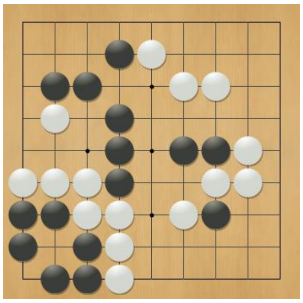
Garry Kasparov vs. Deep Blue (1997)

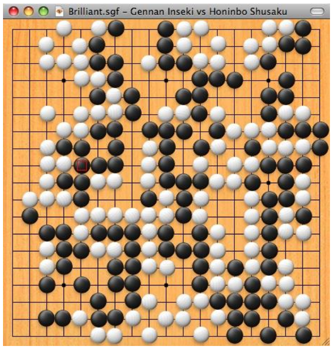
Deep Mind’s AlphaGo vs. Lee Sedol (2016)

Watson vs. Ken Jennings (2011)

Computer Go



9x9 (smallest board)



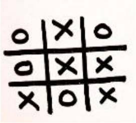
19x19 (standard board)

- “Task Par Excellence for AI” (Hans Berliner)
- “New Drosophila of AI” (John McCarthy)
- “Grand Challenge Task” (David Mechner)

A Brief History of Computer Go

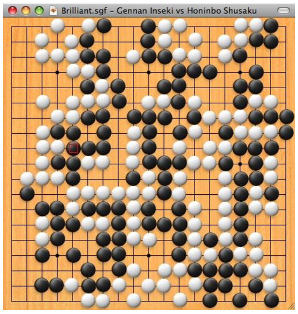
- *1997:* Super human Chess w/ Alpha-Beta + Fast Computer
- *2005*: Computer Go is impossible!

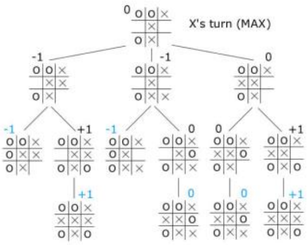
Why?



VS

VS





MiniMax Tree

- Branching Factor
  - Chess ≈ 35
  - Go ≈ 250
- Required search depth
  - Chess ≈ 14
  - Go ≈ much larger
- Leaf Evaluation Function
  - Chess – good hand-coded function
  - Go – no good hand-coded function

A Brief History of Computer Go

- *1997:* Super human Chess w/ Alpha-Beta + Fast Computer
- *2005*: Computer Go is impossible!
- *2006*: Monte-Carlo Tree Search applied to 9x9 Go (bit of learning)
- *2007*: Human master level achieved at 9x9 Go (bit more learning)
- *2008*: Human grandmaster level achieved at 9x9 Go (even more)

Computer GO Server rating over this period:
1800 ELO → 2600 ELO

- *2012*: Zen program beats former international champion Takemiya Masaki with only 4 stone handicap in 19x19
- ***2015: DeepMind's AlphaGo Defeats European Champion 5-0 (lots of learning)***
- ***2016: AlphaGo Defeats Go Legend Lee Sedol 4-1 (lots more learning)***

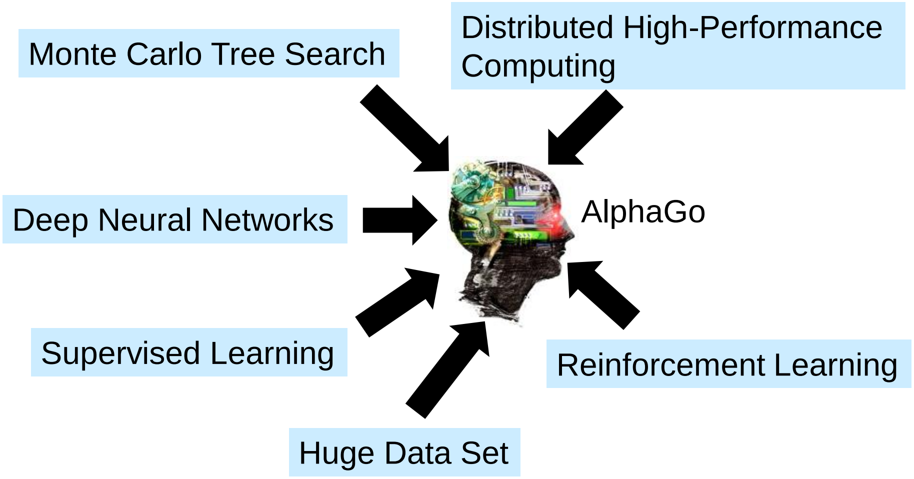
AlphaGo

- Deep Learning + Monte Carlo Tree Search + HPC
- Learn from 30 million expert moves and self play
- Highly parallel search implementation
- 48 CPUs, 8 GPUs (scaling to 1,202 CPUs, 176 GPUs)

