Maximizing Player Engagement in a Global Warming Sensitization Video Game Through Reinforcement Learning

Maxime Boudreault
Université du Québec à Chicoutimi
Chicoutimi, Québec, Canada
maxime.boudreault@uqac.ca

Bruno Bouchard
Université du Québec à Chicoutimi, LIARA
Chicoutimi, Québec, Canada
bruno.bouchard@uqac.ca

Kévin Bouchard
Université du Québec à Chicoutimi, LIARA
Chicoutimi, Québec, Canada
kevin.bouchard@uqac.ca

Sébastien Gaboury
Université du Québec à Chicoutimi, LIARA
Chicoutimi, Québec, Canada
sebastien.gaboury@uqac.ca

ABSTRACT

Global warming's consequences must be known and humanity has to take action. In spite of the efforts already taken toward that goal, the challenge remains. Of all media used to achieve that goal, video games are not as used as their increase in popularity may suggest; even when they are, their entertainment value is too low to raise interest. In this paper, we present a serious game called "Penguin Panic!", specifically developed to increase sensitization about climate change. In this game, a Dynamic Difficulty Adjustment (DDA) system is used to increase the player's interest by providing him or her with a flow-friendly game world. Using Reinforcement Learning, this DDA system adjusts difficulty in real-time based on the player's skills.

CCS CONCEPTS

• Computing methodologies → Machine learning; **• Applied computing → Computer games**; *Education*;

ACM Reference Format:

Maxime Boudreault, Bruno Bouchard, Kévin Bouchard, and Sébastien Gaboury. 2018. Maximizing Player Engagement in a Global Warming Sensitization Video Game Through Reinforcement Learning. In *International Conference on Smart Objects and Technologies for Social Good (Goodtechs '18), November 28–30, 2018, Bologna, Italy*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3284869.3284920

1 INTRODUCTION

Global warming certainly constitutes one of the biggest challenges humanity has ever faced [14]. Its consequences are real, if not alarming; sea-ice extent has significantly decreased from 1978 to 2013 and species are threatened to go extinct. Humans must acknowledge the problem, take responsibility, and adapt their behaviour accordingly. This is why a lot of private [2] and public [12] initiatives have been taken to raise people's awareness about the importance of climate change. However, despite these efforts, the problem of making individuals more aware and proactive about the situation remains a significant challenge [7].

In the last decade, campaigns have been conducted in different Medias promoting an evolution of our societal behaviour concerning climate change [12]. Nevertheless, it seems that the media that constitutes video games, despite its growing popularity [15], has been really underexploited for that purpose. More specifically, Serious Games [3], a new type of digital games specializing in other purposes than just entertainment — such as education, leading societal impact on specific subjects, enhancing individual user's aptitudes and more — seem to be really well suited for the task. Although few serious games cover climate change (educational and advertising games alone account for 57% of serious games [9]), most are text-heavy and unappealing; something important to avoid in order to keep a player motivated [4]. Moreover, to the best of our knowledge, none of these games included new kinds of game mechanics that help adapting the challenge to the player's abilities. Such mechanisms, which are called Dynamic Difficulty Adjustment (DDA) systems [17], aim to maximize the player's engagement, something not yet done in climate change video games. However, to be effective at promoting efficiently the consequences of climate change, the player needs to play for a sufficient amount of time and to be engaged deep enough in the game to feel concerned [1]. Therefore, to maximize the effectiveness of a serious game dedicated to education about climate change, it is necessary to incorporate an effective and well-integrated DDA system. In summary, it consists in developing a model of the player's skills, which is adjusted according to the performance observed in the game session (the player produces data when playing the game such as health, velocity, game states, etc.), and to define an algorithm that determines what kind of changes should be made in the game's variables in order to enhance the player's experience (e.g. increasing the time the player has) [17].

In this paper, we present a new serious game developed at our university for promoting awareness toward climate change, called "*Penguin Panic!*", which includes a new adapted system of DDA based on reinforcement learning algorithms [16] used in artificial intelligence. That game won a prestigious award entitled "Best

Goodtechs '18, November 28–30, 2018, Bologna, Italy

ACM ISBN 978-1-4503-6581-9/18/11...$15.00
https://doi.org/10.1145/3284869.3284920

Innovation and Technical Challenge" at Ubisoft's Game Labs competition of 2018[1]. More specifically, we present in Section 2 an overview of the existing works on related serious games and on dynamic difficulty adjustment. In Section 3, we present the game design and implementation (with Unity) of the game and how the game mechanics are used for promoting awareness about climate change. Section 4 presents the core of DDA system and its algorithm. Section 5 presents our actual validation efforts and the upcoming experiments with the proposed experimental protocol. Finally, Section 6 concludes the paper and discusses about future works.

