The Games Computers (and People) Play Revisited

Jonathan Schaeffer
University of Alberta

Thank you!

End of Intermission

- AAAI 2000:

  The Games Computers (and People) Play (part 1)

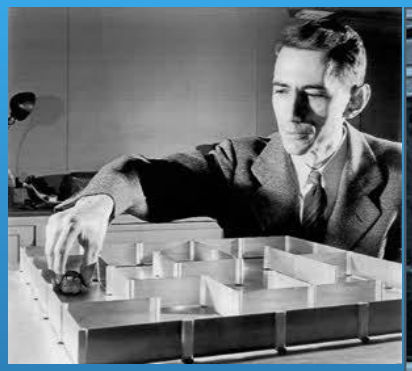
- Intermission (11 years) – Thank you for your patience

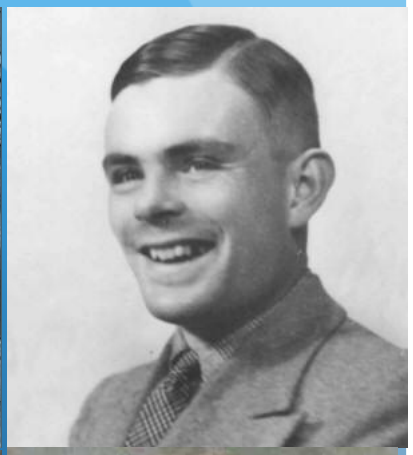
- IJCAI 2011:

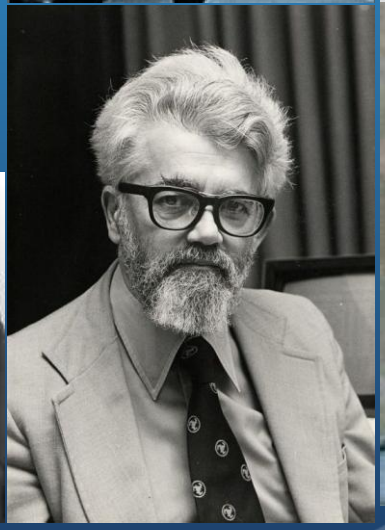
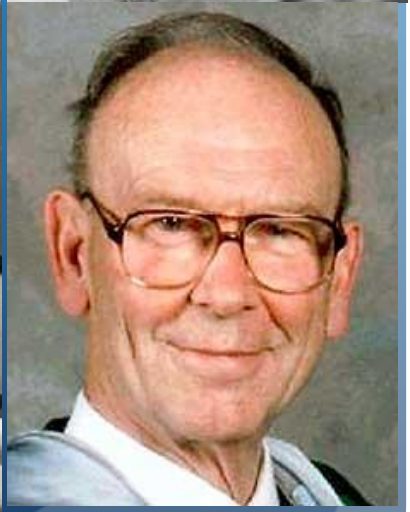
  The Games Computers (and People) Play (part 2)

AI Grand Challenge Problem

- Claude Shannon
- Arthur Samuel
- Alan Turing
- Donald Michie
- John McCarthy
- Allan Newell
- Herb Simon





An AI Success Story

- 60 years after the dawn of computers…
  - Most classic board and card games have strong (even perfect) computer players
  - The research led to fundamental contributions in AI
- DEEP BLUE (1997)
  - Feng-hsiung Hsu, Murray Campbell, and Joe Hoane
  - IBM's "$250 million of free favorable publicity"
  - AI's biggest media event

Objective?

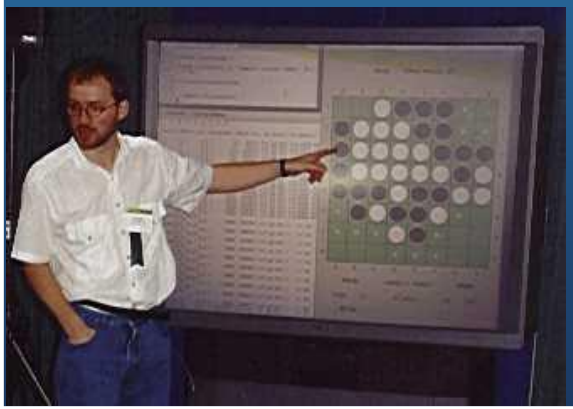
Defeating the strongest human players

AAAI 2000: Secrets of Success

- Search algorithms
  - Deep alpha-beta-based search
  - Checkers, chess
- Knowledge
  - Parameter learning algorithms
  - Backgammon, Othello
- Statistical sampling
  - Play out “likely” scenarios to the end of the game
  - Bridge, poker

AAAI 2000: Computer Successes

- Backgammon: Strong Grandmaster
- Checkers: World Champion
- Chess: Strong Grandmaster
- Crossword puzzles: Strong
- Othello: Super human
- Scrabble: Possibly super human



IJCAI 2011 Agenda

- Discuss the new secrets of success
  - Search (Monte Carlo)
  - Game Theory
  - Technology Advances
- Note the lack of Knowledge!
- Highlight the performance accomplishments
- Proposals for new research

Evolution of Applications

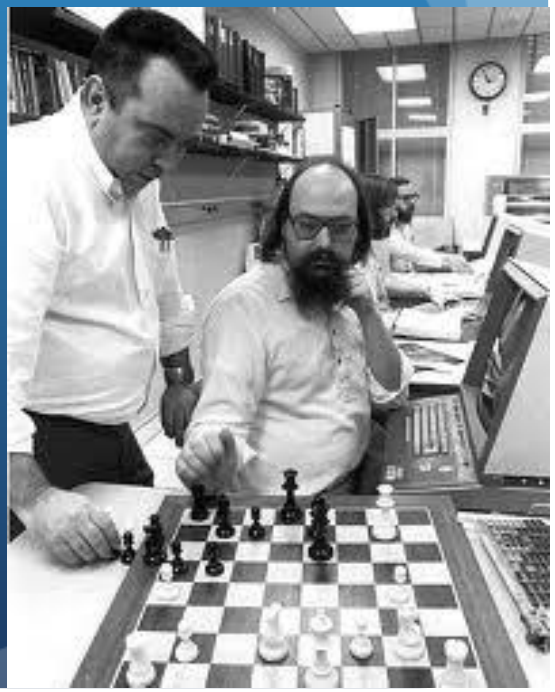
- Two-player perfect information (checkers, chess, go)
  - Difficulty of the domains has grown
- Chance (backgammon, poker)
- Hidden information (bridge, poker)
- Multiple players (Chinese checkers, poker)
- …
- Natural language processing (Jeopardy)
- General game playing (GGP)

Search

2000: Alpha-Beta Search

- Ken Thompson (1982) and his innocuous 2-page paper
- Alpha-beta performance scales with search depth
- Recipe for success: more computing resources

	D=4	D=5	D=6	D=6	D=8	D=9	Rating
D=4	X	5.0	0.5	0.0	0.0	0.0	1235
D=5	15.0	X	3.5	3.0	0.5	0.0	1570
D=6	19.5	16.5	X	4.0	1.5	1.5	1826
D=7	20.0	17.0	16.0	X	5.0	4.0	2031
D=8	20.0	19.5	18.5	15.0	X	5.5	2208
D=9	20.0	20.0	18.5	16.0	14.5	X	2328



