



Lee Sedol,  
9 Dan professional

*Faculty of Science*

# Mastering the Game of Go - ~~Can~~ ~~How~~ Did a Computer Program Beat a Human Champion???

Martin Müller  
Computing Science  
University of Alberta



---

# In this Lecture:

---

- ❖ The game of Go
- ❖ The match so far
- ❖ History of man-machine matches
- ❖ The science
  - ❖ Background
  - ❖ Contributions in AlphaGo
  - ❖ UAlberta and AlphaGo
- ❖ The future



Source: <https://www.bamssoftware.com>

# The Game of Go

---

# Go

---

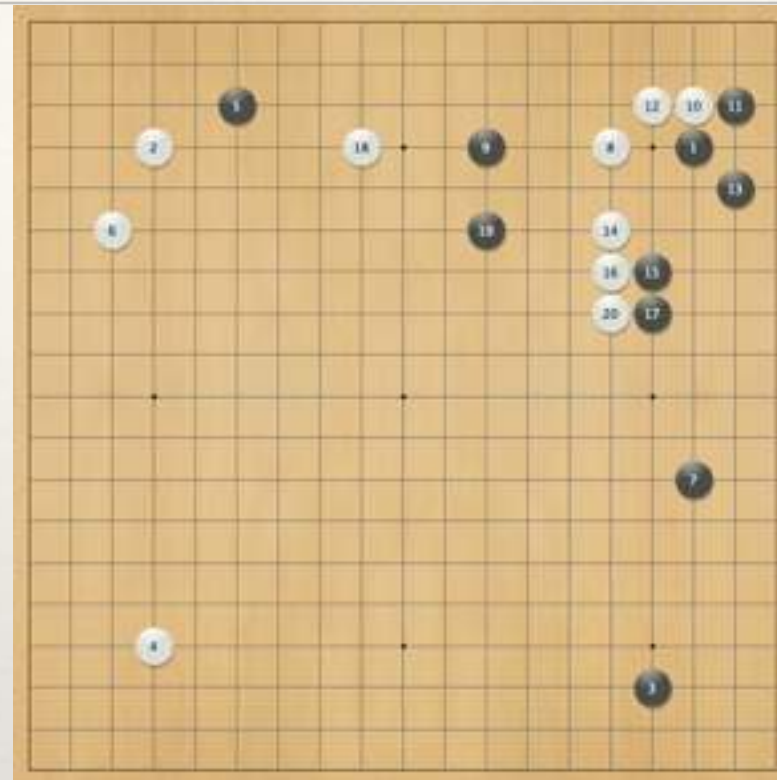
- ❖ Classic Asian board game
- ❖ Simple rules, complex strategy
- ❖ Played by millions
- ❖ Hundreds of top experts - professional players
- ❖ Until now, computers weaker than humans



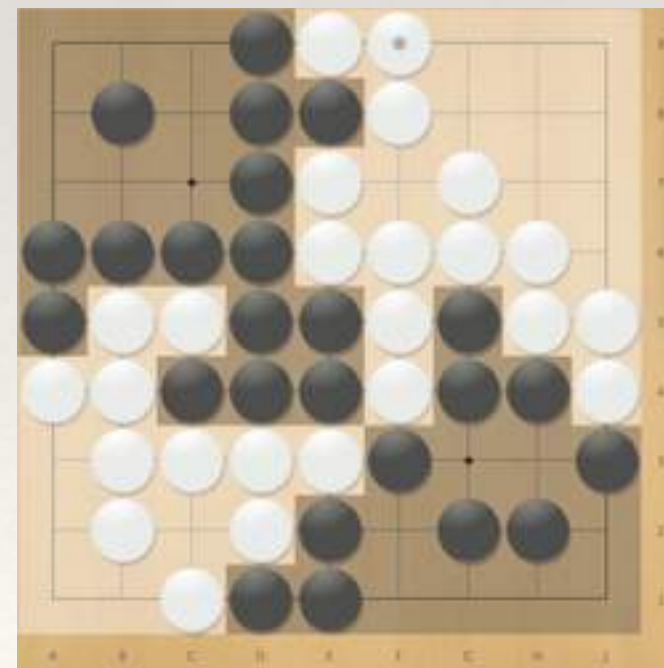


# Go Rules

- ❖ Start: empty board
- ❖ Move: Place one stone of your own color
- ❖ Goal: surround
  - ❖ Empty points
  - ❖ Opponent (capture)
- ❖ Win: control more than half the board
- ❖ *Komi*: compensation for first player advantage



The opening of game 1



Final score on a 9x9 board

# The Match

---

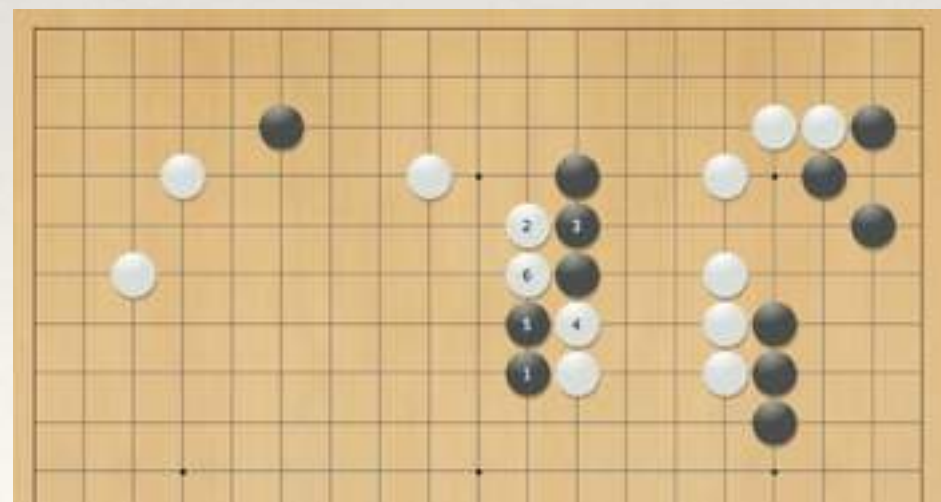
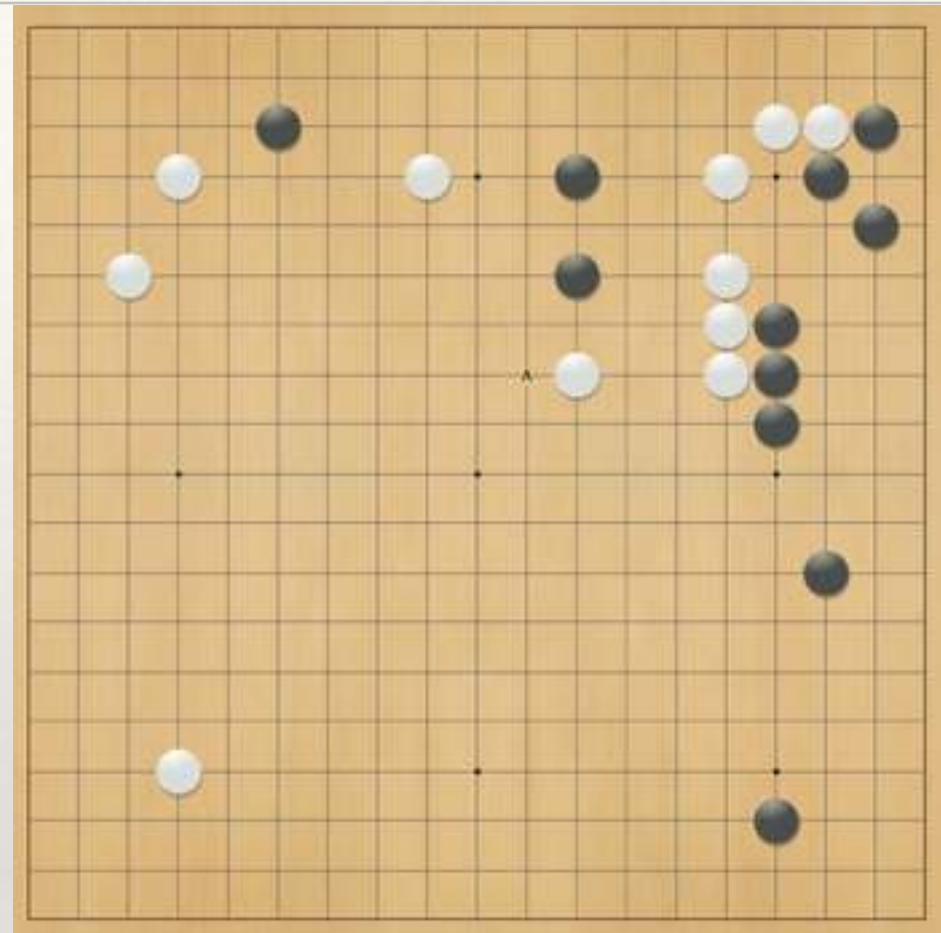
# Lee Sedol vs AlphaGo

---

Name	Lee Sedol	AlphaGo
Age	33	2
Official Rank	9 Dan professional	none
World titles	18	0
Processing Power	1 brain	about 1200 CPU, 200 GPU
Match results	loss, loss, loss, <b>win</b> , ?	<b>win, win, win</b> , loss, ?
Go Experience	Thousands of games against top humans	Many millions of self-play games

