From Deep Blue to Monte Carlo: An Update on Game Tree Research

Akihiro Kishimoto and Martin Müller

AAAI-14 Tutorial

Image sources: britannica.com, wikimedia.org

1997: Deep Blue vs Kasparov

- https://www.youtube.com/watch?v=NJarxpYyoFI

Speakers and their Backgrounds

- Akihiro Kishimoto (aka Kishi), IBM Research Ireland
  - artificial intelligence, parallel computing, high-performance game-playing, planning, risk management systems, computer shogi (Japanese chess)
- Martin Müller, University of Alberta
  - computer games, domain-independent planning, combinatorial game theory and algorithms, computer Go, Monte Carlo Tree Search, Random Walk planning, Fuego open source program

Computer Games Tutorial in One Slide

- We focus on “classical” two player games such as chess, checkers, Othello, 5-in-a-row, Go,…
- Can solve games, or just try to play well
- Huge successes with classical minimax methods such as alphabeta (αβ)
- Recently much progress in Monte Carlo Tree Search (MCTS) methods
- How does it work?

Goals of Tutorial

- Up to date overview of research techniques for classical two player games
- Main Algorithms
  - Minimax and Alphabeta search
  - Proof number search
  - Monte Carlo Tree Search
- Techniques we touch upon
  - Representation and implementation issues, Parallel search, machine learning, program tuning and optimization, Testing

Organization of the Day: Morning

- 9 - 10 Tutorial 1: Overview, introduction, general concepts (Martin)
- 10 - 10:30 Tutorial 2: Solving and playing games (Kishi)
- *10:30 - 11 Coffee break*
- 11 – 12:30 Tutorial 3: Alphabeta and enhancements (Kishi)
- 12:30 - 1 Tutorial 4: Proof Number Search (Kishi)

Organization of the Day: Afternoon

2 - 3	Continue Proof Number Search (Kishi)
3 - 3:30	Tutorial 5: Monte Carlo Tree Search (Martin)
3:30 – 4	*Coffee break*
4 - 5:30	Continue Monte Carlo Tree Search
5:30 - 6	Tutorial 6: State of the art in specific games. Wrap-up (Martin)

Some Questions We Address

- How did game tree search develop since Deep Blue?
- What are the ideas behind current methods?
- Which successes have they achieved in games and elsewhere?
- What are the biggest open problems in games research?

What we Won’t Talk About

- Single-agent games, puzzles
- Multi-player games
- Games of chance (Poker, dice, backgammon,…)
- Classical game theory, Nash equilibria,…
- Combinatorial game theory, sums of games
- General Game Playing (GGP)

From Deep Blue to Monte Carlo: An Update on Game Tree Research

Akihiro Kishimoto and Martin Müller

AAAI-14 Tutorial 1:
Overview,
Introduction,
General Concepts

Image source: ebay.com

Presenter:
Martin Müller, University of Alberta

Prehistory – Game Theory

- Zermelo (1913) - existence of a winning strategy
- von Neumann (1928) - first proof of general minimax theorem with mixed strategies
- von Neumann and Morgenstern (1944) – Theory of Games and Economic Behavior
- Nash (1950) - concept, existence proof of Nash equilibria
- Many applications to decision-making, economics, biology
- At least *twelve* Nobel prizes for game theorists!

Short History of Chess Programming

- 1950 **Shannon** "Programming a Computer for Playing Chess" - *evaluation function*, *selective* and *brute force* search strategies
- 1951 **Turing** - *algorithm* for playing chess, simulates it by hand
- 1956 **McCarthy** *alphabeta* pruning
- 1967 Greenblatt chess program, *transposition tables*
- 1968 First Levy bet, human vs computer
- 1981 *Cray Blitz* achieves Master rating
- 1982 **Ken Thompson**'s *Belle*, *hardware* accelerated chess program, earns US Master title
- 1988 *Deep Thought* becomes Grandmaster strength
- 1996 Kasparov beats *Deep Blue*
- **1997 Deep Blue beats Kasparov**
- today: Mobile phones at strong grandmaster level.
  Programs such as *Stockfish*, *Komodo* far surpass all humans on ordinary PCs

State of the Art in Computer Game-Playing

- Games solved
- Super-human strength
- Human world champion level
- Strong play
- Weak or intermediate-level play

Solved by Search and/or Knowledge

- Four-in-a-row, Connect-four (Allis; Allen 1988)
- Qubic (Patashnik 1980)
- Gomoku - 5 in a row (Allis 1995, Wagner and Virag 2000)
- Nine Men's Morris (Gasser 1994)
- Awari (Romein 2002)
- Checkers (Schaeffer et al 2007)
- Fanorona (Schadd 2007)

