International Journal of Computer Engineering and Information Technology

VOL. 9, NO. 10, October 2017, 246–253
Available online at: www.ijceit.org
E-ISSN 2412-8856 (Online)

Virtual Reality Games for Rehabilitation

Adomas Paulauskas

Juozas Balčikonis Gymnasium, Panevėžys, Lithuania

paulauskas.adam@gmail.com

ABSTRACT

The main problems of applying virtual reality to rehabilitation are related to the compatibility of different system components, big data processing and adaptation to individual users' needs. The objective of the research is to develop and apply virtual reality practises to games for use in lower body motor rehabilitation. A new rehabilitation game adaptation model which accommodates for the player's physical abilities and their level of fatigue has been suggested, implemented and analysed. The adaptation algorithm still only partially adapts to the physical abilities and the fatigue of the player. To improve the adaptiveness of the algorithm it could be further refined. Another way to improve it would be to see how the algorithm behaves when more data is introduced into the system. The scientific novelty of this research is the analysis of the level of fatigue of a person and the adaptation to it.

Keywords: *Adaptation Algorithm, Level of Fatigue, Rehabilitation, Reinforcement Learning Algorithm, Virtual Reality.*

1. INTRODUCTION

In the recent years, many groups of researchers have been working towards applying virtual reality to rehabilitation [1-3]. Virtual reality improves the effectiveness of the rehabilitation process because of these factors: (1) serious games-based methodology ensures control over the person-to-virtual-object interaction; (2) it provides a feedback loop which grants the ability to follow the rehabilitation process; (3) it raises the patient's motivation [3].

Despite the fact that there are many works in this domain, many technological, medical and engineering problems are yet to be solved. The key issues are related to: (1) the testing of virtual reality techniques and the selection of the most relevant ones; (2) the creation and evaluation of the rehabilitation system models; (3) the development and/or application of big data processing algorithms; (4) the testing and evaluation of the system's effectiveness and adaptation possibilities for an individual person's needs.

The objective of the research is to develop and apply virtual reality practises to games for use in lower body motor rehabilitation.

In order to achieve the objective, the following tasks have to be performed:

- Analysis of the main properties, capabilities and weaknesses of serious games-based rehabilitation systems.
- Development and testing of the virtual game environment.
- Creation and evaluation of the adaptation model for accommodating an individual person's abilities.
- Application and analysis of three adaptation strategies of the rehabilitation game.

The scientific novelty of this research is the analysis of the level of fatigue of a person and the adaptation to it.

2. METHODS OF RESEARCH

A modified constructive research method [4] which consists of five stages has been used:

(1) Identifying a practically relevant problem to be solved;
(2) Analyzing research related to the identified problem;
(3) Constructing an idea for solving the problem;
(4) Experimentally demonstrating that the solution works;
(5) Examining the application scope of the proposed solution.

Experimental data was processed using statistical methods available in Microsoft Excel and open-source software JASP version 0.8.0.1 (https://jasp-stats.org/).

3. RELATED WORK

Related research has been divided up into three categories: (1) the comparative analysis of virtual reality systems for rehabilitation; (2) serious games for rehabilitation; (3) application of machine learning algorithms and methods in rehabilitation.

In summary:

- Virtual reality systems are being mainly used for upper body rehabilitation. The main problems are related to the compatibility of different system components, big data processing and adaptation to an individual user's needs [1, 2].
- The utilisation of serious games boosts the motivation of the patients and warrants the ability to select the relevant parameters for an individual person's needs. The adaptation problem has been discussed in the majority of the analysed research [5-8].
- Machine learning methods and algorithms are widely used in the systems for patients with reduced mobility and/or after a stroke. Supervised learning and semi-supervised learning methods are dominant [9, 10].

4. VIRTUAL GAME ENVIRONMENT

The game development framework (see Fig. 1) consists of the technical, developer and user parts. The technical part is comprised of the Kinect sensor, virtual environment and the adaptation system. The Kinect sensor was chosen for its accessibility and suitability for the implementation of the game idea. The idea of the game came from the analysis of the related research.