# The Match So Far: Game 1

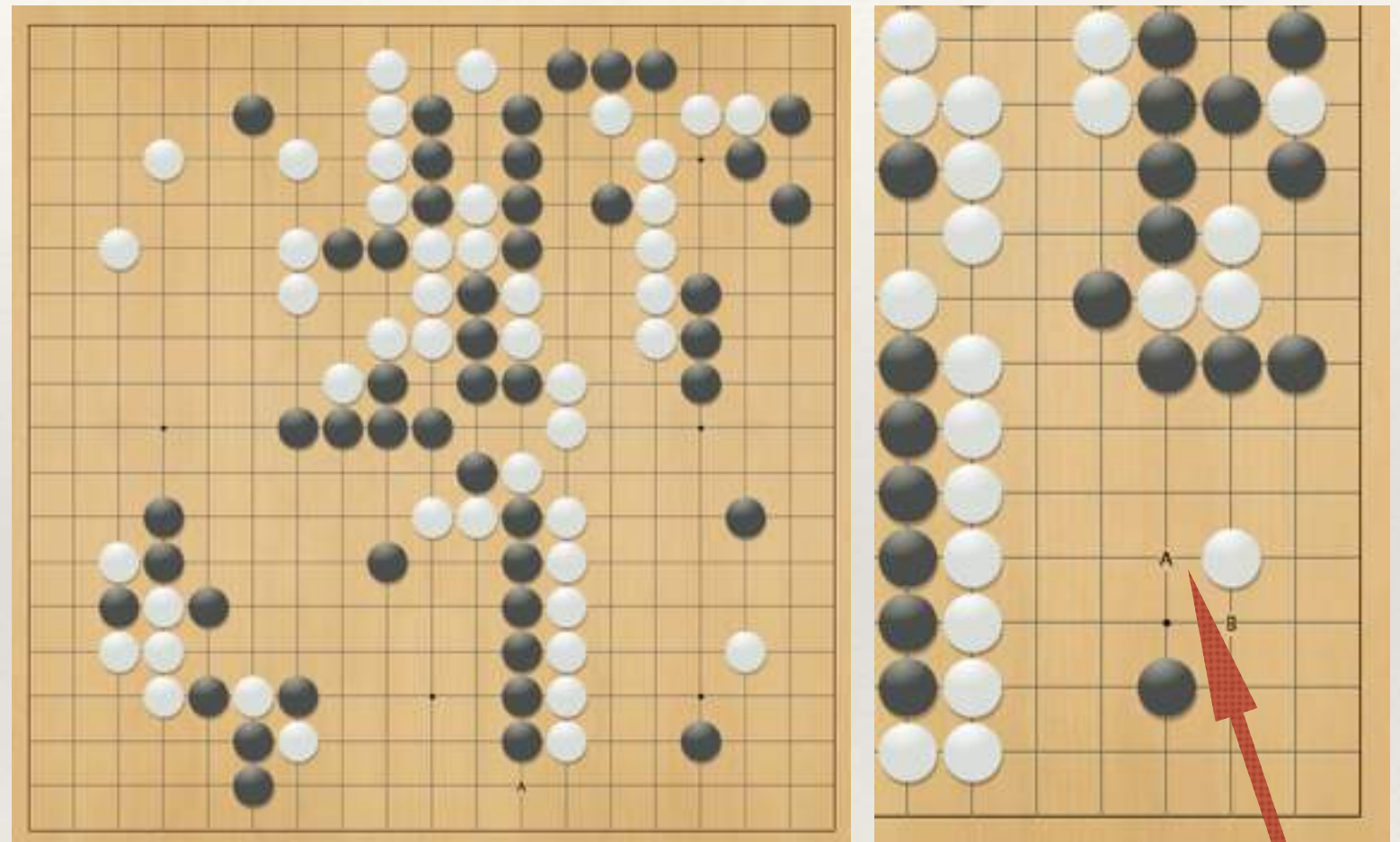
- ❖ Black: Lee Sedol
- ❖ White: AlphaGo
- ❖ Lee Sedol plays all-out at A on move 23. “Testing” the program?
- ❖ AlphaGo counterattacks very strongly and gets the advantage





# Game 1 continued

- ❖ AlphaGo makes a mistake in the lower left. Lee is leading here
- ❖ AlphaGo does not panic and puts relentless pressure on Lee
- ❖ Lee cracks in the late middle game. Move A on the right side may be the losing move



186 moves.

AlphaGo wins by resignation



# Game 1 Reactions

- ❖ Shock. Disbelief.
- ❖ Huge media interest worldwide



About 79,800 results (0.45 seconds)



**Go Grandmaster Lee Sedol Grabs Consolation Win Again...**

WIRED - 11 hours ago

But Lee Sedol's win in Game Four is a reminder that even the most ... before the game began, one big question remained: Does AlphaGo have ...

Go champion Lee Se-dol strikes back to beat Google's DeepMind AI ...

The Verge - 11 hours ago

**AlphaGo beats Lee Sedol in third consecutive Go game**

The Guardian - Mar 12, 2016

Google's AlphaGo Has Won Its Third Match Against Go World ...

Opinion - Gizmodo - Mar 12, 2016

Google's AlphaGo isn't taking over the world, yet

In-Depth - CNET - Mar 12, 2016

Google's A.I. Beats Human Champ at Go for Third Straight Time

Blog - Slate Magazine (blog) - Mar 12, 2016



YouTube



euronews



The Verge



The Guardian



Business Ins...



ZDNet

**Explore in depth** (752 more articles)



**Google DeepMind, humanity and a freakishly hard game**

CNBC - Mar 8, 2016

On Wednesday, Google's AI system AlphaGo defeated Lee Sedol, one of the world's best players of the ancient (and incredibly complex) ...

Google's DeepMind AlphaGo beats world Go champion in first of five ...

Daily Mail - Mar 9, 2016

**AI Challenger Defeats Go Grandmaster Lee**

International - KBS WORLD Radio News - Mar 8, 2016

Google's AlphaGo AI defeats human in first game of Go contest

In-Depth - The Guardian - Mar 9, 2016

Google's AI Has Won Its First Match Against Go World Champion ...

Opinion - Gizmodo - Mar 9, 2016

**A.I. 2, Human Go Champion 0**

Blog - Slate Magazine (blog) - Mar 10, 2016



YouTube



The Guardian



WIRED



Daily Mail



CBC.ca



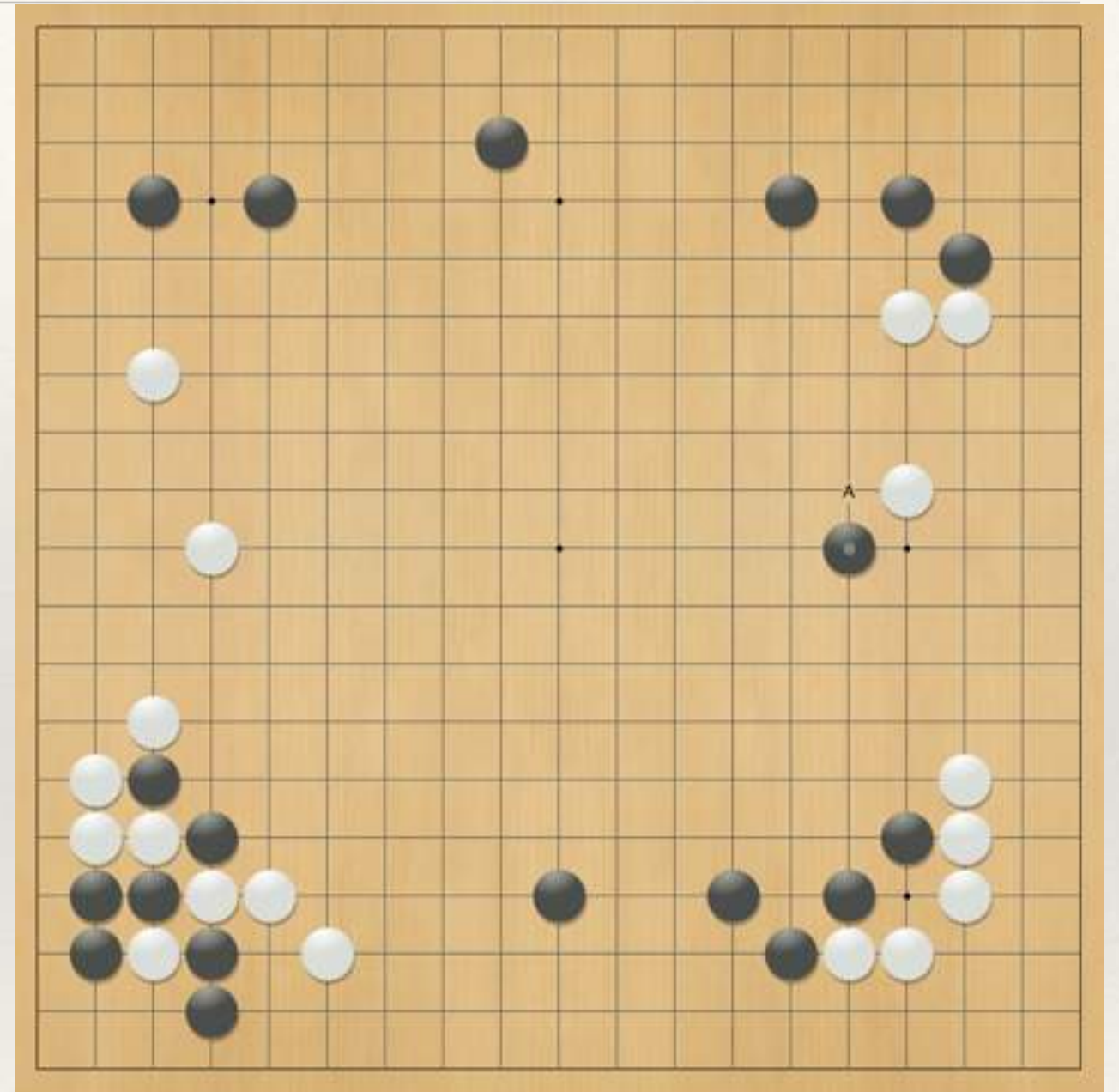
Slate Magazl...

**Explore in depth** (885 more articles)



# The Match So Far: Game 2

- ❖ Lee completely changes his style
- ❖ With white, he plays very safe, solid moves
- ❖ AlphaGo as Black plays creative, flexible moves and gradually gets ahead
- ❖ A masterpiece for AlphaGo



