and acquisition of knowledge, skill or competence [16].

This article presents how DDA techniques were used to create two strategic, goal-oriented computer-controlled (CC) players in order to deliver a higher level of competitiveness for the user in Transform@, a Serious Game aimed at developing entrepreneurship skills, and the achieved results.

2. Dynamic Difficulty Adjustment (DDA)

A well-designed Dynamic Difficulty Adjustment system provides a consistent, perfectly paced game, which brings a greater sense of accomplishment for the player. To create a flexible interactive experience, which adjusts automatically to the player, different DDA systems can be used, while always bearing in mind the importance of decreasing the costs related to the development of adaptive games by using the most effective system for a specific game.

In his review of DDA techniques, Zoahib presents the following classification of techniques [17]:

- Probabilistic Methods
- Single and multi-layered perceptrons
- Dynamic scripting
- Hamlet System
- Reinforcement Learning
- Bound for Trees and Artificial Neural Networks
- Self-organizing System and Artificial Neural Networks

Although not a full-fledged taxonomy, this reference anticipates that the implementation of DDA relies either on probabilistic/statistical analysis of the user behavior or in more artificial intelligence processes, namely neural networks and machine learning.

An example of the application of DDA can be found in [18], where the researchers extracted traces from previously played games of Tetris, built a case base, predicted the skill level of a player and adjusted the difficulty of the game based on that analysis. The first 10 rounds of the game were dedicated to gathering tactical decisions of the player and keeping track of the height of the board, as the skill level prediciton was made only when both conditions were fulfilled. Determining a number of rounds to gather information was complicated, as more rounds give a better estimation, but also require more time to detect the skill level and adjust the difficulty of the game. In this case, the DDA method looked for clusters in the case base to identify different groups of players and used that to predict the skill level of a new player. The number of clusters was selected with the use of k-means algorithm (varying k from 2 to 6). Based on the research, they decided to establish three clusters because of the simplicity of the game. When the system decides to help the player, it checks for how good each type of piece is in the current board for the player and selects one of three good pieces randomly. The lower the level of the player, the more help the system provides. Researchers have concluded that players obtained higher scores and had a higher level of satisfaction in games in which DDA was active.

Another great example of effective DDA usage is described in [19], where researchers analyzed and adjusted the supply and demand of a game inventory in order to control overall game difficulty Game developers analyzed the system by tracking specific identifiable patterns and iteratively refined those systems until the game was balanced. Researchers used the Hamlet system, which is primarily a set of libraries embedded in the Half Life game engine, to monitor incoming game data and estimate player's future state. They monitored the player's movement throughout the game world and estimated the player's future state based on that game data. By observing the player's inventory (health, weapons, ammunition, etc.) they could predict potential shortfalls and thus adjustment opportunities.

The goal was to keep the player in a state where his skills match the difficulty of the game by encouraging certain states, and discouraging others, as described in Csikszentmihalyi's flow model. In order to support continued engagement, they had to match the challenge with the player's skill (Figure 1). Researchers used two types of adjustment actions: reactive and proactive. Reactive actions manipulate elements that are in play (accuracy, damage of attacks, level of health etc.), while proactive actions adjust elements that are not yet in play (changing properties of off-stage entities). Proactive adjustments give more power over game behaviour, but are harder to evaluate, as they happen at a greater distance. They can often require additional reactive adjustments, which can lead to a spiralling loop of changes that can result in chaotic behaviour. Reactive changes, on the other hand, are simple to execute and have a straightforward impact on the game, but can be noticed by the player because they happen closer to the interaction.

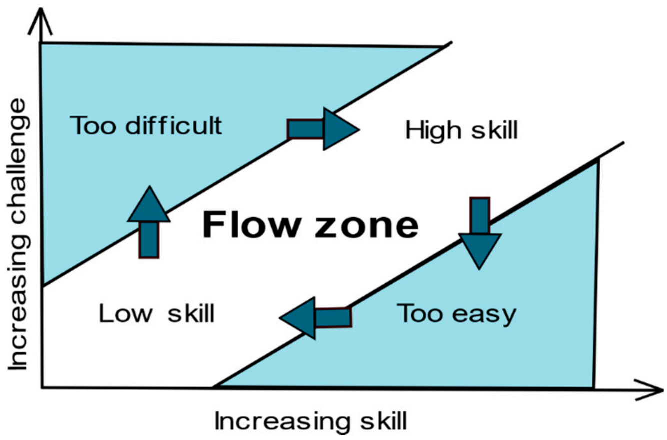

Figure 1. Flow zone, adapted from [15].

When the system decides to intervene and control the supply and demand of the goods, it can perform two different policies: comfort and discomfort zones. The comfort zone policy is characterized by steady demand and predictable supply, as the goal is to keep the player's health between 25% and 75%. If enemies overwhelm the player, they are tuned to shoot less often, less accurately and the player is supported with additional health packages. The player in the comfort zone feels challenged, but safe. The discomfort zone was developed for more experienced players, as its goal was to keep the player on the edge of his seat, fighting back from 15% or 20% health and always expecting enemies to appear. Enemy accuracy increases continually, while ammo and health become more and more scarce. The supply is thus gradually lowered, while the demand continues to increase.

Another popular approach to DDA, used in many games, is rubber banding. The term describes the relation between the player and his opponent, which are held together as a rubber band. In whichever direction the player "pulls" (playing better or worse than the opponent), the opponent will follow in that direction (starts to play better or worse). At first glance the idea is logical and seems to be effective, but can often suffer from misbalance and exploitability. As described in [20], the exploitability can be especially noticed in the racing games (Need for Speed or Mario Kart). The further ahead the player car is, the faster the opponent will drive and the further behind the player car is, the slower the opponent will drive. In Mario Kart, players with lower skill level also get more powerful weapons. The rubber banding system includes all players, regardless of their skills and makes the game more competitive while insuring that the opponent is close to the player. The concern with this technique is the exploitability, as the best tactic would be to drive slowly to ensure that opponents do not go too fast and to make sure you get all the best weapons, which is not what the games are striving for (mastering a skill).