2 RELATED WORKS

2.1 Serious Games

For the purpose of this paper, we identified three categories based on the serious objectives served by serious games: education, physical exercise and social change.

2.1.1 Education. Known as *educational games*, those games mainly aim for the acquisition of academic knowledge by its players. Websites such as Funbrain[2] are common and give access to many educational games. Others, such as Khan Academy[3] offer a gamified learning environment, with quizzes, videos and more. Keeping players motivated — or "tricking" them into learning — has a direct influence on the efficiency of those games. This makes methods such as DDA appealing, as a player facing a challenge up to his or her level will learn better [6]. DDA has been known to affect motivation positively when playing educational games [13]. These games do a great job at teaching, but fail at having an impact on society's current challenges.

2.1.2 Physical Activity. Games from this category are usually called *exergames*, a portmanteau made up from the words *"exercise"* and *"game"*. Some of those, such as the one in [8], will target the population at large as they try to encourage people to adopt an healthier lifestyle. Others will have a more specific target audience, like people suffering from a mental illness [5] or people doing physical therapy exercises [10]. Since keeping players motivated is central to the success of such games, approaches such as the one used in [8] use DDA for the same reasons mentioned earlier. These games can have an indirect impact on pollution — when they encourage the use of bicycles or public transportation, for example — but that's about it. They are usually centered on the player as an individual rather than on the context he or she lives in.

2.1.3 Social Change. The games in this category are those that want to have a positive social impact. *Penguin Panic!* inserts itself in this category as an ecogame (serious game about ecology). A wide range of social issues are tackled by games for social change, including immigration[4] and poverty[5] among others. To our knowledge, no games for social change make use of DDA. This is why it has been used in *Penguin Panic!*. If DDA can have a significant positive impact, as seen in other categories, then it should.

[1]https://montreal.ubisoft.com/en/ubisoft-game-lab-competition-2018-winners/
[2]https://www.funbrain.com/
[3]https://www.khanacademy.org/
[4]http://papersplea.se/
[5]http://playspent.org

2.2 Dynamic Difficulty Adjustment

As computational power and data volume both increase, it becomes more and more interesting to use types of artificial intelligence that can directly benefit from those. Reinforcement Learning fits that profile.

DDA can be done in many different ways [11]. Its core concept stays the same: adjusting the game's difficulty to the player's skills so that it is never too boring nor too frustrating [18]. The cycle shown in Figure 1 shows how DDA iterates, as can be understood from [11].

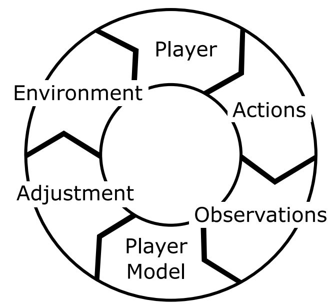

Figure 1: Dynamic difficulty adjustment cycle

Each element of this cycle affects the next one. As the game environment presents challenges and goals to the player, decisions on which actions to take ensue. Choices made by the player, as well as quality of execution, are observed by the DDA system. From those observations, we derive a player model, which in turn is used to adapt challenge accordingly to the player so it never becomes too hard or too easy. A player model is a simplified representation of the player currently playing. It is made by extrapolating data gathered from observing the player's in-game behaviour and represents our comprehension of the player based on that data. It is important to have an accurate player model as this is what our DDA system will adjust to.

3 *PENGUIN PANIC!*

3.1 Creation Context

Using the Unity game engine[6], we developed "*Penguin Panic!*", an online cooperative two-player game that won the "Best Innovation and Technical Challenge" at Ubisoft's Game Labs competition in 2018. It was also nominated in the following categories: 3C (Character, Camera and Controls) and jury's prize. The competition's theme being "Change the world", we chose to weave our game around global warming; the game hence educates in a soft and fun way about the consequences of global warming.

