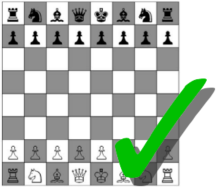
Game AI Challenges: Past, Present, and Future

Professor Michael Buro
Computing Science, University of Alberta, Edmonton, Canada



www.skatgame.net/cpcc2018.pdf

AI / ML Group @ University of Alberta

Edmonton, Alberta, Canada

- **Interested in World-class AI or ML research and spending time in Canada?**
- **15 Professors with 90+ graduate students focusing on AI and ML**
- **Our group is growing, so we are looking for more graduate students!**

UofA's Game AI Group

Jonathan Schaeffer
Heuristic Search, Computer Checkers

Martin Müller
Heuristic Search, Computer Go

Michael Buro
Heuristic Search, Video Game AI

Mike Bowling
Imperfect Information Game AI, Computer Poker

Vadim Bulitko
Real-Time Heuristic Search

Rich Sutton
Reinforcement Learning

Ryan Hayward
MiniMax Search, Computer Hex

Nathan Sturtevant
Single-Agent Search, Pathfinding

and 40+ grad students

My Research Interests

- Heuristic Search
- Game Theory
- Machine Learning (Deep RL in particular)
- Adversarial and Hierarchical Planning

Application Areas:

- Abstract Board Game AI
- Video Game AI
- Traffic Optimization

AI Goal

Overall goal: **Achieve Artificial General Intelligence (AGI)**

I.e., mastering any intellectual task that a human can

Current approach: Achieve **narrow** Artificial Intelligence in distinct problem domains

1. Pick intellectual task in which humans dominate
2. Work on AI system that performs equally good or better
3. Goto 1

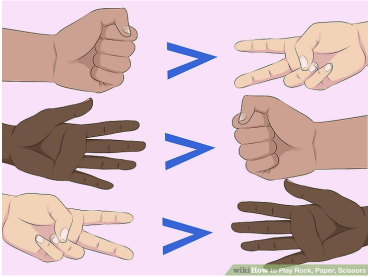
The hope is that this process leads to AGI

AI Research and Games

Games are convenient **testbeds** for studying most AI problems

They can be easily **tailored** to focus on individual aspects and human experts are often easily accessible

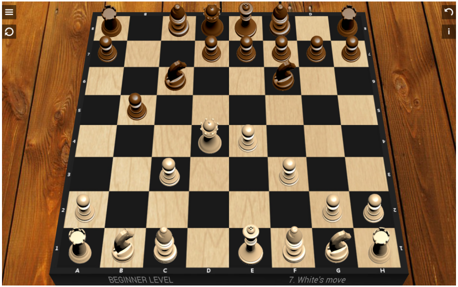
Example: Rock-Paper-Scissors (Study Imperfect Information Games)



Play video `vids/rps.mp4`

Challenge 1: Can machines think like humans?

First AI benchmark problem: Chess
Became the "Drosophila of AI" [Play video vids/ChessBlitz.mp4]



- Classic 2-player perfect information zero-sum game
- There are ≈ 36 legal moves on average
- Games last ≈ 80 moves on average
- There are $\approx 10^{44}$ reachable positions

Chess AI Timeline

194x J. von Neumann, A. Turing, C. Shannon: can a machine be made to think like a person, e.g. play Chess?

1951 First Chess programm (D. Prinz)

1962 MIT program can defeat amateur players

1979 **Chess 4.9** reaches Expert level (mostly due to faster hardware)

1985 **Hitech** reaches Master level using special purpose Chess hardware

1996 IBM's **Deep Blue** reaches Grand Master level

1997 **Deep Blue** defeats World Champion G. Kasparov 3.5-2.5

...

Kasparov vs. Deep Blue

Play video vids/KasparovDeepBlue.mp4

Man vs. Machine in 1997

G. Kasparov	Name	Deep Blue
1.78m	Height	1.95m
80kg	Weight	1,100kg
34 years	Age	2 years
50 billion neurons	Computers	512+64 processors
2 pos/s	Speed	200,000,000 pos/s
Extensive	Knowledge	Primitive
Electrical/chemical	Power Source	Electrical
Enormous	Ego	None

The Secret?