Another interesting example of the Dynamic Difficulty Adjustment system, also described in [20], is employed in "God Hand" by Clover Studio. It follows the strategy "observe and adjust" like the other systems, but does not hide anything from the player. The difficulty level meter is actually presented in the bottom left corner of the screen for the player to observe. This meters fills with player's progression through the game and increases the difficulty when reaching the top. When the player sustains damage, the meter drops. Therefore, the game adjusts to the player, but lets him know that adjustments are happening. The player is rewarded with mastering a new skill in a very demanding environment and simultaneously receiving additional points, which can be used for new moves to help the player defeat the enemy.

Araújo et al present a very interesting study where DDA (through a fuzzy controller) was used to control the generation of levels for visual sequences, but using a robotic system [21]. Roughly, the system worked as follows: four touch-sensitive cubes acted as inputs to the system (through actuators), but also as outputs (LEDs and buzzer) to the user. By touching one of the cubes, the robot can generate a stimulus to the user, represented by the color of the LED of the cube (red, green, blue or yellow). The obtained results showed that participants were interested in using the proposed system because of the attractive human-machine interface, implemented with the robot and smart objects and achieved a greater level of satisfaction and immersion.

3. DDA in Transform@

The Serious Game Transform@ was originally developed at GILT-ISEP (a research group in the field of games, interaction and learning technologies) and Virtual Campus (a software development company) in the scope of the homonym European project. The main objective and purpose of the game is the development of entrepreneurship and e-business competences of the players [22], who are required to build up an e-commerce business in a rural context by collecting financial resources, human resources (HR) and potential clients (PC) while facing other competitors with the same goals.

Transform@ is a turn-based digital board game that can be played in either single-player or multi-player (online) mode as shown in Figure 2. In each turn, the player rolls a dice, picks a tile to move to and performs a business action. Victory goes to the player that first reaches the end tile (company registration booth) with the minimum (configurable) required set of resources (for example: 5000 €, 4 Potential Clients and 4 Human Resources) and at least 50% of company equity.

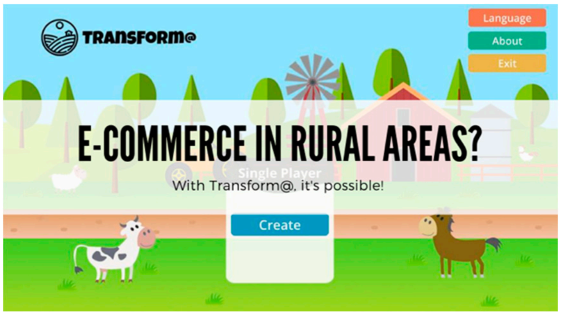

Figure 2. Transform@ entry screen.

In the board, there are three types of tiles: neutral, quiz and luck tile. In quiz tiles, the player selects the difficulty, answers a question and receives (or loses) money depending on the correctness of the answer. Luck tiles can increase or decrease the amount of resources (money, PC or HR). In relation to the business actions, the player can choose from four different categories: Friendly deals, Attack deals, Financial deals and Business deals:

- In the Friendly deals, a player can offer money to the other players to get information about their PC or hire their HR. The player can also put his/her own PCs information for auction. The opponent can either accept or reject the offer.
- In the Attack deals, the player offers money to convince opponent's PC or HR to join his/her company but the opponent doesn't receive any money for the transaction. The player pays the amount he chooses regardless of actually receiving PC or HR, so it is riskier than friendly deals, but non-beneficial for the opponent.
- The Financial deals are used for selling and buying company equity, taking loans, merging with other companies or buying them out.
- The Business deals are the most important for company growth. Here the player pays money to get more HR, PC or invest in something and get more money later.

In the single-player version, the player's opponents are computer controlled characters (1–3 players). Initially there were three difficulty levels of computer operated players: Easy Going, Entrepreneur and Tycoon. They differed by the style of negotiation, negotiation hardness, difficulties and correctness of quiz answers. In terms of decision making and strategy, all three levels operate at the turn-level without a long-term strategy which means that in almost most cases the computer player's decisions will not be based on its current state. This was revealed to be a problem as the challenge that players were faced with was not correctly adjusted to player ability, and so therefore engagement level was not at the maximum. The solution was to incorporate mechanisms that would dynamically adjust the difficulty according to the current state of the game.

This study goal was to create an opponent that decides strategically and presents a real threat to the user player thus increasing the players' engagement. Predetermining the difficulty level of the computer players doesn't consider the individuality of the human player. Game difficulty is predetermined and consequently does not adjust to uniqueness of the human player, which can lead to mismatches between the player's abilities and challenges that the game imposes on him. Having a wide range of possible difficulties that would cover every individual player's skill level imposes a problem for the player, when he has to choose one. On the other hand, a small amount of difficulty levels can hardly cover all the different types of players. Dynamic Difficulty Adjustment (DDA) offers a solution to this problem by modulating in-game systems that respond to an individual player's abilities over the course of the game. We thus provide the computer with the ability to change the difficulty for all the different skill levels without complicating things for the user. By using DDA, the player does not have to switch difficulties during the game, if the game is too hard or too easy, as the computer does the adjusting for him. With that, we can keep the immersion and flow of the game and ensuring the game enjoyability.

