Human-Machine Cooperation in General Game Playing

Maciej Świechowski*, Kathryn Merrick†, Jacek Mańdziuk‡, Hussein Abbass†
*Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland
Email: m.swiechowski@ibspan.waw.pl
†School of Engineering and Information Technology, University of New South Wales, Canberra, Australia
Email: (k.merrick, h.abbass)@adfa.edu.au
‡Faculty of Mathematics and Information Science, Warsaw University of Technology, Warsaw, Poland
Email: j.mandziuk@mini.pw.edu.pl

***Abstract*—This paper presents a framework for cooperation between a human and a general game playing agent. Cooperation is defined as two entities causing each other to modify their behaviour to achieve some mutual advantage. Such human-computer cooperation has the potential to offer insights that can help us improve the performance of artificial agents, as well as improving the performance of humans during certain kinds of strategic interactions. This paper focuses specifically on game playing as a form of strategic interaction. By proposing a framework for cooperation between a human and a general game playing agent, our aim is to create a flexible system that may be applicable to cooperation in other kinds of problem solving and strategic interactions in the future. We evaluate the framework presented in this paper by means of a human study. We observe humans playing games with and without the cooperation of a general game playing agent. We present experimental results of the pilot study as well as proposed changes in the experiment. These changes aim to verify the hypothesis that human-machine cooperation within our framework can indeed lead to mutual advantage.**

Keywords–human-machine study; cooperation; General Game Playing; Monte Carlo Tree Search.

I. INTRODUCTION

General Game Playing (GGP) has been claimed as "The AI Grand Challenge", since it is seen as a step towards strong human-like intelligence [1]. The design and study of approaches that permit cooperation between humans and GGP agents is thus an important, complementary research stream. Such human-computer cooperation has the potential to offer insights that can help us improve the performance of artificial agents, as well as improving the performance of humans during certain kinds of strategic interactions. We borrow a concept of cooperation from [2] stating that it takes place when two systems cause each other to modify their behavior to achieve some mutual advantage. The type of strategic interaction we will consider is game playing. We will consider the type of machine cooperator as a GGP [3] agent as proposed by the Stanford Logic Group [4]. This is currently the most prominent embodiment of the multi-game playing idea, which aims to create systems capable of playing a variety of games (as opposed to agents that can only play single games). The specific type of GGP machine cooperator we will consider is a Monte Carlo Tree-Search (MCTS) based player. The MCTS is used as the main routine of the strongest state-of-the-art GGP players and is also widely applied to other games such as Go [5], Arimaa [6] as well as other areas of Artificial Intelligence (AI) [7]. We will conduct a human user study to validate our approach to human-machine cooperation. In this paper, we present two pilot studies we have performed. The aims of these pilot studies were to (1) verify our setup for cooperation and (2) provide preliminary verification of our research hypothesis. A large-scale experiment is the next step to undergo. Apart from providing the circumstances for the cooperation, we are also interested in measuring the effects of such cooperation, i.e., how it affects the average quality of play. Human-machine interaction has been a hot research area outside the scope of games, e.g., in the areas of aviation [8] or surgery [9]. In games, however, the task of creating machine players has been challenging enough on its own [5][10]. To our knowledge, there has been no related work concerning human-machine cooperation in GGP or in any other MCTS-based game playing. We believe that the way we approach the problem of cooperation can contribute to the area of general knowledge-free and learning-based methods in games [11], because we can examine the way humans learn from machines and provide a basis for automatic methods by which machines can learn games from humans.

The remainder of the paper is organized as follows: the next two sections contain brief descriptions of GGP, MCTS and our cooperation platform within the MCTS framework. In Sections IV and V, we formulate the research hypothesis and the experimental methodology, respectively. Section VI describes the two particular setups tested in the two pilot studies and Section VII discusses the results. The last section is devoted to conclusions and directions for future work.

