An Experiment in Students' Acquisition of Problem Solving Skill from Goal-Oriented Instructions

Matej Guid, Ivan Bratko
Artificial Intelligence Laboratory
Faculty of Computer and Information Science, University of Ljubljana
Ljubljana, Slovenia
{*matej.guid, ivan.bratko*}*@fri.uni-lj.si*

Jana Krivec
Department of Intelligent Systems
Jožef Stefan Institute
Ljubljana, Slovenia
jana.krivec@ijs.si

***Abstract*—In this paper, we investigate experimentally the efficacy of semi-automatically constructed instructions for solving problems that require search. The instructions give advice to the student, in terms of what sub-goals should be attempted next in the process of solving a problem. Our chosen experimental problem domain was the chess endgame of checkmating with bishop and knight, which occasionally presents difficulties even to chess grandmasters. Our subjects, little more than complete beginners, were given the task of learning to win this endgame by studying the instructions. We were interested in two questions: (1) How effective were the goal-oriented instructions as an aid to the student towards mastering this domain? (2) What was the form of the students' "internal representation" of the acquired knowledge? The latter question was studied by our method for automatically identifying so-called procedural chunks from the students' games. Given the simplicity of the chunk detection method, the reconstructed chunks of students' acquired knowledge reflected the goal structure of the instructions amazingly well.**

Keywords-cognitive models; procedural knowledge; procedural chunks; problem solving; perception; memory; chess.

I. Introduction

People operate under constant attacks from lots of external information. If we want to react properly and orient in the world that surrounds us, we need to use this information selectively and effectively. In doing so, we need to incorporate knowledge, stored in long-term memory, which we can recall to short-term memory when needed. Information is usually stored in the memory in the form of chunks - completed units of logically related information clusters that facilitate their retrieval and use, and allow better utilization of a limited working-memory capacity ([1], [2]). Most of the previous research efforts have been devoted to chunks in declarative knowledge, while little is known about the chunks in procedural knowledge. The nature of chunks still remains very elusive, especially with understanding chunks in procedural knowledge. Our attempt is to show the existence of chunks in procedural knowledge, define them, and see how they are incorporated in ones memory.

In this paper, we intend to verify the following claim: people learn procedural knowledge by (sub-consciously) constructing meaningful units of procedural knowledge. To emphasize the difference with respect to well-known chunks in declarative knowledge, we will refer to these meaningful units of procedural knowledge as *procedural chunks*. In chess, for example, a procedural chunk is a sequence of chess moves that all together belong to a chess concept, and are therefore memorized by a player as a whole. We also intend to demonstrate that compared to the traditional approach that usually provides declarative knowledge, using the approach that emphasizes developing students' *procedural knowledge* can greatly improve their skills.

In our previous study [3], we indicated the existence of procedural chunks by using *reconstruction* of chess games. In the experiments presented in this paper, however, the participating students – chess beginners – were actually *playing* chess against a computer. The identification of chunks was performed by combining existing methods for chunk recognition, slightly adapted for detection of chunks in procedural knowledge. The times spent for execution of individual moves played by the students served as the most valuable information for determining chunks, similarly as in several studies related to chunks in declarative knowledge (see, for example: [4], [5], [6], and [7]).

As a case study, we considered teaching students how to play a difficult KBNK (king, bishop, and knight vs. king) chess endgame, by providing the students with goal-oriented procedural knowledge in the form of a manual (textbook instructions) supported with example games. We have chosen chess for our research domain due to the following reasons:

- because of its complexity, clearly defined rules, built-in scales for measuring chess players' knowledge (*Elo* rating system), and generalization to other areas often made possible;
- chess endgames with a few pieces have an additional benefit as an experimental domain: availability of *perfect information* in the form of chess tablebases [8], enabling optimal play by a computer and – more importantly – easier tracking of students' learning progress.