3.1. Data Gathering

In the beginning, 15 data gathering tests were performed in the original Transform@ game for each difficulty level: 5 games with 3 opponents per difficulty. In order to get the most accurate results, the user player didn't perform any moves during the data gathering. On average the player reaches the end tile after 10 rounds, so the data was collected in the initial 10 rounds of play. The observed parameter was the company value of the players, as it represents how much money, HR and PC the player has. For each PC/HR the company value increases by 750 €. The percentage of accomplished goals was not considered relevant, as it was clear that with random move picking it would not be possible to come closer to the desired objective of 5000 €, 4 PC and 4 HR (the company value in that case would be 11,000 €).

As is clear from the results, the three levels do not differ much from one another (Figure 3). The results were in alignment with the expectations, as you cannot have much diversity with a mostly-random choice picking. Surprisingly, what would be the middle difficulty level (Entrepreneur) had a lower average company value than the supposedly weakest one (Easy Going) which allowed to immediately identify an incorrect approach. Nevertheless, the Easy Going player had frequently a bankruptcy issue.

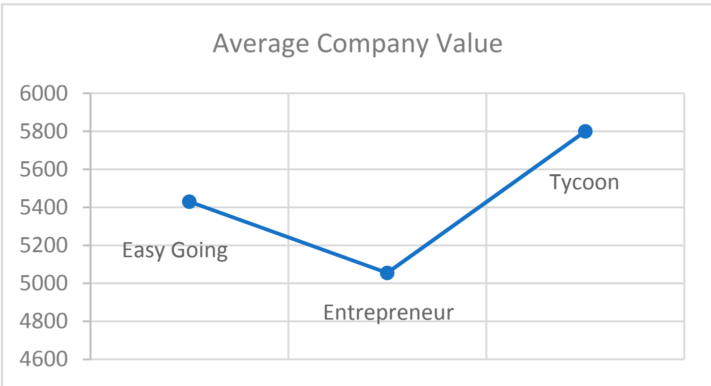

Figure 3. Average company values of the three difficulty levels.

3.2. Creating the "Strategic" Player

In this case, the goal was to improve the weakest difficulty level: Easy Going player. As mentioned before, the Easy Going player had a high bankruptcy probability (on average one in three bankrupted in a 4 player game with 3 Easy Going players). Even without this problem, its chances of winning were minimal. In order to create a strategic version of the Easy Going player (named Strategic player), first its "personality" was defined; that is, how it reasoned, what its mindset was and what kind of "person" it was. In real life an Easy Going player would be a person who has a full-time job and is running a small company on the side. He/she does not want to take too much risks, as this is more of a hobby than a business, he/she enjoys having a small company and wants it to last. But after some years it all becomes unchallenging and tedious, so he/she wishes for expansion and then Easy Going player becomes the Strategic player. The company is growing, but he/she still has one eye on the money, because in the end it is his/her money invested in the company.

It was then decided that the Strategic player should not be aggressive, but be conservative in risk taking and also slowly move towards the game goals. After constructing its personality, the construction of the decision tree was rather self-evident (Figure 4).

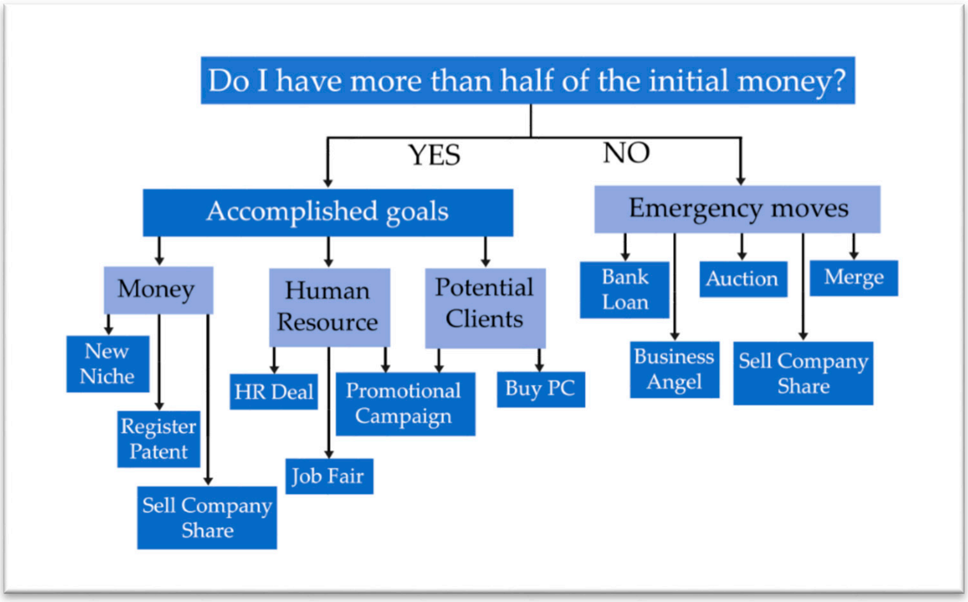

Figure 4. Strategic player's decision tree.

The first thing that the Strategic player checks every turn is its bank account status. If it is in danger of bankruptcy, it chooses a move from the category Emergency moves (Figure 5), that brings instant money and provide survival (*Selling Company Share*, *Auctioning PC Information*, *Bank Loan* or *Merge with Another Company*). In case its financial position is secure, it looks for expansion and growth by investing in the area that is lagging behind (PC, HR or money).

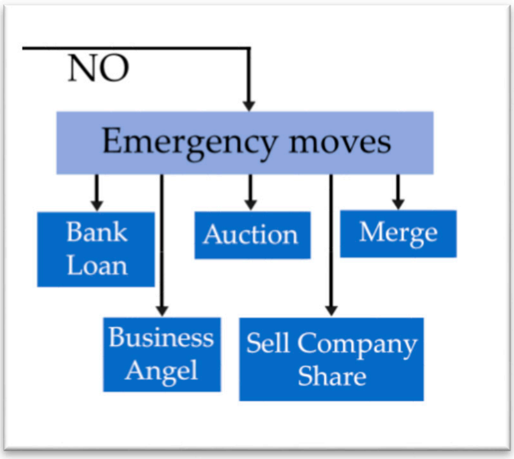

Figure 5. Emergency moves.