Brute-force search

- Consider all moves as deeply as possible (time permitting)
- Some moves can be provably eliminated
- 200,000,000 moves per second versus Kasparov's 2 (using special purpose Chess hardware)
- 99.99% of the positions examined are silly by human standards

  Most considered playing lines are of the form: I make a bluder, followed by you making a blunder, etc.
- Lots of search — and little knowledge

Tour de force for engineering

Knowledge — Sort Of

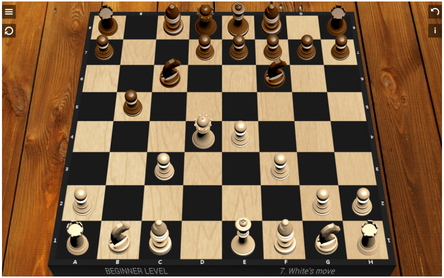
- Opening moves prepared by Chess experts
- Simple evaluation features evaluated in parallel by hardware (material, mobility, King safety, etc.)
- A few parameters tuned using self-play

Chess AI Epilogue

- Since 2007 man is no longer competitive in Chess
- Playing strength of Chess programs increased steadily by using machine learning to improve evaluation and search parameters
- In 2017 Deepmind's **AlphaZero-Chess** program soundly defeated **Stockfish** — the reigning World Champion program by using **Monte Carlo Tree Search** and **deep neural networks** trained via self-play

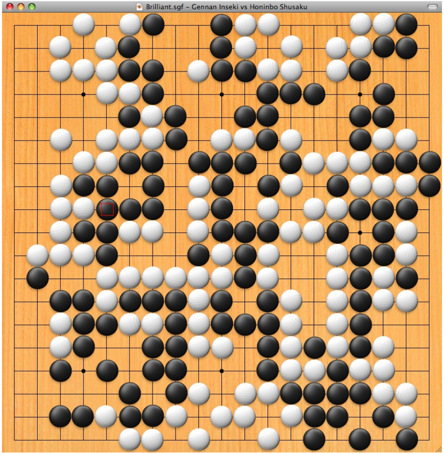
Challenge 2: Can machines handle much more complex games?

Chess



- ≈ 36 legal moves
- ≈ 80 moves per game
- $\approx 10^{44}$ positions

19$\times$19 Go (wéiqí)



- ≈ 180 legal moves
- ≈ 210 moves per game
- $\approx 10^{170}$ positions

The Problem? (2006)

Brute-force search will not work, there are too many variations!

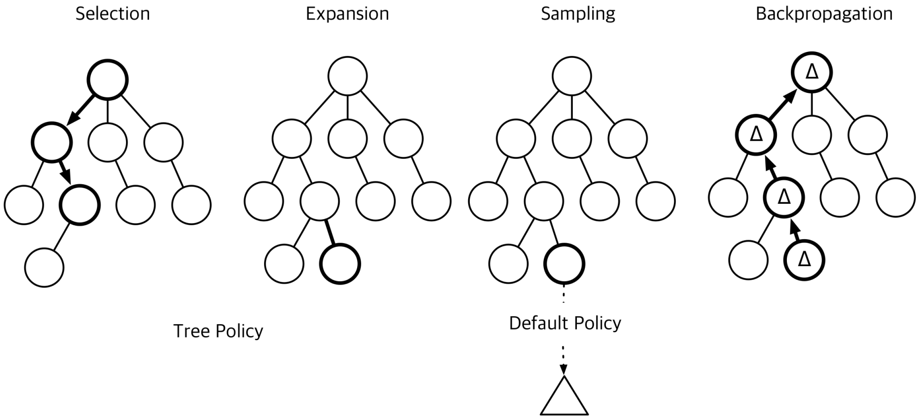
- The only approaches we knew of involved extensive knowledge
- Roughly 60 major knowledge-based components needed
- Program is only as good as the weakest link
- Game positions couldn't be evaluated accurately and quickly like in Chess

Even after 20 years of research we had no idea how to tackle this domain effectively with computers

It took two breakthroughs ...

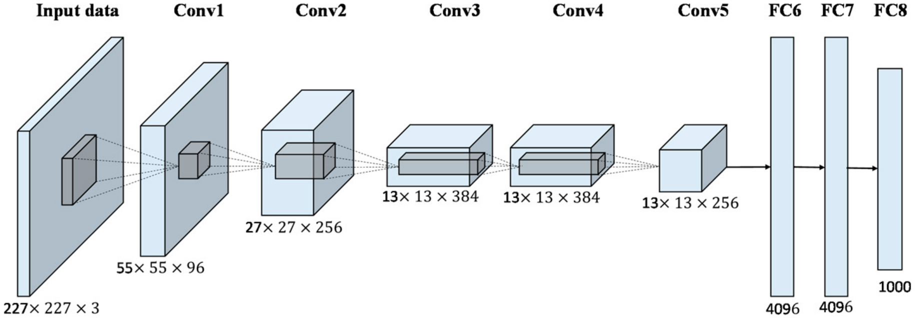
Breakthrough 1: Monte Carlo Tree Search

UCT (2006), MCTS (2007)



Breakthrough 2: Deep Convolutional Networks

AlexNet (2012)



Putting Everything Together ...