3.2 Game Rules

Penguin Panic! is set at the North Pole; scientists are collecting data and, seeing that the North Pole detached itself from its glacier and is now missing, try to find solutions. To their surprise, penguins are pushing the North Pole towards the north to put it back in its rightful place. The scientists decide to film the penguins' efforts and stream them on Twatch.tv to gather funds to help the fight against global warming.

[6]https://unity3d.com/

Figure 2: Penguins reaching for their goal.

The game's mechanics are reminiscent of Super Mario 64 and Pikmin. Each player controls a penguin leader in a three dimensional environment where water and ice dominate. The goal of the game is to bring back an iceberg — clearly indicated by a "North Pole" sign, as seen in Figure 2 — to the north — the direction the sign is pointing — by pushing it through water. Although the goal in itself is linear, the player reaches multiple intersections throughout the game and can choose which way to go. The game support XBox controller and can be beaten by using the left thumbstick to walk or swim and the A button to jump. The use of other features is greatly encouraged, both to complete subquests and enrich one's experience of the game. Thus, a penguin can also slide on its belly; which makes its horizontal movements more physics based, but greatly increases speeds and halves the character's height. By sliding, a penguin can gain speed to jump further or access secret areas.

A penguin leader — controlled by a player — can recruit penguins to be part of his or her colony. Those recruits serve as both torpedoes and "hit points" (both described further). The full recruitment process is done in three steps:

(1) Eat a fish school by swimming in it and pressing the B button. This will make an egg appear on the North Pole iceberg.
(2) Jump on the the North Pole iceberg and reach for the egg, which will make it pop after a second, revealing a lone penguin.
(3) Move next to the penguin to recruit it as part of the leader's colony.

It is possible to encounter eggs and lone penguins in the wild, making it possible to skip steps. Recruits will follow their leader everywhere and will be teleported back to him or her if they ever are too far away. The player can shoot a recruit as a torpedo. When such a thing happens, the penguin torpedo quits the colony and is propelled forward at great speed. If a valid target appears in its range of sight, it will harshly change its direction to go towards the target. Valid targets include killer whales and cracked ice, both described further. After a time, if no target is encountered, the torpedo penguin reverts back to a neutral state where it ceases to pursue any target.

Apart from penguins, the game includes a number of gameplay features. Among the most important ones, the North Pole iceberg. This iceberg slowly melts during the game. An indicator shows its integrity to the player. Using his leader and colony, a player can find ice chunks and bring them back to the iceberg by pushing them, giving the iceberg some of its integrity back. Failure to maintain integrity over zero ends the game; the players are considered to have failed. Success in bringing the iceberg to its rightful place, the actual North Pole, results in a victory for both players.

The next most important feature is the presence of killer whales throughout the game world, as can be seen in Figure 3. Killer whales wander around their own scouting center, moving exclusively in water. As soon as a penguin enters their field of view, the orca will chase after it until it either leaves the orca's field of view or the orca is scared off. An orca's scouting center is automatically updated to the last position it saw a penguin. Another important note: the killer whale always targets a leader's followers first; that's how recruits are considered to be "hit points" in a way. If a leader has no follower, it might be targeted and eaten. When eaten, a leader will respawn on the North Pole iceberg after a short delay. To avoid an orca's attack, a player can shoot a penguin as a torpedo towards it. If the orca is hit, it will flee for a while. A player who wants to avoid any losses may instead find another route, if any, devoid of predators. Figure 3 shows the aforementioned dilemma, as the player stays safe on the iceberg while pondering on how to deal with the orcas.

Figure 3: A player facing two dangerous orcas, pondering on what to do next.

Figure 4: A player trying to reach a golden snowflake from atop a snowball he or she just rolled.

Other features include the following:

- cracked ice, which can be destroyed with a penguin tornado or otherwise may block the players' progression;
- snowballs, which get bigger the more they are rolled around, as seen in Figure 4;
- adrift polar bear cubs, which can be rescued by bringing them back to the North Pole iceberg;

- hidden golden snowflakes that give much points upon collection, as the one seen in Figure 4;
- and button activated doors, which require both players to place their leader on two button platforms related to the door to open it.