Since the essence of the Easy Going player is present in the Strategic player, it does not use moves from the Attack deals and does not buy out other companies; it is like your friendly neighbor that cuts his grass every Sunday and by doing that makes you cut yours so that your yard and house does not look neglected in comparison to his. This is a healthy competitiveness in which both sides win.

3.2.1. Implementation

In the beginning of each turn, the Strategic player checks whether it still has at least half of the initial money. If not, it randomly chooses an Emergency move (*Call Business Angel*, *Get a Loan*, *Auction Information about a PC*, *Sell Company Share* or *Merge*). Every move has the same probability of being picked, except *Merge* (50% less probable of being picked).

Every move has its limitations: for instance, *Calling a Business Angel* (and selling 25% of the equity), is only possible when owning 75 % of company equity, otherwise the player would no longer own the majority of shares and would not be in charge (therefore losing the game). The same goes for *Sell Company Share* (sell 10 % of your company) which can only be used when owning at least 60% of the company equity. A player can only *Auction Information about a PC* if he/she has at least one and he can only *Take a Loan* if there isn't another loan already outstanding. So, in case the Strategic player cannot perform a move, it simply uses the next one from the Emergency moves (Figure 5). If it cannot make any of the four moves, it proposes a *Merge* to the opponent player. The play is more interesting with more players, so the *Merge* is the last option.

Most of the time the Strategic player will have more than half of the initial money. In that case it will check the percentage of accomplished goals for each of its categories (HR, PC and money) and pick the one that is the least accomplished (Figure 6).

Figure 6. Accomplished goals-decisions.

In case the least accomplished goal is the number of its PC, it picks between *Promotional Campaign* and *Buying PC Information* from another player. *Promotional Campaign* gets the player PCs and HRs in exchange for money and can be used only twice. If all the promotional campaigns have already been used, the Strategic player then tries to buy PC information from another player (Figure 6).

If the least accomplished goal is the number of HR, it picks between three moves (*Promotional Campaign, Job Fair* and *HR Deal*). *Job Fair* brings the player HRs in exchange for money and *HR Deal* allows the player to contract an HR from another player. *Promotional Campaign* and *Job Fair* have limited amounts of usage, so in case they cannot be used, an *HR Deal* with another player is performed (Figure 6).

In case the least accomplished goal is the amount of Strategic player's money, it randomly chooses one of three moves (*Register a Patent*, *Invest in a New Niche* or *Sell Company Share*). If the player *Invests in a New Niche* market, he/she pays 100 € every turn for five turns and on the sixth he/she gets 1000 €. By *Registering a Patent* the player pays 500 € for the registration but then gets 200 € for each of the next five turns (as royalties). As a player cannot focus on two *New Niches* or two *Registered Patents* at the same time, these two moves are not always possible. In that case the *Selling of Company Share* is performed (Figure 6).

3.2.2. Testing the "Strategic" Player

The new Strategic player performed very well in the first tests. The tests were made with the same parameters as the initial tests (5 tests with 3 players for 10 rounds). The average company value was approx. 10,300 €, it achieved 89 % of money goals, 117 % HR goals and 82 % of PC goals. The statistics were really good, as the race to the end tile became much more interesting (the possibility of the human player defeat was much more likely now). The Strategic player did not just prevent bankruptcy, but actually had a rather good chance of winning.

However, even though the results were excellent, one problem arose: in the start of the game each player has 2500 €, 1 HR and 0 PC, which means that the first move is quite predictable: the player has more than half of the initial money (of course), so he checks at his accomplished goals. The least accomplished goal in the beginning of the game would always be PC, so there were two only possible moves in the first round (Figure 6). That made the initial moves of the game quite obvious particularly when playing against three Strategic players. The solution was implementing structured randomness: The Strategic player will make X number of its moves random, just like with the Easy going player, except it will only pick between the moves that it uses during the game (no attack moves or buy out). The question that arose was, how many rounds should be random, to keep as much strength as possible, but increasing the diversity of the moves at the same time. The results were quite interesting as can be seen in Figure 7.

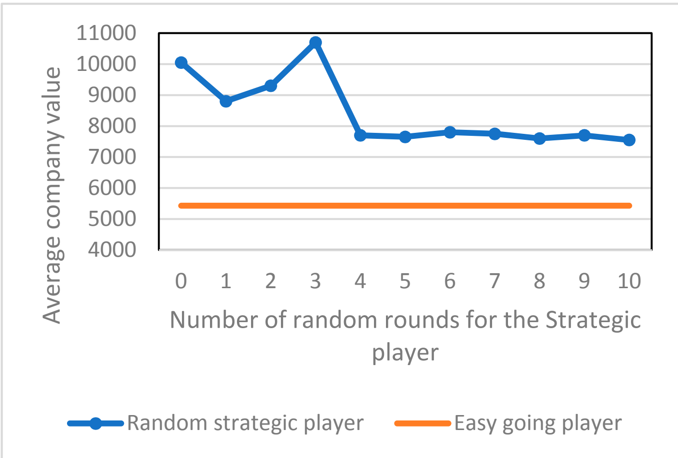

Figure 7. Average company value of Strategic player.

By making the first round random, the company value of the Strategic player dropped for approx. 15%. The company value rose with each added random round and reached its peak at 3 rounds. It uses 33% more moves than the regular Strategic player, while its company value is almost the same (which was quite unexpected). The expectation was that with each added random round the value will drop. As it turned out the player experiences more Emergency moves in the regular Strategic gameplay, because all his first moves consume a lot of money. If the player wants to increase the number of his PC, it must do a *Promotional Campaign*, which costs 1000 €, so if the player does two campaigns it'll be in the "Emergency zone".