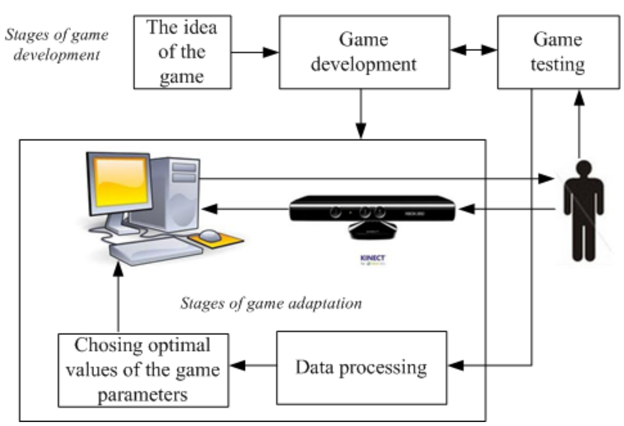

Fig. 1. Game development framework.

A game for training a person's legs was created in which a constant stream of balls fall from the top of the screen. The goal of the player is to perform the exercise of kicking as many balls as possible during a specified period of time.
The first experiment was carried out in order to determine the accuracy of mapping a person into the game environment. Leg position changes were measured before and after a person took a step forward (see Fig. 2).

Mapping data was compared using the Paired Samples T-Test. The results of the comparison are shown in Fig 3. and Tables 1 and 2.
The standard error means of the left and right leg position changes are larger than the standard error mean of the distance between the legs because leg position changes are measured after the legs move and the distance between the legs is measured when the person is standing still (see Table 1).

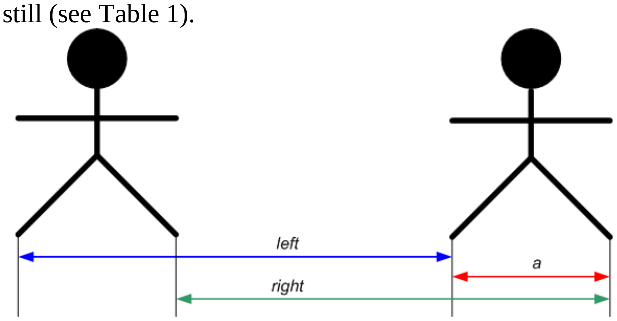

Fig. 2. The distances of/between legs in different positions

The correlation coefficient, which is 0.97 - 0.99, describes the accuracy of mapping a person into the game environment (see Table 1).
The 2-tailed significance values (see Table 2) show that the mapping is accurate.

Table 1: Statistical characteristics of the virtual-real pairs

	Pair 1		Pair 2		Pair 3	
	V_left	R_left	V_right	R_right	V_a	R_a
Mean	94.0	90.8	91.5	88.8	30.7	33.5
Std. Dev.	28.5	27.5	25.2	25.6	12.2	12.1
Std. Error Mean	6.7	6.5	6.0	6.0	2.9	2.9
Corre-lation	0.994		0.989		0.973	
Sign.	0.000		0.000		0.000	

Table 2: Statistical characteristics of the paired differences

		Pair 1	Pair 2	Pair 3
Mean		3.2	2.7	-2.8
Std. Deviation		3.3	3.9	2.8
Std. Error Mean		0.8	0.9	0.7
95% Confidence Interval of the Difference	Lower	1.6	0.8	-4.3
	Upper	4.9	4.6	-1.4
t		4.118	2.947	-4.229
df		17	17	17
Sig. (2-tailed)		0.001	0.009	0.001

The second experiment was carried out to determine the scale of the vertical axis of the virtual environment. The height of a raised leg in the real world and in the virtual environment was measured and the scale was calculated. The numerical value of the scale is equal 1.8.
The third experiment is related to the evaluation of the stability of the environment. On the basis of the

experimental data, in which the speed distribution is a Gaussian (a total of 4374 measurements, 23 participants of the experiment), the ball's lifetime (t) as a function of its average speed (v) has been determined (see also Fig. 4 and Fig. 5):

$$t = \frac{4.52}{v} \quad (1)$$

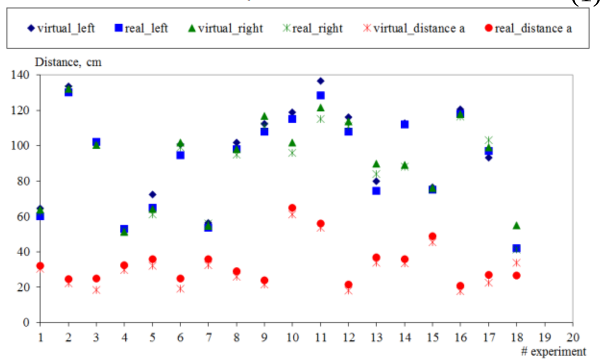

Fig. 3. The accuracy of mapping a person into the game environment

The coefficient 4.52 defines the average distance the ball travels from first contact with the leg until disappearing from the field of view. After comparing the experimental times with the ones calculated from function (1) using the experimental speeds, the correlation coefficient, which shows how accurately function (1) describes the experimental data, was calculated and is equal to 0.98. The larger the time values the more scatter there is in the data because of ball.