The paper is organized as follows. In Section II, we

 ISBN: 978-1-61208-218-9

Table I
THE 11 GOALS PRESENTED IN THE TEXTBOOK INSTRUCTIONS.

1	Deliver checkmate.
2	Prepare the knight for checkmate.
3	Restrain black to a minimal area beside the right corner.
4	Build a barrier and squeeze black king's area.
5	Approach black from the center.
6	Block the way to the wrong corner.
7	Push black towards the right corner.
8	Push black towards the edge.
9	Approach with the king.
10	Bring the knight closer.
11	Keep the kings close.

describe in more detail the domain of KBNK, the teaching materials used in the experiments, and the experimental procedure. Section III introduces the methods used for identification of procedural chunks. Section IV presents the results of our experiments, particularly in terms of students' learning progress during the playout games and procedural chunks identified. We conclude the paper and point out directions for further work in Section V.

II. EXPERIMENTAL SETUP

A. Domain Description

Our domain of choice was the KBNK (king, bishop, and knight vs. a lone king) chess endgame, which is regarded as the most difficult of the elementary chess endgames. The stronger side can always checkmate the opponent, but even optimal play may take as many as 33 moves. Several chess books give the general strategy for playing this endgame as follows. Since checkmate can only be forced in the corner of the same color as the squares on which the bishop moves, the opponent will try to stay first in the center of the board, and then retreat in the wrong-colored corner. The checkmating process can be divided into three phases: (1) driving the opposing king to the edge of the board, (2) forcing the king to the appropriate corner, and (3) delivering checkmate. However, only knowing this basic strategy hardly suffices for anyone to checkmate the opponent effectively. There are many recorded cases when strong players, including grandmasters, failed to win this endgame.

B. Textbook Instructions and Example Games

In our experiments, we used *"textbook instructions"* in the form of goals for delivering checkmate from any given KBNK position. These instructions were semi-automatically derived using an interactive procedure between a chess teacher (a FIDE master of chess) and the computer [9], using argument-based machine learning (ABML) approach combined with an algorithm for semi-automated domain conceptualization of procedural knowledge [10].

The textbook instructions consist of 11 goals listed in Table I (see [11] for details). The chess-player is instructed to always try to execute the highest *achievable* goal. The goals are listed in order of preference, goal 1 being the most preferred. In the textbook instructions presented to the students, these goals were supplemented with detailed explanations and illustrative diagrams.

The students also had *example games* demonstrating the checkmating procedures at their disposal. The example games were supplemented with the goals given as instructions, in terms of which is the preferable (and also achievable) goal in a particular position. An instruction is given each time the previous suggested goal was accomplished.

Figure 1 is taken from the textbook instructions. It demonstrates the execution of one of the goals. This goal is supplemented by the following explanation in the textbook instructions: *"When the defender's king is already pushed to the edge of the board, the attacker's task is to constrain as much as possible the defending king's way to the wrong-colored corner. At the same time, the attacker should keep restraining the enemy king to the edge of the board."*

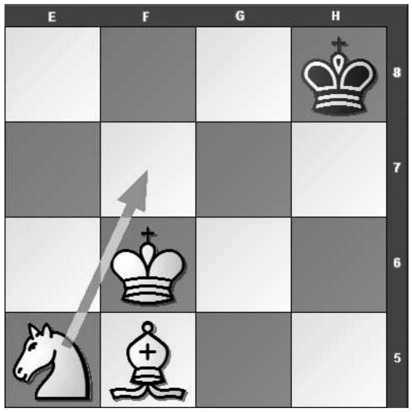
Both the textbook instructions and the example games that were used in our experiments are available in a web appendix at [11].

C. Experimental Procedure

Three students – chess beginners of slightly different levels – were involved in the experiment. Student 1 is a *Class B* player, Student 2 is a *Class C* player (*i.e.*, slightly weaker than Student 1 in terms of his chess strength), and Student 3 is a *Class D* player (a complete beginner, however, well familiar with the rules of chess). According to the ELO rating scales, the *Class B*, *Class C*, and *Class D* represent ELO rating ranges of 1600-1800, 1400-1600, and 1200-1400, respectively [12].