With three initial random rounds, the moves are more diverse, so the players *Take out Loans*, *Sell Company Share*, *Contract HR* etc. Interestingly enough this has its limits, and it is exactly three. By implementing another random round (4 random rounds at the start of the game) the average company value dropped significantly (Figure 7). With four random rounds things become too unpredictable and the Strategic player is left to correct the mistakes throughout the entire game and fails to reach the full potential.

With even more random rounds (4+) the company value doesn't drop significantly, as the chaos is already present since the fourth round, but the company value becomes less consistent. For instance, with 10 random rounds in the beginning, the Strategic player scored as high as 9250 and as low as 6300, but the average is still similar to the other 4+ random rounds gameplays (Figure 4). The more random the choices the more random the outcome.

The 10-random rounds Strategic player is similar to the Easygoing player. Both pick random moves, with the exception that 10-random rounds Strategic player randomly picks its moves out of the set of moves that have been deemed as useful. Therefore, it is no wonder that he scores almost 50% higher than the Easygoing player. Since they are both random, the accomplishment of goals doesn't improve significantly.

3.3. The "Monopolist" Player

The goal of the "Monopolist" player was to keep the player in the Csikszentmihalyi 's "Flow zone" [16], where the player's skills correlate with the difficulty of the challenge. The player should be as difficult to play with as possible, but keeping the human player away from the too-difficult or too-easy zone, thus ensuring that his/her entertainment is at optimal level. By setting the challenge too high, the player would fall out of the "Flow Zone" and the game would prove to be too difficult for him/her to continue. Similarly, we didn't want to keep the challenge too low, as the game would not be interesting. As player's skills increase so should the difficulty of the challenges that are presented to him/her. In order to adjust the difficulty, the game must know when the player leaves the "Flow zone", so that it can adjust the challenge. The game must keep track of the player's progress and establish the right time to change the difficulty.

3.3.1. Implementation

The idea was to develop a grand competitor that keeps track of the human player's moves and applies as much pressure (with Attack deals) as the human player can take. As opposed to the Strategic player this new "Monopolist player" would keep track of the player stats and attack him, if it the game is going too well for him/her, otherwise it just focuses on its own strategy. Transform@ revolves around the decision making of the player, so we cannot make the game easier for the human player, as we cannot help him/her with the decisions or give him boosts to be used against the opponent player. The DDA thus ensures that the game is meant for a great spectrum of stronger players eager for a challenge.

Like with the creation of the Strategic player, the first thing was to define the personality of the Monopolist player. The Monopolist player should be very aggressive, willing to take high risks and ensuring that it was on top. In real life a Monopolist player would be a very big company, that wants to control the whole market and doesn't like competition, so they try to eliminate them. They are always looking for chances to make money, by constantly selling and buying their shares depending on the market value. They take high risks, because in the worst-case scenario they will just take another loan.

The Monopolist's behavior is more complicated than the Strategic player's so its strategy is divided in 3 stages (Table 1).

Table 1. Monopolist player stages.

STAGE 1	STAGE 2	STAGE 3
Gathering wealth	Manipulating the market	Attacking the human player

STAGE 1: For the first 4 rounds or until its money is lower than 1200 €, the Monopolist player makes moves that will increase the company value as it needs it to be very high in order to effectively sell his shares in stage 2. By the end of the first stage it *Takes a Loan*, which additionally increases the company value (Figure 8).

STAGE 2: It lasts until the seventh round, consisting of *Selling and Buying Company Shares* that simultaneously provide the Monopolist player big quantities of money during the process. If it is not a profitable time to buy or sell a share, it will instead increase its lowest accomplished goal (PC, money or HR), similar to the Strategic player's decision tree (Figure 8), but with added Attack deals (Figure 9).

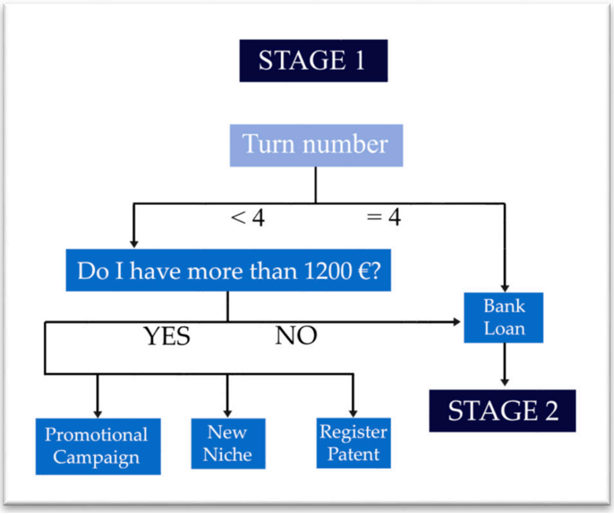

Figure 8. Stage one.