II. GENERAL GAME PLAYING

A. Basics

GGP is a trend in AI which involves creating computer systems, known as GGP agents, capable of playing a variety of games with a high level of competence. The range of games playable within the GGP framework is any finite deterministic game. Unlike specialized playing programs, GGP systems do not know rules of the games being played until they actually start. The concept of designing universal game playing agents is also known as multi-game playing or metagaming, but as stated in the introduction, we refer to the Stanford's definition of GGP [3] which is the most recent one. The official GGP Competition, which is *de facto* the World Championship Tournament, is also part of the GGP specification. The machine player used in this research is our entry in the latest installment of the competition (2014). Borrowing from the GGP terminology, we will use the term *play clock* for the time (in seconds) available to make a move by a player. To enable matches between our GGP program and humans, we

 ISBN: 978-1-61208-382-7

had to slightly loosen the official specification. For instance, GGP agents are normally penalized for not responding with a legal move in time by having the move chosen for them at random. In our scenario, human participants can think about moves as long as they want to without any penalty and the machine players always respond in time.

B. The Tree-Search Algorithms Used

MCTS is an algorithm for searching a game tree in a quasi-random fashion in order to obtain as accurate an assessment of game states as possible. In general, the assessment is computed statistically as the average score - **Q** - which is defined by the total score of simulations going through a state divided by the number of visits to that state. The total score is a sum of the outcomes of simulations. For all games considered in this article, the value of 1.0 denotes a win, 0.5 denotes a draw and 0.0 denotes a loss in a single simulation. The input to the method is the current game state. Then, the algorithm gradually searches the game tree starting from the current state in a series of iterations adding one node in each of them. An iteration consists of the following four steps:

1) **Selection.** Start from the root and go progressively down. In each node, choose the child node with the highest average score until reaching a leaf node.
2) **Expansion.** If a state contained in the leaf node is not terminal, choose an action which would fall out of the tree. Allocate a new child node associated with that action; simulation.
3) **Simulation.** Starting from a state associated with the newly expanded node, perform a full game simulation (i.e., to a terminal state).
4) **Backpropagation.** Fetch the result of the simulated game. Update statistics (average scores, numbers of visits) of all nodes on the path of simulation, starting from the newly expanded node up to the root node.

The algorithm can be stopped at any time. The final output of the search is the action with the highest average score **Q** for the player who is currently to make a move in a game. A significant improvement over the pure MCTS is the Upper Confidence Bounds Applied to Trees (UCT) algorithm [12]. The purpose of the algorithm is to maintain balance between the exploration and exploitation ratio in the selection step. Instead of sampling each action uniformly (as is the case of MCTS) or greedily, the following selection formula is applied:

$$a^* = arg \max_{a \in A(s)} \left\{ Q(s,a) + C\sqrt{\frac{ln\,[N(s)]}{N(s,a)}} \right\} \qquad (1)$$

where ***s*** is the current state; ***a*** is an action in this state; ***A(s)*** is a set of actions available in state ***s***; ***Q(s,a)*** is an assessment of performing action ***a*** in state ***s***; ***N(s)*** is a number of previous visits to state ***s***; ***N(s,a)*** is a number of times an action ***a*** has been sampled in state ***s***; *C* is the exploration ratio constant.