Our assumption was that none of the three students possessed procedural knowledge sufficient for successfully delivering checkmate in the KBNK endgame at the beginning of the experiment. In order to verify the correctness of this assumption, the participants were first asked to try to deliver checkmate in three games against the computer. The computer was defending "optimally", *i.e.*, always randomly choosing among moves with the longest distance to mate (using chess tablebases). The time limit was 10 minutes per game. Each game started from a different starting position: mate-in-30-moves or more assuming optimal play. The moves and times spent for each move were recorded automatically using *Fritz 13* chess software by *Chessbase*.

None of the students were able to deliver checkmate at this first stage of the experiment. While the students were occasionally able to force the defending king towards the edge of the board, it turned out that the most difficult part was to block the way to the wrong corner and push the king towards the right corner – the corner where checkmate



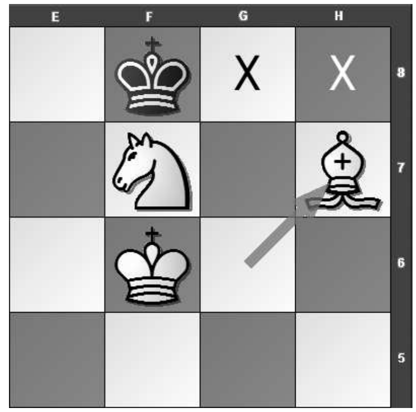


Figure 1. Demonstration of the execution of the goal 6: *"Block the way to the wrong corner."* In the position on the left, white pieces lure the defending king out of the wrong corner (note that a light squared bishop cannot deliver mate in a dark square corner): 1.Ne5-f7+ Kh8-g8 2.Bf5-g6 Kg8-f8 (this is the only available square to the black king, since h8 is attacked by the knight) 3.Bg6-h7! The last move in this sequence takes under control square g8, and sets up the blockade one square farther from the wrong corner.

can be delivered. They had no idea how to establish the appropriate barrier around the right corner in order to deny the defending king an escape from there.

Then the students were given the textbook instructions and access to the example games that contain instructions as commentary to particular move sequences. The students were reading the instructions and observing the example games until they felt they are ready to challenge the computer once again. None of them spent more than 30 minutes for reading the instructions and observing the example games at this point.

In the second stage of the experiment, the students were again trying to checkmate the optimally defending computer. The time limit per game was again set to 10 minutes. The starting position of each game was chosen randomly in such way that each of the four pieces occupied one corner square, with the white bishop always being placed on a white square. Similarly as in the first stage, the moves and times spent for each move were recorded automatically. The textbook instructions and example games were not accessible to the students at this stage. If a game ended in a draw, the student was again granted the access to the textbook instructions and example games for up to ten minutes before starting a new game. This procedure was then being repeated until the first win was recorded. In order to verify the quality of the learned knowledge, the students were asked to challenge the computer again, this time with the white bishop being placed on a black square (*i.e.*, the opposite square than in their all earlier games).

Since it is our conjecture that people learn procedural knowledge by using procedural chunks, we attempted to verify whether the students learned any procedural chunks during the process of learning how to deliver checkmate in the difficult KBNK chess endgame. Thus, we needed some methods to identify procedural chunks. These methods are presented in the following section.

III. Identification of Procedural Chunks

The chunks were identified on the basis of a hypothesis stated by Chase and Simon [4], which states that longer time interval during the reconstruction of a meaningful unit of material (*i.e.*, the material about which we have some relevant knowledge) reveals the recall of a new structure/chunk from the long-term memory. In our experiments, the students did not deal with *reconstructions* of chess positions (as in Chase and Simon [4], and Bratko *et* al. [5]) or particular move sequences (as in Krivec *et* al. [3]), but were actually *playing* against the computer.