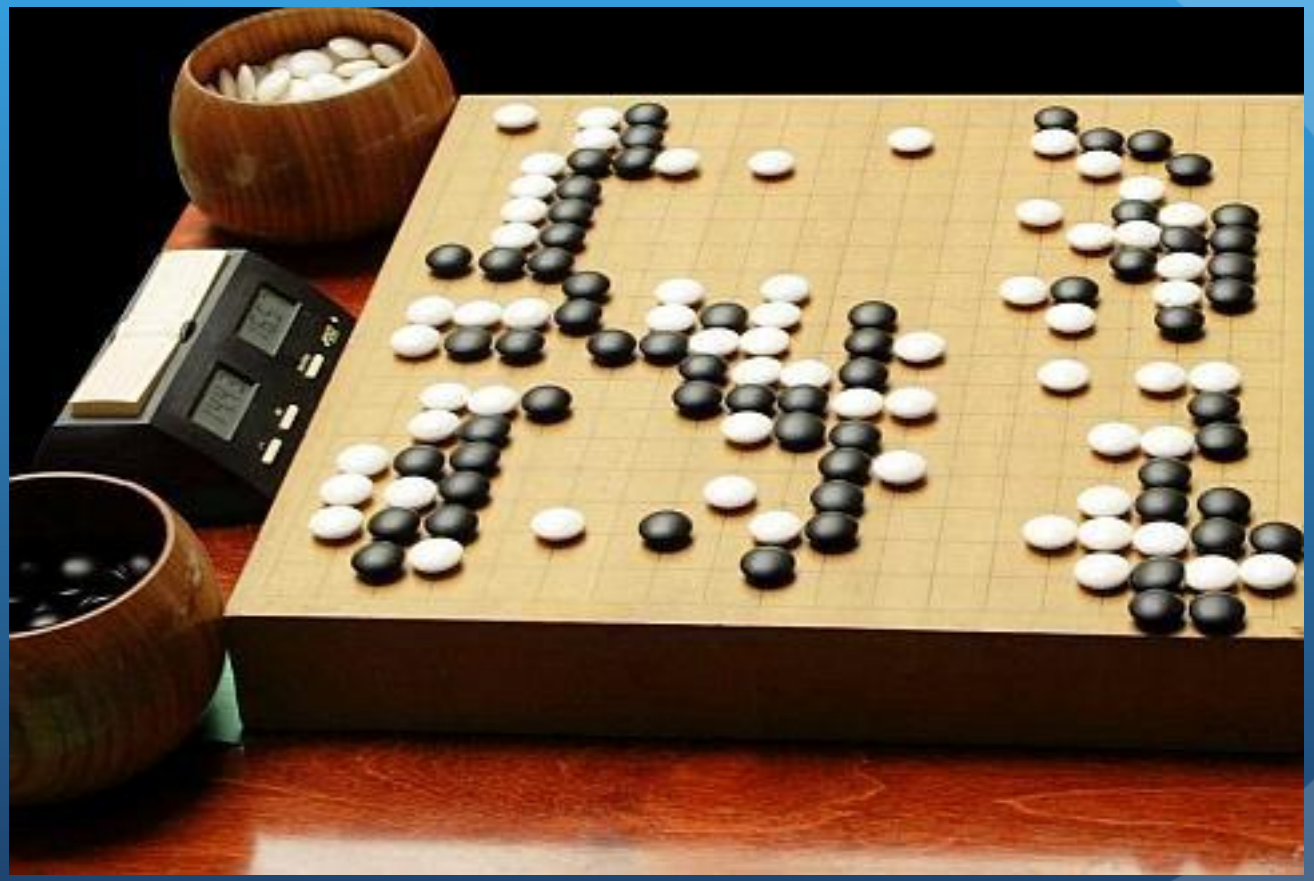
Results (2011)

- Checkers
  - 1994: World champion
  - 2007: Game solved; computers will never lose
- Chess
  - 1997: World-champion-level caliber
  - 2002: Deep Junior draws Garry Kasparov
  - 2005: Hydra crushes Michael Adams
  - 2006: Deep Fritz defeats Vladimir Kramnik
  - Humans may disagree, but the evidence is overwhelming that computers are super-human

2011: Monte Carlo Search

- Exploit the power of statistics
  - Make “random” move sequences to the end of the game
  - Game rules define value of terminal state
  - Acquire statistics for the success rate of each legal move in the root position
  - Play the move with the highest probability of success
- Can random move sequences really lead to strong play?
  - Limited success in bridge (Ginsberg 1998) and poker (Billings *et al*. 1999)
  - Important role in backgammon (Tesauro 2002) and Scrabble (Sheppard 2002)

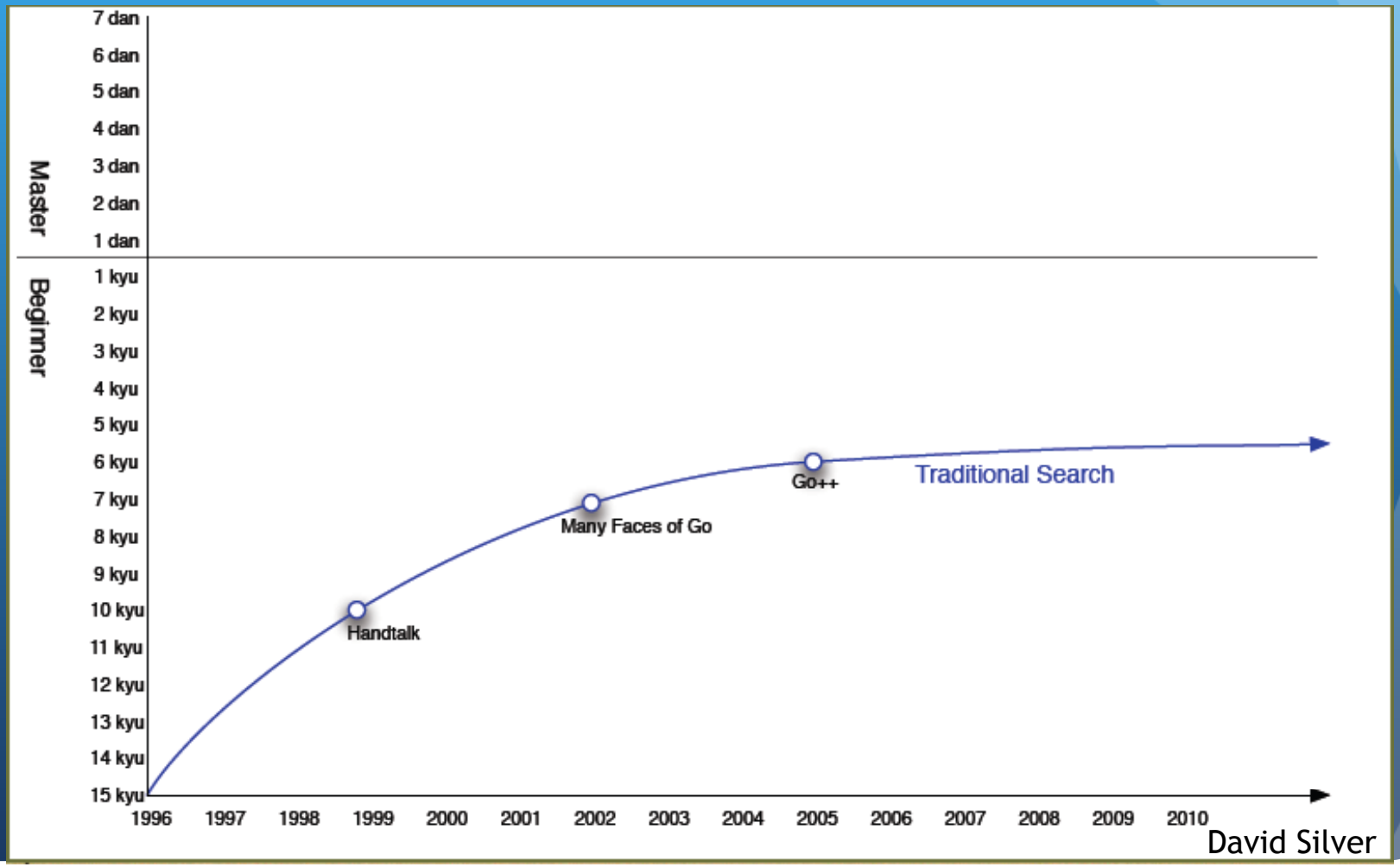
Go



Go (2000)

