Foundations of Artificial Intelligence

41. Board Games: Introduction and State of the Art

Martin Wehrle

Universität Basel

May 23, 2016

41.1 Introduction

41.2 State of the Art

41.3 Summary

Classification

classification:

Board Games

environment:

- static vs. dynamic
- deterministic vs. non-deterministic vs. stochastic
- fully vs. partially vs. not observable
- discrete vs. continuous
- single-agent vs. multi-agent (opponents)

problem solving method:

- problem-specific vs. general vs. learning

Board Games: Overview

chapter overview:

- 41. Introduction and State of the Art
- 42. Minimax Search and Evaluation Functions
- 43. Alpha-Beta Search
- 44. Monte-Carlo Tree Search: Introduction
- 45. Monte-Carlo Tree Search: Advanced Topics
- 46. AlphaGo and Outlook

41.1 Introduction

Why Board Games?

Board games are one of the oldest areas of AI
(Shannon 1950; Turing 1950).

- abstract class of problems, easy to formalize
- obviously "intelligence" is needed (really?)
- dream of an intelligent machine capable of playing chess is older than electronic computers
- cf. von Kempelen's "Schachtürke" (1769), Torres y Quevedo's "El Ajedrecista" (1912)

German: Brettspiele

Games Considered in This Course

We consider board games with the following properties:

- current situation representable by finite set of positions
- changes of situations representable by finite set of moves
- there are two players
- in each position, it is the turn of one player, or it is a final position
- final positions have a utility
- utility for player 2 always opposite of utility for player 1 (zero-sum game)
- "infinite" game progressions count as draw (utility 0)
- no randomness, no hidden information

German: Positionen, Züge, am Zug sein, Endposition,
Nutzen, Nullsummenspiel

Example: Chess

Example (Chess)

- positions described by:
  - configuration of pieces
  - whose turn it is
  - en-passant and castling rights
- turns alternate
- final positions: checkmate and stalemate positions
- utility of final position for first player (white):
  - $+1$ if black is checkmated
  - 0 if stalemate position
  - -1 if white is checkmated

Other Game Classes

important classes of games that we do not consider:

- with randomness (e.g., backgammon)
- with more than two players (e.g., chinese checkers)
- with hidden information (e.g., bridge)
- with simultaneous moves (e.g., rock-paper-scissors)
- without zero-sum property ("games" from game theory
  $\rightsquigarrow$ auctions, elections, economic markets, politics, ...)
- ...and many further generalizations

Many of these can be handled with similar/generalized algorithms.

Terminology Compared to State-Space Search

Many concepts for board games are similar to state-space search.
Terminology differs, but is often in close correspondence:

- state $\rightsquigarrow$ position
- goal state $\rightsquigarrow$ final position
- action $\rightsquigarrow$ move
- search tree $\rightsquigarrow$ game tree

Formalization

Board games are given as state spaces $\mathcal{S} = \langle S, A, cost, T, s_0, S_\star \rangle$
with two extensions:

- player function $player : S \setminus S_\star \to \{1, 2\}$
  indicates whose turn it is
- utility function $u : S_\star \to \mathbb{R}$ indicates utility of final position
  for player 1

other differences:

- action costs *cost* not needed

We do not go into more detail here as we have previously seen sufficiently many similar definitions.

Specific vs. General Algorithms

- We consider approaches that must be tailored
  to a specific board game for good performance,
  e.g., by using a suitable evaluation function.

$\rightsquigarrow$ see chapters on informed search methods

- Analogously to the generalization of search methods
  to declaratively described problems (automated planning),
  board games can be considered in a more general setting,
  where game rules (state spaces) are part of the input.

$\rightsquigarrow$ general game playing: annual competitions since 2005

Why are Board Games Difficult?

As in classical search problems, the number of positions of (interesting) board games is huge:

- Chess: roughly 10^{40} reachable positions; game with 50 moves/player and branching factor 35: tree size roughly $35^{100} \approx 10^{154}$
- Go: more than 10^{100} positions; game with roughly 300 moves and branching factor 200: tree size roughly $200^{300} \approx 10^{690}$

In addition, it is not sufficient to find a solution path:

- We need a strategy reacting to all possible opponent moves.
- Usually, such a strategy is implemented as an algorithm that provides the next move on the fly (i.e., not precomputed).

Algorithms for Board Games

properties of good algorithms for board games:

- look ahead as far as possible (deep search)
- consider only interesting parts of the game tree (selective search, analogously to heuristic search algorithms)
- evaluate current position as accurately as possible (evaluation functions, analogously to heuristics)

41.2 State of the Art

State of the Art

some well-known board games:

- Chess, Go: $\rightsquigarrow$ next slides
- Othello: Logistello defeated human world champion in 1997; best computer players significantly stronger than best humans
- Checkers: Chinook official world champion (since 1994); proved in 2007 that it cannot be defeated and perfect game play results in a draw (game "solved")

German: Schach, Go, Othello/Reversi, Dame

Computer Chess

World champion Garri Kasparov was defeated by Deep Blue in 1997 (6 matches, result 3.5–2.5).

- specialized chess hardware (30 cores with 16 chips each)
- alpha-beta search (⇝ Chapter 43) with extensions
- database of opening moves from millions of chess games

Nowadays, chess programs on standard PCs are stronger than human world champions.

Computer Chess: Quotes

Claude Shannon (1949)

The chess machine is an ideal one to start with, since

1. the problem is sharply defined both in allowed operations (the moves) and in the ultimate goal (checkmate),
2. it is neither so simple as to be trivial nor too difficult for satisfactory solution,
3. chess is generally considered to require "thinking" for skillful play, [. . .]
4. the discrete structure of chess fits well into the digital nature of modern computers.

Alexander Kronrod (1965)

Chess is the drosophila of Artificial Intelligence.

Computer Chess: Another Quote

John McCarthy (1997)

In 1965, the Russian mathematician Alexander Kronrod said, "Chess is the Drosophila of artificial intelligence."

However, computer chess has developed much as genetics might have if the geneticists had concentrated their efforts starting in 1910 on breeding racing Drosophilae. We would have some science, but mainly we would have very fast fruit flies.

Computer Go

Computer Go

- The best Go programs use Monte-Carlo techniques (UCT).
- Until recently (autumn 2015), Zen, Mogo, Crazystone played on the level of strong amateurs (1 kyu/1 dan).
- Until then, Go has been considered as one of the "last" games that are too complex for computers.
- In October 2015, Google's AlphaGo defeated the European Champion Fan Hui (2p dan) with 5:0.
- In March 2016, AlphaGo defeated world-class player Lee Sedol (9p dan) with 4:1. The prize for the winner was 1 million US dollars.

⇝ We will discuss AlphaGo and its underlying techniques in more detail later in the course.

41.3 Summary

Summary

- Board games can be considered as classical search problems extended by an opponent.
- Both players try to reach a final position with (for the respective player) maximal utility.
- very successful for a large number of popular games
- AlphaGo recently defeated one of the world's best players in the game of Go.