An Upper Bound on the Complexity of Tablut

Andrea Galassi

Department of Computer Science and Engineering (DISI)
University of Bologna, Bologna, Italy
a.galassi@unibo.it

Abstract. Tablut is a a complete knowledge, deterministic, and asymmetric board game, which has not been solved nor properly studied yet. In this work, firstly its rules and its characteristics are presented, then a study on its complexity is reported. Dividing the state-space of the game into subspaces according to specific conditions, an upper bound to its complexity is eventually found. Since the upper bound seems to be comparable to the one found for Draughts, the open challenge of solving this game seems to require a great computational effort.

Keywords: Board game · game-playing · game complexity.

1 Introduction

According to [1], a game can be solved according to several level:

- ultra-weakly solved: the game-theoretic value for the initial position(s) is known
- weakly solved: a strategy is known to obtain the game-theoretic value of the game
- strongly solved: for any state reachable from the initial position(s), a strategy to obtain the game-theoretic value of that state is known

2 Rules and Properties

Many different variants of Tablut rules do exists. In this work, the rules of the game are described mostly following the work of Ashton [2].

2.1 Terminology, Material, and Setup

The game is played by two players on a square board of 9×9 cells depicted in Figure 1. The central cell is called royal citadel, or castle, or throne. On each board side, there are 4 groups of 4 cells arranges with a t-shape that are called citadels or camps. For a better comprehension, the former will be addressed as *castle*, the latter as *camps*, while the term *citadel* will be used with the meaning "either of them". Any non-camp and non-corner cell along the edge of the board will be addressed as *escape*, the reason will be clear in the next Subsection.

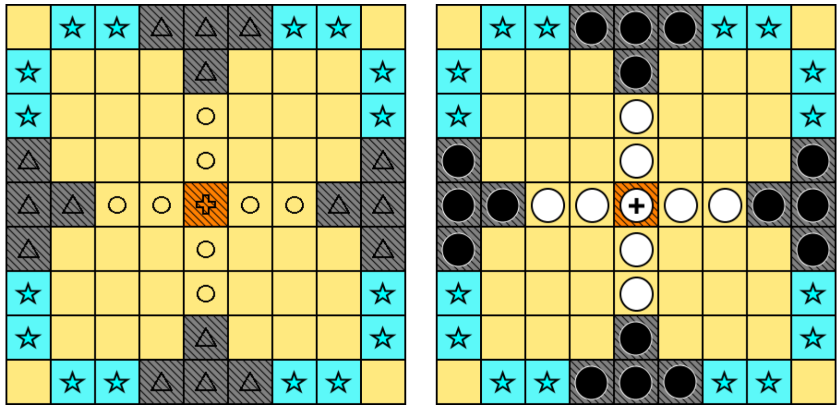

Fig. 1. The Tablut empty board (left) and the initial setup (right). The *castle* cell is represented with the orange color and a cross. The *camp* cells are represented in grey with a triangle. The *escape* cells are in blue, with a star. The white soldiers starting cell are marked with a circle. The *king* checker is marked with a black cross.

Two cells are considered adjacent if and only if they are aligned horizontally or vertically and they share an edge. The term *side* of a checker will be used to indicate any cell which is adjacent to the cell where the checker is placed.

One players moves the white checkers, which represent the defenders or Swedes, while the other moves the black checkers, which represent the attackers or Muscovites. There are 16 black checkers and 9 white checkers.[1] One of the white checker is the *king* and it is marked. Any non-king piece will be addressed as *soldier*. The checkers are placed as in Figure 1, with the king in the castle, the black soldiers in the camps, and the remaining white soldiers aligned by 2 on each side of the king.

2.2 Endgame

The game ends when one of the following conditions is met:

1. The king reaches one of the escape cells. This results in the winning of the white player.
2. The king is captured. This results in the winning of the black player.
3. The games reaches the same state twice. This results in a draw.[2]
4. When a player has no possible moves. This results in the winning of the other player. This is an addition with respect to [2].

[1] In this work, the terms *checker*, *stone*, and *piece* will be used as synonyms.

[2] This rule is added to avoid endless sequence of repeated moves and to simplify the game with respect to [2].

2.3 Movement and Capture

The two players alternate their turns, which consist of a single movement of a checker. The white player starts first. A checker can be moved along a single straight line, horizontally or vertically, by any number of cells. The movement must not pass over nor end into a cell which is occupied by another checker. The same holds for cells which are part of a citadel, unless the checker starts its movement from a cell of the same citadel. This implies that the only checkers which may move inside a citadel are the black checkers, but only in their starting citadel and only if they have not ever left it.