3.3 HUD related features

Figure 5: Screenshot processed to highlight and number different parts of the HUD: The quest log (1), the compass (2), the scores (viewer count and money raised) (3) and the iceberg integrity indicator (4).

The head-up display's (HUD) main features, presented in Figure 5, are the compass, the quest log and the iceberg integrity indicator. The compass indicates in which direction are the goal, the North Pole iceberg and the polar bears. The iceberg integrity indicator, as previously mentioned, indicates how much of the North Pole iceberg is left before it completely melts away. Players' score is also shown, although it has no direct impact on gameplay.

3.4 Where global warming fits in

The game attempts to educate mainly through its gameplay; the detached North Pole sign, melting ice and the possibility to save endangered polar bears are all examples of such educational gameplay elements. In a subtle, but comical way, the game can make one think about the fact that penguins will not save us from global warming, that such a responsibility belongs to humanity.

3.5 Dynamic Adjustments

A very basic ad hoc dynamic adjustment system was part of the first design of the game for the purpose of validating that the game architecture supported it. That system was based on the dynamic between players and not their skills. It affected spawn rates of game entities as well as iceberg integrity loss rate and ice chunks' healing power. It was not meant to accommodate the players' skills, but rather the way they played. A more collaborative duo's iceberg would melt quicker, forcing them to search for ice. More fishes would spawn, so one could hunt while the other protected the iceberg. On the opposite, a more exploration based or competitive duo's concern would be less about healing their iceberg and more about fending off orcas. They would find more penguins and eggs than fishes. This part was removed from the game and replaced by a new DDA system.

4 ADJUSTING DIFFICULTY

Reinforcement Learning [16] (RL) is used as the core of our dynamic difficulty adjustment algorithm, which inserts itself in the cycle seen in Figure 1. RL is a type of machine learning. It selects an action based on its expected reward from the current state to maximize rewards. It can be implemented in such a way that can either prefer short-term rewards or long-term ones. Also, a balance must be kept between reward maximization and exploration to take into account variations in rewards.

Actions that the player can take are those previously described. We don't observe the actions directly but rather their effect on the game environment and whether or not it had a positive outcome for the player. For example, failure to hit target when launching a torpedo results in a loss of one penguin for the player, which is negative. On the other hand, hitting breakable ice with that same torpedo would result in the loss of a penguin, but also being one step closer to reaching a goal, which is a positive outcome. Through the observation of variables, we interpret the following skills:

- danger management (ability to avoid or fend off orcas);
- progression focus (progress towards quests completion and gathering golden snowflakes).

Skill observation is refreshed at each AI episode, an episode being a period of time during which data is gathered without further processing. The end of an episode is reached upon accumulating a combined amount of "Win" and "Fail" points. This amount is arbitrary; while a smaller amount allows for quicker iterations (and learning) of our RL algorithm, a larger one will bring to a better understanding of the player's skills in the long run. At the end of an episode, the "Win" points ratio is evaluated, resulting in a number between 0 and 1 inclusively. The player's performance is then interpreted from that ratio as a whole number ranging from 0 (worst) to 4 (best). Each value is associated with a specific range within the win ratio continuum. "Win" and "Fail" points are awarded when targeted events occur. Multiple instances of our DDA system can coexist, allowing for different aspects of the game to be adjusted independently.

Considering a player's skill may improve during a gameplay session, the player model has to be updated at runtime in a way that allows such progression. It also has to be stable enough to avoid unwanted difficulty spikes. To have complete stability of a variable, we would simply keep a mean and update it at each AI step. For complete consideration of progression, we would only take into account the value the variable had at the last AI episode. To find a value V at episode t, we use a discount rate γ ($0 < \gamma \leq 1$) exponentially so that the older an episode is, the less of an impact the variable's value v_{t-n} has on V_t:

$$V_t = \gamma\left[v_t + (1-\gamma)v_{t-1} + (1-\gamma)^2 v_{t-2} + \ldots + (1-\gamma)^{t-1} v_1\right] + (1-\gamma)^t v_0$$

Which is equivalent to $V_t = \gamma v_t + (1-\gamma)V_{t-1}$. Both win ratio and actions expectations follow this formula, thus allowing change over time without completely ignoring the past.