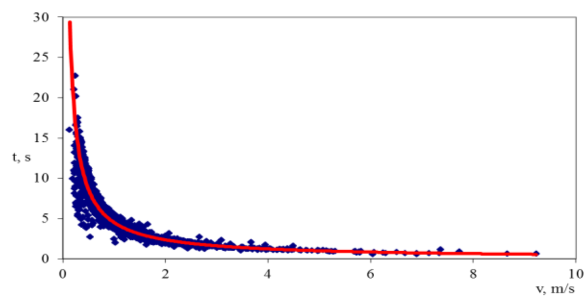

Fig. 4. Ball lifetime as a function of its average speed (4374 measurements)

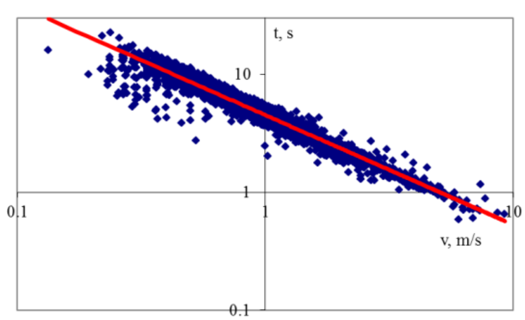

Fig. 5. Ball lifetime as a function of its average speed on a logarithmic scale

To verify that the system is stable the experiment was carried out a second time. 12 test subjects took part in the repeated experiment. Average speeds, lifetimes of the balls, angles of the lower and upper parts of the leg were measured. The distributions of the measured parameters values are presented in Fig. 6a-d.

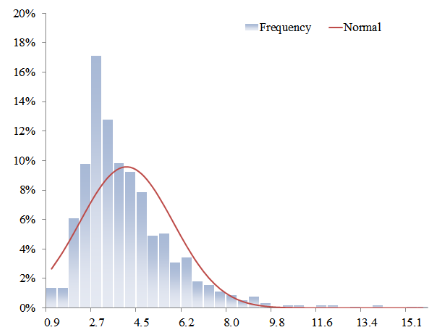

Fig. 6a. The distribution of average ball speeds

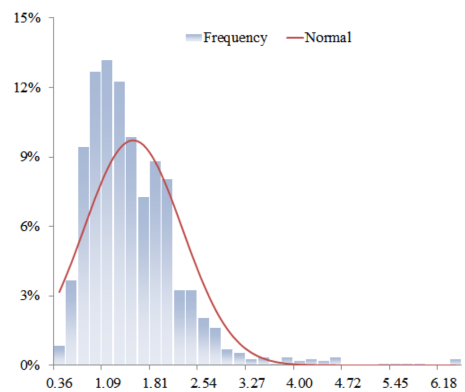

Fig. 6b. The distribution of ball lifetimes

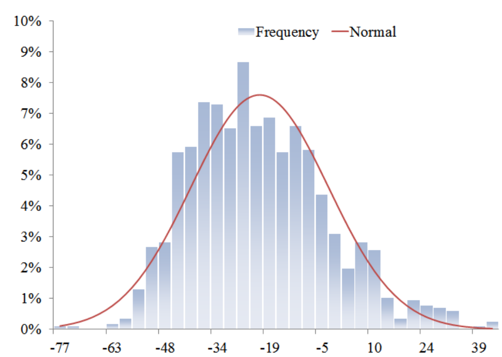

Fig. 6c. The distribution of the angles of the lower part of the leg

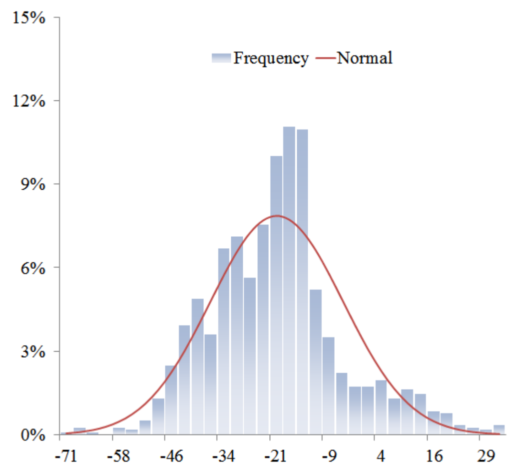

Fig. 6d. The distribution of the angles of the upper part of the leg

The results of the performed experiments show that the system is stable and it can be assumed that the constant environment parameters were selected properly.