Solved by Mathematical Techniques

Using Combinatorial game theory:

- Nim (Bouton 1908)
- Hackenbush, Domineering, … (Winning Ways)
- Go endgame puzzles (Berlekamp and Wolfe 1994)

Games Solved only on Small Boards

➚	Hex	6x6 (Enderton 1994), 7x7, 8x8, 9x9 (Yang; Hayward et al)
➚	Go	5x5 (van der Werf 2003), 7x4, …(v.d. Werf & Winands, 2009)
➚	Othello	6x6 (Feinstein 1993)
➚	Domineering	10x10 (Bullock 2002)
➚	Amazons	5x5 (Müller 2001), 5x6 (Song & Müller 2014)
➚	Dots and Boxes	up to 4x6 (Wilson) several variations on rules

Not Solved, Super-human Strength

- Backgammon (Tesauro - TD-Gammon, 1995)
- Chess (Deep Blue 1997)
- Othello (Buro - Logistello, 1997)
- Scrabble (Sheppard - Maven, 2002)

World Champion Level

- 9x9 Go (Fuego 2009, MoGo 2009, Zen)
- Shogi - Japanese chess
- Xiangqi - Chinese chess (?)
- 10x10 draughts? (?)
- Heads-up (2 person) Poker (Alberta team 2008)
- Amazons? (Invader - Lorentz)

Master Level

- 19x19 Go (Zen, Crazy Stone, 6 Dan amateur)
- 14x14, 19x19 Hex?
- Bridge?
- Poker with 3 or more players?
- Arimaa?
- Havannah?

Weak to Intermediate Level

- General Game Playing (GGP) - relative strength varies by game

So, are we Almost Done?

- Huge successes
- If we “solve Go”, are we done?
- Of course not!
  - Playing better than humans is not solving a game
    - Go will not be “solved”
  - Games are great test beds to study algorithms
  - The games people play evolve over time

Why “Classical” Games?

- Simple, controlled environment
- Still hard to solve or play well
- Interesting for many people
- Domains easy to understand
- For humans, playing games well requires intelligence

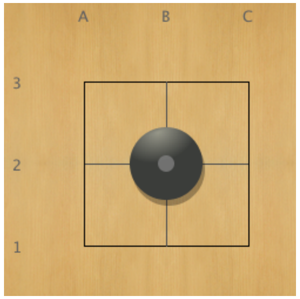
Two Player Games – Basic Concepts

- Two players, often called *Black* and *White*
- Move alternately: I play, you play, I play,…
- Zero-sum: my win is opponent's loss
- Perfect Information:
  - Both players know state of the game
  - No hidden information
- No chance:
  - no dice rolls, card draws, other random events
  - Games of pure skill

Sample Game: NoGo

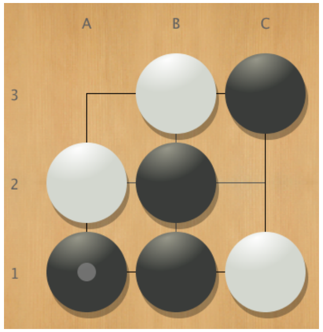
- NoGo, short for No Capture Go. Also called Anti Atari Go
- Rules similar to Go
  - Gameplay is *completely* different
- Played on a grid, initially empty
- Two players Black and White
- One move = put one stone of your own color on board
- Last player who can move wins

Brief Demo

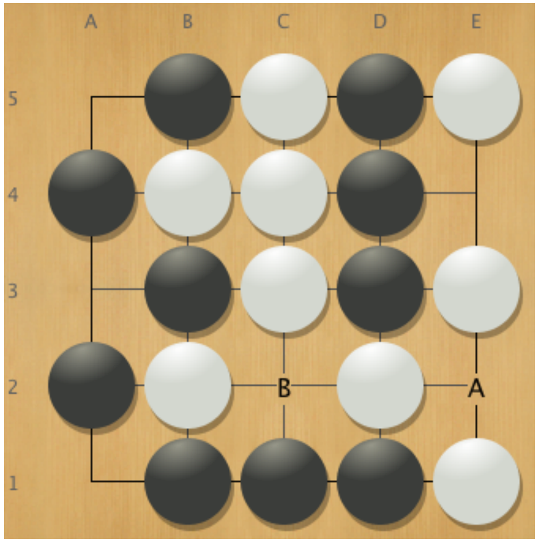


NoGo Concepts and Rules

- *Block* – connected set of stones of same color
  - Example: two black blocks:
    three Black stones {A1, B1, B2},
    single stone {C3}