Two instances of our DDA system are used in *Penguin Panic!*, each for one aspect of the game. One of them, called "progression adjustment" observes the player's progression over time; how quickly the player completes quests and pushes the North Pole iceberg toward its final destination. In response to the player's performance,

Progression Adjustment					
Difficulty	**0**	**1**	**2**	**3**	**4**
Chunk distance	5-10	5-12.5	7.5-12.5	7.5-15	10-15
Chunk arc	0-90	30-90	45-120	60-150	60-180
Restoration power	23-32	20-27	17-23	14-20	12-17
Iceberg melting rate	0.25	0.29	0.33	0.38	0.44

Table 1: How progression difficulty affects iceberg.

Danger Adjustment					
Difficulty	**0**	**1**	**2**	**3**	**4**
Orca speed (meters / second)	8	9	11	12.5	15
Orca detection radius (meters)	4	4.5	5	6	7
Torpedo homing radius (meters)	7	6	5	4.5	4

Table 2: How danger difficulty affects orca encounters.

the system influences the melting rate of the iceberg, the spawning distance of ice chunks, the convenience of their placement and how much they restore the iceberg's integrity, such as seen in Table 1.

Table 2 shows the other instance of our DDA system, called "danger adjustment". This one influences encounters with orcas: orcas' speed, orcas' detection range and penguin torpedoes' homing feature. Observed events are recruitment of a penguin, loss of a penguin to an orca and player's accuracy when launching torpedoes. In both cases, instead of greatly influencing a single variable, we slightly change multiple of them; this makes changes more transparent to the player. The numbers contained within both tables were hand adjusted through short test sessions.

Algorithm 1 shows how our DDA system works and how it integrates with the game loop. From outside the system, within initialization logic, we read a config file to register Win and Fail points attribution on game events, so that when those events happen, points will be cumulated. Events include recruiting a penguin, hitting an orca with a torpedo, and so on. Also, since our system affects game entities, every adjustable entity in the game implements a common interface, allowing them to be registered in our DDA system upon creation. Their difficulty level is then adjusted whenever necessary (upon registration and at the beginning of every episode thereafter), changing the entity's statistic and/or behaviour as specified by its implementation of the interface.

The DDA system itself exposes both Win and Fail procedures and manages everything else. The "NextEpisode" procedure, called when an arbitrary threshold of Win and Fail points is reached, proceeds to adjust difficulty. First, it deduces the performance level from the win points ratio and applies a discount to both win and fail points (the discount should be a value lower than 1 and greater or equal to 0). Then, the last executed action is rewarded depending on the state reached. Each combination of a difficulty level and a performance level is a state. Since we have five of each, this means we have twenty-five states. The nearer the performance level is to the target, the higher the reward. Then, from the current state (the one with last's episode difficulty and our newly deduced performance level), the RL algorithm selects an action, whose execution may change the difficulty level, either by increasing or decreasing it by one. Finally, that difficulty level is applied to every registered entity.

Figure 6 shows an example of our RL algorithm's distribution of probabilities based on a player model. In the example, our player has been facing the highest level of difficulty (4) for a complete episode. His ratio of Win points is set at 0.75, a value considered to be within the range of level 1 performance, which is under our target performance of 2. Based on its knowledge, our RL algorithm considers that lowering the difficulty might bring player performance to level 2, but still considers the possibility that keeping the current difficulty level could have the desired outcome. Since the difficulty is already at its maximum, there are no chances for it to be raised. For a player whose skills are between two difficulty levels, the algorithm could learn that oscillating between those two levels is the best way to achieve optimal results.

Algorithm 1 Simplified dynamic difficulty adjustment loop

```
procedure Initialize
    Register Win(p) and Fail(p) on specific game events.
end procedure
procedure Update
    if an adjustable game entity is created, then
        register it to the DDA system.
    end if
end procedure
procedure Win(p)
    winPoints ← winPoints + p
    CheckForEpisodeEnd
end procedure
procedure Fail(p)
    failPoints ← failPoints + p
    CheckForEpisodeEnd
end procedure
procedure CheckForEpisodeEnd
    if winPoints + failPoints ≥ threshold, then
        NextEpisode
    end if
end procedure
procedure NextEpisode
    winRatio ← winPoints / (winPoints+failPoints)
    performanceLevel ← GetPerformanceLevel(winRatio)
    winPoints ← winPoints × discount
    failPoints ← failPoints × discount
    previousState.RewardLastAction(currentState.reward)
    adjustment ← currentState.SelectAndExecuteAction
    difficultyLevel ← difficultyLevel + adjustment
    for all entity ∈ registeredEntities do
        entity.difficulty ← difficultyLevel
    end for
end procedure
```