III. Cooperation in the MCTS Framework

The machine cooperator used in this paper is an adapted MiNI-Player [13][14] - a GGP program equipped with additional features to enable cooperation. First and foremost, the machine provides statistics to help humans choose which move to play. During cooperative play, it is always a human who makes the final choice with or without taking advantage of the provided statistics. The second means of cooperation is by permitting interference with the MCTS. In this way, we propose an interactive process of building the game tree, while playing the game, involving both the machine and human. In the original MCTS, the same four-phase algorithm is repeated all the time during the *play clock*. For cooperative purposes we split this time into three equal intervals T1 + T2 + T3 = *play clock*. Between any two consecutive intervals (T1 and T2 or T2 and T3) humans can interact with the MCTS based on statistics presented to them. The statistics include: each action ***a*** available to the player to make a move with the ***Q(s,a)*** and ***N(s,a)*** values from (1). These values are scaled to the [0%, 100%] interval to be more readable by the participants. The final statistic is the actual number of simulations which ended with a win, draw and loss for the subject, respectively. The MCTS can be directed by the human in two ways: enabling/disabling actions available in the current state or toggling priorities of the actions on/off. If an action is disabled, the MCTS will ignore this action in the selection step, which means that no simulations will start with a disabled action. Changing the priority is equivalent to changing the value of the *C* parameter in (1) from 1 to 10. Participants are allowed to make any number of the aforementioned interventions at each step and once they are done, they click the simulate button to submit all of them in one batch and observe how the statistics have changed. By doing so, they can help the machine to focus on the most promising actions and avoid presumably wasteful computations. On the other hand, the feedback from the machine supports or questions the above-mentioned human player's choices. Our experimental design is justified based on two observations. First of all, in many well-established games, it has been found that the experts can intuitively discard unpromising actions and focus on the few best ones. Such behavior is manifested by human playing experience and intuition and is one of the aspects in which humans are better than machines. Provided that the human choice is correct, the process can converge faster to the optimal play. The introduction of action priority is a similar, but slightly weaker, modification to the MCTS algorithm. The second observation (or assumption) we made is that the cooperation has to be easy for participants to understand.

IV. Research Hypothesis

To focus the study of performance of human-machine cooperation we formulated the following research hypothesis: **a human cooperating with a machine GGP agent is a better player than human or machine agent individually**. We write this thesis in a shortened form of H + M $>$ M and H + M $>$ H, where H denotes a human player; M denotes a machine player and M + H denotes a hybrid player comprising a cooperating machine and human. We attempt to verify this hypothesis in a devoted experiment. The main research question is whether a mutually beneficial cooperation can originate and develop between human and machine players. In order to verify the above-listed hypotheses, we gathered samples from people playing without any machine assistance (H vs. M) and with such assistance (H+M vs. M). The first case involves a human simply playing a match against our GGP agent named MINI-Player [13] [14]. The second case involves a human playing

 ISBN: 978-1-61208-382-7

against the same opponent but this time with assistance of a "friendly" GGP agent running in the background.

V. Pilot Studies

This paper reports on the results of two pilot studies that we have run to refine our experimental setup as well as to gather preliminary evidence regarding the research hypothesis. In this section, we present a technical setup and introduce one of the games used in the experiment. Because a well-played game is time consuming, we limited the number of games a single person can play to three. The experiment was performed separately for each human subject, so no information could be exchanged in the process, e.g., looking how other people play. The program participants used to play, and the opponent program were run on the same computer, both having access to two physical CPU cores. We set the *play clock* for the two machines (the cooperator and adversary) to 30 seconds in the first pilot study and 9 seconds in the second one. In order to avoid time-outs resulting from the human player, we discarded the concept of random moves if a player fails to respond in time. The matches were played only during weekdays anytime from the morning to the late afternoon. The age of participants varied from 21 to 30 with only one exception of 31 to 40. Most of them were PhD students of computer science. In the experiment, we used three games but one of them, named Tic-Tac-Chess, was discarded after the Pilot Study 1. Figures 1, 2 and 3 show screenshots of the program operated by participants for Inverted Pentago, Nine Board Tic-Tac-Toe and Tic-Tac-Chess respectively.

Inverted Pentago is a game played on a 6x6 board divided into four 3x3 sub-boards (or quadrants). Taking turns, the two players place a marble of their color (either red or blue) onto an unoccupied space on the board, and then rotate any one of the sub-boards by 90 degrees either clockwise or anti-clockwise. A player wins by making their opponent get five of their marbles in a vertical, horizontal or diagonal row (either before or after the sub-board rotation in their move). If all 36 spaces on the board are occupied without a row of five being formed then the game is a draw. Participants play as blue and are the second player to have a turn.