5. ADAPTATION STRATEGIES

The analysis of pre-existing applications of adaptation models has led to a classification of three adaptation strategies:

- The first strategy is based on manual assignment of the game parameter values by a rehabilitation specialist. This principle is applied in the majority of systems.
- The second strategy is based on pre-processing already collected data, from which rule sets that define the parameter value bounds are created. If the parameters of a certain patient fall outside the bounds, then they are changed and the rule set is modified.
- The third strategy is based on processing large amounts of collected data and the use of artificial intelligence algorithms for predicting the correct parameter values for the specific needs of individual patients.

5.1 The First Strategy

When implementing the first adaptation strategy, an experiment was conducted in which the length of the game was 10 minutes and the time between two ball spawns was set to 3 seconds. 18 people participated in the experiment. The average ball speeds and the average speeds of the upper and lower parts of the leg were measured. For each ball, participants were grouped into subsets of all their corresponding parameter values, which were then averaged. The Fourier transform was applied as the main method of analysis to assess the dynamics of the data set. The Discrete Fourier Transform (DFT) decomposes a time series and transforms it into the frequency domain. The inverse transform recovers a simplified time series which describes the original data set. The results produced by the inverse Fourier transform are presented in Fig. 7. Until ball 60-70 the trend of the speeds of the upper part of the leg differs from the trend of the speeds of the ball and the lower part of the leg. During this stage players utilise the upper part of the leg more, but the lower part is used to actually kick the ball, therefore it has a stronger effect on the speed of the ball. This is the stage of the player acclimatising to the virtual game environment which last for 3 - 3.5 minutes. During the rest of the game the trends of the lower and upper parts of the leg and the ball speeds are similar and the process follows a periodic pattern. Even though the number of extrema is low, a lengthening trend of the period can be observed. This result can be explained by the player becoming tired.

The periodic trend obtained from the results grants the ability to apply the second adaptation strategy and to create rule sets which define the parameter value bounds.

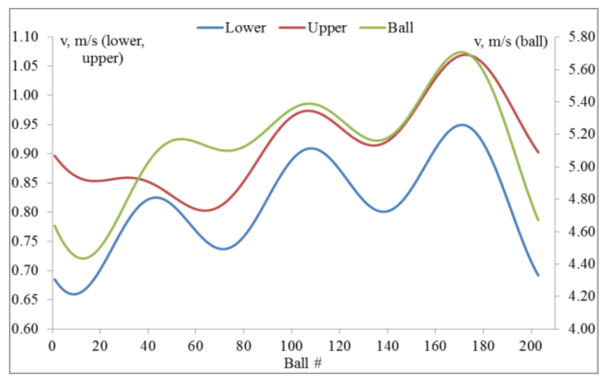

Fig. 7. Fourier transformation results based on average speeds of balls, upper and lower parts of the leg

5.2 The Second Strategy

After applying the first adaptation strategy a lengthening trend of the period was observed which is caused by the player becoming tired, thus an adaptation algorithm was chosen which changes the time between two ball spawns.

The adaptation algorithm uses a heuristic formula the purpose of which is to determine the limit of the player's physical abilities:

$$t(n) = ak^{n-1} \quad (2)$$

where n – the number of the ball, a – the time of the first ball spawn, k – the coefficient for decreasing the time between ball spawns.

The optimum interval for the time between two balls spawns is determined by a given player's abilities. If the player kicks the balls fast enough and no two balls collide, then the time between two ball spawns is decreased according to t(n) and the player can kick more balls over the total game time. On the other hand, when two balls collide, the time between ball spawns is increased to adapt both to the physical abilities of the player and to them becoming tired. The generalized reactive planning adaptation algorithm is shown in Fig. 8.

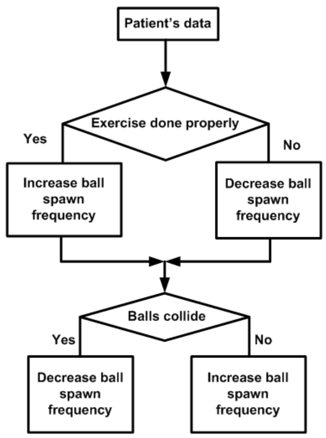

Fig. 8. Generalized reactive planning adaptation algorithm

An experiment was carried out on 12 test subjects to analyse the algorithm's adaptiveness to the abilities of a player. The length of the game was set to 2 minutes and the times between two ball spawns were chosen using the generalized algorithm (see Fig 8). When the player was active, the values of the following parameters were measured: 1) the height of the collision point between the leg and the ball; 2) the speed and the angle of the lower and upper parts of the leg; 3) the average speed of the ball over its lifetime; 4) the height at which the ball left the field of view; 5) the lifetime of the ball; 6) full time of the game when the ball leaves the field of view.