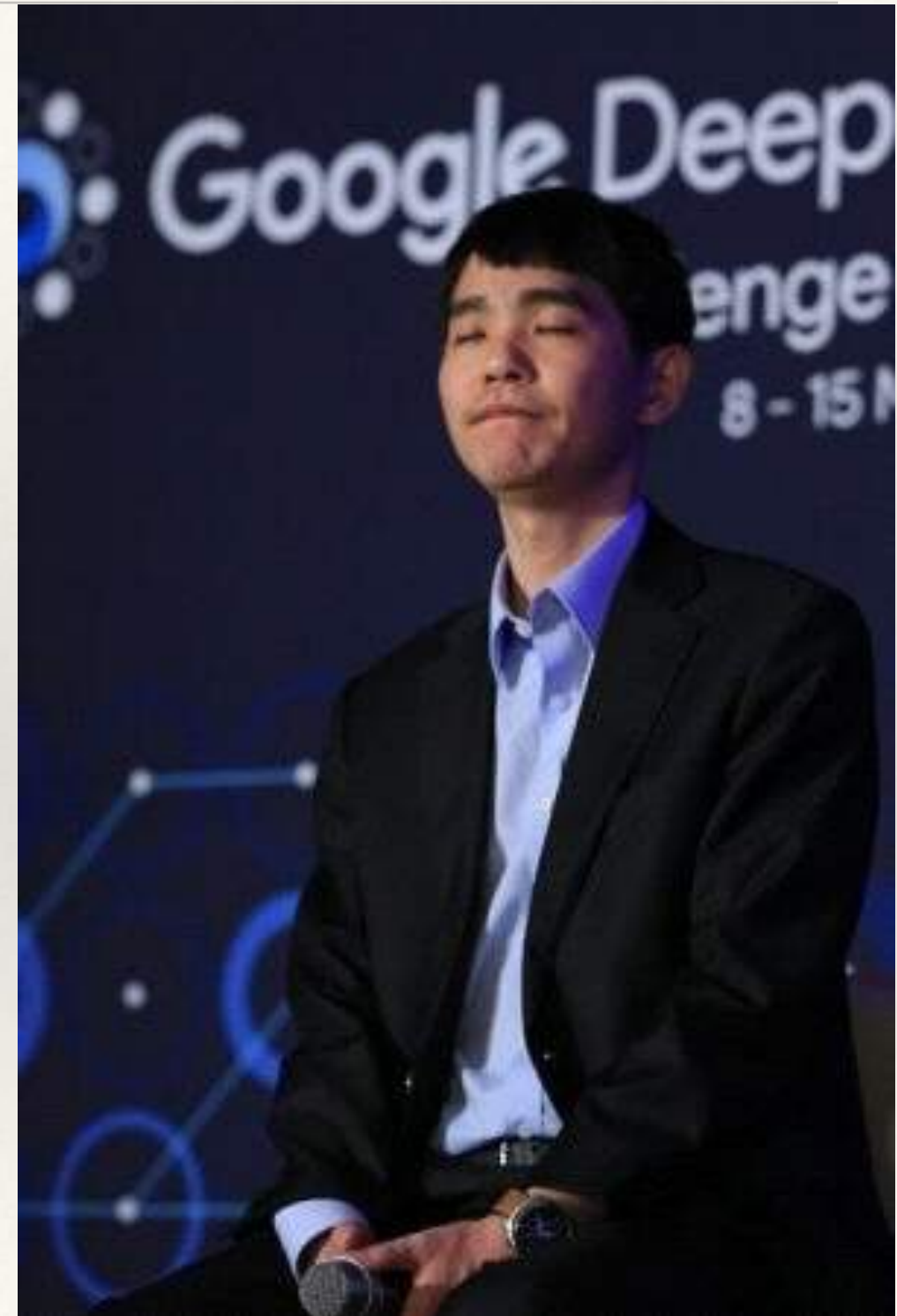
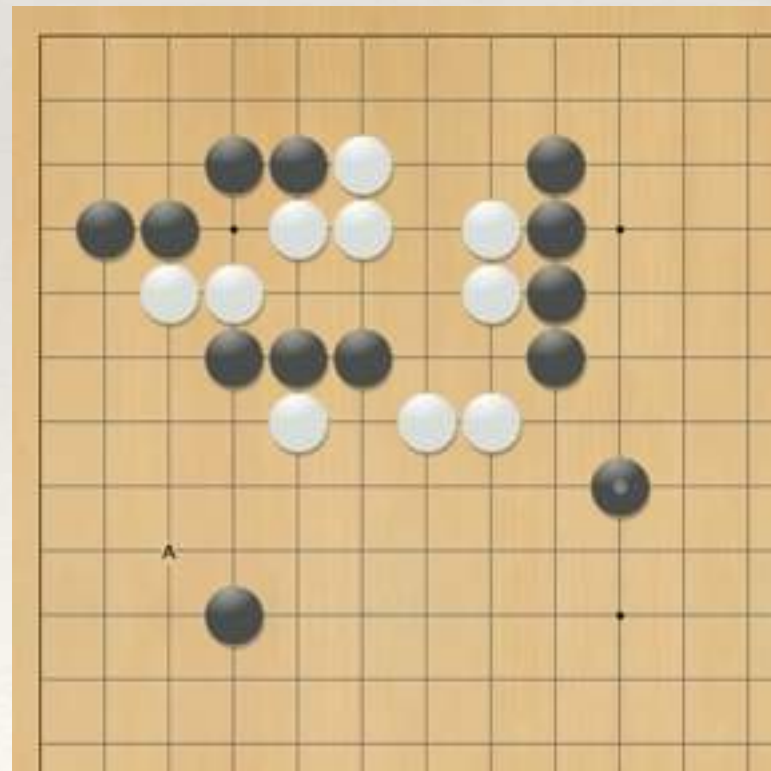
211 moves.

AlphaGo wins by resignation

# The Match So Far: Game 3

- ❖ Almost flawless game by AlphaGo
- ❖ Lee strongly attacks in the first corner, but AlphaGo turns the tables step by step
- ❖ AlphaGo “relaxes” after that but keeps a safe lead
- ❖ Professionals:

- ❖ “It played so well that it was **almost scary**”
- ❖ “Could 31 be the losing move?”





# The Match So Far: Game 4

- ❖ Lee's new strategy: take lots of profit, then stake the game on invading the center
- ❖ Lee came very close to losing all the center
- ❖ Then he produced a fantastic "tesuji"
- ❖ AlphaGo needed to compromise here. But it still thought it could get everything, and made things much worse for itself



Lee Sedol wins by resignation

# Game 4 Discussion

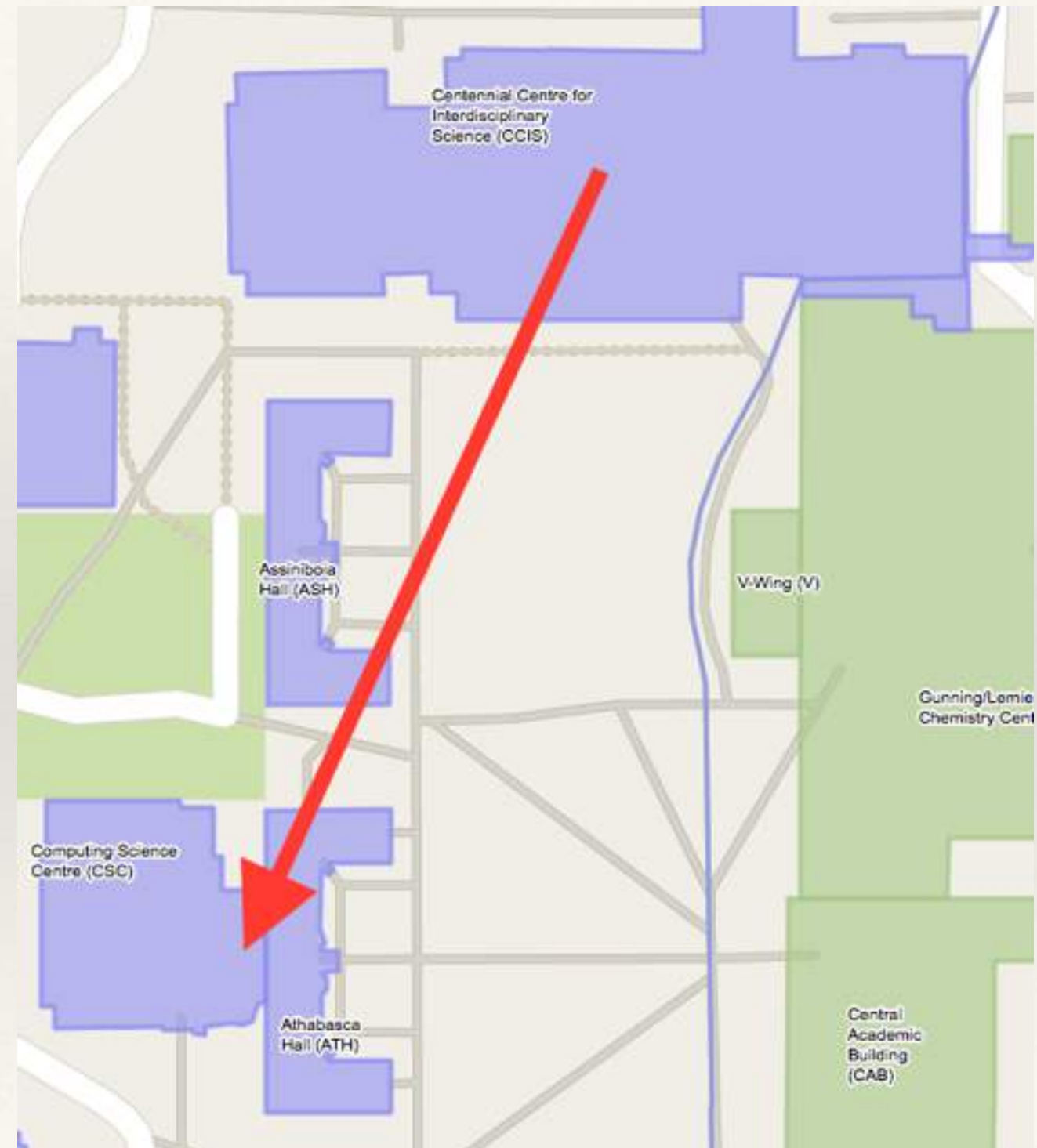
- ❖ Why did AlphaGo miss this?
- ❖ Complex tactical fight
  - ❖ Multiple targets
  - ❖ Many threats
  - ❖ No subset of threats works, but all together they work
- ❖ Computers lost in combinatorial explosion?
- ❖ Humans can precisely plan
- ❖ Human's only (?) hope: out-calculate the computer (!)





# Game 5 Tonight - Watch With Us

- Game 5 is the last game of the match
- It is very important:
  - Was game 4 a “fluke” ...
  - ...or did Lee figure out how to beat AlphaGo?
- Tonight from **10pm**
- Viewing party on campus, in room CSC 3-33
- Live Youtube feeds with professional commentaries



# History of Computer Games Research and Man-Machine Matches



---

# Prehistory

---



Ernst  
Zermelo



John von  
Neumann



Alan Turing

- ❖ Many pioneers of Computing Science worked on game theory or program designs
- ❖ Basis for all future work