Nine Board Tic-Tac-Toe. In nine board tic-tac-toe, nine 3x3 tic-tac-toe boards are arranged in a 3x3 grid. Participants play as 'O' and are the second player to have a turn. The first player may place a piece on any board; all moves afterwards are placed in the empty spaces on the board corresponding to the square of the previous move. For example if a piece was placed were in the upper-right square of a board, the next move would take place on the upper-right board. If a player cannot place a piece because the indicated board is full, the next piece may be placed on any board. Victory is attained by getting 3 in a row on any board.

Tic-Tac-Chess is a game played on a 7x7 board. Players start with one piece marked by a red or blue square in their respective starting location. Participants are the second player to have a turn. The starting locations are outside the movable area of the board which is defined by the inner 5x5 square. On their turn, each player may move a piece as though it were a Chess knight or capture with a piece as though it were a Chess king. Capturing is possible only with pieces belonging to the center 5x5 square. Pieces from the starting locations do not disappear when moved, so moving a piece from the starting location effectively spawns a new one on a destination square. The first player to get three pieces in a row, column, or diagonal in the center 3x3 square wins.

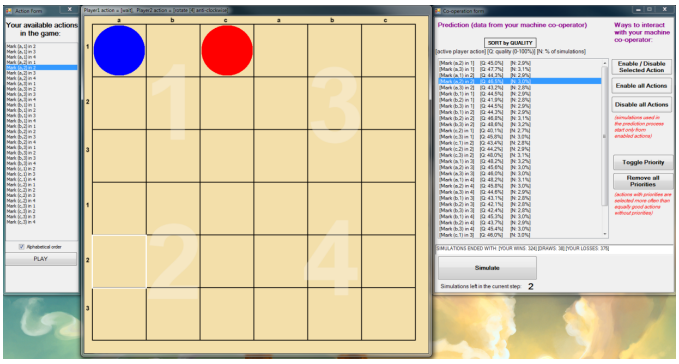

Figure 1. Screenshot of a program used to play Inverted Pentago (version with the cooperation).

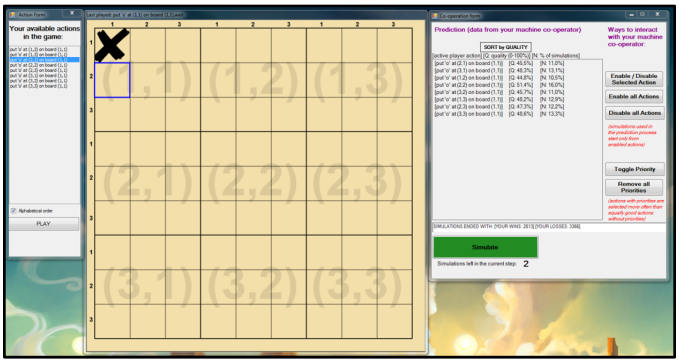

Figure 2. Screenshot of a program used to play Nine Board Tic-Tac-Toe (version with the cooperation).

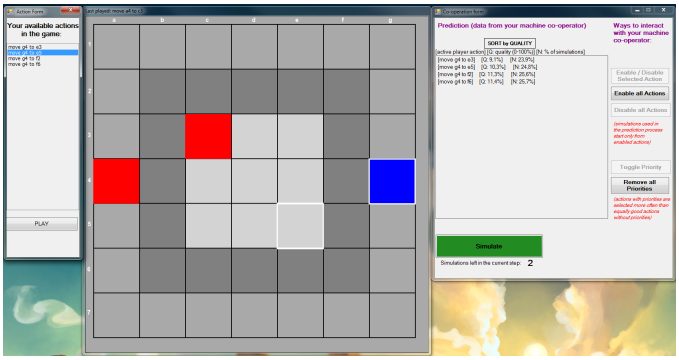

Figure 3. Screenshot of a program used to play Tic-Tac-Chess (version with the cooperation).