- Alpha-beta search
  - Large branching factor
  - Combination of global and local searches
  - Defining the boundaries of local search can be hard
- Selective search
  - Using domain knowledge to severely restrict the set of moves considered
- Extensive use of application-dependent knowledge
  - Human go expertise required
  - Brittle knowledge requiring extensive (manual) tuning
- Progress was slow with each advance requiring substantial development effort

Traditional Approaches



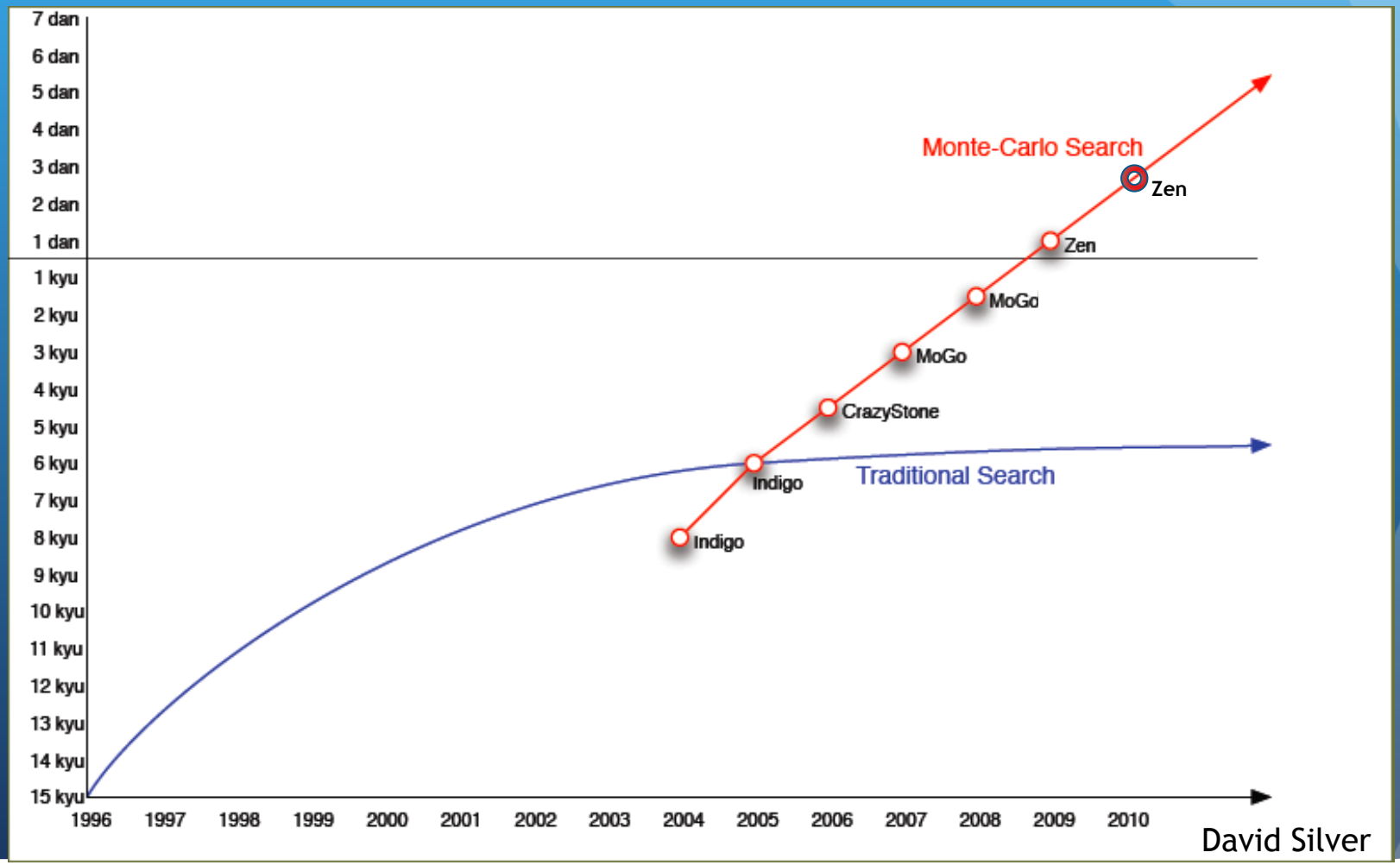
Go (2011)

- Monte Carlo search
  - Brugmann (1993), Bouzy and Cazenave (2001)
  - Limited success in Go
- UCT (Upper Confidence bounds applied to Trees)
  - Kocsis and Szepesvari (2006)
  - Disciplined way to balance exploration and exploitation
- Monte Carlo Tree Search (MCTS)
  - Combination of Monte Carlo, UCT, and selective search
  - Expand a node and use random move sequence to determine a value

Where to Search?

- Choose the position where we stand to gain the most information
  - Best move so far? Exploitation
  - Moves that have not been analyzed much? Exploration
- UCT is a compromise
  - Select move to explore where
    value + uncertainty
    is the highest
  - Keep count of #times a node is visited

Monte Carlo Approaches



Should We Be Surprised?

- Monte Carlo methods are common in physics, mathematics, finance, and other disciplines
- Confession: I was surprised
- Why? Sampling can miss key moves
  - E.g., move A is only refuted by B
  - If sampling does not include B, a bad result can arise
- Don’t underestimate the power of statistics!

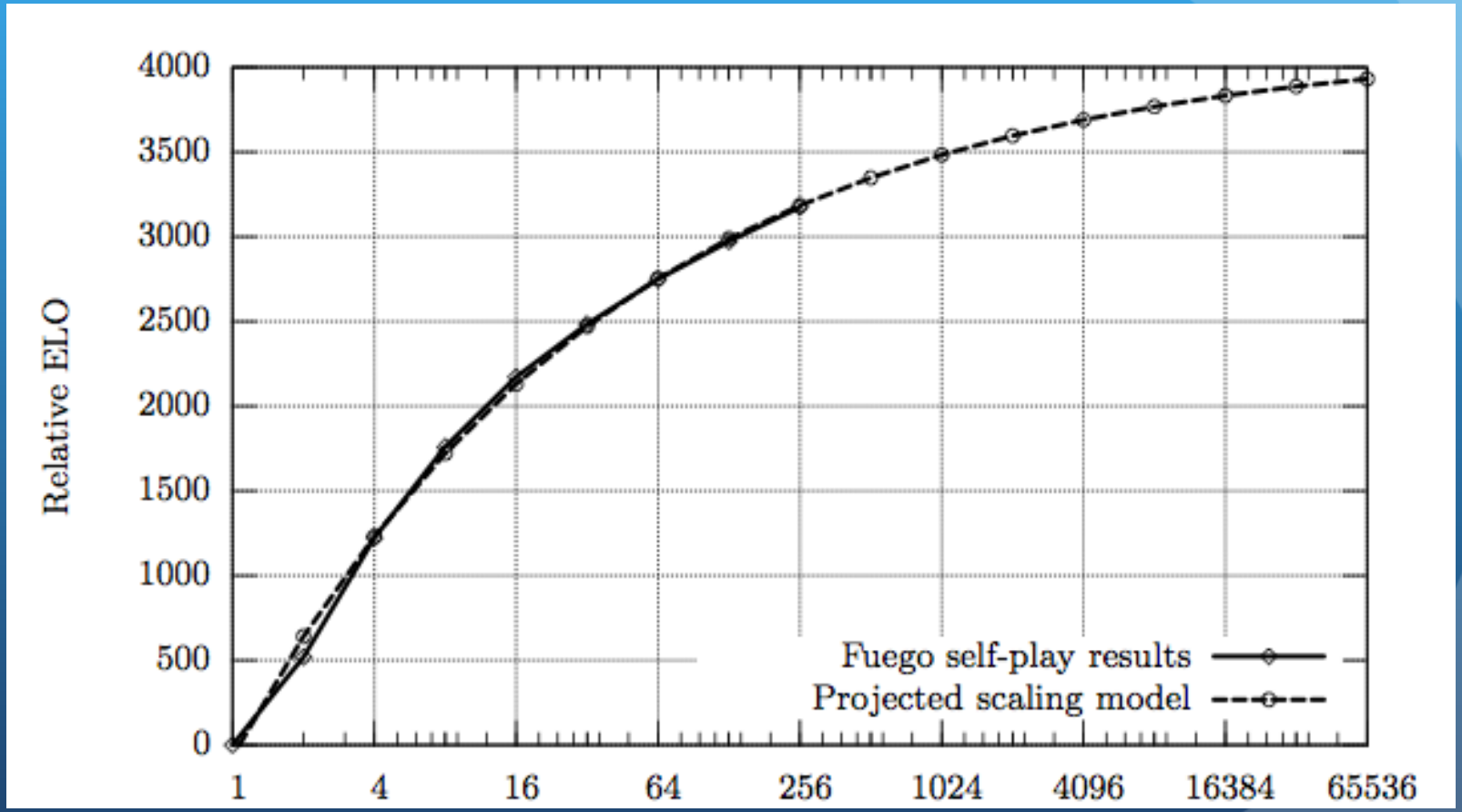
Why is Go So Different than Chess?