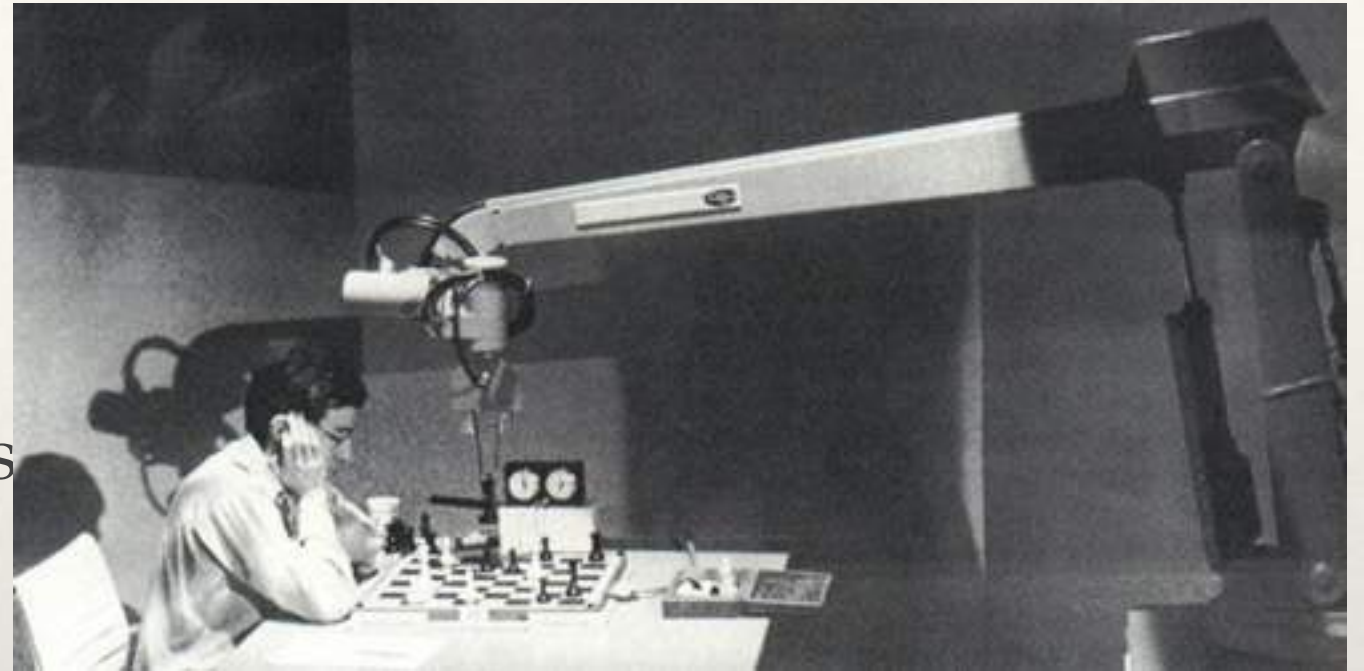
Claude  
Shannon



Norbert  
Wiener

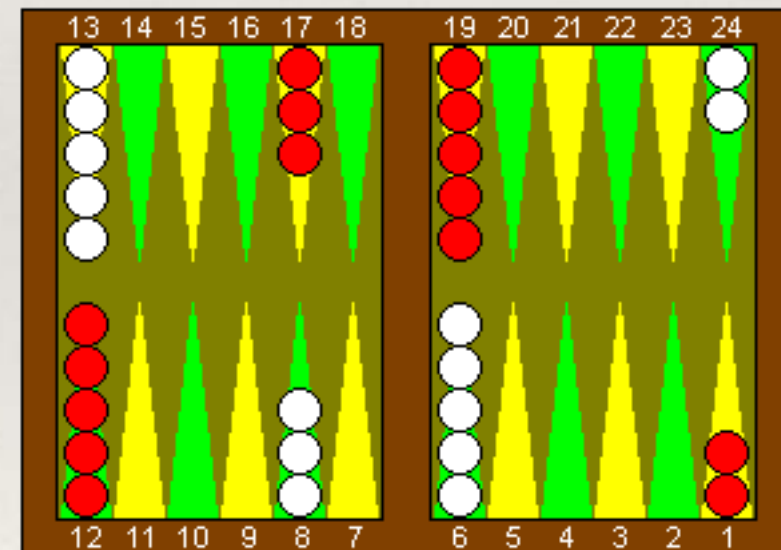
# Chess Man vs Machine

- ❖ David Levy's bet - no program can defeat me in 10 years
- ❖ Easy wins in 1977, much closer in 1978 and 1979 but David Levy wins
- ❖ 1989: Deep Thought easily defeats Levy, 4-0
- ❖ 1996, Kasparov wins 4-2 vs Deep Blue
- ❖ 1996, Kasparov loses 2.5-3.5 vs Deep Blue



# Backgammon Man vs Machine

- ❖ 1979: Berliner's BKG wins short exhibition match against Villa
  - ❖ Lucky with the dice...
- ❖ 1992: Tesauero, TD-gammon
  - ❖ Very close to top human experts
- ❖ Current programs are almost perfect



**Figure 2.** An illustration of the normal opening position in backgammon. TD-Gammon has sparked a near-universal conversion in the way experts play certain opening rolls. For example, with an opening roll of 4-1, most players have now switched from the traditional move of 13-9, 6-5, to TD-Gammon's preference, 13-9, 24-23. TD-Gammon's analysis is given in Table 2.



# Checkers Man vs Machine

- ❖ Chinook program developed over decades by Jonathan Schaeffer and his group
- ❖ 1992+1994: Chinook vs Tinsley
- ❖ 2007: Checkers solved





# Othello Man vs Machine

- ❖ Logistello program developed by Michael Buro
- ❖ 1997:  
Logistello vs Murakami  
6 - 0



# Poker Man vs Machine

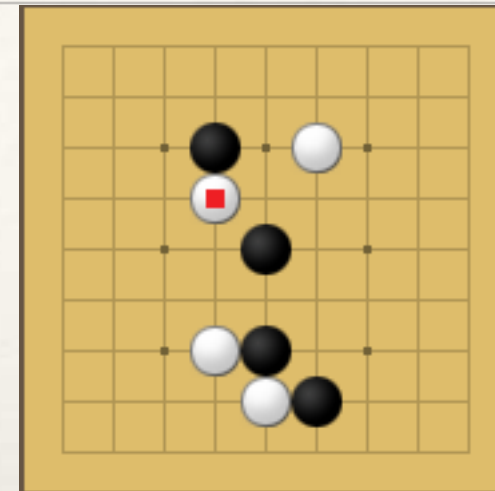
- ❖ UAlberta Poker research group, led by Mike Bowling
- ❖ 2007, 2008: Polaris vs Poker pros, Polaris wins in 2008
- ❖ 2015: Heads-up limit Texas hold'em is solved





# Go Man vs Machine

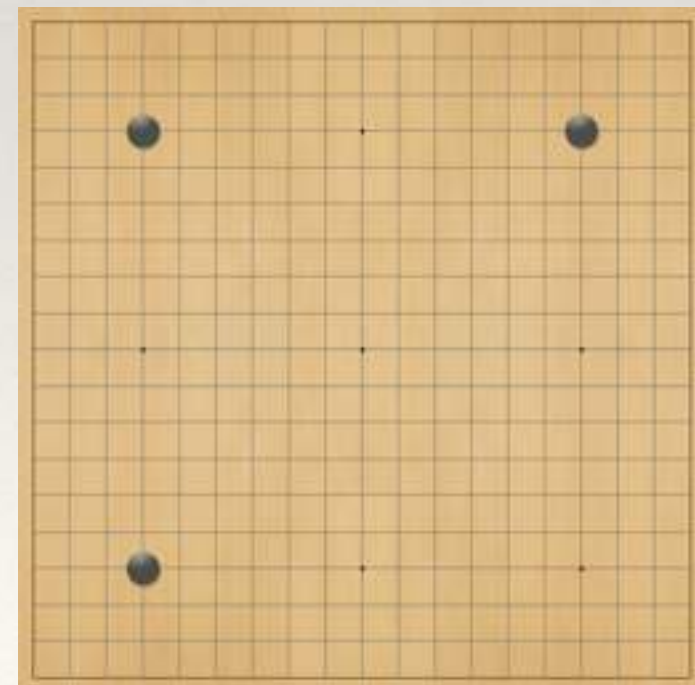
- ❖ 2009: Fuego (open source, mostly UAlberta)
  - ❖ First win against top human professional on 9x9 board
- ❖ 19x19 board: Many handicap matches (computer starts with an advantage)
- ❖ Before AlphaGo, about 3-4 handicap stones
- ❖ 2015: AlphaGo beats Fan Hui 2 Dan professional, no handicap
- ❖ 2016: AlphaGo - Lee Sedol



White: Fuego

Black: Chou Chun-Hsun 9 Dan

White wins by 2.5 points



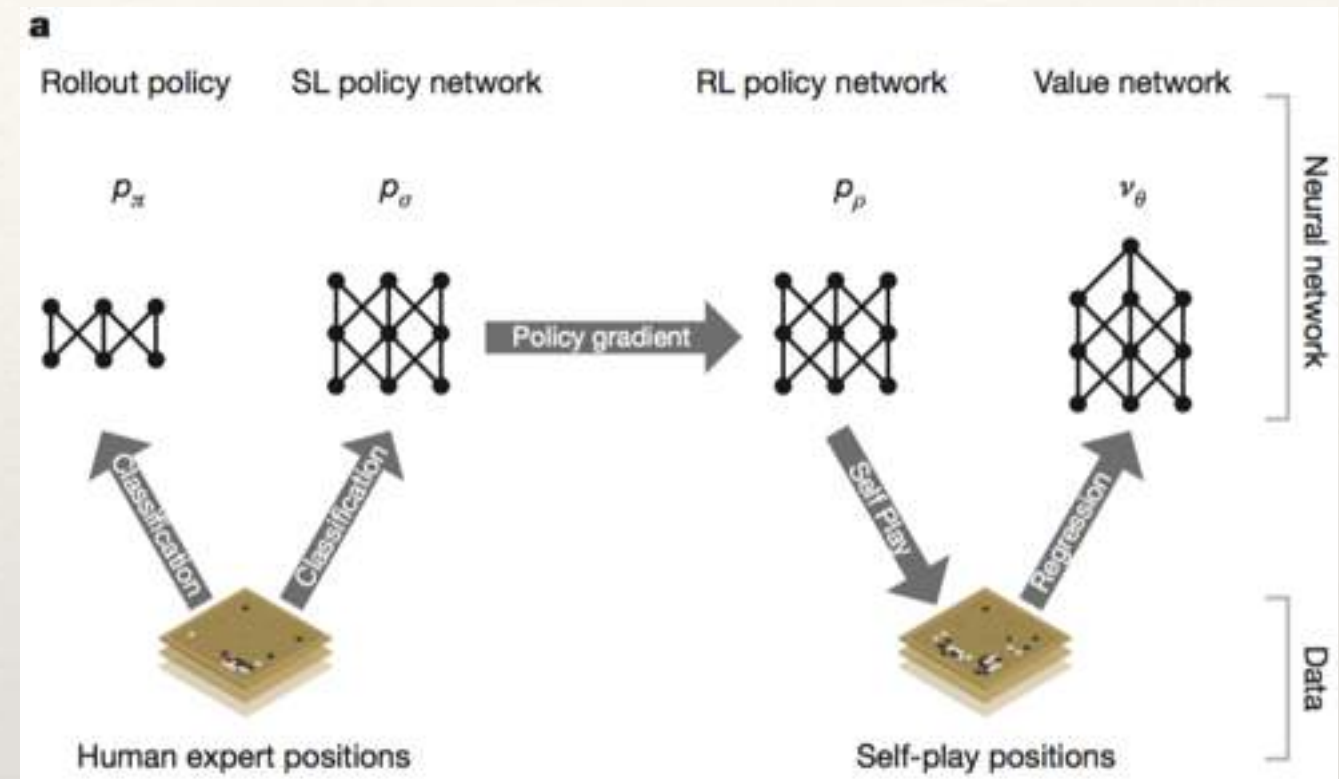
3 handicap  
starting  
position

# The Science



# The Science Behind AlphaGo

- ❖ AlphaGo builds on decades of research in:
  - ❖ Building high performance game playing programs
  - ❖ Reinforcement Learning
  - ❖ (Deep) neural networks



UAlberta is a world leader

# The Science - Background

# Making Complex Decisions

- ❖ We make decisions every moment of our lives
- ❖ What is the process that leads to our decisions?
- ❖ How to make good decisions?
- ❖ Consider many alternatives
- ❖ Consider short-term and long-term consequences
- ❖ Evaluate different options and choose the best-looking one