To make a capture, a player must move one of its own piece so to surround an adversary piece. A checker is considered surrounded according to different criteria:

- When the king is in the castle, it is considered surrounded if there are enemy pieces on all four of its sides.
- When the king is adjacent to the castle, it is considered surrounded if there are three enemy pieces on three of its side and the castle on the fourth.
- When a soldier is adjacent to a citadel, or when the king is adjacent to a camp, it is considered surrounded if there is an enemy piece on the opposite side with respect to the citadel/camp.
- In any other case, any piece is considered surrounded if there are two enemy pieces on two opposite sides of its cell, so that the three pieces are aligned horizontally or vertically.

Capture can happen only in a active way, which means that if a player move his/her piece so to make it surrounded, the piece is not captured. It is possible to capture multiple pieces (up to 3) with a single move, if that move allows to surround more than one piece.

2.4 Properties

In any moment of the game the two players know everything about the state of the game. Given a state of the game, it is also known how any possible move will change it, since there are no random components. Finally, the two players have different starting position, checkers, and goal. Therefore this is a complete knowledge, deterministic, and asymmetric game.

The board is symmetric along 4 axes, so a single board configuration can have up to 7 symmetric configurations.

3 Computing an Upper Bound for the State Space Size

To compute the complexity of the game it is useful to divide the state-space into subspaces and compute their size separately. Firstly two subspaces are distinguished: the space without endgame states and the space with endgame states. Then, additional subspaces are considered according to specific conditions. The symmetries of the problem will not be taken into account.

3.1 State Space without Endgames

The relevance of computing the complexity without the endgame positions is due to the fact that in some cases, according to how the solver is modeled, it could be sufficient to look at the possible move to know the the game is ended, without computing the new state, reducing the computational footprint. A move that declares an escape cell as destination for the king is an example of this.

Two possible endgame are not taken into account in this work: the endgame by draw and neither the endgame given by the impossibility for a player to move if he has still checkers on the board. The former are naturally included in the subspace without endgame state. The latter are not investigated in this work.

it is possible to make the following considerations:

- The king has to be on the board, it can not be in a corner cell, and it can not be on escape cells, otherwise it would mean that the one of the players has already won (respectively, the black or the white). Therefore it can be in any of the $7 \times 7 = 49$ cells in the center of the board. Excluding the camps, there are only 45 possible cells, one of which is the castle.
- There has to be at least one black soldier on the board. Otherwise, it would mean that the white player has won. Indeed, the black player could lose its last checker(s) only due to a capture by the white player. The next turn would be of the black player, who would not be able to move any checker, and therefore would lose.
- The castle can host only the king.

A naive initial upper bound can be computed considering the values that any cell can assume: 2 for the castle (with and without king), 2 for the 16 camps cells (with and without the black soldier), 3 for the 20 edge cells (with a black soldier, a white soldier or neither), 4 for any other of the remaining 44 cells (with a black soldier, a white soldier, the king, empty). This would result in 10^{41} possible states as given by Equation 1.

$$UB_{naive} = 2 \cdot 2^{16} \cdot 3^{20} \cdot 4^{44} \approx 10^{41} \tag{1}$$

To reduce the upper bound, it is possible to split the state space according to particular conditions. It is possible indeed to differentiate two scenarios: the king in the castle and the king outside of the castle.
In the first scenario, each possible configuration has $1 \leq b \leq 16$ black pieces, $0 \leq w \leq 8$ white soldiers, and $e = 80 - w - b$ empty cells. It can therefore be modeled as a permutation with repetitions as in Equation 2.

$$P_{80}^{(b,w,e)} = \frac{80!}{b! \cdot w! \cdot e!} \tag{2}$$

The second scenario can be defined similarly, but the king would occupy one of the 44 cells (the castle is excluded). The remaining cells are therefore 79, so the number of states (for each combination of b, w, and e value) is $44 \cdot P_{79}^{(b,w,e)}$.