Fig. 9 presents t(n) without the rest of the algorithm, the results of one player with the full algorithm at work and the simplified time series generated from the experimental data.

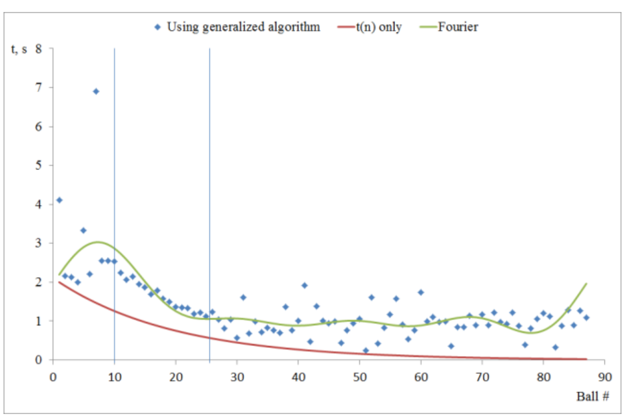

Fig. 9. t(n) without the algorithm, the results of one player with the algorithm at work and its simplified time series (Fourier)

Three areas depicted in Fig. 9 mark the different stages of the behaviour of the adaptation algorithm. In the first area the scatter is large due to the player learning how to do the exercise properly. The trend in the second area is the shifted formula (2). The high amount of scatter in the third area shows the activity of the full algorithm.

A simplified time series was generated for the experimental data by applying the inverse Fourier transform (see Fig. 9). The third area, where the full algorithm is working, contains a periodic trend. The length of the period is stable which shows that the algorithm adapts to the player becoming tired.

The sum of squared errors (square difference between the experimental and the generated values) were calculated for each of the simplified time series of the 12 players. It was observed that a periodic pattern emerges in all of the test. Three time series which over all the experimental data produced the least amount of squared error have been selected (see Fig. 10). Until ball 10-15 all three time series are similar because the algorithm adapts to the physical abilities of the player. Until ball 30 the time series diverge because the players learn how to do the exercise. After that an increasing trend is observed which is caused by fatigue.

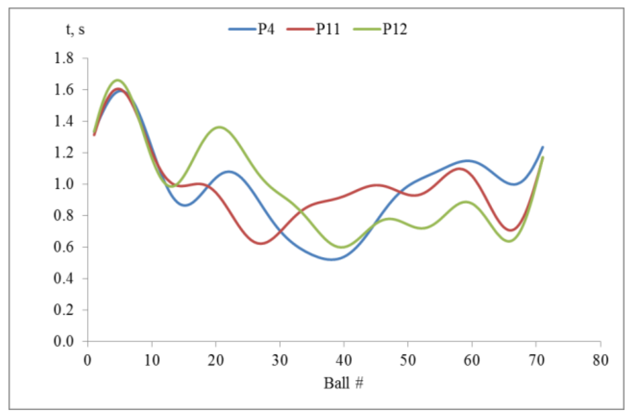

Fig. 10. Simplified functions produced by the inverse Fourier transform of three patients

The second experiment, which was 20 minutes long, was conducted to determine whether the periodic pattern observed before emerges over a longer period of time. Fig. 11 presents a simplified time series generated from the 20 min data by the inverse Fourier transform. The time series is partitioned into 5 areas based on the extrema.

The first area corresponds to the trend of the data in Fig 9. The second one has a maximum of 1.27 s and shows that the player is resting. In the third area the time between two ball spawns is decreasing and the game becomes more intense. The time value reaches a minimum equal to 0.54 s. In the fourth area the time between two balls has an increasing trend which indicates that the player is tired, but the player still continues to do the exercise. The last

area depicts a process analogous to the one in the first area, but the time magnitude is larger due to fatigue.
9 periods can be observed in Fig. 11, the length of which differs depending on the magnitude of the change between two local minima/maxima: the larger the magnitude of the change, the longer the period.