A. Pilot Study 1

We gathered 6 human participants for the first pilot study. They were divided into two groups of 3 people each. These two groups formed our two samples of data: playing with machine assistance (H+M) and without (H). During the experiment, we started each game with a short training session. We also gave participants a transcript explaining what they are asked to do and how the user-interface works. When participants were ready, they started playing a serious (i.e., not training)

 ISBN: 978-1-61208-382-7

game and when they finished all three matches they were asked to complete a short questionnaire to obtain a profile of the subjects. The assignment of human players to games was based on the Latin Square Design with 3 games, 6 participants and two playing modes, i.e., with machine assistance being switched ON or OFF. Using this design, the minimum required number of participants for a full experiment is 12, but in the pilot study we stopped at 6 participants.

B. Pilot Study 2

At this point, we decided to revisit the experimental setup slightly and continue the experiment, called pilot study 2, to mitigate some problems that arose. Instead of asking people to play each game once, we asked them to play one game three times in order to enable learning by experience. The first match played includes a training session. The training session was extended to be a full match to let participants learn from their mistakes in endgames (late phases), which are often the most tricky to play. It is also often the case that people learn how to play better from the way they lost. We also excluded Tic-Tac-Chess from the set of games for giving too much advantage to the first player to have a turn. As a consequence, each subject lost their match very quickly in the same way leaving us with no relevant data to work on. Although there exist certain strategies to avoid a quick loss, it is unlikely to be seen by players unfamiliar with the game. Having only one type of game per participant, we modified the players' assignment in such way that we have all combinations of participants playing at least one of the three consecutive matches with the co-operation of the machine. In order to deal with the problem of long experiments, which was mainly caused by the simulation time needed to get meaningful results, we decided to write highly-optimized dedicated interpreters for rules of the chosen games. We were able to reduce the *play clock* just to 9 seconds.

VI. Results

We make the following observations based on numerical outcomes and human players' behavior during the experiments:

- The score between samples is even.
- All games appear to be very demanding for participants.
- There were no wins for Inverted Pentago and for the discarded game of Tic-Tac-Chess. There were 2 wins for Nine Board Tic-Tac-Toe, one with the cooperation and one without.
- The main reason for poor performance as specified by subjects in the questionnaire (and said after the experiment) was the lack of experience playing the given games. The rotations in Pentago were commonly mentioned as something being particularly difficult.
- Despite understanding the role of the program and the advice provided to them, the participants often seemed not to have desire to cooperate. If they had an assumption about which action was the best, they just opted to play it instead of investing time for more simulations.
- The participants seemed to enjoy playing the game but some stress was caused by the level of difficulty and the expectation to win.

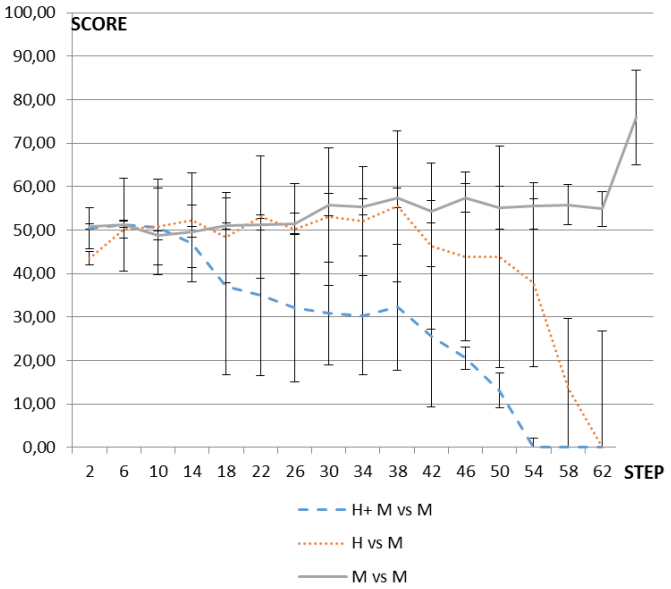

Figure 4. Graph showing the average scores obtained by the cooperating participants (H+M) and not cooperating participants (H) against the machine in Inverted Pentago.