- Chess
  - Pretty good at evaluating states
  - Dominated by short-term considerations (material) which can be evaluated well, but long-term subtleties (pawn weaknesses) can cause problems
- Go
  - We do not know how to evaluate states well
  - Dominated by long-term subtleties
- UCT helps us select good moves
  - Applying application-dependent knowledge to further narrow the possibilities can be effective

Magic Recipe

- Fast!
  - Need only move generator and terminal node evaluator
- Knowledge
  - No need for application-dependent knowledge
  - Knowledge can accelerate convergence by focusing the search on more likely scenarios
- Scaling
  - UCT/MCTS: performance scales with search depth
  - Recipe for success: more computing resources

Scaling



Segal (2011)

Outcome?

- Chess: is alpha-beta enough to reach the top?
  - 1985: “maybe but I don’t think so” (Jonathan Schaeffer)
  - 1997: DEEP BLUE defeats Kasparov
- Go: is MCTS enough to reach the top?
  - 2011: “maybe but I don’t think so” (Martin Mueller)
  - 2023: ????

Computer Go (2011)

- 9x9: competitive with the best
- 13x13: strong play
- 19x19: good amateur player

Zhou Junxun 9p losing at 9x9 Go to MoGo (no handicap), 2009

General Game Playing

- Genesereth *et al.* (2005)
- Given only the rules of a game (in GDL), learn to play the game as well as possible
  - One-player games (puzzles)
  - Two or more player games (with perfect information)
- Annual competitions (typically 10-15 entries)
- Similar in spirit to the biennial planning competitions

Advances?

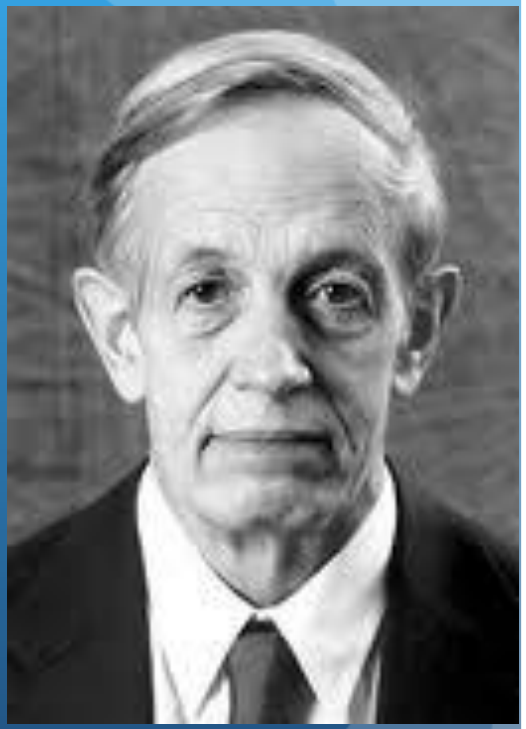
- Similar to the planning competition…
  - State generation is very slow
  - Creating useful evaluation functions is very hard
- Search
  - UCT is the catch-all for everything
  - “Solves” the evaluation problem
- Progress is slow
- GGP is an important domain that deserves more attention from the AI community

Feeling Lucky?

Game Theory

Nash Equilibrium

- Solution to a game of two or more players; ideal for domains with imperfect information
- Equilibrium when no player can gain by changing their strategy
- Can be a recipe for breaking even; does not say how to exploit an opponent who is using an inferior strategy



Poker

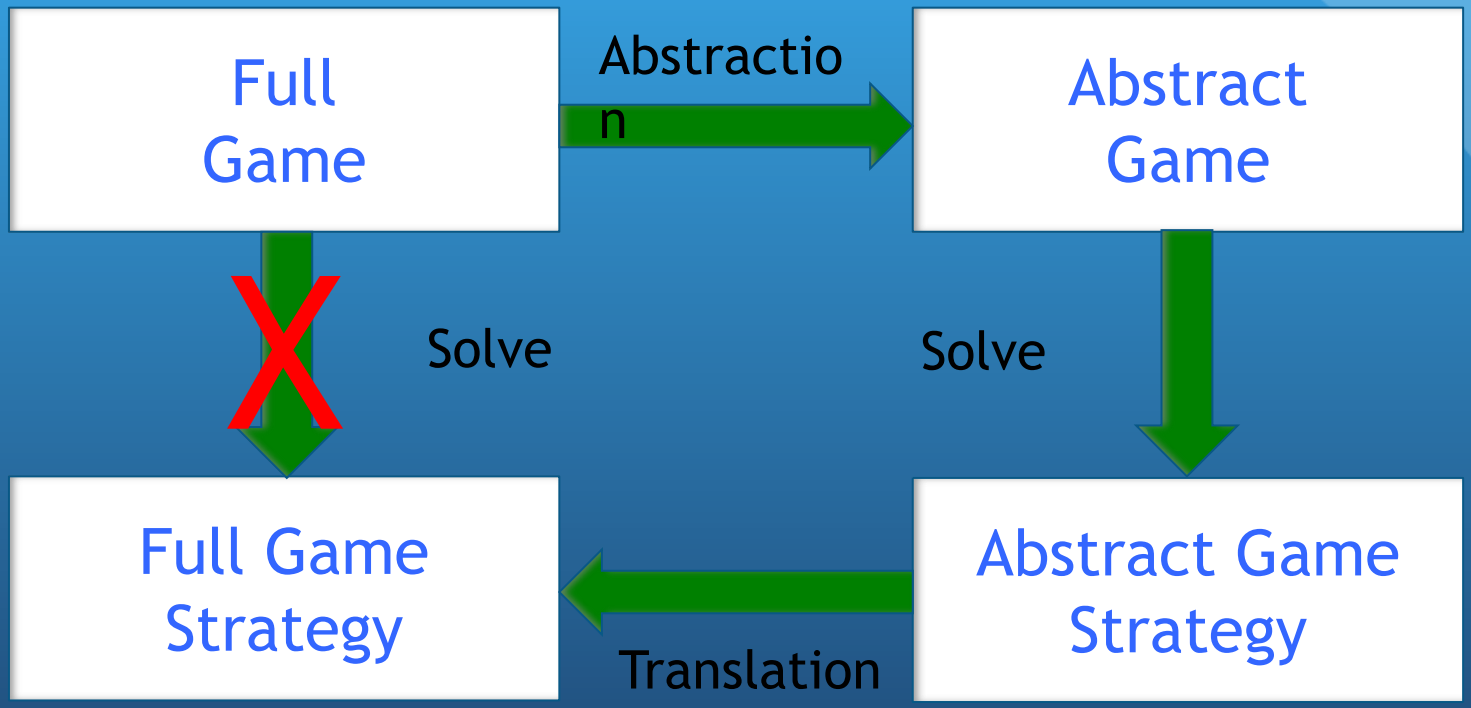
2000: Poker

- Challenges:
  - Hidden information
  - Stochastic
  - Multiple players
- Limit Texas Hold'em poker
  - Limited gains from Monte Carlo sampling
  - Limited gains from opponent modeling
  - Limited success handling more than two players
  - Able to complete favorably for small stakes
- No success with no-limit poker