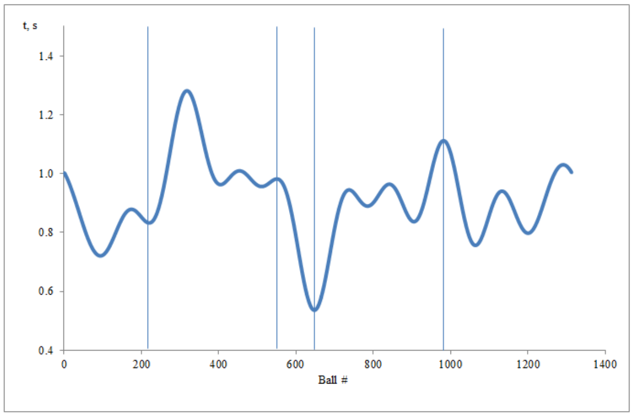

Fig. 11. The simplified time series of the 20 minute experiment when the second adaptation strategy is used

According to the analysis of the experimental data, it can be concluded that the reactive planning adaptation algorithm is unable to foresee drastic changes in the player's activity. Therefore, the length of the periods of the experimental data is not stable which means that the algorithm does not fully adapt to the player's fatigue. The next step in improving the adaptation of the rehabilitation game is applying the third strategy which is based on using artificial intelligence algorithms for processing large amounts of collected data.

5.3 The Third Strategy

The reactive planning algorithm based on the second adaptation strategy was changed to a reinforcement learning agent. This new algorithm is capable of adapting to an individual person's abilities by learning from a previous experimental data. The same result can be achieved using supervised learning algorithms, but they require more computer resources.
A reinforcement learning agent changes its behavioural strategy depending on its knowledge about environment. The agent maximizes the predefined reward function and the values of the learnt utility function. The internal reward function defines short-term consequences whereas the utility function defines long-term consequences.
Temporal difference learning was used for predicting more accurate game parameter values for each player. Temporal difference learning is based on the use of raw experience without having a dynamical model of system.
During the game a binary tree is being generated. Every node of the tree defines a specific state of the game. The state is characterized by a set of parameters: (1) the time between two ball spawns; (2) the total number of balls already spawned; (3) number of balls that have not collided with other balls; (4) the time from the start of the game. The edges of the tree describe the change between states which is expressed in time between two ball spawns. The depth of the tree is equal to the total number of balls (n). A fragment of the tree is presented in Fig. 12.

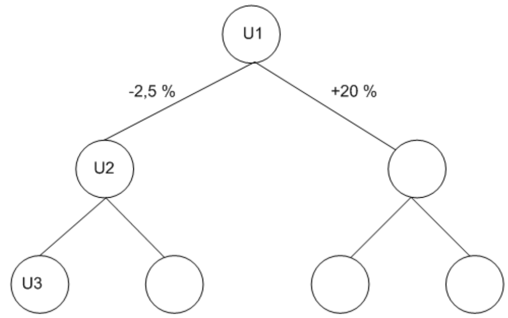

Fig. 12. A fragment of the binary tree generated by the reinforcement learning algorithm

The depth of the tree is dynamic and depends on the time of the experiment. A path-finding problem is being solved. The path is chosen depending on the calculated values of the reward and utility functions. If the value of the utility function is not defined, then the values of the reward function are used. The value of the reward function is calculated from the parameters of the next state s':

$$R(s') = \frac{1}{t+1}, \tag{3}$$

where s' - the next state, t - the time between two ball spawns.
When the game is in state s' in which the reward function has the value of $R(s')$ and two balls collide, the value of $R(s')$ is reduced by one for $R(s')$ to move to the interval of [-1; 0]. Reduction by one was chosen to reduce the growth of complexity.
The utility function is defined as follows:

$$U(s) = U(s) + N(s)(R(s) + \gamma U(s') - U(s)), \tag{4}$$

where s' - the next state, s - the current state, $N(s)$ – function of significance based on the interactions of s, γ - discounting factor for reward at distant time step to be less important.
The values of the utility function are re-calculated by regressing through the tree. When the tree is being generated, the value of the utility function is equal to zero and the reward function is used until the first regression through the specific state.
The optimum interval for the time between two balls spawns is determined by a given player's abilities. If the player kicks the balls fast enough and no two balls collide, then the time between two ball spawns is decreased by 2.5 % given that the value of the utility function, which

was calculated based on actions taken by previous players, for decreasing the time is more than that of increasing the time. Otherwise, the time between ball spawns is increased by 20 %.