- *Liberty* – empty intersection adjacent to a block
  - Each block needs at least one liberty
  - Illegal moves: *capture* and *suicide*
    - Example: only A, B legal for White



Basic Concepts, Terminology

1. Game position, board state
2. Game state, state
3. Move, action
4. Move sequence, history, game record
5. State space, game graph, game tree
6. Score, value, evaluation, result

1. Game Position, Board State

- "Snapshot" of game in progress
- Examples:
  - board state, board position: which pieces on which locations?
  - cards held, cards open, on stacks, discarded,…
  - money, etc held by each player or public

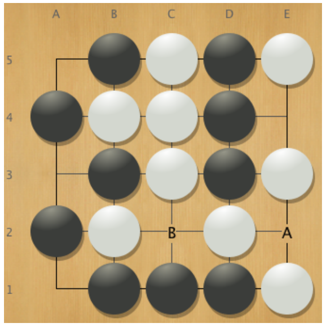


Image sources: texasnoregrets.com, everythingbrisa.blogspot.ca

2. Game State

- Complete description of current game situation
- Includes:
  - Game position
  - *ToPlay* - whose turn it is
  - Often includes (parts of) *history*:
    sequence of moves from start of game
    - When is history needed?
      Depends on rules, structure of search space
      Example: detect position repetition

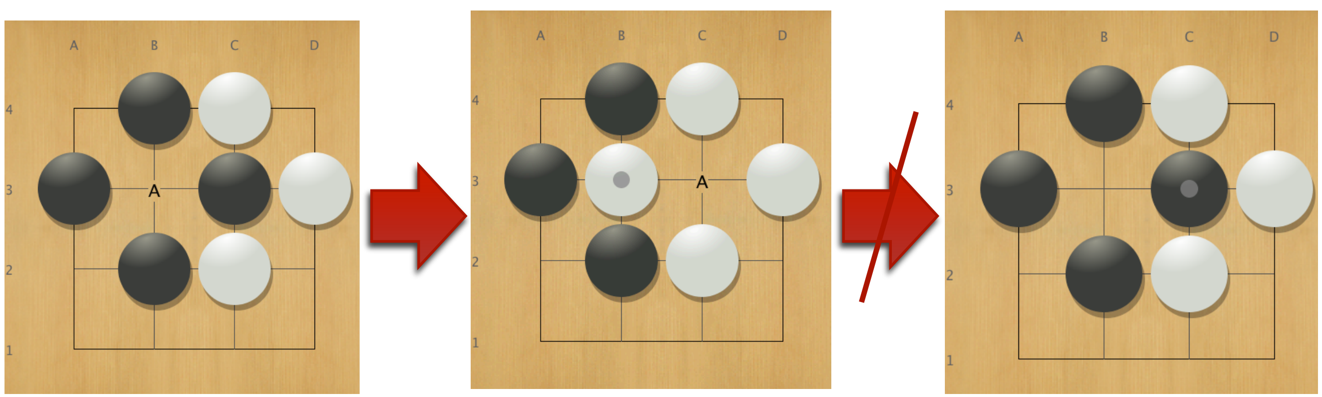
Image source: www.theguardian.com

What's in a Game State ?

- All information needed to
  - Determine set of *legal moves*
  - Determine if game is finished or not
    - A player might also *resign* to end the game
  - Determine the result when finished
    - Who wins, and by how much?

Game State Examples

- NoGo: state = (position, toPlay)
  - Does not need history (why?)
- Go: *Ko* rules prevent repetitions
  - Need history in state to determine legal moves