Figure 4 shows the average scores (0 meaning loss and 100 meaning victory) obtained by the cooperating participants (H+M) and non-cooperating participants (H) against the machine in Inverted Pentago whereas Figure 5 shows the same graph for Nine Board Tic-Tac-Toe. Vertical error bars denote 95% confidence intervals. The X axis denotes game step (ply). The error bars overlap so the results cannot be used yet to formally verify the hypothesis. There were not enough participants in the pilot study to make any statistically significant claims. However, the trend so far is that the participants who did not cooperate played slightly better average games. This is reflected in the **H vs M** curve, starting from step 10, being above the **H + M vs M** one. However, both curves eventually meet at a common point which means that the average game results of both samples are even and equal to zero (which means a loss). The same properties are valid in the Nine Board Tic-Tac-Toe game. Because in the pilot studies, the participants rarely and quite chaotically used the cooperation possibilites, a conclusion that cooperation does not help would be an overstatement. The sample is too small, the participants would use the provided statistics when already behind in the game and because the cooperation options were shown only every second move, the machine was not able to help with a coherent line of actions.

Based on things we have learned during the pilot studies, these are the changes we want to make before moving to the final phase of the experiment:

- Each subject should play more than three times, preferably at least five. We have to make room for more learning possibilites, because it turns out that three games are not enough to learn how to play previously uknown games well (e.g., Inverted Pentago and Nine Board Tic-Tac-Toe). With more repeats we can also slightly reduce (though not eliminate) the effect of personal predispositions.

 ISBN: 978-1-61208-382-7

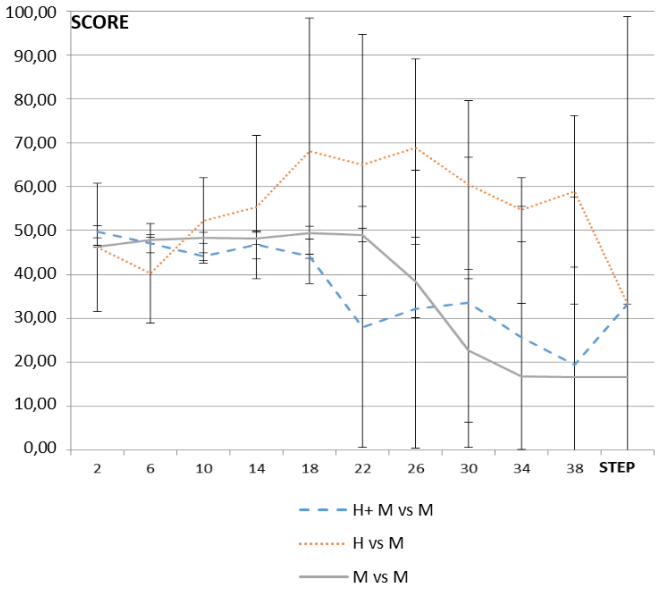

Figure 5. Graph showing the average scores obtained by the cooperating participants (H+M) and not cooperating participants (H) against the machine in Nine Board Tic-Tac-Toe.

- The cooperation options should definately be shown all the time for players playing with the help of a machine.
- We plan to remove actions' priorities and leave only enabling and disabling actions because the latter has more influence on the game tree and should be used more often. We have to make sure that all the participants understand why and when it is beneficial to disable actions.
- We will ask participants to play two games with the machine cooperation in the middle (e.g., the second and the third ones) to be able to observe, in the remaining games, the effects of learning from those games.

VII. Conclusions and Future Work

We analyzed the average outcomes of matches for the H + M vs. M and H vs. M samples of data as well as the average evaluation observed by the machine in every 4 steps of games. We computed 95% confidence intervals using the t-student test. It shows that the number of participants in the pilot study is not enough to make any significant claims regarding the hypothesis. Therefore, we plan to repeat the experiment for a larger sample of participants and with setup slightly modified.

We have presented a complex competitive environment in which human and machine can cooperate during strategic interactions. In general it appeared that subjects not having machine assistance fare slightly better, yet still worse than the machine opponent alone. The reason for this could, most likely, be attributed to the lack of continuous cooperation option (which was shown only at every other move). The other reasons include games' difficulty compounded by the lack of experience and possibly stressful activity of playing a game which is recorded. We believe that the way of introducing the cooperation into MCTS is a good idea, but the design of the experiment should be revisited.