Image: AP/ Lee Jin-man

March 2016 :
AlphaGo beats Lee Sedol 4-1

Arsenal of AlphaGo



Monte Carlo Tree Search

Distributed High-Performance Computing

Deep Neural Networks

AlphaGo

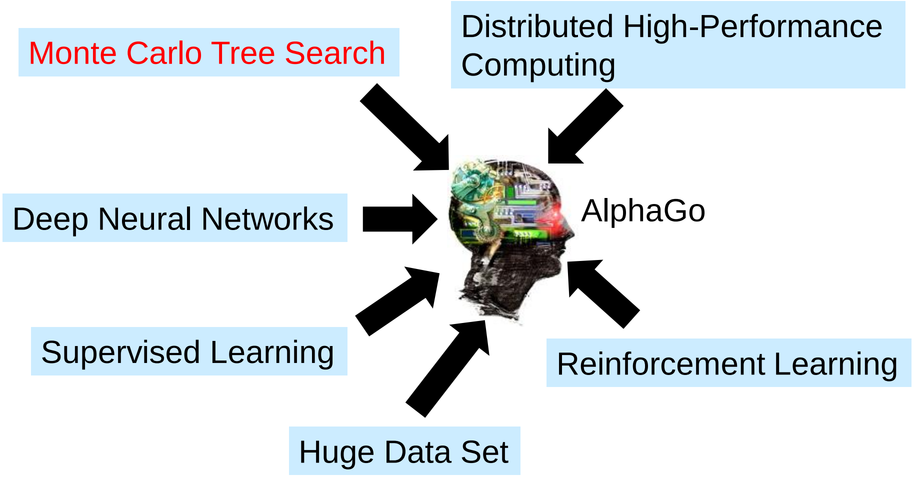
Supervised Learning

Reinforcement Learning

Huge Data Set

Mastering the game of Go with deep neural networks and tree search
Nature, 529, January 2016.

Arsenal of AlphaGo



Monte Carlo Tree Search

Distributed High-Performance Computing

Deep Neural Networks

AlphaGo

Supervised Learning

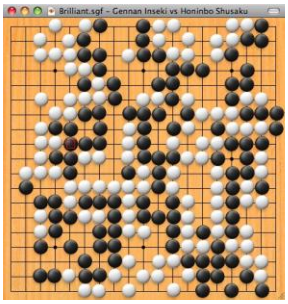
Reinforcement Learning

Huge Data Set

Mastering the game of Go with deep neural networks and tree search
Nature, 529, January 2016.

Monte Carlo Tree Search

Idea #1: board evaluation function via random rollouts

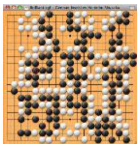


Evaluation Function:

- play many random games
- evaluation is fraction of games won by current player
- surprisingly effective

Even better if use rollouts that select better than random moves

Monte Carlo Tree Search



Idea #2: selective tree expansion

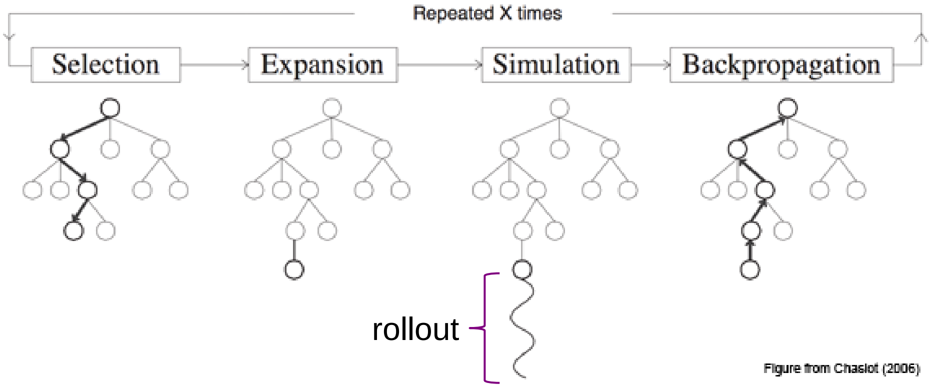
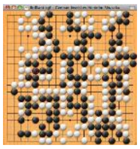
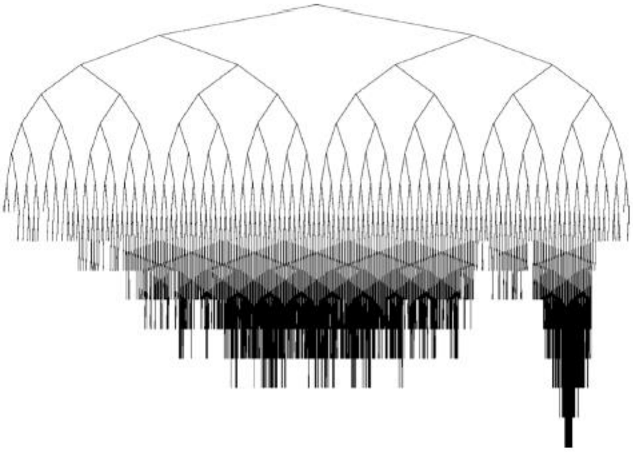