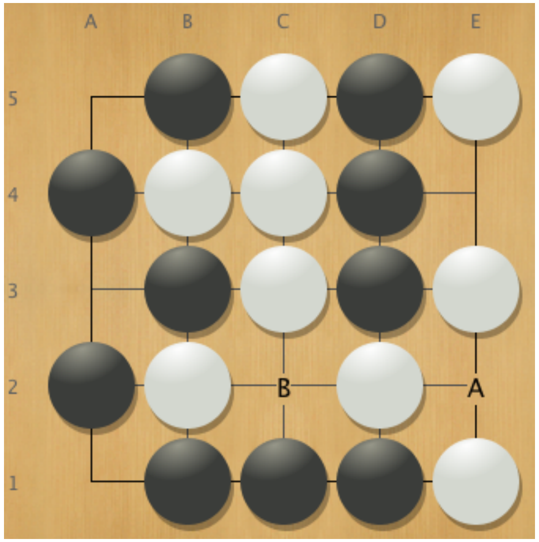


Go: legal capture

Go: illegal ko re-capture

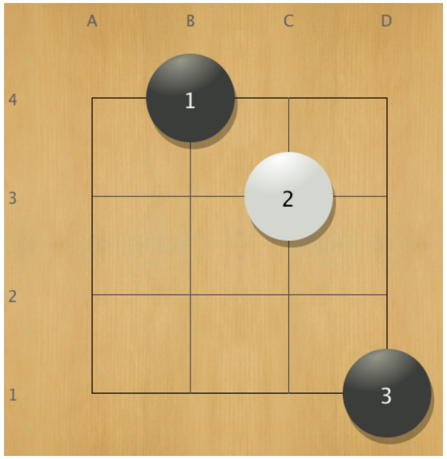
3. Move, Action

- Move: from one state to the next
  - Usually: small, simple description
    - Example: White E2
  - Change of full state is often implicit and has to be computed
- Partial move: if move is complicated it may make sense to split it up
  - Examples:
    - chess: from-to squares
    - Arimaa: each move moves four pieces



4. Move Sequence, History, Game Record

- Move sequence:
  - zero or more consecutive moves
  - Leads from one state to another through a sequence of intermediate states
    - Abstract example:
      - Move sequence <m_1, m_2, m_3>
      - Start state s_0
      - Playing m_1 in s_0 leads to s_1
      - Playing m_2 in s_1 leads to s_2, etc.
    - Concrete example: see figure <B B4, W C3, B D1>



  - A move sequence is legal if each move is legal in the corresponding state

History, Game Record

- History – move sequence from start of game to current state
- Game record – history of a complete game
  - Popular file formats: pgn, sgf,…
  - Support variations, annotations, comments,…

5. State Space, Game Graph, Game Tree

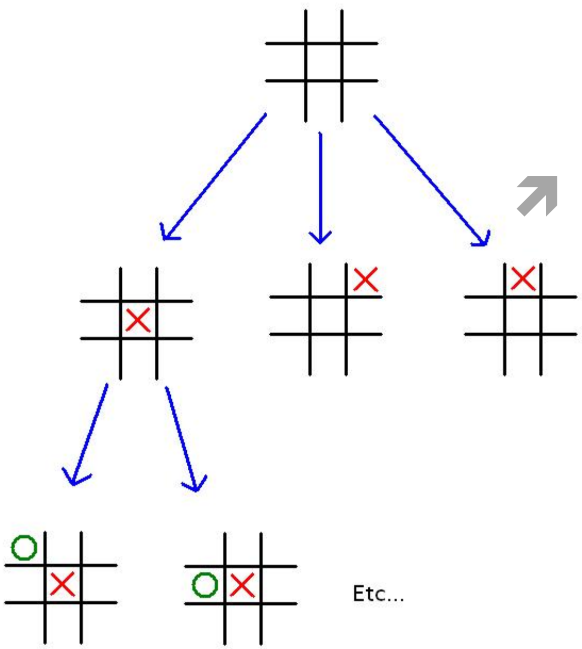
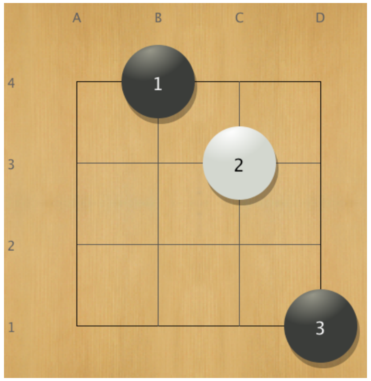


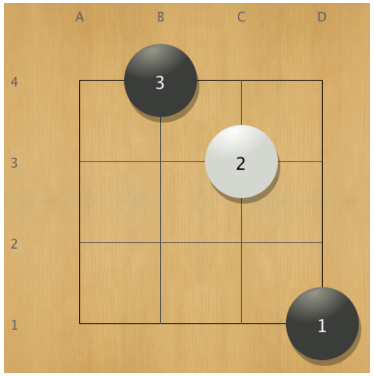
Image source: sciencefair.math.iit.edu

- State space:
  - Which states exist?
  - How are they connected by moves?
- Game graph, game tree:
  - Each state is a node in a graph
  - Connect states s1, s2 by a directed edge iff there is a legal move from s1 to s2
  - Can different move sequences lead to the same state? If no, the game graph is a tree