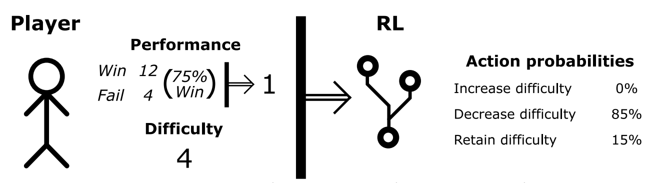

Figure 6: RL decision making example

5 VALIDATION AND FUTURE EXPERIMENTS

Being in the process of writing an application for ethical approval, no experimentation has yet been done. We plan on recruiting about 30 people for our experiment following the protocol below:

(1) The experimented takes five minutes to explain the setting, goals and main features of the game to the player.
(2) The player plays the game a first time until victory or defeat, coached by the experimenter.
(3) The player plays the game a second time, without coaching, until victory or defeat.
(4) The player fills out a survey.

We implemented a DDA system that acts in response to its observation of the player's skills. Humans being much more than the sum of their skills, we would be interested in broadening the nature of our observations in a future experiment and try to dig more into the player's psychology to base our dynamic adjustments on his or her gameplay and difficulty preferences by guessing those through observation of the player's behaviour in game. Such features should increase the player's engagement even further. The more engaging the game, the more meaningful will become improvements to its educational segment [6].

6 CONCLUSION

Climate change is a problem affecting us on a global scale. It can be hard to see the big picture from the right perspective when we are part of it. The population needs to adapt its behaviour to climate change and to adopt a more sustainable style of living. Over the years, many initiatives [2, 12], in different forms of media, tried to raise people's awareness about this important issue. Video games, and a more specific kind of game called Serious Games, have the potential of educating and leading societal impact [3]. In this paper, we presented a new serious game, called "*Penguin Panic!*", developed at our university and aiming to educate people about the problem of climate change. That game won a prestigious award entitled "Best Innovation and Technical Challenge" in Ubisoft's Game Labs competition of 2018. We pointed out that, to maximize the effectiveness of a serious game dedicated to education about climate change, it is necessary to incorporate an effective and well-integrated system of Dynamic Difficulty Adjustment to ensure the player's engagement in the game. Since the state of flow (i.e. the immersion in the game) helps in learning, providing a way to reach it facilitates transmission of a perspective on climate change's consequences. Therefore, we presented a new adapted system of DDA [17], based on a reinforcement learning algorithm [16], which has been implemented in the game. We detailed the game design and implementation (with Unity) and showed how the game mechanics are used for promoting awareness about climate change. Finally, we presented our actual validation efforts and upcoming experiments with the proposed experimental protocol.

Of course, this work is only the beginning and many more efforts are needed in the near future. A representative number of people, following our proposed protocol, must carefully test the game. That is our main priority. Only then will we be able to clearly evaluate the impact of the game and the efficiency of the DDA system.

7 ACKNOWLEDGMENTS

I wish to thank my colleagues, with whom I worked day and night for two months to accomplish the *Penguin Panic!* game: Sébastien Corfa, Camille Cros, Guilhem Cichoki, Vincent Pera, Grégoire Baron, Noémie Artigouha and Matthieu Le Boucher. I also wish to thank Isabelle Lacasse from Ubisoft Montréal for giving us the opportunity to surpass ourselves with the Game Labs competition, as well as Jessika Gagné from Ubisoft Saguenay for bringing us together with media contacts to increase our reach. At last, I wish to thank both Jared Pearson and Dany Savard, mentors entitled to us by Ubisoft Québec and Ubisoft Saguenay respectively.