Figure from Chaslot (2006)

Monte Carlo Tree Search

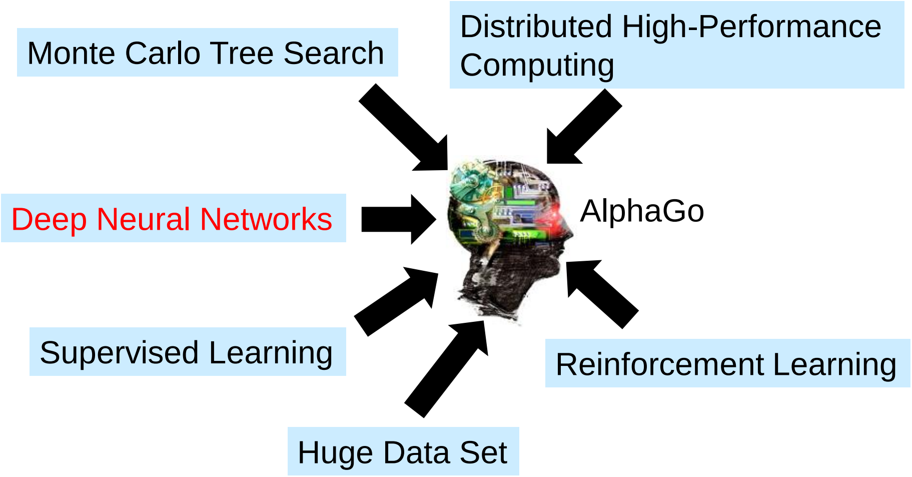


Idea #2: selective tree expansion



Non-uniform tree growth

Arsenal of AlphaGo



Monte Carlo Tree Search

Distributed High-Performance Computing

Deep Neural Networks

AlphaGo

Supervised Learning

Reinforcement Learning

Huge Data Set

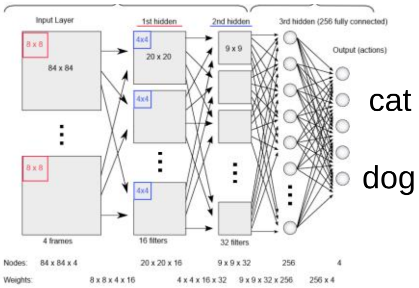
Mastering the game of Go with deep neural networks and tree search
Nature, 529, January 2016.

Deep Neural Networks

How can you write a program to distinguish cats from dogs in images?

Machine Learning: show computer example cats and dogs and let it decide how to distinguish them

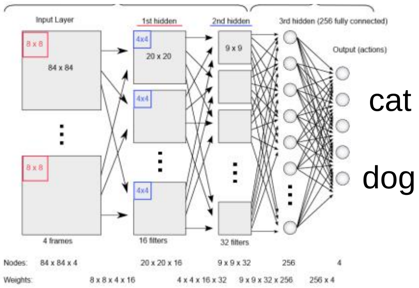
Deep Neural Network


Input Layer
1st hidden
2nd hidden
3rd hidden (256 fully connected)
Output (actions)
8 x 8
84 x 84
4x4
20 x 20
9 x 9
4 frames
16 filters
32 filters
cat
dog
Nodes: 84 x 84 x 4 20 x 20 x 16 9 x 9 x 32 256 4
Weights: 8 x 8 x 4 x 16 4 x 4 x 16 x 32 9 x 9 x 32 x 256 256 x 4

Deep Neural Networks

State-of-the-Art Performance: very fast GPU implementations allow training giant networks (millions of parameters) on massive data sets