Game Tree, DAG, DCG

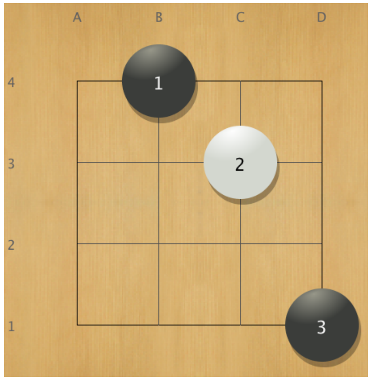
- Tree:
  - Root = starting position
  - Using full history will always result in a tree
  - Following edges on path from root gives the move sequence
- DAG – directed acyclic graph
  - Different sequences may lead to same state
  - Acyclic: non-empty move sequence cannot lead back to the same node
  - Example: NoGo game graph is a DAG
- DCG – directed cyclic graph, may have loops

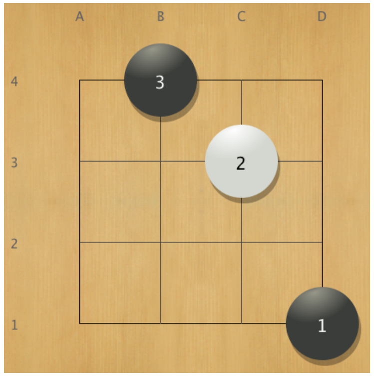




DAG Examples

- Sequence <B B4, W C3, B D1>
- Sequence <B D1, W C3, B B4>
- Both lead to same game state





(Tricky) quiz for chess players: do

a) 1. e4 e5
   2. Nf3

b) 1. Nf3 d5
   2. e4

Lead to the same game state?

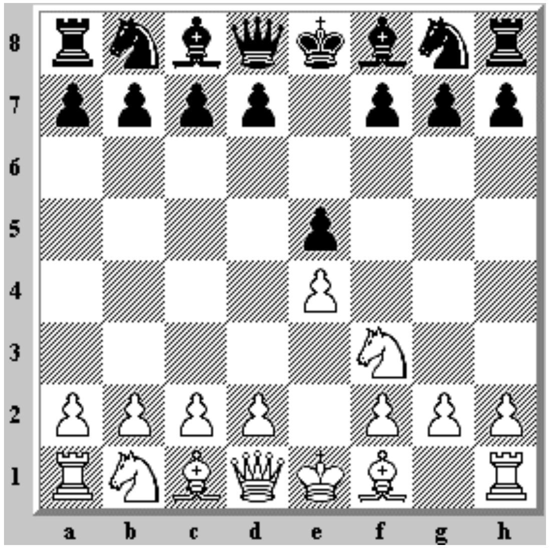


Image source: www.mark-weeks.com

DCG - Directed **Cyclic** Graph

- In NoGo, we cannot have loops
  - Each move adds exactly one stone to the board
- In many games we can repeat the board position
  - But: is it really the same game state???
  - Usually not, but depends on game rules
  - We will come back to this question
- Example in chess, checkers: move pieces back and forth
- Example in Go: ko rule (forbids repetition)

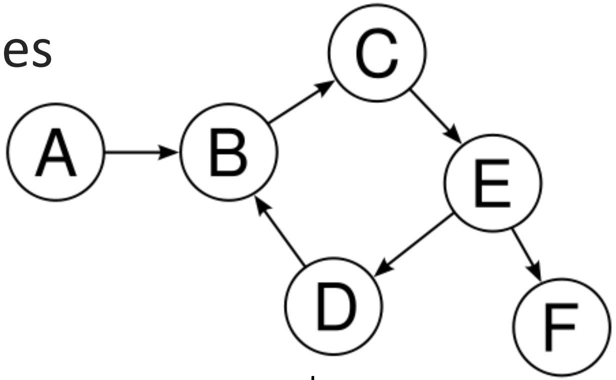


Image source: commons.wikipedia.org

6. Game Score, Value, Evaluation, Result

- *Result*: at end of game
  - Simplest: win / loss (1/0)
  - Often: win / loss / draw (1/0.5/0)
  - Point-scoring games:
    score = size of win / loss
    - Example: Black wins by 3.5 points
- *Value* of a position:
  - Game-theoretic value:
    result assuming best play by both
- *Evaluation*: what a player (program) *estimates* the value to be
  - Examples: winning probability, expected score

Image source:
beautyandthefeastblog.com

Summary of Tutorial 1

- Brief history of computer players, state of the art in playing and solving games
- Sample game: NoGo
- Basic concepts:
  - game position, game state
  - move, move sequence
  - state space, game graph
  - result - value – evaluation

Image source: www.ps4site.com