The algorithm was tested in the Health and Rehabilitation Centre on 23 patients (13 women and 10 men) aged 16 to 75 with limited motor movement after trauma or surgery. A rehabilitation specialist set the duration of the game to one minute because of the patients' physical disabilities.

The times between two ball spawns selected by the algorithm were measured during the experiment. The results of one patient are presented in Fig. 13.

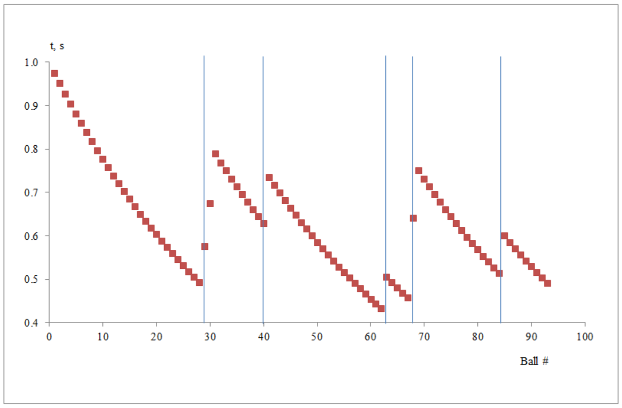

Fig. 13. Results of one patient of the time between two ball spawns as a function of ball number using the reinforcement learning algorithm

Until two balls collide (the positions of collision are marked by vertical lines in Fig. 13) the algorithm selects the time between ball spawns from the binary tree generation process. The time between two ball spawns can be selected from a previously generated branch or from branch that is generated during random exploration.

The values of the reward function are modified after two balls collide. Because the algorithm maximizes the reward and utility functions, it often chooses to decrease the time between ball spawns. Due to the algorithm selecting the optimal policy independently of a controlling human, the causes for a certain choice can be only partially inferred.

After comparing patients of different levels of motor movement it can be concluded that the algorithm adapts to the physical abilities of the patient. The trend of increasing values in Fig. 14 shows that algorithm is trying to find the optimal time between two ball spawns for the given patient. In the case of Fig. 14, the optimal time is around 3.0-4.0 seconds. The optimal time for different patients depends on their individual abilities. Fig. 14 shows that the results of one patient differ. Possible causes for this might be that the algorithm learns from previous data or the wellbeing of the patient on the day of the test. However, the attempt to define the factors for these differences was unsuccessful because the collected data set is too small due to short game times and not being able to repeat tests with the same patient.

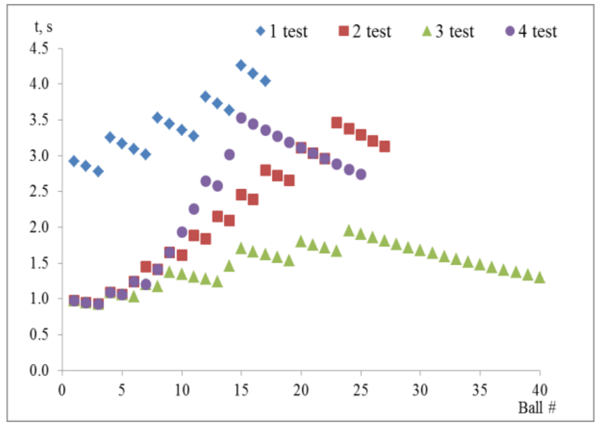

Fig. 14. Four tests of a single patient using the reinforcement learning algorithm

Experiments were conducted, which were 5 and 20 minutes long, to analyse how the algorithm behaves when the length of the game is increased. Fig. 15 presents the results of a single patient which played 2 games of 5 minutes and one game of 20 minutes. It can be observed that the trends of all the graphs are the same. The graphs of the 5 minute tests are slightly shifted, but the general trend of the graphs is the same.

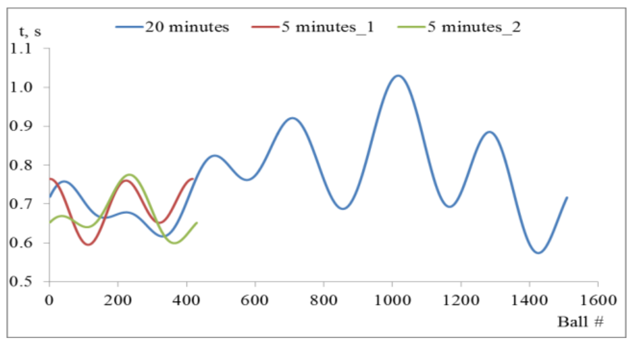

Fig. 15. The simplified time series of a single test subject using the reinforcement learning algorithm

By comparing the graph of the 20 min experiment from the second adaptation strategy (see Fig. 11) and the graph of the 20 min experiment from the third strategy (see Fig. 15) it can be seen that the magnitude of the changes of times between two ball spawns in the third strategy is smaller than the one in the second strategy. However, the graph in Fig. 15 displays the same pattern as in Fig. 11 where the length of the period differs depending on the magnitude of the change between two local minima/maxima: the larger the magnitude of the change, the longer the period. To reduce the differences between the lengths of the periods the algorithm can be refined.