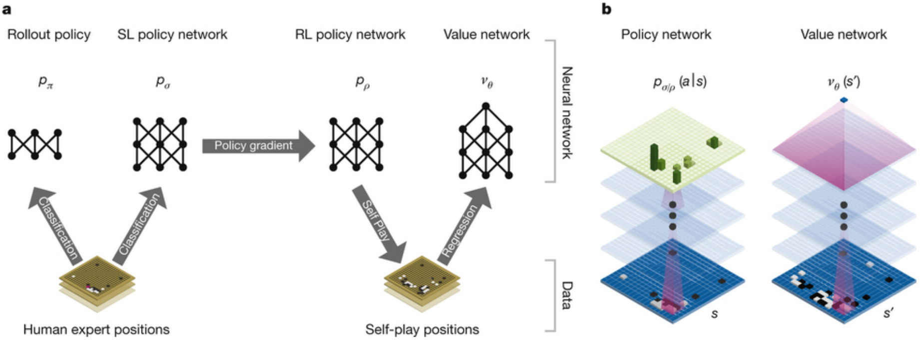
After 2 years of work on **AlphaGo** led by D. Silver (**UofA alumnus**) Google Deepmind challenges Lee Sedol — a 9-dan professional Go player in March 2016

AlphaGo wins 4-1

A historic result — AI mastered man's most complex board game!

The Secret?

- Training policy and value networks with human master games and self-play (networks have hundreds of millions of weights)
- Fast network evaluations using 176 GPUs
- Distributed asynchronous Monte Carlo Tree Search (1,200 CPUs)

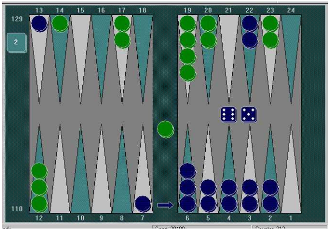


Go AI Epilogue

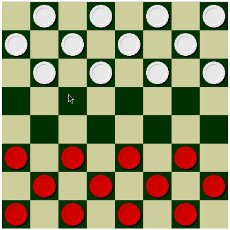
- After the Sedol match **AlphaGo-Master** wins 60-0 against strong human players (playing incognito on a Go server)
- **AlphaGo-Zero** wins 100-0 against **AlphaGo-Lee** in 2017 (not depending on human expert games)
- Human Go experts don't understand how **AlphaGo-Zero** plays

Man is no longer competitive in Go

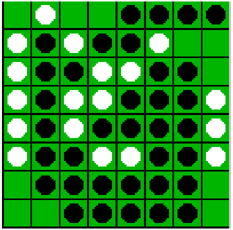
Some other classic games ...



Backgammon



Checkers



Othello/Reversi

Scrabble

... and their respective AI milestones

1992 G. Tesauro's **TD-Gammon** uses TD-learning to teach itself to play Backgammon at expert level via self-play

1994 UofA's J. Schaeffer's **Chinook** wins the Checkers World Champion title. Its strenghts stems from using a large pre-computed endgame database

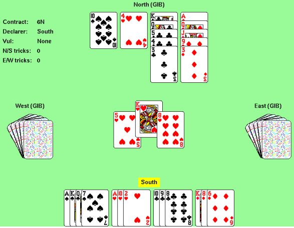
1997 M. Buro's **Logistello** defeats reigning Othello World Champion T. Murakami 6-0. It's evaluation consists of hundred-thousands of parameters optimized by sparse linear regression. It also uses aggressive forward pruning and a self-learned opening book

1998 B. Sheppard's **Maven** wins 9-5 against A. Logan, an expert Scrabble player. **Maven** uses a 100,000 word dictionary and letter rack simulations

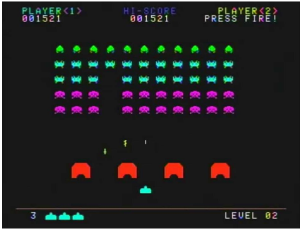
2007 **Chinook**, now using a 10-piece endgame database (13 trillion positions), solves Checkers: it's a draw