The relative time of particular person was considered instead of an absolute 2-second limit used by Chase and Simon. We defined a *longer time interval* in the following way. Times of each participant were normalized using two different methods: (1) we calculated the percentage of time used for a certain move with regard to the time spent for a whole game, and (2) by converting them into z values. The quantity z represents the distance between the raw score and the population mean in units of the standard deviation. The value of z is negative when the raw score is below the mean, and positive when above.

All relevant symmetries were taken into account when processing individual moves in the identification of procedural chunks. For example, the move Bh7 (the bishop moves to h7) in the sequence 1.Nf7+ Kg8 2.Bg6 Kf8 3.Bh7 (see Fig. 1) is equivalent to the move Ba2 (the bishop moves to a2) when the enemy king is in the opposite "wrong" corner of the board and thus the sequence is actually 1.Nc2+ Kb1 2.Bb3 Kc1 3.Ba2.

For all moves in each game, an average value and standard deviation of normalized time medians was calculated. All the moves that exceeded the boundary of the average value plus one standard deviation were considered as a "long time interval" and as such candidates for the beginning of a new procedural chunk. If such a candidate appeared in the majority of the games, it was considered as the beginning of a procedural chunk.

Validity of this method of chunk identification was statistically verified as follows. We randomly generated input data, having the same number of chunks as in the original data. After this, we calculated the co-occurrence of the chunks beginning. We repeated this 100 times. Then we calculated 95-percentile of the sum of co-occurred chunk beginning. We compared the possibility that the original results (*i.e.*, the beginning of the chunks) are the result of a chance. That is, if the co-occurrences of the beginning of the chunks in the results of the real play represented more than 95% co-occurrences in randomly generated results, it was considered that it is highly improbable that the co-occurrences happened by chance only.

IV. Results

A. Deviation from Optimal Play

In order to track the progress by the students as they were more and more exposed to the textbook instructions and example games between (but not during) their trial games against the computer, we observed the deviations of the moves they played from an optimal play. Chess tablebases served us for this purpose, providing the number of moves required to deliver checkmate from any given position assuming optimal play by both players. Although the goal of the conceptualized procedural knowledge included in the textbook instructions is not to teach students how to play "optimally," but merely to enable them to achieve a step-by-step progress towards the ultimate goal – delivering checkmate – deviation from optimal play was chosen as a sensible measure of quality of their play.

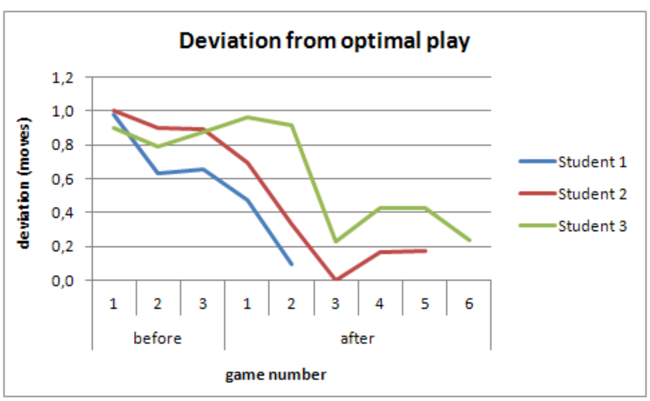


Figure 2. The average students' deviations from optimal play in the games of both stages of the experiment, *i.e.*, before and after they were first given access to the textbook instructions and the example games.

The results of these observations are presented in Figure 2. They demonstrate the average students' deviations from optimal play in all their games until they successfully checkmated the optimally-defending computer for the first time (this game is included in the graph). The lower deviation from optimal play means a better performance of the player.