An upper bound to the number of possible states without endgame positions is therefore 6×10^{27}, as given summing these two scenarios for any configuration of b, w, and e, as in Equation 3.

$$UB_{no\_end} = \sum_{b=1}^{16} \sum_{w=0}^{8} P_{80}^{(b,w,80-b-w)} + 44 \cdot \sum_{b=1}^{16} \sum_{w=0}^{8} P_{79}^{(b,w,79-b-w)} \approx 6.1 \times 10^{27} \quad (3)$$

It is now considered an additional characteristics: the white checkers cannot occupy a camp. Therefore, it is useful to treat camps and non-camps cells separately. A new variable c is now defined as the number of black checkers inside any camp. The number of black checker outside the camp is then $b - c$. The number of possible configuration of black checkers inside the camps is now given by the permutation $P_{16}^{(c,16-c)}$, with $0 \leq c \leq b$. The number of possible configuration on the non-camps cells can be computed as previously (considering 16 cells less). According to this new consideration, an upper bound on the size of the two discussed scenarios is then computed as:

$$UB_{no\_end}^{castle} = \sum_{b=1}^{16} \sum_{w=0}^{8} \sum_{c=0}^{b} P_{16}^{(c,16-c)} \cdot P_{64}^{(b-c,w,64-b-w+c)} \approx 3 \times 10^{25} \quad (4)$$

$$UB_{no\_end}^{no\_castle} = 44 \cdot \sum_{b=1}^{16} \sum_{w=0}^{8} \sum_{c=0}^{b} P_{16}^{(c,16-c)} \cdot P_{63}^{(b-c,w,63-b-w+c)} \approx 9.2 \times 10^{26} \quad (5)$$

Their sum, (Equation 6) provides an upper bound $UB'_{no\_end} \approx 9.5 \times 10^{26}$

$$UB'_{no\_end} = UB_{no\_end}^{castle} + UB_{no\_end}^{no\_castle} \approx 9.5 \times 10^{26} \quad (6)$$

Another fact that could be taken into account is that if three black pieces are in the ends of a camp, also the fourth black piece has to be in that camp, but this is will not be considered in this work.

3.2 State Space of Endgames

To make a proper comparison with other board games, an upper bound has to be computed considering also the endgame positions. This is done considering all the possible endgame scenario and separately computing the upper bound given by each scenario.

Excluding the case when the king is captured, a first endgame scenario is obtained considering the states when all the black soldier have been captured. As done previously, it is possible to divide the scenarios where the king is inside or outside the castle.

$$UB_{\alpha} = \sum_{w=0}^{8} P_{64}^{(w,64-w)} + 44 \cdot P_{63}^{(w,63-w)} \approx 2.0 \times 10^{11} \quad (7)$$

Another endgame scenario is given by the successful escape of the king to one of the 16 cells. The cell adjacent to the cell used to escape has to be empty, therefore only 62 cells can host black and white soldiers. In these cases, the lower bound on b increases once again to 1, since without black soldiers the game would have already been finished.[3] This contributes to the final upper bound with a term

$$UB_{\beta} = 16 \cdot \sum_{b=1}^{16} \sum_{w=0}^{8} \sum_{c=0}^{b} P_{16}^{(c,16-c)} \cdot P_{62}^{(b-c,w,62-b-w+c)} \approx 2.3 \times 10^{26} \tag{8}$$

The last endgame condition is given by the capture of the king, which could occur in many different cases. In the following scenarios the variable b will assume the meaning of number of black soldier on the board which are not involved in the capture of the king.

1. When the king is inside of the castle, 4 black soldiers are necessary to capture it, therefore those checkers and those cells are determined. The number of cells which can host any other checker is therefore reduced from 64 to 60, while the highest possible value of b is 12. The contribution of this case is:

$$UB_{\gamma} = \sum_{b=0}^{12} \sum_{w=0}^{8} \sum_{c=0}^{b} P_{16}^{(c,16-c)} \cdot P_{60}^{(b-c,w,60-b-w+c)} \approx 2.8 \times 10^{22} \tag{9}$$

2. When the king is adjacent to the castle (4 possible positions), 3 black soldiers are necessary to capture it. As for the previous case, the cells surrounding the king are for sure occupied by 3 black checkers. This reducing the number of cells to consider to 60 and the highest value of b to 13. The upper bound is therefore computed as follows:

$$UB_{\delta} = 4 \cdot \sum_{b=0}^{13} \sum_{w=0}^{8} \sum_{c=0}^{b} P_{16}^{(c,16-c)} \cdot P_{60}^{(b-c,w,60-b-w+c)} \approx 5.1 \times 10^{23} \tag{10}$$

3. When the king is adjacent to a camp (12 possible positions), 1 black soldiers is sufficient to capture it. In 8 case, the capturing checker can be in 2 positions, while in 4 it has to be in a specific position, therefore the possible scenarios are 20. The highest value of b is 15 and the number of cells to consider are 62. The possible configurations for this scenario are:

$$UB_{\epsilon} = 20 \cdot \sum_{b=0}^{15} \sum_{w=0}^{8} \sum_{c=0}^{b} P_{16}^{(c,16-c)} \cdot P_{62}^{(b-c,w,62-b-w+c)} \approx 8.0 \times 10^{25} \tag{11}$$

[3] It is possible for the king to make a capture in the same movement in which it reaches the escape, but this does not change the number of cases. On the opposite, this consideration would lower the number of states.