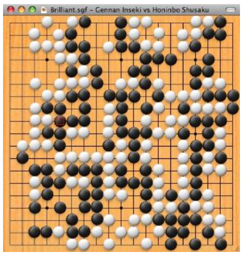
Deep Neural Network


Input Layer
1st hidden
2nd hidden
3rd hidden (256 fully connected)
Output (actions)
8 x 8
84 x 84
4x4
20 x 20
9 x 9
4 frames
16 filters
32 filters
cat
dog
Nodes: 84 x 84 x 4 20 x 20 x 16 9 x 9 x 32 256 4
Weights: 8 x 8 x 4 x 16 4 x 4 x 16 x 32 9 x 9 x 32 x 256 256 x 4

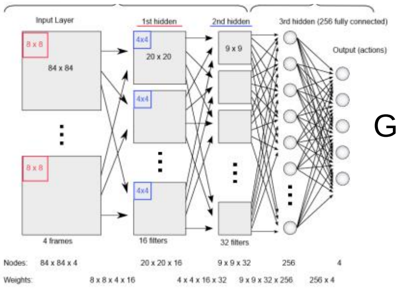
Deep Neural Networks

State-of-the-Art Performance: very fast GPU implementations allow training giant networks (millions of parameters) on massive data sets

Could a Deep NN learn to predict expert Go moves by looking at board position? Yes!

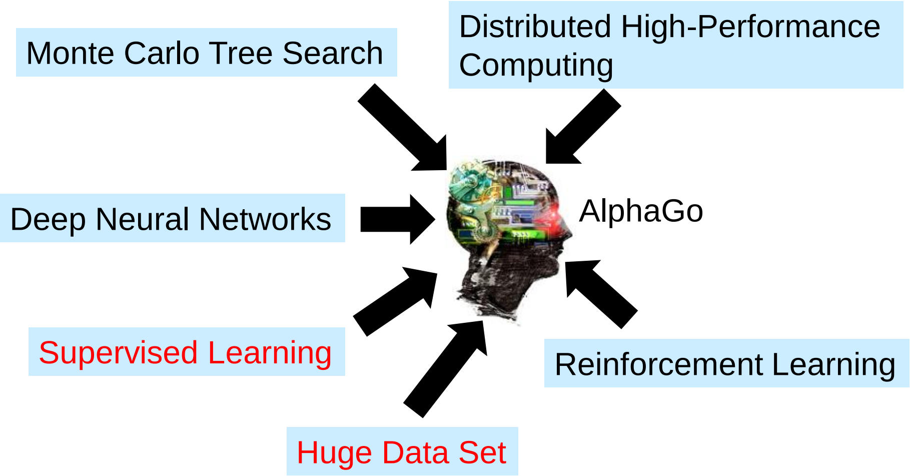


Deep Neural Network



Go Move

Arsenal of AlphaGo



Monte Carlo Tree Search

Distributed High-Performance Computing

Deep Neural Networks

AlphaGo

Supervised Learning

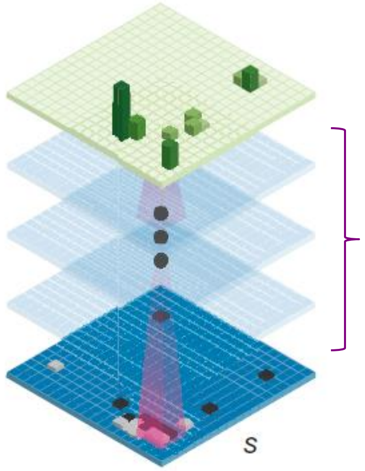
Reinforcement Learning

Huge Data Set

Mastering the game of Go with deep neural networks and tree search
Nature, 529, January 2016.