The deviations from optimal play for a particular move played by the player were calculated as follows:

$$DTM_{dev} = DTM_{played} - DTM_{optimal} \tag{1}$$

where "DTM" represents the distance to mate in moves (rather than *plies* or half-moves). $DTM_{optimal}$ is the value of DTM of an optimal move, and DTM_{played} is the value of DTM of the move played by the student. DTM_{dev} (deviation from optimal play) value of 0 therefore means that the student played a given move optimally, the value of 1 means that the distance to mate remained the same after the execution of the moves by the player and (optimally defending) computer, and DTM_{dev} values higher than 1 for a given move mean that on the next player's move DTM even increased.

The results clearly suggest that the speed of achieving mastery of this difficult chess endgame is correlated with the chess-playing strength. Student 1, the strongest of the three players, not only successfully checkmated the opponent already in the second game after having studied the teaching materials – he also made less inferior moves earlier than the other two students. The other two students checkmated in games 5 and 6, respectively. Once they achieved the win the students had no problems at all achieving it again, even with the white bishop being placed on the opposite square color than in all previous games. The results also demonstrate a vast progress of all three students after they got acquainted with (sub)goals and procedures presented in the teaching materials.

It is particularly interesting that the second student played optimally(!) in his third game of the second stage of the experiment. He actually played 22 optimal moves in a row – an achievement that a chess grandmaster could be proud of. Moreover, it happened in less than an hour after he was first given access to the textbook instructions and example games. This result would be very hard or even impossible to achieve without an effective way of memorizing particular (sub)goals or concepts of procedural knowledge required in order to master this difficult endgame.

B. Procedural Chunks Identified

We identified procedural chunks separately for each of the three phases mentioned in II-A (repeated here for clarity):

Phase I:
: Driving the opposing king to the edge of the board (the black king aims towards the "wrong" corner).

Phase II:
: Forcing the king from the "wrong" corner to the "right" corner (where checkmate can be delivered).

Phase III:
Delivering checkmate (once a “barrier” is set up).

The students actually tended to spend more time on moves that indicate a borderline between two phases. Based on this observation, we determined the start of Phase II just before a waiting move with the bishop before blocking the way of the black king to the wrong corner (see the move 2.Bf5-g6 in Figure 1), and the start of Phase III just after the barrier is established. Tables II, III, and IV show identified procedural chunks in the games of the second stage of the experiment, *i.e.*, after the teaching materials were presented to the students, and frequencies of their occurrence in the playouts. The chunks are described by meaningful descriptions.

Table II
PROCEDURAL CHUNKS DETECTED IN PHASE I.

#	Chunk Description	Freq.
1	Finding the path for the knight to attack the corner square.	14
2	Bringing the bishop into the game.	9
3	Bringing the knight into the game.	8
4	Using the king to push the enemy king towards the edge.	6

Let us take a look how the chunks identified in Phase 1 (Table II) are associated with the goals (see Table I) given in the textbook instructions. Chunk #1 represents preparation for execution of Goal 6 (“Block the way to the wrong corner.”). To achieve this goal, White must first bring the knight to the square from which it attacks the wrong corner square (see Figure 1). While it may take a fraction of a second for a master to spot the path with the knight to a given square, the chess beginners involved in our experiments typically paused for a while before executing a sequence of moves that brought the knight to a desired square. Chunks #2 and #4 can be associated with Goal 8 (“Push black towards the edge.”), and the latter is also associated with Goal 5 (“Approach black from the center.”), Goal 9 (“Approach with the king.”) and Goal 11 (“Keep the kings close.”). Chunk #3 is associated with Goal 10 (“Bring the knight closer.”).

Table III
PROCEDURAL CHUNKS DETECTED IN PHASE II.