Beyond Classic Perfect Information Games I

Poker



Contract Bridge



Atari 2600 Games

DOTA 2

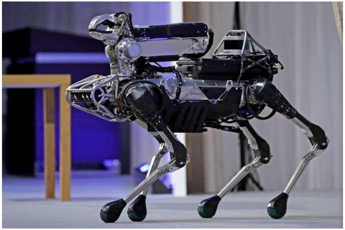
Quake 3

StarCraft 2

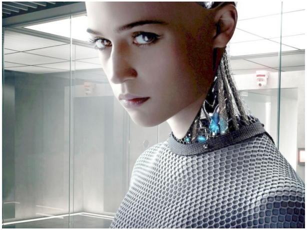
Beyond Classic Perfect Information Games II

Jeopardy! [Watson]

Autonomous Cars [Waymo]



Agile Robots [Boston Dynamics]



Smart Robots [Ex Machina]

Jeopardy!

- General knowledge clues are given to contestants
- They have to answer in the form of a question, quickly
- Example:

  Category: Rhyme Time

  Clue: It's where Pele stores his ball.

  Answer: What's a soccer locker?

Some Recent Milestones

2008 UofA's limit Texas Hold'em Poker program **Polaris** wins against human experts

2008 UofA's Skat program **Kermit** reaches expert level

2011 IBM's **Watson** defeats the best Jeopardy! players

2015 Google Deepmind creates an AI system that plays 49 Atari 2600 video games at expert level using DQN learning

2015 A UofA team led by M. Bowling solves 2-player limit Texas Hold'em Poker

2017 UofA's **DeepStack** and Carnegie Mellon's **Libratus** no-limit-Texas Hold'em Poker programs defeat professional players

2018 OpenAI creates a system that can play DOTA-2 at expert level

2018 Google Deepmind builds Quake 3 bots that coordinate well with teammates

Recent Game AI Trends

- Train deep neural networks by supervised and reinforcement learning at HUGE scale
  (E.g., OpenAI used 128,000 CPU cores for DOTA-2)
- Networks often have hundreds of millions of weights
- They are trained using millions of self-played games
- Clever feature encoding is less relevant, having more training data currently seems more important

  AlphaZero-Chess learned to play super-human Chess via self-play without feature engineering
- Focus is on making machine learning more data efficient, and to figure out how to deal with large action sets in real-time

More Challenges

- More than two agents
- Non-zero sum payoffs (e.g., agent cooperation)
- Partial state observability
- Huge action sets
- Modeling agents and adjust quickly
- Real-time decision constraints
- Acting in the world requires learning suitable abstractions
- ...

New AI Challenge Problem

After Chess and Go, the next big milestone is defeating a World-Class player in Real-Time Strategy (RTS) video games, e.g., StarCraft 2

Play video `vids/combat.wmv` Play video `vids/rts-pros.mp4`

Obstacles

- Partial observability ("Fog of War")
- Huge branching factor, rendering traditional search useless
- Action effects often microscopic and rewards are delayed
- Real-time constraints (if no command issued, game proceeds)
- No explicit forward model exists which complicates search

Can neural networks be trained to play RTS games well?

- Blizzard Entertainment released over 1 million human game replays!

We are working on it, but Google DeepMind is on the case, too

Does anyone have 100k idle CPU cores and 100 GPUs to help us?

State of the Art

- Build order optimization (**"build things quickly"**)
- Small-scale combat using Minimax search (**"micro"**)
- Scripted "macro" strategy (**"what to build and when?"**)

StarCraft AI systems are not competitive yet

Things being tried:

- Training networks for mini games (e.g., small-scale combat)
- Learning "macro" strategies from game replays
- Hierarchical search mimicing military command and control

Also: Multi-Player Card Games

In "simple" abstract imperfect information team games such as Spades, Contract Bridge, Skat, or Dou Dizhu

- Humans can quickly infer hidden state information
- Humans quickly discover opponent and partners' strengths and weaknesses and act accordingly
- Human players use sophisticated signalling schemes

Computers don't yet, but will hopefully soon

Conclusions I

- Game AI is a main driver of AI research since the 1950s
- It allows us to compare AI systems with human experts head on
- It is competitive and fun and can model many aspects of human decision making in adversarial and cooperative settings
- Neural networks and search form a powerful combination – human experts are baffled by how **AlphaGo-Zero** and **AlphaZero-Chess** play

Conclusions II

- Game AI research is now moving towards much more difficult problems such as tackling multi-player video games with imperfect information and huge action sets
- To compete with the DeepMinds of the World, academics need help: we need thousands of CPU cores and hundreds of GPUs to replicate existing research and to test our own ideas
- Please join us, working on Game AI is FUN and REWARDING!

"What do you mean, no questions?"

[The New Yorker]