An additional caveat is to maintain a proper balance of the experiment's difficulty. Games cannot be too easy for humans, because the machine cooperation would not be needed and, at the same time, cannot be too difficult to avoid a majority of games ending with a loss (which actually happened). We will restart the experiment with increasing chance to make the human participants learn the games. The participants also need to be clearly told that winning the match is not the exclusive goal of the experiment.

Acknowledgment

M. Świechowski would like to thank the Foundation for Polish Science under International Projects in Intelligent Computing (MPD) and The European Union within the Innovative Economy Operational Programme and European Regional Development Fund. The research was financed by the National Science Centre in Poland, grant number DEC-2012/07/B/ST6/01527. This work was performed while Maciej Świechowski and Jacek Mańdziuk were visiting UNSW Canberra. The ethics approval number granted from the university is A14-09

References

[1] J. Mańdziuk, "Towards Cognitively Plausible Game Playing Systems," IEEE Computational Intelligence Magazine, vol. 6, no. 2, 2011, pp. 38–51.

[2] C. P. Hoc J-M. and H. E., Eds., Expertise and Technology: Cognition & Human-computer Cooperation. Psychology Press, 2013.

[3] M. R. Genesereth, N. Love, and B. Pell, "General Game Playing: Overview of the AAAI Competition," AI Magazine, vol. 26, no. 2, 2005, pp. 62–72. [Online]. Available: http://games.stanford.edu/competition/misc/aaai.pdf

[4] "Stanford General Game Playing," 2014, URL: http://games.stanford.edu/ [accessed: 2014-12-05].

[5] S. Gelly, L. Kocsis, M. Schoenauer, M. Sebag, D. Silver, C. Szepesvári, and O. Teytaud, "The Grand Challenge of Computer Go: Monte Carlo Tree Search and Extensions," Commun. ACM, vol. 55, no. 3, Mar. 2012, pp. 106–113, DOI: 10.1145/2093548.2093574.

[6] O. Syed and A. Syed, Arimaa - A New Game Designed to be Difficult for Computers. Institute for Knowledge and Agent Technology, 2003, vol. 26, no. 2.

[7] K. Waleędzik, J. Mańdziuk, and S. Zadrożny, "Proactive and Reactive Risk-Aware Project Scheduling," in 2nd IEEE Symposium on Computational Intelligence for Human-Like Intelligence (CHILI'2014), Orlando, FL. IEEE Press, 2014, pp. 94–101.

[8] B. Stevens and F. Lewis, Aircraft Control and Simulation. New York: Wiley, 1992.

[9] C. G. Eden, "Robotically assisted surgery," BJU International, vol. 95, no. 6, 2005, pp. 908–909.

[10] M. Buro, "The Evolution of Strong Othello Programs," in Entertainment Computing, ser. The International Federation for Information Processing, R. Nakatsu and J. Hoshino, Eds. Springer US, 2003, vol. 112, pp. 81–88.

[11] J. Mańdziuk, Knowledge-Free and Learning-Based Methods in Intelligenet Game Playing, ser. Studies in Computational Intelligence. Berlin, Heidelberg: Springer-Verlag, 2010, vol. 276.

[12] L. Kocsis and C. Szepesvári, "Bandit based Monte-Carlo planning," in Proceedings of the 17th European conference on Machine Learning, ser. ECML'06. Berlin, Heidelberg: Springer-Verlag, 2006, pp. 282–293.

[13] M. Świechowski and J. Mańdziuk, "Self-adaptation of playing strategies in general game playing," IEEE Transactions on Computational Intelligence and AI in Games, vol. 6, no. 4, Dec 2014, pp. 367–381.

[14] ——, "Fast interpreter for logical reasoning in general game playing," Journal of Logic and Computation, 2014, DOI: 10.1093/logcom/exu058.

 ISBN: 978-1-61208-382-7