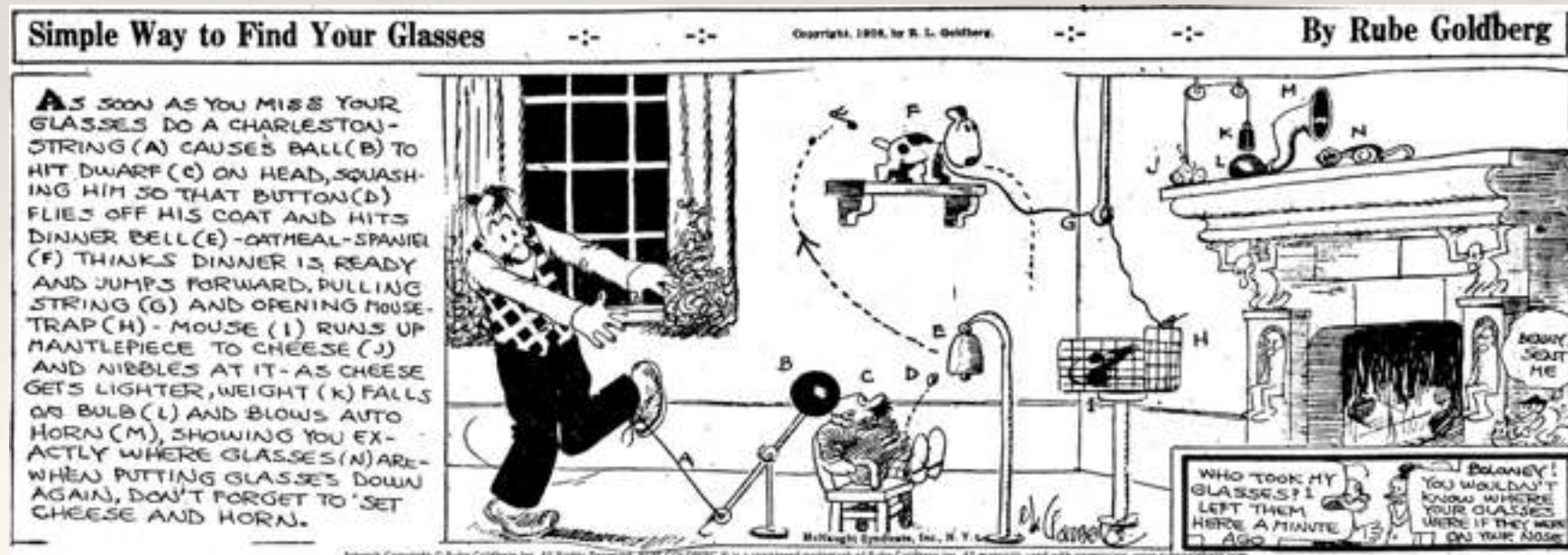


Image Source:  
<https://www.rubegoldberg.com>



# Making Sequential Decisions

- ❖ Loop:
  - ❖ Get current state of world
  - ❖ Analyze it
  - ❖ Select an action
  - ❖ Observe the world's response
  - ❖ If not done: go back to start of loop

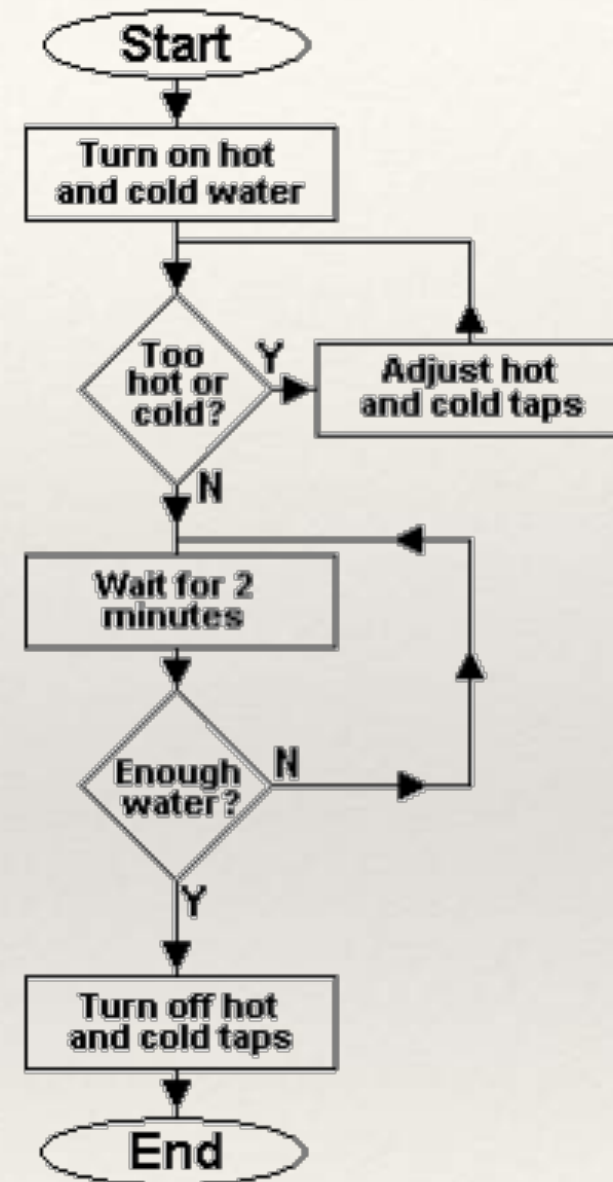
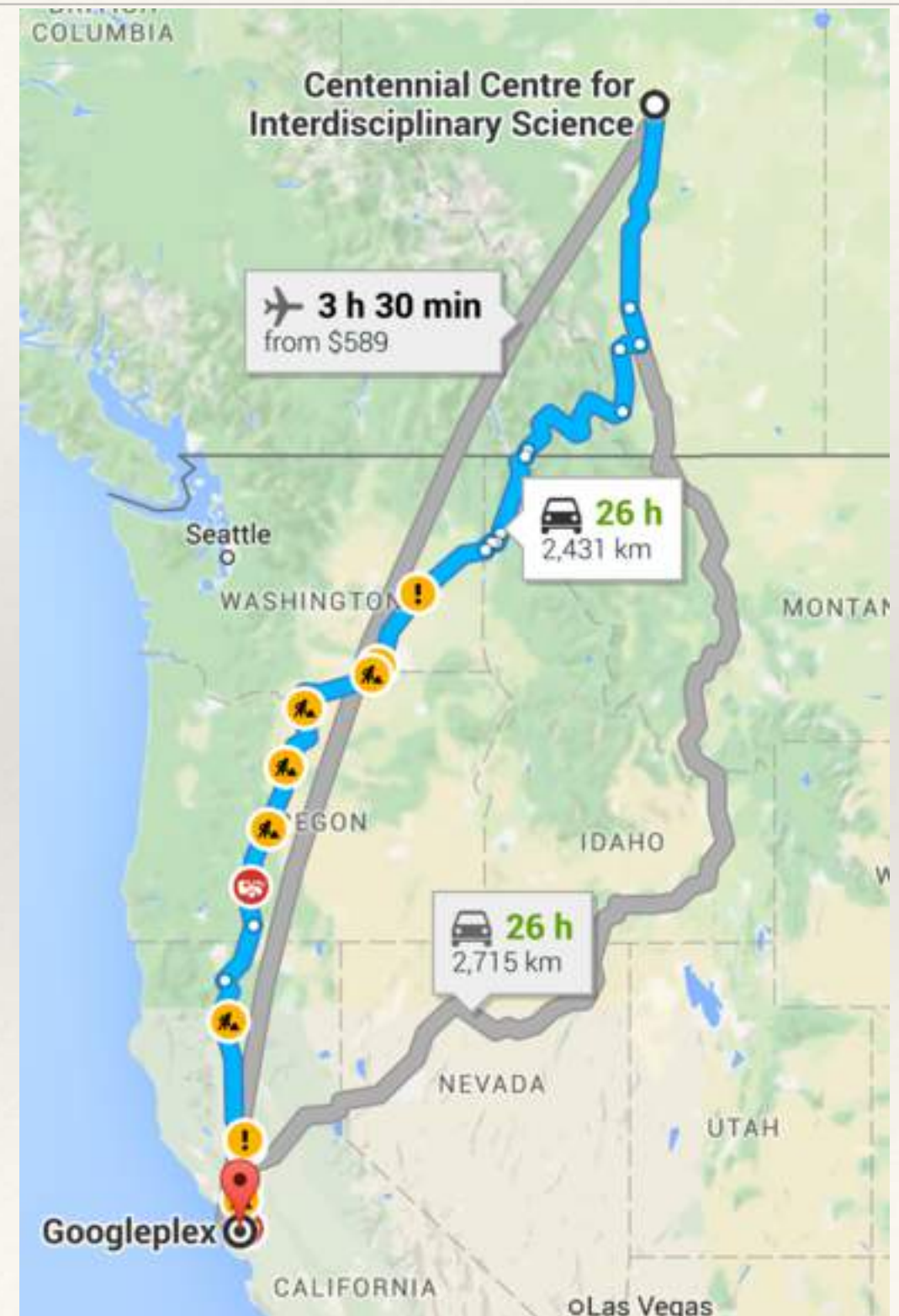


Image Source:

<http://www.mind-development.eu>

# Heuristic Search

- ❖ Heuristic search is a research area in computing science
- ❖ It is considered a part of the field of Artificial Intelligence
- ❖ It can be used for sequential decision-making problems
- ❖ Applications: automated planning, optimization problems, pathfinding, games, puzzles,...



---

# The Three Plus One Pillars of Modern Heuristic Search

---

- ❖ Three main ingredients:
  - ❖ Search
  - ❖ Knowledge
  - ❖ Simulations
- ❖ Plus one:
  - ❖ Machine learning to acquire knowledge
- ❖ We will see all of these used in AlphaGo
- ❖ Many other modern heuristic search methods also use those
- ❖ Examples:
  - ❖ planning
  - ❖ robot motion planning
  - ❖ mapping unknown terrain
  - ❖ other games



# Tree Search

- ❖ At each step in the loop:
- ❖ I need to choose one of my actions
- ❖ The world could react in one of many possible ways
- ❖ Drawing all possible sequences results in a (huge) tree