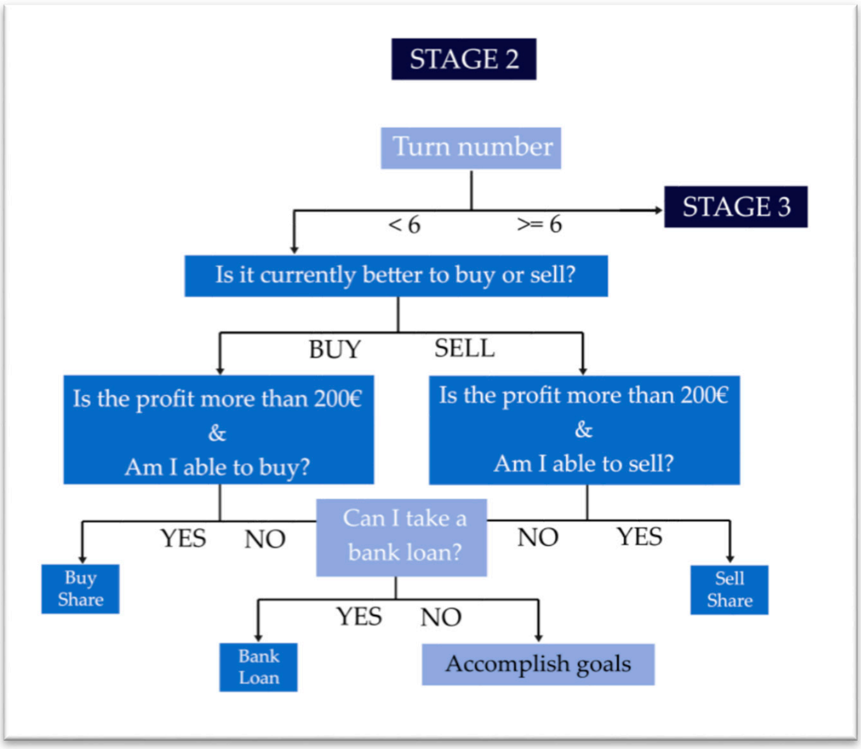

Figure 9. Stage two.

STAGE 3: In round seven it enters the third stage, which emphasizes the surveillance of the user player. The better the human player's stats, the higher the chance of an Attack from the Monopolist player. The form of the attack depends on the player stats. The Monopolist player will always attack the highest goal or attack both (*Negative Campaign*), if they are both accomplished (Figure 10).

If in a particular round the Monopolist player can't attack the user player, it regulates the number of his PCs and HRs, either increasing or lowering them. In case it has completed all three goals, it will lower the highest one by making a move that will benefit the other goals. For example: if it has too many PCs, but barely enough money to win, it will sell two PCs and increase its money while still keeping the number of PCs according to the preset goals. If it has a lot of money, but not enough PCs and HRs, it will make a move from the decision tree (Figure 6). However, if neither of these apply, it will look for money opportunities exactly like in Stage 2 (Figure 9).

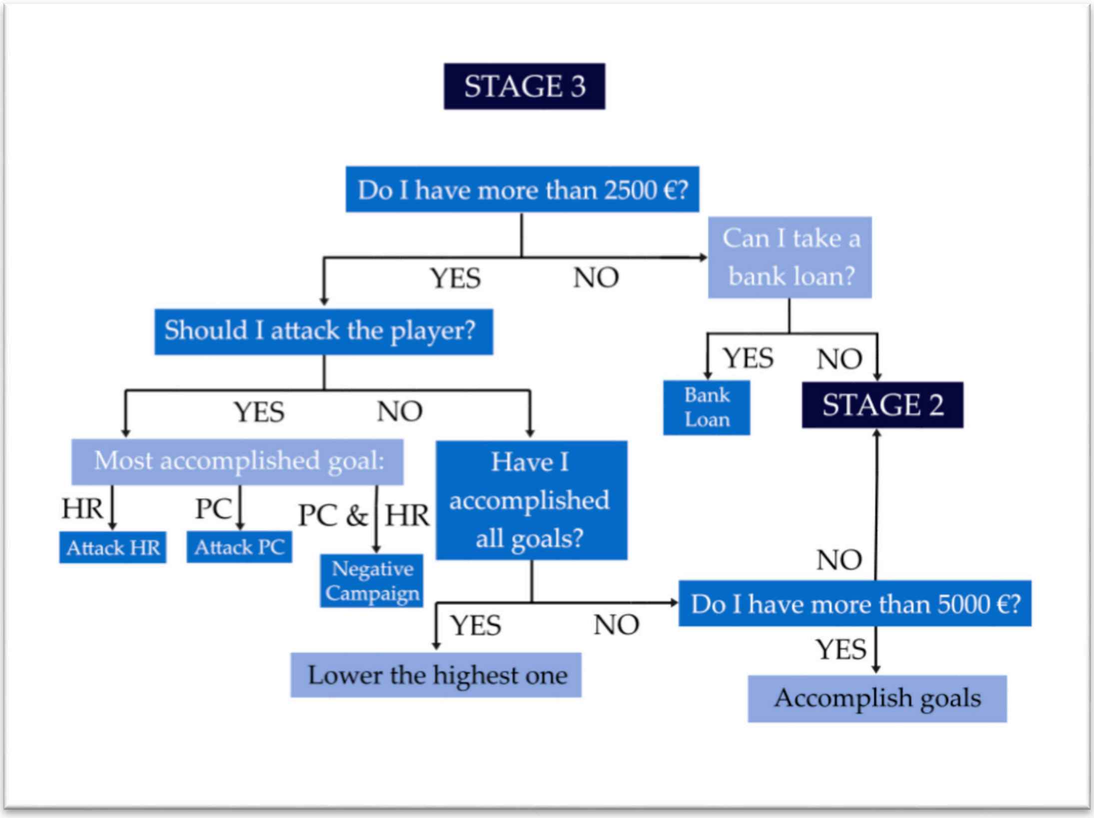

Figure 10. Stage three.

3.3.2. Testing the "Monopolist" player

The Monopolist player was tested and improved many times, as it could quickly lose money because of its high risk taking and then it would just wander till the end of the game without money and possible moves. Besides that, the Monopolist player needed to change the behavior at the moment it had enough money to actually finish the game. It was hard to determine how often and how successfully the monopolist player should attack the human player, because of two things:

- Attacks are expensive: too many failed attacks can mean bankruptcy for the Monopolist player.
- Player enjoyment is dependent of success of attacks: too many successful attacks would discourage the human player and make the game too frustrating to play, but too low would make attacks insignificant.