2003: Poker Breakthrough

- Koller and Pfeffer (1995)
  - Proposed an algorithm for finding optimal randomized strategies that work for “small” games
- Billings *et al.* (2003)
  - Abstract a game to something much smaller that preserves the key properties of the game
  - Express as a linear program
  - Use the solution to the abstract game for the full game

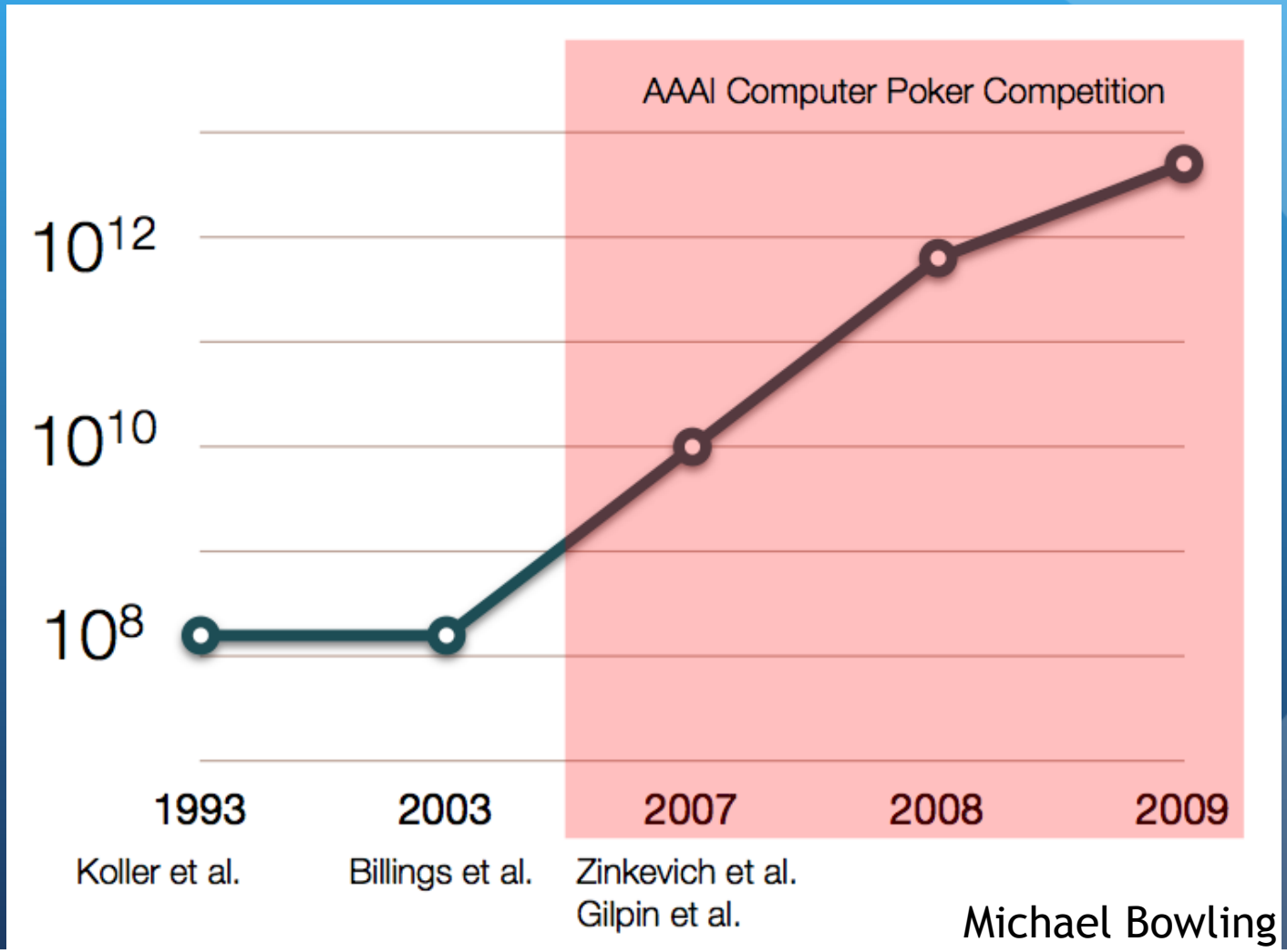
Bringing the Game Down to Size



Progress

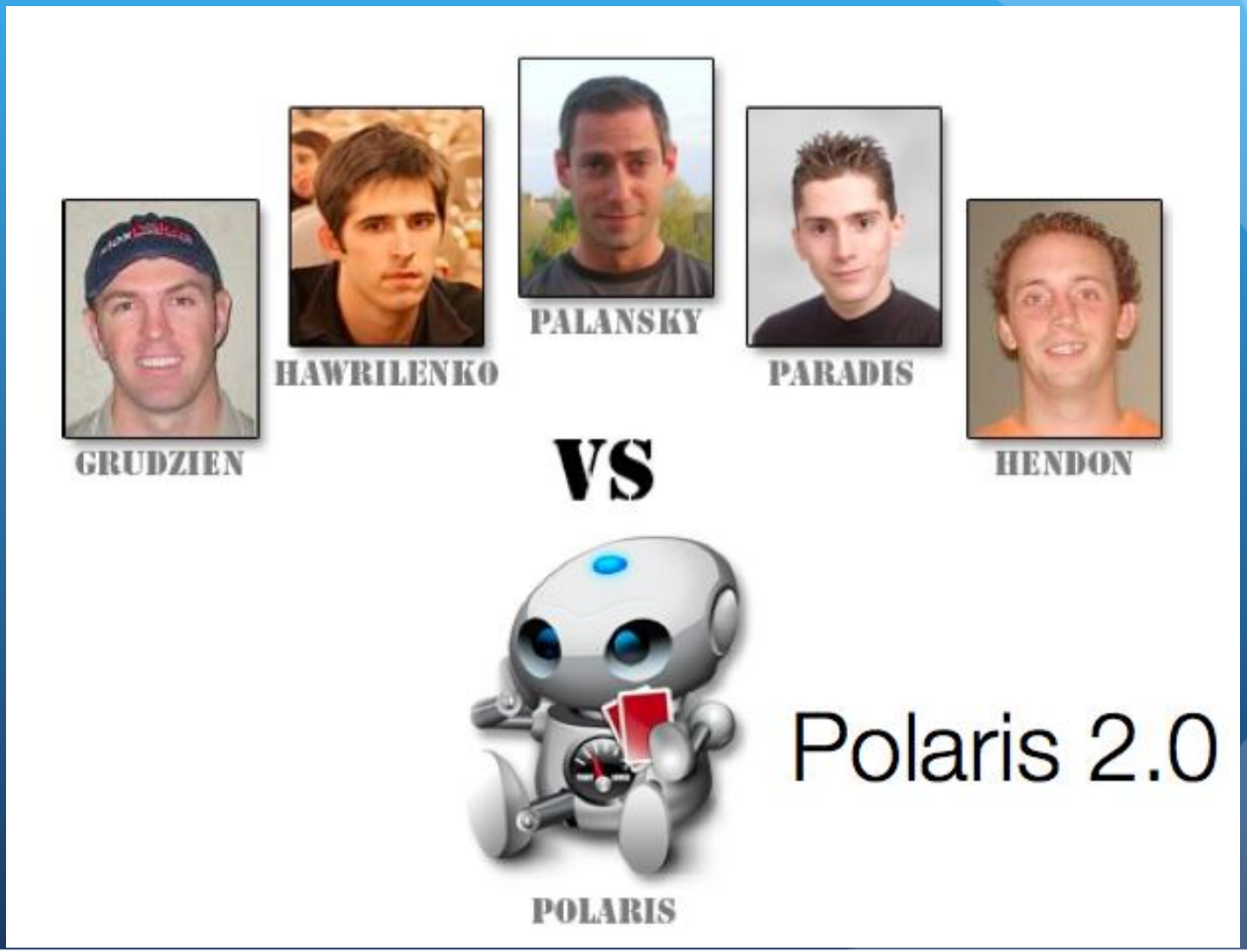
- Limitations of the linear programs
  - Coarse abstraction to keep the state space small
- Counter-factual regret (Zinkevitch *et al.*, 2007)
  - Regret scaled by how often the decision must be made, assuming the agent had tried to reach that decision
  - Iterate and converge on a solution
- GameShrink (Gilpin and Sandholm, 2007)
  - Abstract the game using isomorphisms (preserve optimality)
  - Add in approximate methods (scale up, but lose optimality)