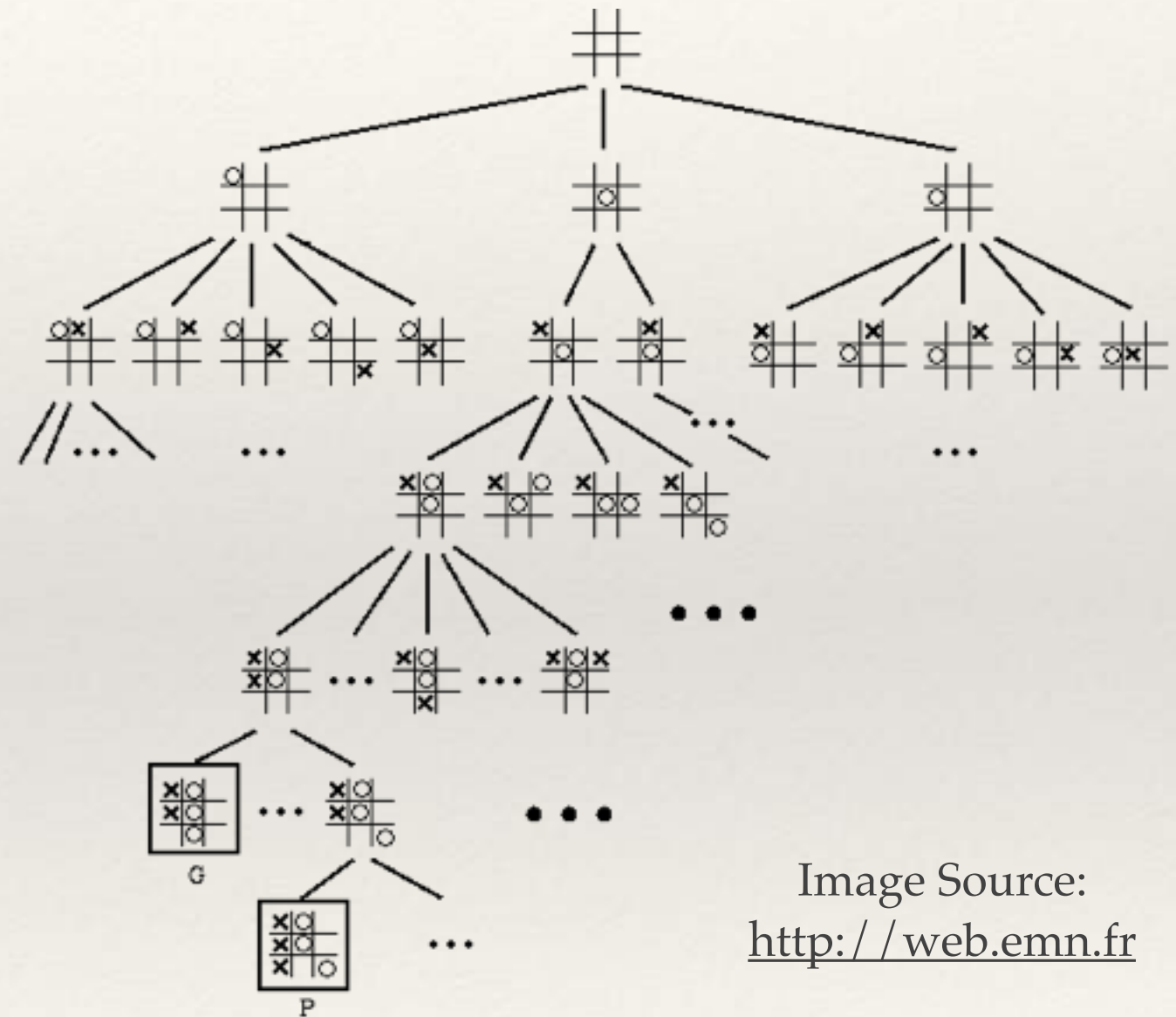
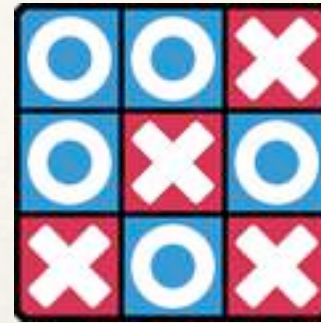


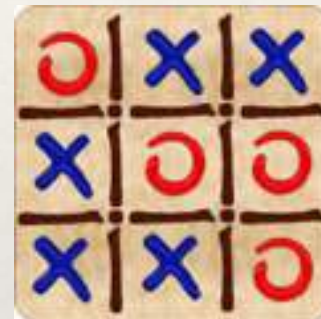
Image Source:  
<http://web.emn.fr>

# Domain Knowledge and Evaluation

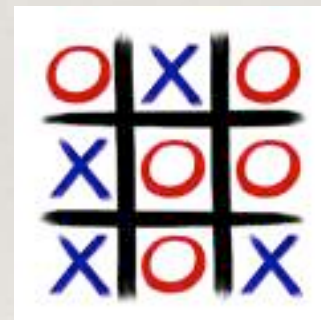
- ❖ We need to know if a sequence led to a good result
- ❖ Exact knowledge: we know the result for sure
- ❖ Heuristic knowledge: an estimate of the result
- ❖ Evaluation: mapping from a state of the world to a number
- ❖ How good or bad is it for us?



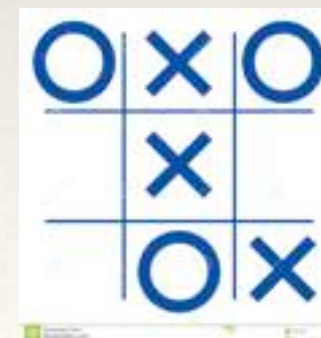
Win for X



Loss for X (Win for O)



Draw



What's your evaluation?

# Simulation

- ❖ For complex problems, there are far too many possible sequences
- ❖ Sometimes, there is no good evaluation
- ❖ We can sample long-term consequences by simulating many future trajectories

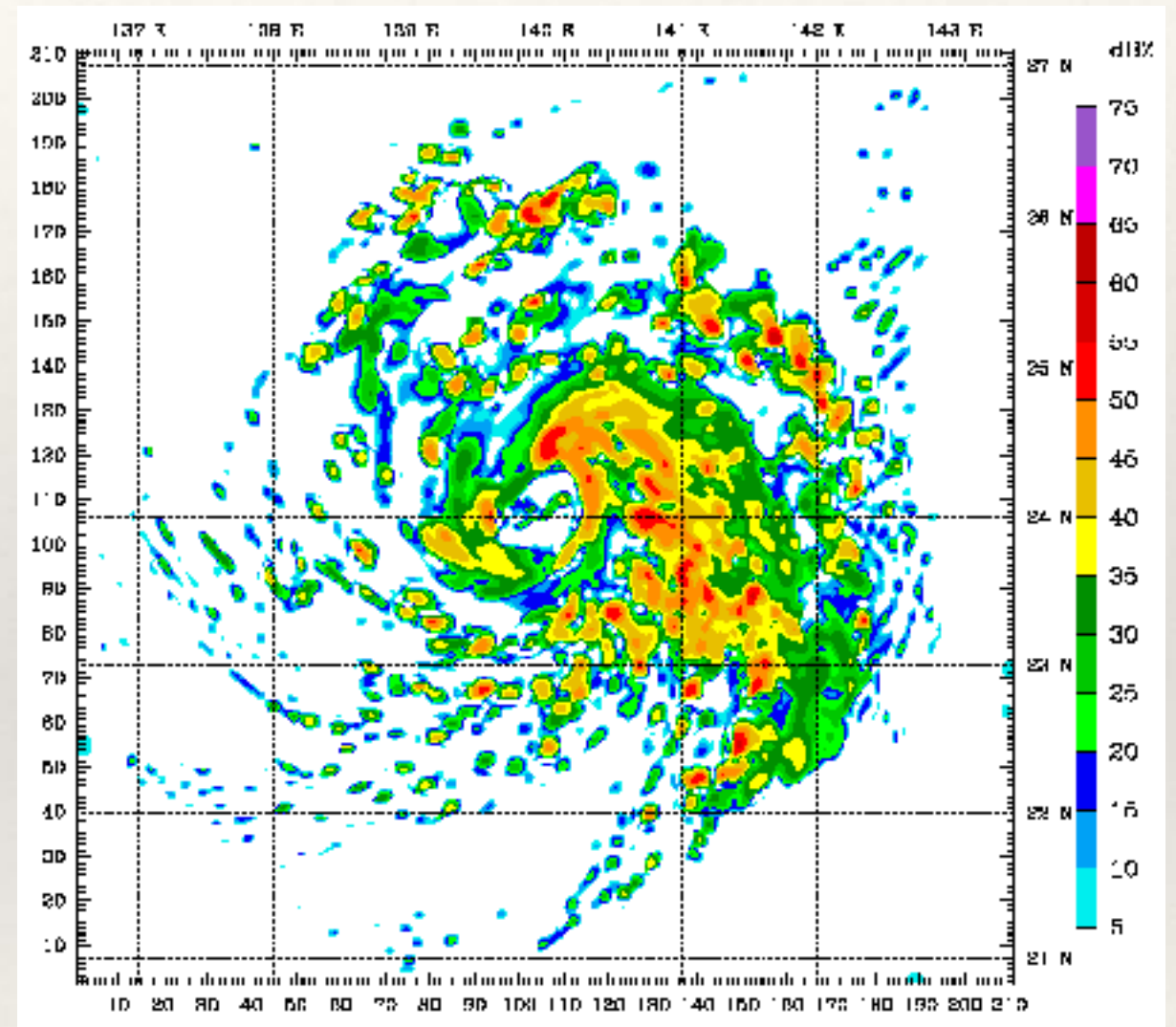


Image Source:

<https://upload.wikimedia.org>



---

# Computer Go Before AlphaGo

---

- ❖ **Search:**

  - Monte Carlo Tree Search

- ❖ Invented about 10 years ago

- ❖ First successful use of simulations for classical two-player games

- ❖ Scaled up to massively parallel (e.g. Fuego on 2000 cores on Hungabee)

- ❖ **Simulation:**

  - ❖ Play until end of game

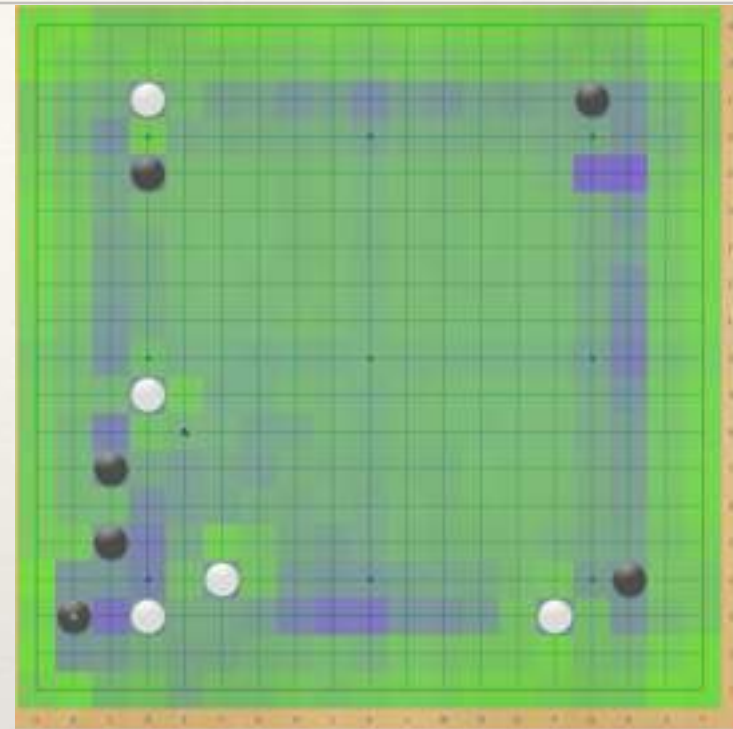
  - ❖ Find who wins at end (easy)

  - ❖ Moves in simulation: random + simple rules

  - ❖ Early rules hand-made, later machine-learned based on simple features

# Computer Go Before AlphaGo

- ❖ **Knowledge:**
- ❖ Fast, simple knowledge: used for move selection in simulation (“rollout policy”)
- ❖ Slower, better knowledge: used for move ordering in tree search (“SL policy”)
- ❖ Since 2015: even better slow knowledge from deep convolutional neural networks



