Some facts on finite two-player games with perfect information

Definition. *For the purpose of this handout, a* game *will always be a game between two players of the following form. The game starts in some* starting position. *The player alternate in making their move, thus bringing the game in a new* position. *It is part of the position which player is to move next, so the position "empty board and Player 1 moves next" is different from "empty board and Player 2 moves next". Both players have at every time full information about the rules of the game and of the current position. (This is called* perfect information. *Many board games are perfect information games, while most card games are not.)*

Some positions are endings. *Each ending implies either Player 1 or Player 2 as the* winner *of the game (so we do not allow draws). We call the game* finite *if there is no infinite sequence of moves without reaching an ending.*

Examples for such games are Bridg-It, Go, and Nim games (e.g.: from a pile of 100 coins, the players may take in their turn 1, 2, or 3 coins. The player who takes the last coin wins).

Theorem. *For every finite two-player information game as defined above, and every position P, exactly one of the two players (call him W) has a* winning strategy *meaning that for all strategies of the other player (call him L) there exists a strategy of W that will lead to an ending where W wins. In this case, we call P a* winning position *for W.*

Proof (sketch). Let P be a position of the game. Assume that player 1 moves next. Then it is easily determined whether P is a winning position for Player 1 by the following recursive algorithm.

DETERMINE_WINNER(P)

- For each possible move M of Player 1, let $P'(M)$ be the resulting position. For each M, compute DETERMINE_WINNER($P'(M)$).
- If DETERMINE_WINNNER($P'(M)$) $==$ Player 2 for all moves M then return "Player 2".
  $\#$ No matter what Player 1 does, Player 2 can win in all cases.
- If there is a move M such that DETERMINE_WINNNER($P'(M)$) $==$ Player 1, then return "Player 1".
  $\#$ There is a winning move for Player 1.

Note that this algorithm is terribly inefficient, but for finite games it will terminate and return the winner. It is also called "Minimax Algorithm". To see that this algorithm is well-defined, that is, it will eventually terminate, it is enough to observe that we cannot reach position P from $P'(M)$, as otherwise there would exist an infinite sequence of moves without reaching an end position. $\square$

Bridg-It

Recall that the board of Bridge-It consists of 25 "crossings" which can be either claimed by the blue or the red player.

So we can describe a position by a triple (B, R, b), where B and R are two disjoint subsets of [5]x[5], encoding the crossings that have been claimed so far by the blue and the red player, respectively; and b is a bit that encodes which player is the next to move. Note that in this way we even encode more positions than we can reach from the standard starting position ($B = R = \varnothing$).

A move of the blue player (the red player) consists of adding one free crossing (= a crossing neither contained in B nor R) to B (to R), and flipping the bit b. An end position is reached when all 25 crossings have been claimed. Then blue is a winner if and only if B contains a path from left to right.

Lemma. *The game is monotone in the following sense. If (B, R, b) is a winning position for blue, and $B' \supseteq B$ is a set of crossings disjoint to R, then (B', R, b) is also a winning position for blue.*

Proof. We use an inductive argument on the number of moves yet to play. In an end position, the statement is trivially true.

If (B, R, b) is not an end position and it is red's turn, then we need to show that all moves of red lead to a winning position for blue. Since (B, R, b) is a winning position for blue, for all free crossings c the position $(B, R \cup \{c\}, \bar{b})$ is also a winning position for blue. Hence, by induction hypothesis, $(B', R \cup \{c\}, \bar{b})$ is also a winning position whenever B' and $R \cup \{c\}$ are disjoint. Thus (B', R, b) is a winning position for blue.

If (B, R, b) is not an end position and it is blue's turn, then there exists a free crossing c such that $(B \cup \{c\}, R, \bar{b})$ is a winning position for blue. Now we need to find a winning move for position (B', R, b). If c is not contained in B', blue takes c as his next move, otherwise he takes an arbitrary free crossing. (If no free crossing exists then (B', R, b) is already an end position won by blue.) In both cases, blue can achieve a position $(B'', R, \bar{b})$, where $B'' \supseteq B \cup \{c\}$. By induction hypothesis, he has thus reached a winning position. □

Theorem. *The blue player (= the starting player) has a winning strategy for Bridge-It.*

Proof. Assume otherwise, i.e., assume the position $P_0 := (\varnothing, \varnothing, blue)$ is a winning position for red. Then by symmetry, $(\varnothing, \varnothing, red)$ must be a winning position for blue. Let c be an arbitrary crossing. Then by the lemma, $P_1 := (\{c\}, \varnothing, red)$ is also a winning position for blue. But blue can reach P_1 from P_0 by claiming the crossing c. Thus P_0 is also a winning position for blue (he has a winning move), in contradiction to the assumption. □