REFERENCES

[1] Elizabeth A. Boyle, Thomas M. Connolly, Thomas Hainey, and James M. Boyle. 2012. Engagement in digital entertainment games: A systematic review. *Computers in Human Behavior* 28, 3 (2012), 771 – 780. https://doi.org/10.1016/j.chb.2011.11.020
[2] Substainable Business Council. 2015. Business survey on climate changes. (November 2015), 1–15.
[3] Damien Djaouti, Julian Alvarez, Jean-Pierre Jessel, and Olivier Rampnoux. 2011. *Origins of Serious Games*. Springer London, London, 25–43. https://doi.org/10.1007/978-1-4471-2161-9_3
[4] J. Firth, S. Rosenbaum, B. Stubbs, P. Gorczynski, A. R. Yung, and D. Vancampfort. 2016. Motivating factors and barriers towards exercise in severe mental illness: a systematic review and meta-analysis. *Psychological Medicine* 46, 14 (2016), 2869–2881. https://doi.org/10.1017/S0033291716001732
[5] Yannick Francillette, Bruno Bouchard, Eric Boucher, Sébastien Gaboury, Paquito Bernard, Ahmed Jérome Romain, and Kévin Bouchard. 2018. Development of an Exergame on Mobile Phones to Increase Physical Activity for Adults with Severe Mental Illness. In *Proceedings of the 11th PErvasive Technologies Related to Assistive Environments Conference (PETRA '18)*. ACM, New York, NY, USA, 241–248. https://doi.org/10.1145/3197768.3201521
[6] Juho Hamari, David J. Shernoff, Elizabeth Rowe, Brianno Coller, Jodi Asbell-Clarke, and Teon Edwards. 2016. Challenging games help students learn: An empirical study on engagement, flow and immersion in game-based learning. *Computers in Human Behavior* 54 (2016), 170 – 179. https://doi.org/10.1016/j.chb.2015.07.045
[7] Dekhtyar P. Migdal A. Henderson R.M., Reinert S. 2017. Climate Change in 2017: Implications for business. (June 2017), 1–39.
[8] Reza Khoshkangini, Annapaola Marconi, and Giuseppe Valetto. 2017. Machine Learning for Personalized Challenges in a Gamified Sustainable Mobility Scenario. In *Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '17 Extended Abstracts)*. ACM, New York, NY, USA, 361–368. https://doi.org/10.1145/3130859.3131321
[9] Fedwa Laamarti, Mohamad Eid, and Abdulmotaleb El Saddik. 2014. An Overview of Serious Games. *Int. J. Comput. Games Technol.* 2014, Article 11 (Jan. 2014), 1 pages. https://doi.org/10.1155/2014/358152
[10] Baihua Li, Mark Maxwell, Daniel Leightley, Angela Lindsay, Wendy Johnson, and Andrew Ruck. 2014. Development of exergame-based virtual trainer for physical therapy using kinect. In *Games for Health 2014*. Springer, 79–88.
[11] Olana Missura. 2015. Dynamic difficulty adjustment. (2015).
[12] Office of Climate Change, Energy Efficiency Government of Newfoundland, and Labrador. 2015. Public Awareness Campaign on Climate Change and Energy Efficiency Baseline and Follow-Up Surveys. , 54 pages.
[13] Jan Papoušek and Radek Pelánek. 2015. Impact of Adaptive Educational System Behaviour on Student Motivation. In *Artificial Intelligence in Education*, Cristina Conati, Neil Heffernan, Antonija Mitrovic, and M. Felisa Verdejo (Eds.). Springer International Publishing, Cham, 348–357.
[14] Ian Simmonds. 2015. Comparing and contrasting the behaviour of Arctic and Antarctic sea ice over the 35 year period 1979-2013. *Annals of Glaciology* 56, 69 (2015), 18–28.
[15] Statista. 2018. Gaming – Statistics & Facts. https://www.statista.com/topics/1680/gaming/. Accessed: 2018-07-23.
[16] RS Sutton and AG Barto. 2017. Reinforcement learning: An introduction, (complete draft).
[17] Jonathan Tremblay, Bruno Bouchard, and Abdenour Bouzouane. 2012. *Understanding and Implementing Adaptive Difficulty Adjustment in Video Games*. IGI Global, Hershey, PA, USA, 82–106. https://doi.org/10.4018/978-1-4666-1634-9.ch005
[18] G. N. Yannakakis and J. Hallam. 2009. Real-Time Game Adaptation for Optimizing Player Satisfaction. *IEEE Transactions on Computational Intelligence and AI in Games* 1, 2 (June 2009), 121–133. https://doi.org/10.1109/TCIAIG.2009.2024533