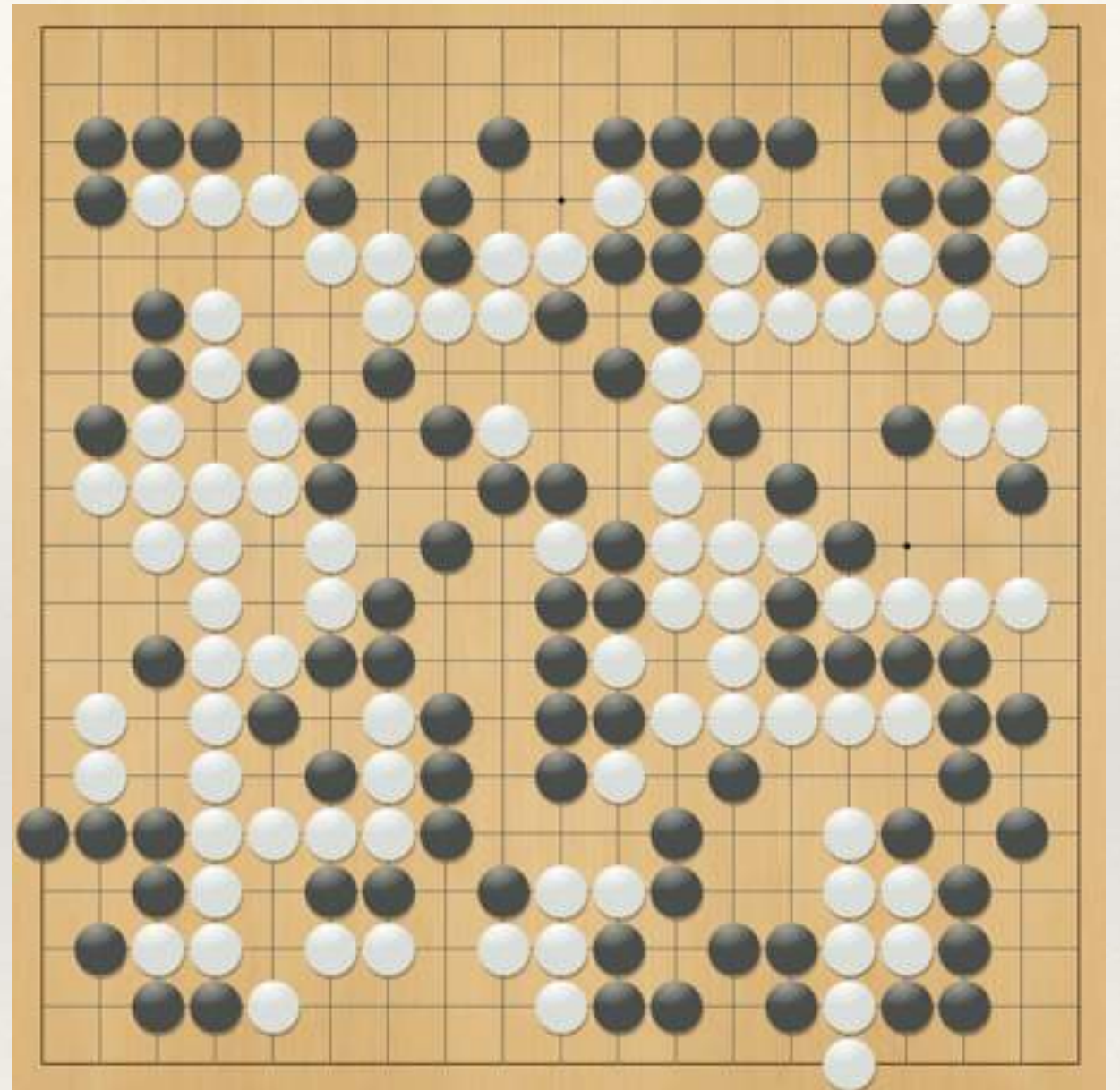
Knowledge based on simple features in Fuego



Storkey + Henrion's Deep convolutional neural network in Fuego

# Computer Go Before AlphaGo

- ❖ Summary of state of the art before AlphaGo:
- ❖ Search - quite strong
- ❖ Simulations - OK, but hard to improve
- ❖ **Knowledge**
  - ❖ Good for move selection
  - ❖ **Considered hopeless for position evaluation**



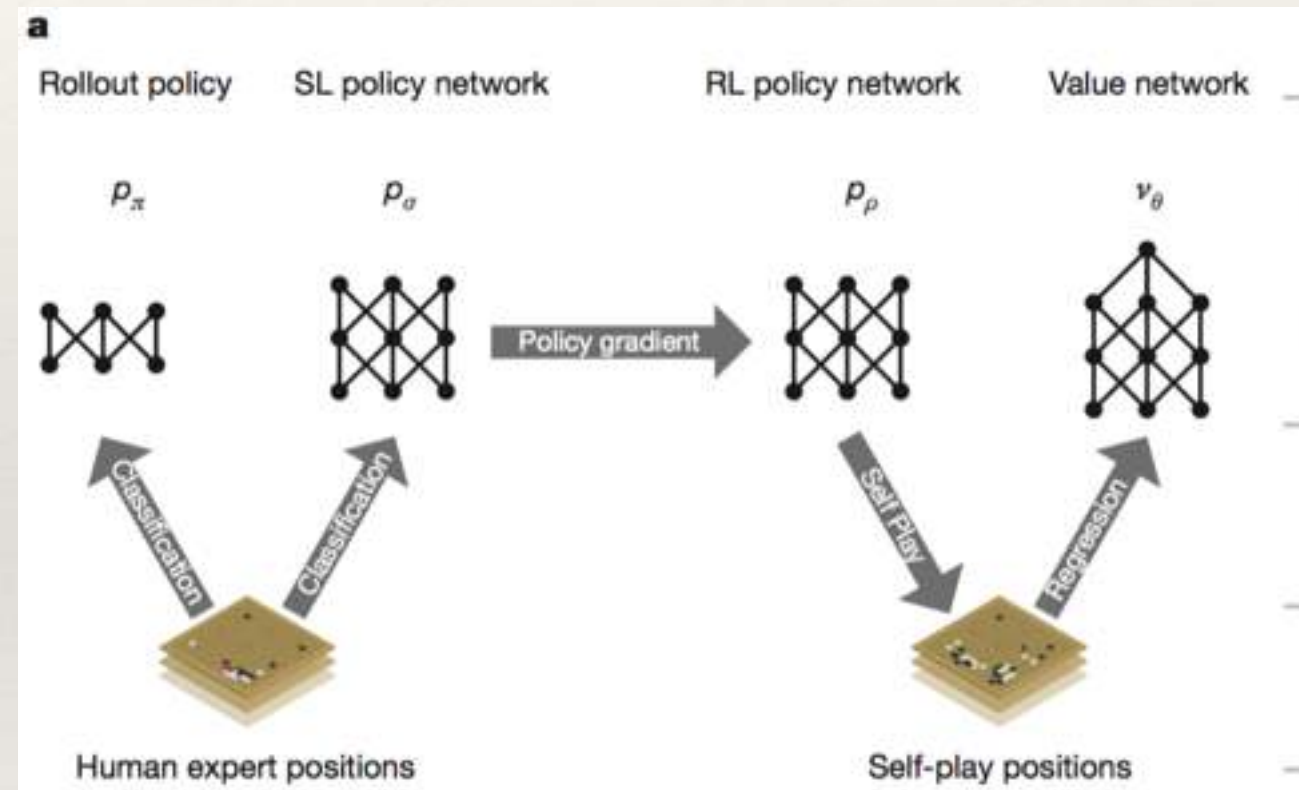
Who is better here?



# The Science - AlphaGo's Contributions

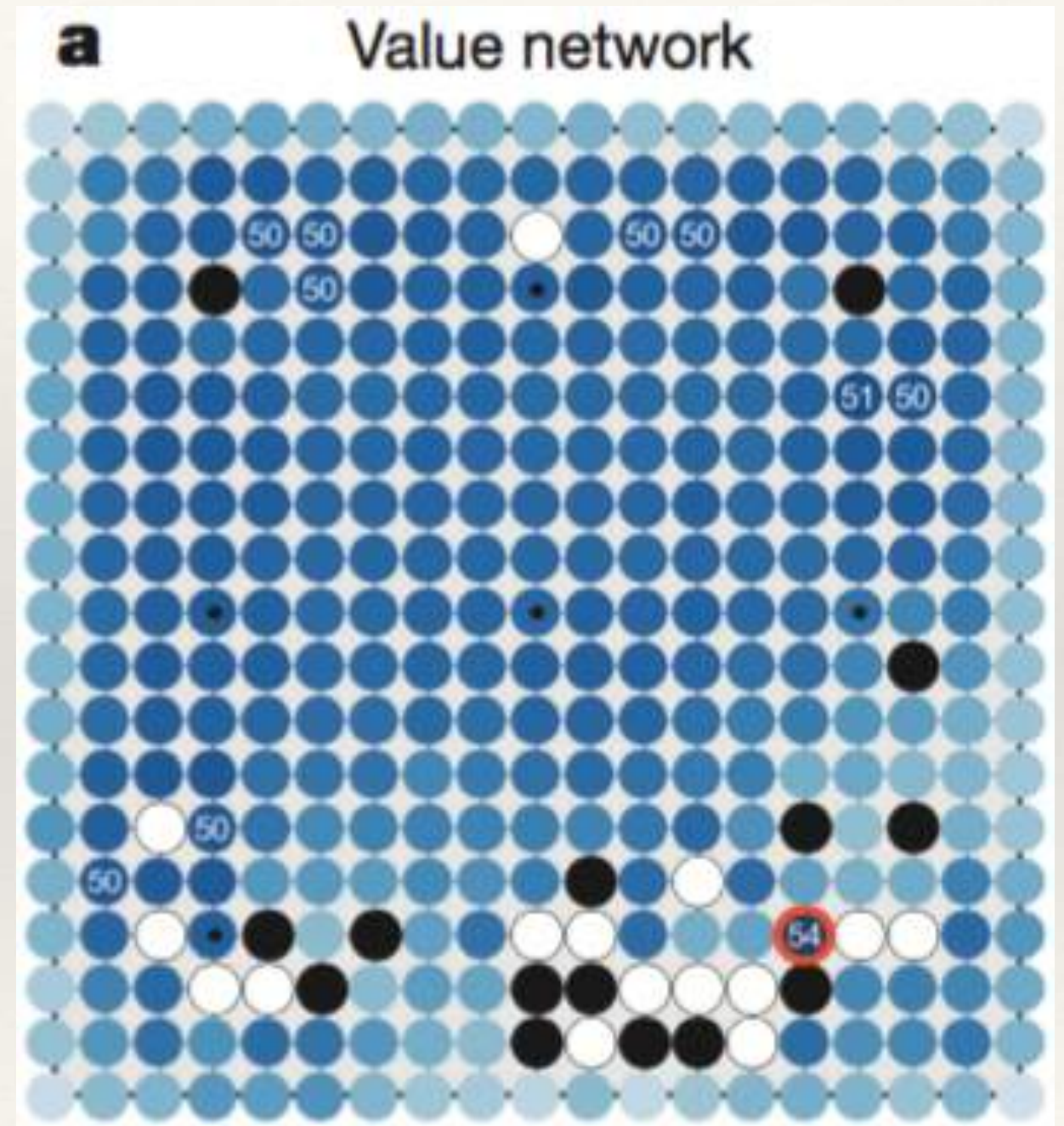
# Alpha Go Design

- ❖ According to paper in Nature
- ❖ Not yet known what changed over the last 5 months, other than much more self-play
- ❖ Search: MCTS (normal)
- ❖ Simulation (rollout) policy: relatively normal
- ❖ Supervised Learning (SL) policy from master games: improved in details, more data
- ❖ **New: Reinforcement Learning (RL) from self-play for value network**
- ❖ **New: Reinforcement Learning (RL) from self-play for policy network**