4. Finally, when the king is captured in any other position (28 possible cells), 2 black soldiers are necessary. For any position, there are 2 possible configurations of black soldiers, so the cases are 56. The highest number of black soldier on the board not involved in the capture is 14, and the cells to be taken into account are 61. So the possible configurations for this scenario are :

$$UB_\zeta = 56 \cdot \sum_{b=0}^{14} \sum_{w=0}^{8} \sum_{c=0}^{b} P_{16}^{(c,16-c)} \cdot P_{61}^{(b-c,w,61-b-w+c)} \approx 1.6 \times 10^{26} \tag{12}$$

Taking all these scenarios into account as in Equation 13, the upper bound on the number of endgame states is $UB_{end} \approx 4.6 \times 10^{26}$.

$$UB_{end} = UB_\alpha + UB_\beta + UB_\gamma + UB_\delta + UB_\epsilon + UB_\zeta \approx 4.6 \times 10^{26} \tag{13}$$

3.3 Total State Space

Summing the contribution of Equation 6 and Equation 13, the final upper bound on the state space is $UB_{end} \approx 1.4 \times 10^{27}$:

$$UB_{end} = UB'_{no_e nd} + UB_{end} \approx 1.4 \times 10^{27} \tag{14}$$

Taking into account possible endgames given by impossibility to move, or some other properties, would lead to an improvement of this estimation. For now, it is fair to assert that would be difficult to find even a weak solution for this game, since games with a similar state space are still unsolved, as illustrated in Table 1.

Table 1. Comparison of upper bound on state-space complexity in different board games

	Tablut	Nine Men's Morris	English Draughts	International Draughts	Othello	Chess	Go
UB	1.4×10^{27}	3×10^{11}	5×10^{20}	10^{30}	10^{28}	10^{43} or 10^{50}	2×10^{170}
Solution	No	Strong	Weak	No	No	No	No
Source		[3]	[4]	[1]	[1]	[1]	[5]

3.4 Conclusion and Discussion

For the first time, an upper bound for the state-space complexity of the board game Tablut has been computed. The game seems to be comparable with the game of Draughts, therefore finding the strong solution of Tablut would probably to require a great computational effort.

Nonetheless, many characteristics of the game have not been taken into account, therefore it is still possible to reduce this upper bound and, since it is not computed any lower bound, it is not possible to exclude a real complexity greatly inferior with respect to this upper bound.

Due to the separation of the sub-spaces of the game, it is possible to know which are the scenarios which contributes the most to the computation of this upper bound. For what concerns the non-endgame subspace, is the case of the states where the king is not in the castle. Among the end-game scenarios, the successful escape of the king is the one with the higher upper bound.

References

1. Allis, L.V.: Searching for solutions in games and artificial intelligence. Ph.D. thesis, Maastricht University (1994)
2. Ashton, J.C.: Linnaeus's game of tablut and its relationship to the ancient viking game hnefatafl. The Heroic Age: A Journal of Early Medieval Northwestern Europe **13**, 1526–1867 (2010), https://www.heroicage.org/issues/13/ashton.php
3. Gasser, R.: Solving nine men's morris. Computational Intelligence **12**(1), 24–41 (1996). https://doi.org/10.1111/j.1467-8640.1996.tb00251.x, https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8640.1996.tb00251.x
4. Schaeffer, J., Burch, N., Björnsson, Y., Kishimoto, A., Müller, M., Lake, R., Lu, P., Sutphen, S.: Checkers is solved. Science **317**(5844), 1518–1522 (2007). https://doi.org/10.1126/science.1144079, http://science.sciencemag.org/content/317/5844/1518
5. Tromp, J., Farnebäck, G.: Combinatorics of go. In: van den Herik, H.J., Ciancarini, P., Donkers, H.H.L.M.J. (eds.) Computers and Games. pp. 84–99. Springer Berlin Heidelberg, Berlin, Heidelberg (2007). https://doi.org/10.1007/978-3-540-75538-8_8