#	Chunk	Freq.
1	A waiting move with the bishop before blocking the way.	7
2	The knight keeps the enemy king on the edge.	7
3	Building a barrier with the knight and the bishop.	6
4	The king keeps the enemy king on the edge.	5

In Phase II (Table III), Chunk #1 represents a part of an execution of Goal 6 (“Block the way to the wrong corner.” – see Figure 1). Chunks #2 and #4 are associated with Goal 7 (“Push black towards the right corner.”) in which White also needs to keep the enemy king at the edge of the chessboard – note that two chunks were learned for an execution of a single goal, which incidentally also turned out to be the most difficult of the goals to be learned (judging from the times spent for its execution). Chunk #3 represents an execution of Goal 4 (“Build a barrier and squeeze black king’s area.”).

Note that Phase II is the most difficult part of the KBNK endgame, since a precise sequence of moves must be executed and a single mistake may have a fatal consequence: the black king may escape to the opposite wrong corner before the barrier is established. The students learned this sequence of moves by remembering meaningful intermediate subgoals, as suggested by the identified procedural chunks.

Table IV
PROCEDURAL CHUNKS DETECTED IN PHASE III.

#	Chunk	Freq.
1	Squeezing the enemy king into the right corner (start).	8
2	Manoeuvring the bishop to set up the “minimal area”.	7
3	Squeezing the enemy king into the right corner (continue).	6
4	Calculating the checkmate procedure.	6
5	Approaching with the knight for delivering checkmate.	5

In Phase III, Chunks #1 and #3 are associated with Goal 4 (“Build a barrier and squeeze black king’s area.”), Chunk #2 closely resembles Goal 3 (“Restrain black to a minimal area beside the right corner.”), and Chunk #5 is associated with Goal 2 (“Prepare the knight for checkmate.”). Finally, Chunk #4 is associated with the highest goal in the hierarchy, Goal 1 (“Deliver checkmate.”).

All the described chunks were detected automatically by using the methods described in Section III. As it can be seen from the descriptions above, they cover all 11 goals presented in the textbook instructions.

Figure 3 shows the progress of the students by means of the different goals executed in their playout games. The results closely resemble the ones demonstrated in Figure 2: Student 1, the strongest of all three students in terms of chess strength, was the first to master all 11 goals given in the textbook instructions, and all three players demonstrated in their games a progress towards the final goal – successfully checkmating the opponent’s king.

In the figure, it can be seen that some goals were only partially executed. This happened on occasions where a particular goal (as presented in Table I) consisted of more than one procedural chunk (as given in Tables II, III, and IV). In the first stage of the experiment, the students were merely able to execute the most intuitive goals. One such goal is Goal 5 (“Approach black from the center.”), which is very intuitive for a human – it is useful for the white king to approach the black king from the central part of the board.

 ISBN: 978-1-61208-218-9

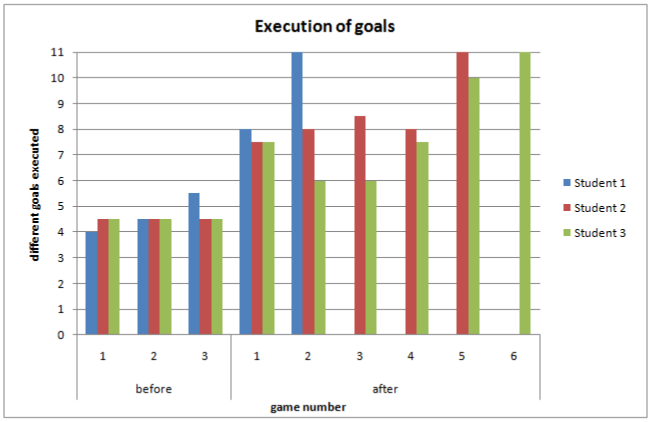


Figure 3. Number of instruction goals the students successfully executed in the playout games against the computer.

V. Conclusion and Future Work

In this paper, we studied how procedural knowledge for solving problems in a domain is learned operationally by students from "textbook instructions" (a manual). In our study, textbook instructions had the form of if-then rules that specify goals to be achieved in solving a particular problem in the domain, ordered according to the degree of ambition of the goals which roughly corresponds to the time order of subtasks.