# Value Network

- ❖ Given a Go position
- ❖ Computes probability of winning
- ❖ No search, no simulation!
- ❖ Static evaluation function
- ❖ Trained by RL from self-play
- ❖ Trains a deep neural network
- ❖ Similarly, the policy network is trained to propose stronger moves





# Putting it All Together

- ❖ A huge engineering effort
- ❖ I only showed the tip of the iceberg here
- ❖ Many other technical contributions
- ❖ Massive amounts of self-play training for the neural networks
- ❖ Massive amounts of testing / tuning
- ❖ Large hardware:  
1202 CPU, 176 GPU used in previous match, “similar hardware” vs Lee Sedol



*University of Alberta and AlphaGo*



# DeepMind and Us

- ❖ AlphaGo is “big Science”
- ❖ Dozens of developers, millions of dollars in hardware and Computing costs
- ❖ What is the role of our university in all of this?
- ❖ We contributed lots of:
  1. Basic research
  2. Training





# UAlberta Research and Training

- Citation list from AlphaGo paper in Nature
- Papers with UofA faculty or UofA trainees in yellow

1. Allis, L. V. *Searching for Solutions in Games and Artificial Intelligence*. PhD thesis, Univ. Limburg, Maastricht, The Netherlands (1994).
2. van den Herik, H., Uiterwijk, J. W. & van Rijswijk, J. Games solved: now and in the future. *Artif. Intell.* **134**, 277–311 (2002).
3. Schaeffer, J. The games computers (and people) play. *Advances in Computers* **52**, 189–266 (2000).
4. Campbell, M., Hoane, A. & Hsu, F. Deep Blue. *Artif. Intell.* **134**, 57–83 (2002).
5. Schaeffer, J. et al. A world championship caliber checkers program. *Artif. Intell.* **53**, 273–289 (1992).
6. Buro, M. From simple features to sophisticated evaluation functions. In *1st International Conference on Computers and Games*, 126–145 (1999).
7. Müller, M. Computer Go. *Artif. Intell.* **134**, 145–179 (2002).
8. Tesauro, G. & Galperin, G. On-line policy improvement using Monte-Carlo search. In *Advances in Neural Information Processing*, 1068–1074 (1996).
9. Sheppard, B. World-championship-caliber Scrabble. *Artif. Intell.* **134**, 241–275 (2002).
10. Bouzy, B. & Helmstetter, B. Monte-Carlo Go developments. In *10th International Conference on Advances in Computer Games*, 159–174 (2003).
11. Coulom, R. Efficient selectivity and backup operators in Monte-Carlo tree search. In *5th International Conference on Computers and Games*, 72–83 (2006).
12. Kocsis, L. & Szepesvári, C. Bandit based Monte-Carlo planning. In *15th European Conference on Machine Learning*, 282–293 (2006).
13. Coulom, R. Computing Elo ratings of move patterns in the game of Go. *ICGA J.* **30**, 198–208 (2007).
14. Baudiš, P. & Gailly, J.-L. Pachi: State of the art open source Go program. In *Advances in Computer Games*, 24–38 (Springer, 2012).
15. Müller, M., Enzenberger, M., Arneson, B. & Segal, R. Fuego – an open-source framework for board games and Go engine based on Monte-Carlo tree search. *IEEE Trans. Comput. Intell. AI in Games* **2**, 259–270 (2010).
16. Gelly, S. & Silver, D. Combining online and offline learning in UCT. In *17th International Conference on Machine Learning*, 273–280 (2007).
17. Krizhevsky, A., Sutskever, I. & Hinton, G. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 1097–1105 (2012).
18. Lawrence, S., Giles, C. L., Tsoi, A. C. & Back, A. D. Face recognition: a convolutional neural-network approach. *IEEE Trans. Neural Netw.* **8**, 98–113 (1997).
19. Mnih, V. et al. Human-level control through deep reinforcement learning. *Nature* **518**, 529–533 (2015).
20. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015).
21. Stern, D., Herbrich, R. & Graepel, T. Bayesian pattern ranking for move prediction in the game of Go. In *International Conference of Machine Learning*, 873–880 (2006).
22. Sutskever, I. & Nair, V. Mimicking Go experts with convolutional neural networks. In *International Conference on Artificial Neural Networks*, 101–110 (2008).
23. Maddison, C. J., Huang, A., Sutskever, I. & Silver, D. Move evaluation in Go using deep convolutional neural networks. *3rd International Conference on Learning Representations* (2015).
24. Clark, C. & Storkey, A. J. Training deep convolutional neural networks to play go. In *32nd International Conference on Machine Learning*, 1766–1774 (2015).
25. Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.* **8**, 229–256 (1992).
26. Sutton, R., McAllester, D., Singh, S. & Mansour, Y. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems*, 1057–1063 (2000).
27. Sutton, R. & Barto, A. *Reinforcement Learning: an Introduction* (MIT Press, 1998).
28. Schraudolph, N. N., Dayan, P. & Sejnowski, T. J. Temporal difference learning of position evaluation in the game of Go. *Adv. Neural Inf. Process. Syst.* **6**, 817–824 (1994).
29. Enzenberger, M. Evaluation in Go by a neural network using soft segmentation. In *10th Advances in Computer Games Conference*, 97–108 (2003), 267.
30. Silver, D., Sutton, R. & Müller, M. Temporal-difference search in computer Go. *Mach. Learn.* **87**, 183–219 (2012).
31. Levinovitz, A. The mystery of Go, the ancient game that computers still can't win. *Wired Magazine* (2014).
32. Mechner, D. All Systems Go. *The Sciences* **38**, 32–37 (1998).
33. Mandziuk, J. Computational intelligence in mind games. In *Challenges for Computational Intelligence*, 407–442 (2007).
34. Berliner, H. A chronology of computer chess and its literature. *Artif. Intell.* **10**, 201–214 (1978).
35. Browne, C. et al. A survey of Monte-Carlo tree search methods. *IEEE Trans. Comput. Intell. AI in Games* **4**, 1–43 (2012).
36. Gelly, S. et al. The grand challenge of computer Go: Monte Carlo tree search and extensions. *Commun. ACM* **55**, 106–113 (2012).
37. Coulom, R. Whole-history rating: A Bayesian rating system for players of time-varying strength. In *International Conference on Computers and Games*, 113–124 (2008).
38. KGS. Rating system math. <http://www.gokgs.com/help/rmath.html>.
39. Littman, M. L. Markov games as a framework for multi-agent reinforcement learning. In *11th International Conference on Machine Learning*, 157–163 (1994).
40. Knuth, D. E. & Moore, R. W. An analysis of alpha-beta pruning. *Artif. Intell.* **9**, 293–326 (1975).
41. Sutton, R. Learning to predict by the method of temporal differences. *Mach. Learn.* **3**, 9–44 (1988).
42. Baxter, J., Tridgell, A. & Weaver, L. Learning to play chess using temporal differences. *Mach. Learn.* **40**, 243–263 (2000).
43. Veness, J., Silver, D., Blair, A. & Uther, W. Bootstrapping from game tree search. In *Advances in Neural Information Processing Systems* (2009).
44. Samuel, A. L. Some studies in machine learning using the game of checkers II - recent progress. *IBM J. Res. Develop.* **11**, 601–617 (1967).
45. Schaeffer, J., Hlynka, M. & Jussila, V. Temporal difference learning applied to high-performance game-playing program. In *17th International Joint Conference on Artificial Intelligence*, 529–534 (2001).
46. Tesauro, G. TD-gammon, a self-teaching backgammon program, achieves master-level play. *Neural Comput.* **6**, 215–219 (1994).
47. Dahl, F. Honte, a Go-playing program using neural nets. In *Machines that learn to play games*, 205–223 (Nova Science, 1999).
48. Rosin, C. D. Multi-armed bandits with episode context. *Ann. Math. Artif. Intell.* **61**, 203–230 (2011).
49. Lanctot, M., Winands, M. H. M., Pepels, T. & Sturtevant, N. R. Monte Carlo tree search with heuristic evaluations using implicit minimax backups. In *IEEE Conference on Computational Intelligence and Games*, 1–8 (2014).
50. Gelly, S., Wang, Y., Munos, R. & Teytaud, O. Modification of UCT with patterns. Monte-Carlo Go. Tech. Rep. 6062, INRIA (2006).
51. Silver, D. & Tesauro, G. Monte-Carlo simulation balancing. In *26th International Conference on Machine Learning*, 119 (2009).
52. Huang, S.-C., Coulom, R. & Lin, S.-S. Monte-Carlo simulation balancing in practice. In *7th International Conference on Computers and Games*, 81–92 (Springer-Verlag, 2011).
53. Baier, H. & Drake, P. D. The power of forgetting: improving the last-good-rep policy in Monte Carlo Go. *IEEE Trans. Comput. Intell. AI in Games* **2**, 303–309 (2010).
54. Huang, S. & Müller, M. Investigating the limits of Monte-Carlo tree search methods in computer Go. In *8th International Conference on Computers and Games*, 39–48 (2013).
55. Segal, R. B. On the scalability of parallel UCT. *Computers and Games* **6515**, 36–47 (2011).
56. Enzenberger, M. & Müller, M. A lock-free multithreaded Monte-Carlo tree search algorithm. In *12th Advances in Computer Games Conference*, 14–20 (2009).
57. Huang, S.-C., Coulom, R. & Lin, S.-S. Time management for Monte-Carlo tree search applied to the game of Go. In *International Conference on Technologies and Applications of Artificial Intelligence*, 462–466 (2010).
58. Gelly, S. & Silver, D. Monte-Carlo tree search and rapid action value estimation in computer Go. *Artif. Intell.* **175**, 1856–1875 (2011).



# The Future

---

# Where do we Go from Here?

---

- ❖ Which other problems can be tackled with this approach?
- ❖ The methods are quite general, not Go-specific
- ❖ We need an internal **model** of the problem in order to learn from self play
- ❖ We may be able to use similar approaches when we have lots of data
- ❖ Can we build a model from data?
- ❖ MCTS started in Go, has found a large number of applications
- ❖ Deep learning techniques have revolutionized many fields such as image recognition, speech recognition, natural language processing, drug discovery...
- ❖ Go was the first combination of MCTS and deep learning
- ❖ Limitless possibilities...



# What Should UAlberta Do?

- ❖ Keep doing world-leading basic research and training
- ❖ Find ways to attract and retain the best students in the field
- ❖ Update our computational infrastructure to *not completely lose touch* with industry
- ❖ Develop applications beyond games? Big science?



# Summary and Outlook

- ❖ DeepMind's AlphaGo program is an incredible research breakthrough
- ❖ Landmark achievement for Computing Science
- ❖ University of Alberta has played very significant roles on the way there
- ❖ We must try to stay relevant in the future!



Watch game 5 with us:  
10 pm, CSC 3-33