

Board Games AI

Tad Gonsalves

Sophia University, Japan

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 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 *Drosophila* 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

DOI: 10.4018/978-1-5225-2255-3.ch013

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.

BOARD GAMES

This section describes the characteristics of two-player board games. The two players taking turns in playing give rise to a “turn game”. The unfolding of the game on the board surrounded by the two players can be represented by an abstract graph called the game tree. These two concepts are defined in the following sub-sections.

Turn Games

In the simplest board games, there are usually two players. These are called turn games, because each player takes a turn in playing. The game proceeds by the two players taking turns alternately. The outcome of a two-player turn game is either a win, a loss or a draw for each of the players concerned. The two-player board games are, therefore, zero-sum games – the net outcome score is zero. Let’s say, the score for winning is +10, for losing is -10 and for drawing the game is 0. The zero-sum turn play necessarily implies that a win for a player results in a loss for the opponent and vice versa. The sum of the scores at the end of the game is zero (+10-10). In the case of a draw situation also, the sum is zero.

Perfect information is another underlying assumption in a two-player turn game. It is assumed that the two players have complete knowledge of the progression of the game till the end result. In other words, any of the two players at any given point in the game knows all the possible move options at that point and all the possible outcomes emanating from each of those moves. This is opposed to random games which may involve the rolling of a die in which none of the players can predict the next move because of its intrinsic non-deterministic nature.

Game Tree

The complete information about the progression of the turn game that each player has access to can be represented in a graph consisting of nodes and edges. This graph is called the game tree. The nodes in the graph represent the board positions (configurations), and the edges represent the possible moves a player can choose to make. The starting position of the game is represented by the *root* node. A single move transports the game from the *parent* node to one of the next level *children* node. The end of the game is represented by the *leaf* nodes.

10 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/board-games-ai/183729

Related Content

An Objective Function for Evaluation of Fragmentation Schema in Data Warehouse

Hacène Derrar, Omar Boussaid and Mohamed Ahmed-Nacer (2015). *Encyclopedia of Information Science and Technology, Third Edition* (pp. 1949-1957).

www.irma-international.org/chapter/an-objective-function-for-evaluation-of-fragmentation-schema-in-data-warehouse/112601

Incorporating Technology Acceptance and IS Success Frameworks into a System Dynamics Conceptual Model: A Case Study in the ERP Post-Implementation Environment

Meg Fryling (2012). *International Journal of Information Technologies and Systems Approach* (pp. 41-56).

www.irma-international.org/article/incorporating-technology-acceptance-success-frameworks/69780

Using Technology to Reduce a Healthcare Disparity

Nilmini Wickramasinghe (2018). *Encyclopedia of Information Science and Technology, Fourth Edition* (pp. 3725-3732).

www.irma-international.org/chapter/using-technology-to-reduce-a-healthcare-disparity/184081

An Adaptive CU Split Method for VVC Intra Encoding

Lulu Liu and Jing Yang (2023). *International Journal of Information Technologies and Systems Approach* (pp. 1-17).

www.irma-international.org/article/an-adaptive-cu-split-method-for-vvc-intra-encoding/322433

Data Mining to Identify Project Management Strategies in Learning Environments

Ana González-Marcos, Joaquín Ordieres-Meré and Fernando Alba-Elías (2018). *Encyclopedia of Information Science and Technology, Fourth Edition* (pp. 1934-1946).

www.irma-international.org/chapter/data-mining-to-identify-project-management-strategies-in-learning-environments/183908