The goal was to find the "Flow zone" so that the game will not be too frustrating due to constant attacks, but that it also wouldn't be boring, as the human player watches the rivals lose all of their money on unsuccessful attacks. The possibility of the attack depends on the user player's accomplished goals. The percentage of accomplished goals is equal to the chance of player being attacked in a turn regardless of the number of Monopolist players. If the player reaches 100% of his goals and there are 3 Monopolist players, the user player will be attacked only once per turn, as every monopolist player has 33% chance of attacking him, or 50% if there are two Monopolist players.

The final tests were made with the same parameters as the initial tests (5 tests with 3 players for 10 rounds). The human player did not make any moves to eliminate the possibility of the Monopolist players attacking him, as the point was to only test its strength.

The company value of the Monopolist player was consistently rising and by round 6 it almost accomplished enough goals to finish the game (money: 104%, PC: 89% and HR: 103%) (Figure 11). After the sixth round the value dropped a little due to buying back shares of its company. In this phase it would otherwise attack the human player and its company value would depend on the success of the attacks.

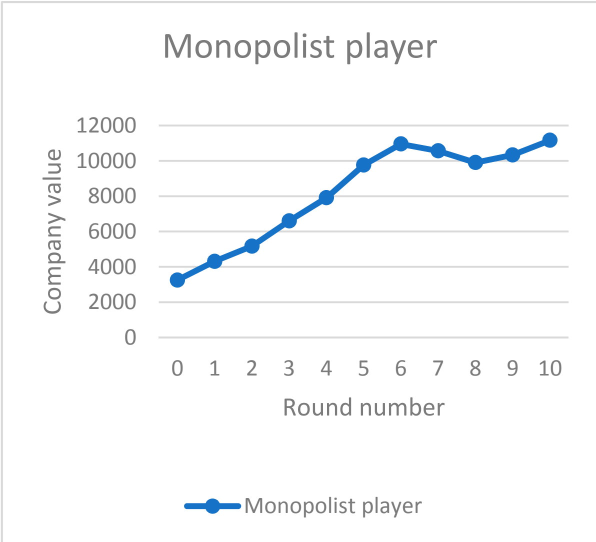

Figure 11. Average company value of Monopolist player.

Besides the strength test, it was necessary to test the success of the attacks. Because the attacks are dependent on the accomplished goals, a custom resource setting was established so that all the players had an abundance of resources, and thus made sure that accomplished goals were always above 100%.

The test was done for 50 rounds. On average, the Monopolist player's attacks were 32% successful (Figure 12). If the human player reaches his/her goals, he/she would on average get attacked once per round (regardless of the number of the players), and successfully every third round.

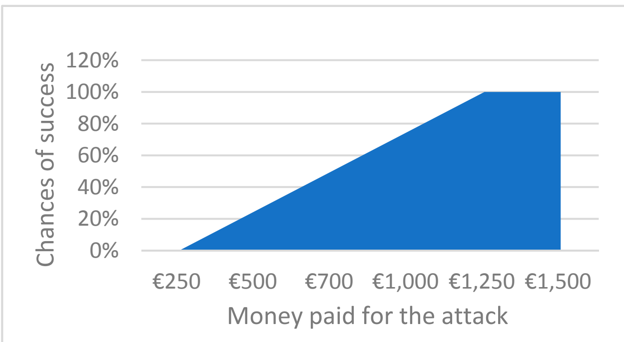

Figure 12. Success of Monopolist attacks.

When the Monopolist player attacks, it deposits a random amount of money for the attack, ranging from 500 to 1500 €. His chances of successful attack are dependent on the amount of money it puts in (Figure 13). The higher the amount, the higher the probability of success.

Figure 13. Chances of successful attack.

4. Discussion

As a final test, a series of games were played with CC players of different levels (Strategic, Monopolist, Tycoon). The company values of each level differ from the values gathered in individual tests, as the interaction between different difficulty levels differs from the interaction of the singular difficulties. Both Monopolist and Strategic player had great average company values throughout the whole game (Figure 14). The improvement of the Tycoon player is insignificant, the Monopolist player was about 4 times stronger after 10 rounds and the Strategic player about 3 times. The Tycoon player did not really had a chance of winning as its company value stays low throughout the whole game. The goal of developing better competitors was succesfully achieved and resulted in strong competitors that keep the player engaged throughout the whole game.

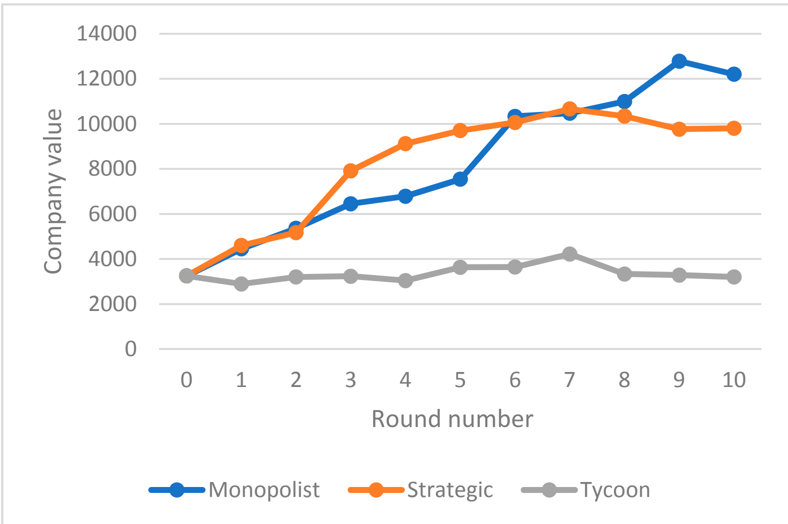

Figure 14. Comparison between different difficulty levels.