Arms Race



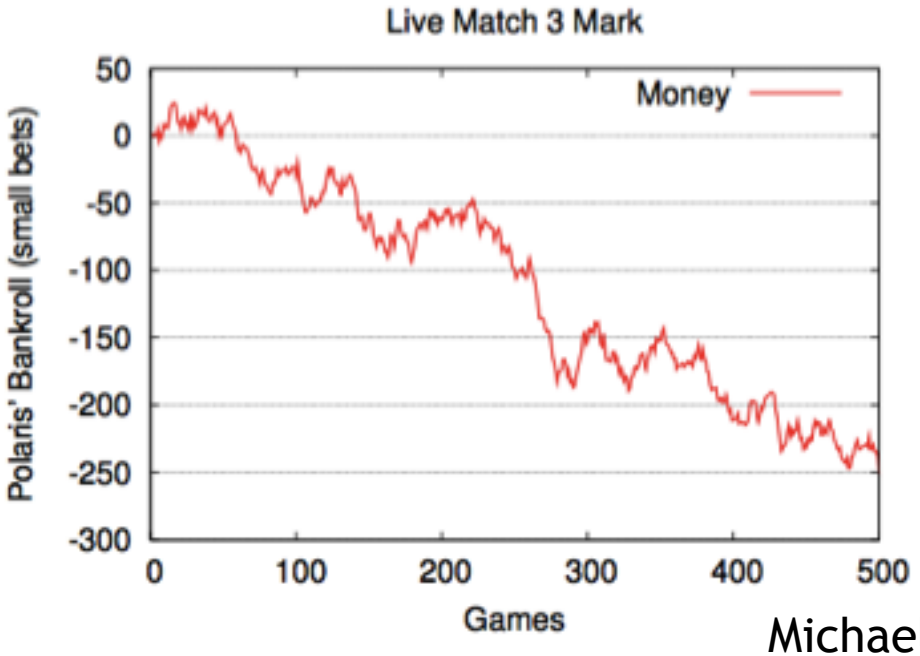
Michael Bowling

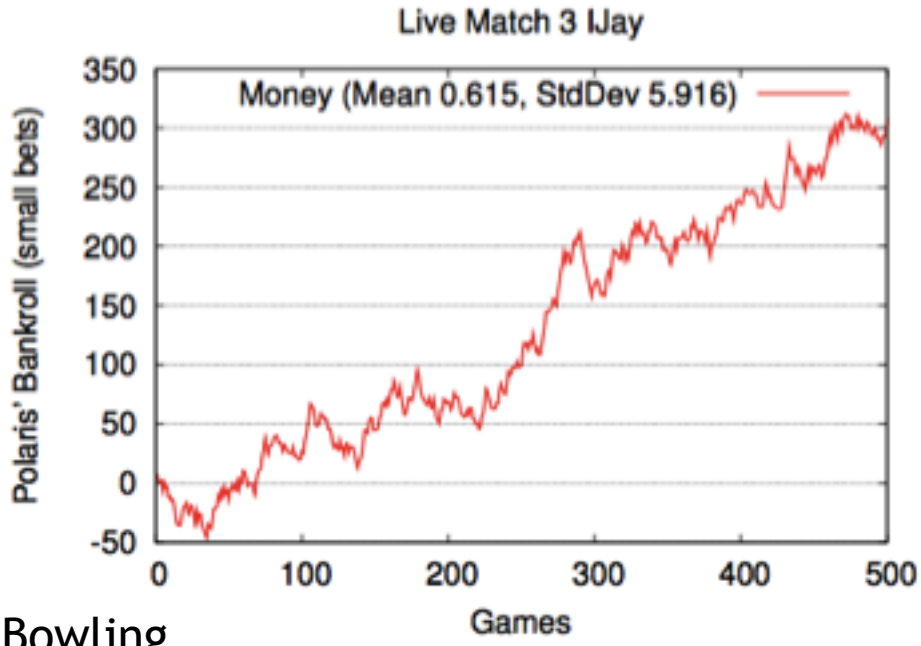
2008: Man-Machine Match



Money

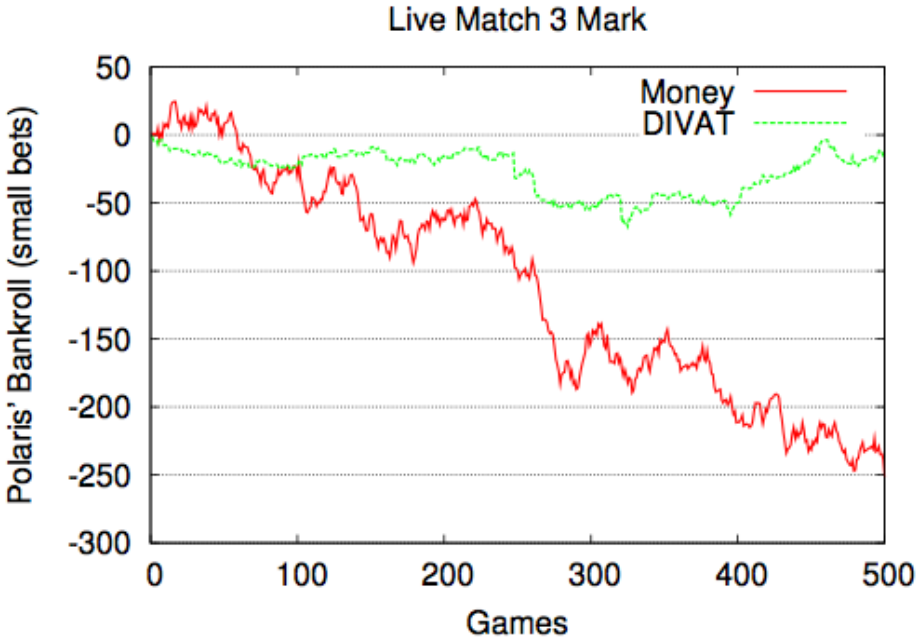
- Only statistic that matters to humans
- 20,000 hands for statistical significance
  - Duplicate matches reduce this to 2,000 pairs of hands
- POLARIS wins by 56 small bets

