Chapter 6
Board Games AI

Tad Gonsalves
Sophia University, Japan

ABSTRACT

The classical area of AI application is the board game. This chapter introduces the two most prominent AI approaches used in developing board game agents—the MinMax algorithm and machine learning—and explains their usage in playing games like Tic-Tac-Toe, Checkers, Othello, Chess, Go, etc. against human opponents. The game tree is essentially a directed graph, where the nodes represent the positions in the game and the edges the moves. Even a simple board game like Tic-Tac Toe (naughts and crosses) has as many as 255,168 leaf nodes in the game tree. Traversing the complete game tree becomes an NP-hard problem. Alpha-beta pruning is used to estimate the short-cuts through the game tree. The board game strategy depends on the evaluation function, which is a heuristic indicating how good the player's current move is in winning the game. Machine learning algorithms try to evolve or learn the agent's game playing strategy based on the evaluation function.

INTRODUCTION

To date, Artificial Intelligence (AI) has produced a plethora of techniques and algorithms in the field of computer game playing. Escaping from the maze being chased by the enemy, firing shots and dodging missiles, etc. coupled with background music, eerie sounds, and animated colors are some of the AI games that fill modern computer screens. Although the flashy and dazzling computer games are increasingly becoming popular, this chapter deals with the description and analysis of the *classic* AI games, where two players (an intelligent computer program and a human player) sit on opposite sides with the game board in the middle. Examples of such classic AI games are tic-tac-toe, checkers, othello, chess, shogi, go, etc.

The classic games, played by two players on a single board taking turns alternatively are known as zero- sum games with perfect information (Millington, & Funge, 2009). The game can be unfolded in the form of a tree structure and the winning path consisting of legal moves through the tree nodes and branches can be traced using an efficient algorithm called the *MiniMax* algorithm. At the heart of this algorithm is the evaluation function, which gives an estimate of the goodness of a board position. Even complex board games like chess rely on the computation of the evaluation function to determine the

DOI: 10.4018/978-1-5225-7368-5.ch006

best possible move. However, due to the combinatorial explosion of the state space an exhaustive search of the game tree is infeasible. This situation is alleviated by the alpha-beta pruning technique and other heuristic search methods.

Machine Learning (ML) techniques employing Artificial Neural Networks is emerging as a powerful technique in game playing. In some applications, ML programs learn the evaluation function itself and then compute the winning moves using the *MiniMax* and the alpha-beta pruning algorithms. Some rely purely on the ML methods for mastering the winning strategy while others also use expert knowledge to supplement the ML process.

This chapter introduces the classic two-player zero-sum (the end result of the winner-loser score is zero), deterministic board games and explores some of the classic AI algorithms for game playing. In particular, it describes the game-tree, the *MiniMax* and the alpha-beta pruning algorithms for searching for the winning moves in real-time. It also introduces the relatively new ML techniques for game playing and the Evolutionary and Swarm Intelligence algorithms for conducting efficient learning. Finally, it indicates the direction of future research in AI game-playing algorithms and techniques.

BACKGROUND

Playing games intelligently and trying to win against human champions has been a grand challenge for *Artificial Intelligence* from its inception. Chess, in particular, has been referred to as the D*rosophila* of AI. The first study in computer chess was published by Claude Shannon in 1950 (Shanon, 1950). However, the first working AI programs were for playing checkers (Samuel, 1960; Samuel, 1967). In the 1970s and 1980s, computer-games research concentrated on chess and the brute-force search approach (Schaeffer, 2002). The triumph of AI in computer games came in the 1990s when to everyone's surprise AI programs began to defeat the reigning world champions. In 1994, the program CHINOOK won the World Man–Machine Championship (Schaeffer, 1997). Three year later, IBM'S DEEP BLUE defeated the World Chess Champion Garry Kasparov (Hsu, 2002) and LOGISTELLO won against the Othello Champion Takeshi Murakami (Buro, 1997). Finally, in 2011, IBM's WATSON defeated the world champions in the quiz game of Jeopardy.

The computer games mentioned above, with the exception of Jeopardy, are all two-player board-games. Any classic board-game playing algorithm makes a static list of all possible moves of the two players called the game tree, at a given stage of the game. It then assigns a relative score to each of the position. The algorithmic game playing strategy consists of efficiently searching through the moves that maximize the score. However, in practice, the number of moves even for any modest game are astronomically large and an exhaustive search is impossible. Specialized super-computers like the DEEP BLUE and WATSON greatly improves the depth and breadth of the search, but in no way is it exhaustive. AI techniques deal with the design and implementation of algorithms that provide short-cuts in the search. These AI algorithms also combine the power of Machine Learning to further reduce the search path. The AI techniques and their future development is described in detail in the following sections.

Related Content