Another way to do this would be to see how the algorithm behaves when more data is introduced into the system.

6. CONCLUSIONS

The main properties, capabilities and drawbacks of virtual reality and serious games-based rehabilitation systems have been analysed in the research. The analysis showed that there is a lack of virtual reality rehabilitation games for the lower part of the body and a lack of adaptation algorithms which take into account the physical abilities and the fatigue of the player.

A new rehabilitation game adaptation model which accommodates for the player's physical abilities and also their level of fatigue has been suggested, implemented and analysed.

Three experiments measuring the accuracy of mapping a person into the virtual environment and the stability of the environment have been performed. The results of the experiments show that the parameters of the game environment have been chosen properly.

During game creation and testing, three adaptation strategies have been explored:

- The experiments while using the first strategy showed that the data displays a periodic trend. A lengthening of the period can be observed which can be explained by the player becoming tired;
- According to the analysis of the experimental data while using the second strategy, the reactive planning adaptation algorithm is unable to foresee drastic changes in the player's activity. Therefore, the length of the periods of the experimental data is not stable which means that the algorithm does not fully adapt to the player's fatigue;
- The reinforcement learning algorithm implemented for the third strategy decreased the magnitude of the changes between the local minima and maxima of the time between two ball spawns.

The adaptation algorithm only partially adapts to the physical abilities and the fatigue of the player. To improve the adaptiveness of the algorithm it could be further refined. Another way to improve it would be to see how the algorithm behaves when more data is introduced into the system.

REFERENCES

[1] B. Andika, J. Tantra, and A. A. P. Wai, Interactive Rehabilitation Game Development Using Natural User Interface Devices, 2012.

[2] [2] M. Gotsis, V. Lympouridis,, D. Turpin, A. Tasse, I. C. Poulos, D. Tucker, ..., and M. Jordan-Marsh, "Mixed reality game prototypes for upper body exercise and rehabilitation", in Virtual Reality Short Papers and Posters (VRW), 2012 IEEE, pp. 181-182.

[3] [3] S. C. Yeh, S. H. Lee, J. C. Wang, S. Chen, Y. T. Chen, Y. Y. Yang, ..., and Y. P. Hung, "Virtual reality for post-stroke shoulder-arm motor rehabilitation: Training system & assessment method", in e-Health Networking, Applications and Services (Healthcom), 2012 IEEE 14th International Conference, pp. 190-195.

[4] [4] E. Kasanen, and K. Lukka, "The constructive approach in management accounting research", Journal of management accounting research, (5), 1993, pp. 243-264.

[5] [5] P. A. Rego, P. M. Moreira, and L. P. Reis, "A Serious Games Framework for Health Rehabilitation", in Gamification: Concepts, Methodologies, Tools, and Applications: Concepts, Methodologies, Tools, and Applications, 2015, 404.

[6] [6] P. Rego, P. M. Moreira, and L. P. Reis, "Serious games for rehabilitation: A survey and a classification towards a taxonomy", in Information Systems and Technologies (CISTI), 2010 5th Iberian Conference, pp. 1-6.

[7] [7] T. H. Costa, N. M. Soares, W. A. Reis, and F. M. Bublitz, "A systematic review on the usage of games for healthcare:, in Consumer Electronics-Berlin (ICCE-Berlin), 2015 IEEE 5th International Conference, pp. 480-484.

[8] [8] M. Graafland, M. Dankbaar, A. Mert, J. Lagro, L. De Wit-Zuurendonk, S. Schuit, ..., and M. Schijven, "How to systematically assess serious games applied to health care", JMIR serious games, 2(2), 2014.

[9] [9] R. C. Deo, "Machine learning in medicine", Circulation, 132(20), 2015, pp. 1920-1930.

[10] [10] T. J. Cleophas, A. H. Zwinderman, and H. I. Cleophas-Allers, Machine learning in medicine, New York: Springer, 2013.