5. Conclusions

When balanced, competition drives humans to improve their skills and abilities. In games, player engagement is partly achieved by delivering competition with the right amount of difficulty. The use of Dynamic Difficulty Adjustment techniques allows the game to, dynamically and in real time, adjust the challenge level to the player performance therefore keeping him/her always immersed and active. In Serious Games, this makes the user more prone to acquire new information, develop new skills and improve existing abilities.

In this paper we described how DDA techniques were used to create computer-controlled (CC) players in order to deliver a higher level of competitiveness for the user in Transform@, a Serious Game meant at developing entrepreneurship skills. As a result, the strength of the computer controlled player increased by more than 100%, which rendered challenges much more motivating for the human players, who were now engaged for longer periods. Learning became fun and unconscious due to high level of entertainment which is the basic goal of any serious game. However, although the specificity of the DDA approaches that must be matched to each individual game does not allow us to extrapolate conclusions to the entire domain, this was certainly a case in favor of the use of these techniques in Serious Games.

Author Contributions: Conceptualization, C.V.d.C.; Data curation, D.K., P.B. and R.C.; Formal analysis, D.K.; Investigation, D.K.; Methodology, D.K.; Software, P.B. and R.C.; Supervision, C.V.d.C.; Writing—original draft, D.K.; Writing—review & editing, C.V.d.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Juzeleniene, S.; Mikelioniene, P.; Escudeiro, P.; Vaz de Carvalho, C. Gaball Project: Serious games based language learning. *Procedia Soc. Behav. Sci.* **2014**, *136*, 350–354. [CrossRef]
2. Bermúdez, M.R.; Rodríguez, M.C.; Nistal, M.L.; Carvalho, C.V.; Nogueira, F. eCity: Virtual city environment for engineering problem based learning. In Proceedings of the 2015 IEEE Global Engineering Education Conference (Educon), Tallinn, Estonia, 18–20 March 2015; pp. 159–166.
3. Clark, C. *Serious Games*, 2nd ed.; Viking Press: New York, NY, USA, 1970; p. 196.
4. Gamasutra. Available online: http://www.gamasutra.com/view/feature/129948/the_chemistry_of_game_design.php (accessed on 20 August 2019).
5. Andersen, G.; Ahdell, R. Games Simulation in Corporate e-Learning. Master's Thesis, University of Illinois, Illinois, IL, USA, 2002.
6. Dondlinger, M.J. Educational video game design: A review of the literature. *J. Appl. Educ. Technol.* **2007**, *4*, 21–31.
7. Vorderer, P.; Hartmann, T.; Klimmt, C. Explaining the enjoyment of playing video games: The role of competition. In Proceedings of the Second International Conference on Entertainment Computing, Carnegie Mellon University, Pittsburgh, PA, USA, 8–10 May 2003; pp. 2–5.
8. Vandercruysse, S.; Vandewaetere, M.; Cornillie, F.; Clarebout, G. Competition and students' perceptions in a game-based language learning environment. *Educ. Technol. Res. Dev.* **2013**, *61*, 927–950. [CrossRef]
9. Cheng, H.N.; Wu, W.; Liao, C.C.; Chan, T.W. Equal opportunity tactic: Redesigning and applying competition games in classrooms. *Comput. Educ.* **2009**, *53*, 866–876. [CrossRef]
10. Malone, T.W.; Lepper, M.R. Making learning fun: A taxonomy of intrinsic motivations for learning. *Aptit. Learn. Instr.* **1987**, *3*, 223–253.
11. Nemerow, L.G. Do classroom games improve motivation and learning? *Teach. Chang.* **1996**, *3*, 356–366.
12. Atanasijevic-Kunc, M.; Logar, V.; Karba, R.; Papic, M.; Kos, A. Remote multivariable control design using a competition game. *IEEE Trans. Educ.* **2011**, *54*, 97–103. [CrossRef]
13. Kohn, A. *No Contest: The Case against Competition*; Houghton Mifflin: New York, NY, USA, 1992.
14. Malone, T.W. Toward a theory of intrinsically motivating instruction. *Cognit. Sci.* **1981**, *4*, 333–369. [CrossRef]
15. Csikszentmihalyi, M. *Flow: The Psychology of Optimal Experience*, 1st ed.; Harper & Row: New York, NY, USA, 1990; p. 303.
16. Zohaib, M. Dynamic difficulty adjustment (DDA) in computer games: A review. *Adv. Hum. Comput. Interact.* **2018**, *2018*, 5681652. [CrossRef]
17. Silva, M.P.; Silva, V.D.N.; Chaimowicz, L. Dynamic Difficulty adjustment through an adaptive AI. In Proceedings of the 2015 14th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames), Piauí, Brazil, 11–13 November 2015; pp. 173–182.
18. Lora, D.; Sanchez-Ruiz, A.A.; Gonzalez-Calero, P.A.; Gomez-Martın, M.A. Dynamic Difficulty Adjustment in Tetris. *Fla. Artif. Intell. Res. Soc. Conf.* **2016**, *12824*, 335–339.

19. Hunicke, R.; Champan, V. *AI for Dynamic Difficulty Adjustment in Games*; University of California: Berkeley, CA, USA, 2004.
20. Missura, O. Dynamic Difficulty Adjustment. Ph.D. Thesis, University Bonn, Bonn, Germany, January 2015.
21. Araújo, V.; Mnedez, D.; Novel, A.; González, A. Approach to working memory training based on robotics and AI. *Information* **2019**, *10*, 350. [CrossRef]
22. Vaz de Carvalho, C.; Costa, R.; Bessa, P.; Monterrubio, L.; Santos, J. Developing entrepreneurship skills with a serious game. In *HCI in Games*; Fang, X., Ed.; LNCS: Berlin, Germany, 2019; Volume 11595, pp. 351–363.