In our experimental study, we measured the students' progress in assimilating this procedural knowledge by observing their skill at the given task (checkmate in the KBNK endgame). Roughly, our subjects were just slightly better than complete chess beginners.

We were interested in two questions:

1) How useful are the goal-oriented instructions to the student as a help towards mastering the play in this domain?
2) What was the form of the student's "internal representation" of the acquired knowledge? Our hypothesis was that it was a goal-based representation with a similar goal structure as in the instructions.

Our hypothesis regarding the students' internal representation of the acquired skill was experimentally tested by means of the identification of *procedural chunks*, using a new method for chunk identification from games played. Roughly, a procedural chunk is a sequence of chess moves that all together belong to a chess concept, and are therefore memorized by a player as a whole. The chunk identification method is based on observing the times between consecutive moves in a game played. Longer time intervals between moves indicate boundaries between procedural chunks.

Our findings concerning the two questions above were as follows:

- Finding regarding question (1): The students learned the skill operationally in up to an hour's time of studying the instructions and testing their skill in actual problem solving (playing the endgame). To put this result in perspective, it should be remembered that even chess grandmasters often have serious difficulties in playing this endgame, occasionally failing to deliver mate at all.
- Finding regarding question (2): Automatically detected procedural chunks in the students' games corresponded almost perfectly to the goal-oriented rules in the textbook instructions. We also measured the dynamics of acquiring these chunks during the learning time, that is the number of different chunks appearing in consecutive games played by a student.

As future work, we intend to strengthen these experimental results by scaling up the experiments in terms of the number of subjects, and by extending the experiments to other domains (other chess tasks and domains other than chess).

References

[1] A. de Groot, *Thought and Choice in Chess*. Mouton, The Hague, 1965.

[2] F. Gobet, P. Lane, S. Croker, P. Cheng, G. Jones, I. Oliver, and J. Pine, "Chunking mechanisms in human learning," *Trends in Cognitive Sciences*, vol. 5, no. 6, pp. 236–243, 2001.

[3] J. Krivec, M. Guid, and I. Bratko, "Identification and characteristic descriptions of procedural chunks," in *Proceedings of the 2009 Computation World: Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns*, ser. COMPUTATIONWORLD '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 448–453.

[4] W. Chase and H. Simon, "Perception in chess," *Cognitive Psychology*, vol. 4, pp. 55–81, 1973.

[5] I. Bratko, P. Tancig, and S. Tancig, "Detection of positional patterns in chess," *International Computer Chess Association Journal*, vol. 7, no. 2, pp. 63–73, 1984.

[6] V. Ferrari and E. Didierjean, Andre an Marmeche, "Dynamic perception in chess," *The Quarterly Journal of Experimental Psychology*, vol. 59, no. 2, pp. 397–410, 2006.

[7] A. Linhares and A. E. Freitas, "Questioning Chase and Simon's (1973) "Perception in Chess": The "experience recognition" hypothesis," *New Ideas in Psychology*, vol. 28, no. 1, pp. 64–78, 2010.

[8] K. Thompson, "Retrograde analysis of certain endgames," *International Computer Chess Association Journal*, vol. 9, no. 3, pp. 131–139, 1986.

[9] M. Guid, M. Možina, A. Sadikov, and I. Bratko, "Deriving concepts and strategies from chess tablebases," in *ACG*, 2009, pp. 195–207.

[10] M. Možina, M. Guid, A. Sadikov, V. Groznik, and I. Bratko, "Goal-oriented conceptualization of procedural knowledge," in *Lecture Notes in Computer Science*, vol. 7315, 2012, p. to appear.

[11] M. Guid, "KBNK," http://www.ailab.si/matej/KBNK/, 2012.

[12] A. Elo, *The Rating of Chessplayers, Past and Present*. Arco, 1978.