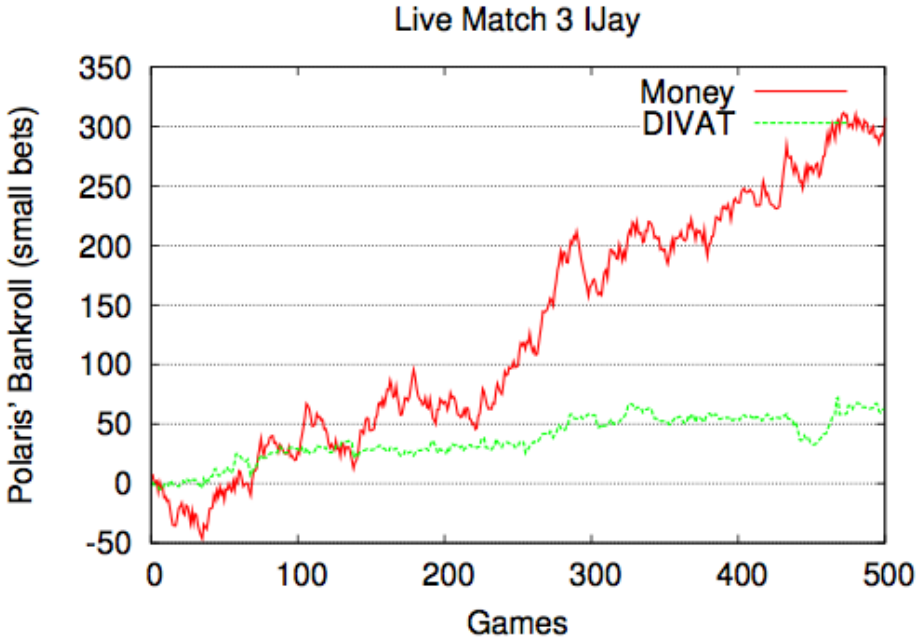




DIVAT

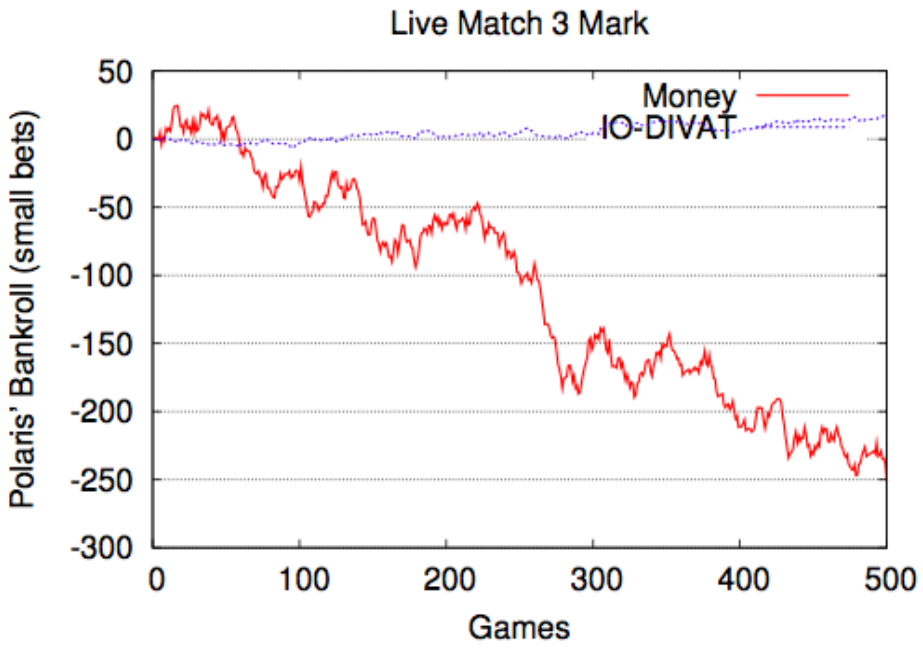
- Eliminate much of the luck factor (Zinkevitch *et al. 2006)*
  - Do not penalize a player for making the right decision
  - 1,200 pairs of hands for statistical significance
- POLARIS wins by 47 small bets

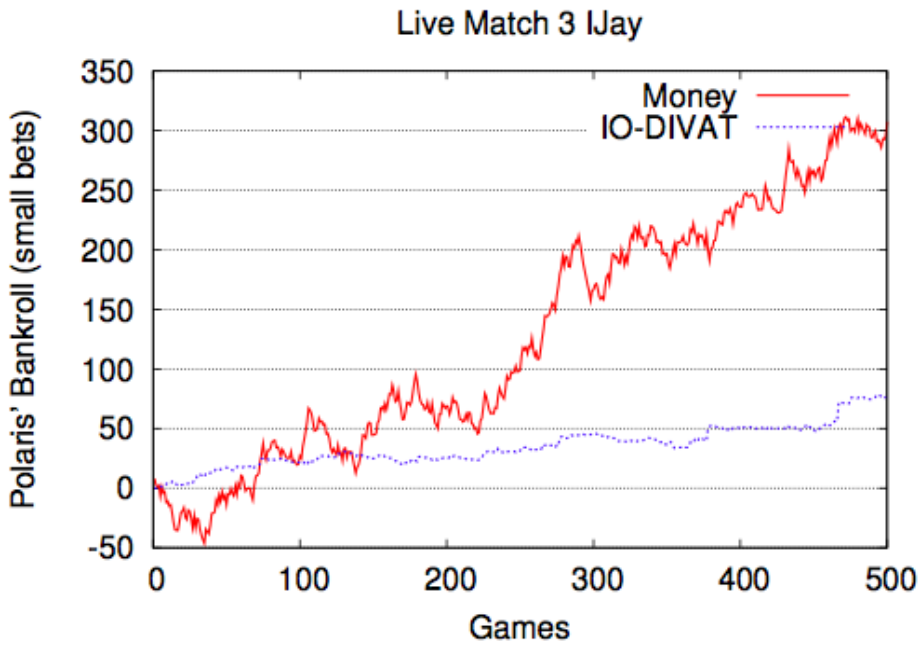




IO-DIVAT

- Imaginary Observations (Bowling *et al*. 2008)
  - Importance sampling with appropriate weighting
  - 500 pairs of hands for statistical significance
- Polaris wins by 89 small bets





The “Real” Result?

Match	Money	DIVAT	IO-DIVAT
1	-25.5 (loss)	+41	+41
2	+120 (win)	+93	+64 (win)
3	+5 (draw)	+8	-10
4	-50 (loss)	-16	+33
5	+56 (win)	+47	+89 (win)
6	+89.5 (win)	+87	+127 (win)

Lies (“you were lucky”)

Damned Lies (“the match was unfair”)

Statistics (“your math is meaningless to us”)

Challenges

- Two-player limit Texas Hold'em: superhuman play
- Three or more players: Weak play
- No-limit Hold'em (*a la* television): Weak play
- Opponent modeling
  - Lots of room for creative solutions
  - Hard but important problem

Technology Advances

Hardware

- Computers: speed advances have stalled, but now have multiple cores per chip
  - Moore's law still holds, although it is not evident in the speed of a single chip
- Storage: gigabytes (10^9) have given way to terabytes (10^{12}) and petabytes (10^{15})
- Network bandwidth: megabits have been replaced by gigabits
- Price: more performance for the same price

Solving a Game

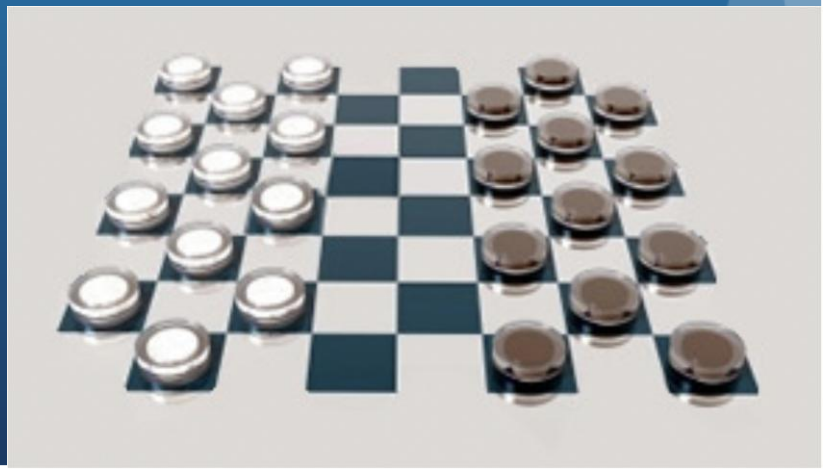
- Strong: play perfectly for both players from any legal position
- Weak: play perfectly for both players from the starting position
- Ultra-weak: determine the result of perfect from the starting position, without showing the perfect play

Awari

- Positions: 889,063,398,406
- Result: Draw (strongly solved)
- Team: Romein and Bal (2002)
- Resources:
  - Retrograde analysis
  - Roughly 1 CPU year
  - 144-processor cluster
  - 72 GB of RAM
  - 1.4 TB of disk

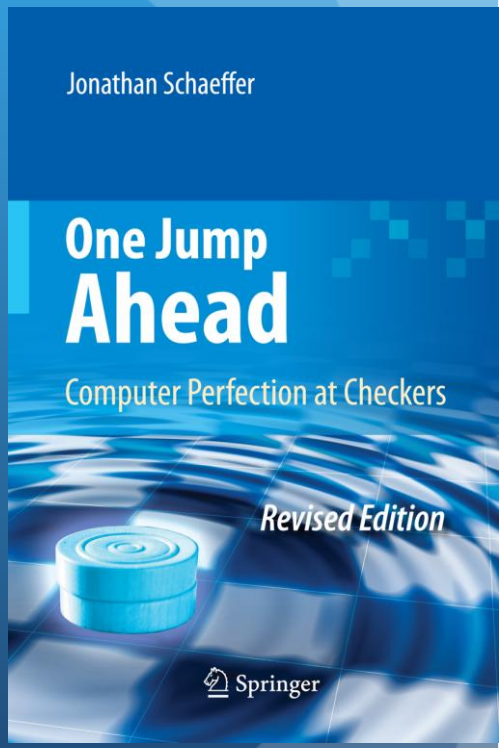
Checkers (8×8)

- Positions: 500,995,484,682,338,672,639
- Result: Draw (weakly solved)
- Team: Schaeffer, Burch *et al.* (2007)
- Resources:
  - Retrograde analysis and forward search
  - 17 years of hardware evolution
  - Varied from 1-200 processors
  - Estimate it could be solved today with 50 processors in five years



Memories

- “Didn’t Samuel solve that game in 1963?”
- In 2000 rolled back the computations seven years because of a single bit
  - Ensure the integrity of large data sets over many years
- “What a moron! Don’t you know how to play the publishing game?”



Efficient Search?

- Positions:
  - $5x10^{20}$
- Data solution:
  - 10^{18} disk and $\geq 10^{21}$ computations
- Compute solution:
  - 0 disk and $\geq 10^{23}$ computations (optimistic)
- Our hybrid solution:
  - 10^{11} disk and 10^{14} computations
- Combination of search and storage is effective

Last Word

"It's been 18 years! …

obsessive-compulsive behavior…

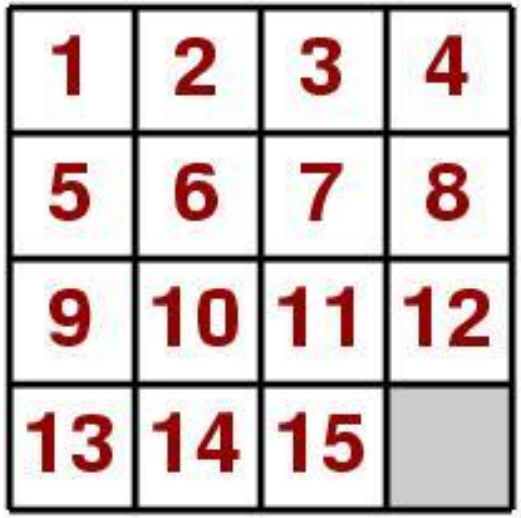
not normal… .

Get a life, Jonathan."

Stephanie Schaeffer

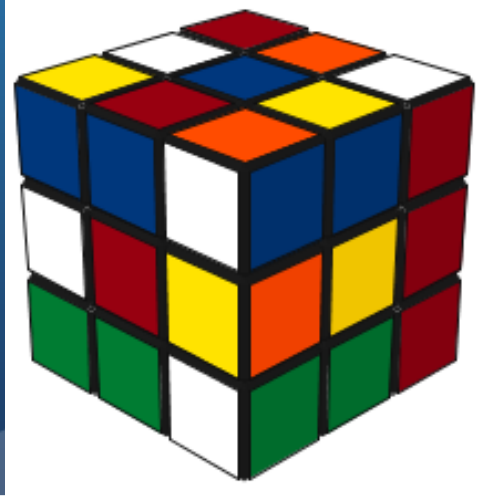
Sliding Tile Puzzle (4×4)

- Positions: 20,922,789,888,000
  - Reduced by a factor of 4 (symmetry and illegal states)
- Result: All positions solved (maximum solution length 80)
- Team: Korf and Schultze (2008)
- Resources:
  - Retrograde analysis
  - 0.2 CPU years
  - 2GB of RAM
  - 1.4 TB of disk



Rubik's Cube

- Positions: 43,252,003,274,489,856,000
  - Reduced by a factor of ≈40 by symmetry and set covering
- Result: All positions can be solved in a maximum of 20 face turns
- Team: Rokicki *et al*. (2010)
- Resources:
  - Forward search
  - 35 CPU years
  - Google infrastructure



Conclusions

Scorecard

- Perfection
  - Awari, Checkers (8x8), Chess (endgames), Hex (small boards),
- Super human
  - Backgammon, Chess, Othello, Poker (2-player limit), Scrabble
- Strong
  - Bridge, Go (9x9), Shogi
- Improvement needed
  - Chinese checkers, Go (19x19), Hearts, Poker (no limit; more than 2 players)

What's Left for Me?

Challenges (Classic Games)

- Creating human-understandable knowledge
  - Data rich and information poor
- Annotating games
  - Explaining decisions
- Opponent modeling
- New genres of games have new challenges
  - Mostly coming out of Germany
  - Multi-player, cooperative, auctions, tile placement, etc.

Challenges (Video Games)

- Hard because of real-time, limited resources, and believability
- Unfortunately little of what is done in the AI community has relevance to the games industry!
- Relatively little progress, but interesting problems
  - Automated story writing
  - Realistic character behavior
- Success
  - Pathfinding algorithms
- Potential success
  - Natural language (Jeopardy)

Research using games as the experimental testbed is fun!

Intermission

- AAAI 2000

  Intermission (11 years)

- IJCAI 2011

  Intermission (10 years)

- IJCAI 2021 or AAAI 2021

  The Games Computers (and People) Play (part 3)

Sources

- Michael Bowling, Martin Mueller, David Silver, Nathan Sturtevant
- Images
  - http://www.gemanigames.com/html/mancala.html
  - http://www.cube20.org/
  - http://www.math.ubc.ca/~cass/courses/m308-02b
  - www.freeimageslive.co.uk
  - http://plus.maths.org
  - http://www.dailymail.co.uk/news
  - http://www-formal.stanford.edu
  - http://www.mcclatchydc.com
  - http://4.bp.blogspot.com
  - http://www.ubergizmo.com
  - http://chess-theory.com
  - http://www.popsci.com
  - http://www.nobelprize.org
  - http://wii.gamerzdb.com
  - http://blogto.com
  - http://britannica .com