Supervised Learning for Go

Output: probability of each move



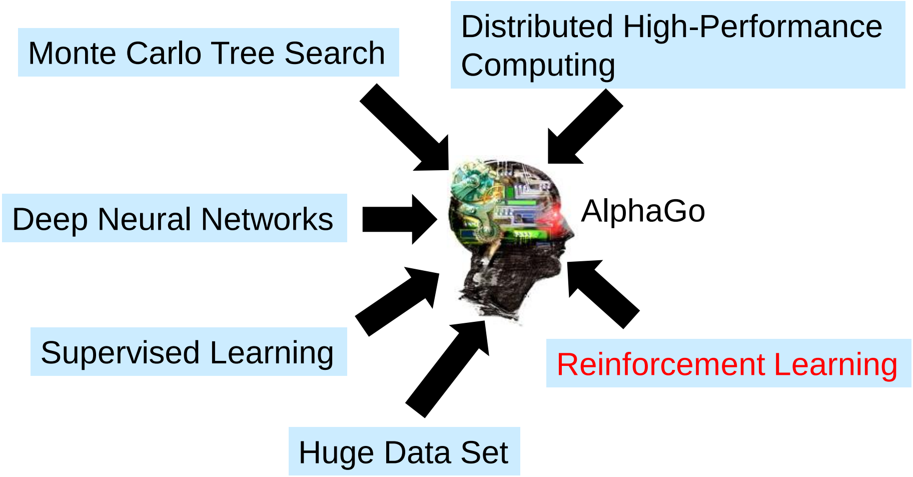
Deep NN Internal Layers

Trained for 3 weeks on 30 million expert moves

- 57% prediction accuracy!

Input: Board Position

Arsenal of AlphaGo



Monte Carlo Tree Search

Distributed High-Performance Computing

Deep Neural Networks

AlphaGo

Supervised Learning

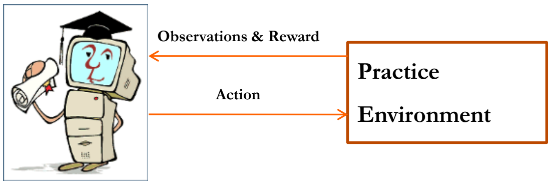
Reinforcement Learning

Huge Data Set

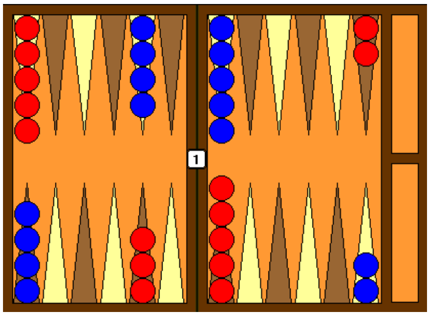
Mastering the game of Go with deep neural networks and tree search
Nature, 529, January 2016.

Reinforcement Learning

Reinforcement Learning: learn to act well in an environment via trial-and-error that results in positive and negative rewards



TD-Gammon (1992)

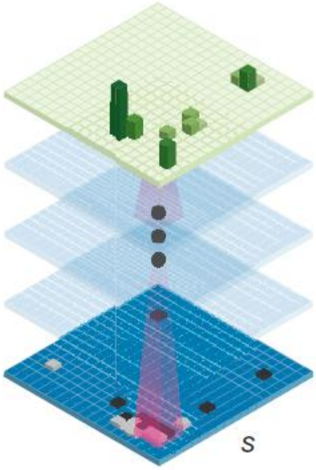


Backgammon

- Neural network with 80 hidden units (1 layer)
- Used Reinforcement Learning for 1.5 Million games of self-play
- One of the top (2 or 3) players in the world!

Reinforcement Learning for Go

Output: probability of each move

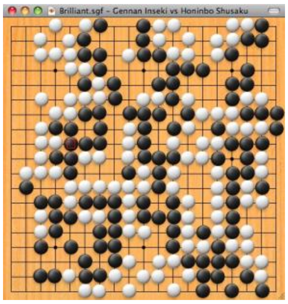


Input: Board Position

- Start with Deep NN from supervised learning.
- Continue to train network via self play.
- AlphaGo did this for months.
- 80% win rate against the original supervised Deep NN
- 85% win rate against best prior tree search method!
- Still not close to professional level

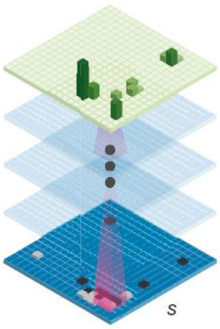
Monte Carlo Tree Search

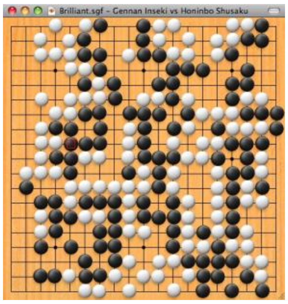
Idea: use deep NN for rollout evaluation

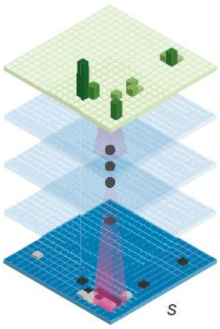


Monte Carlo Tree Search

Idea: use deep NN for rollout evaluation







Monte Carlo Tree Search

Idea: use Deep NN for rollouts in Monte Carlo Tree Search

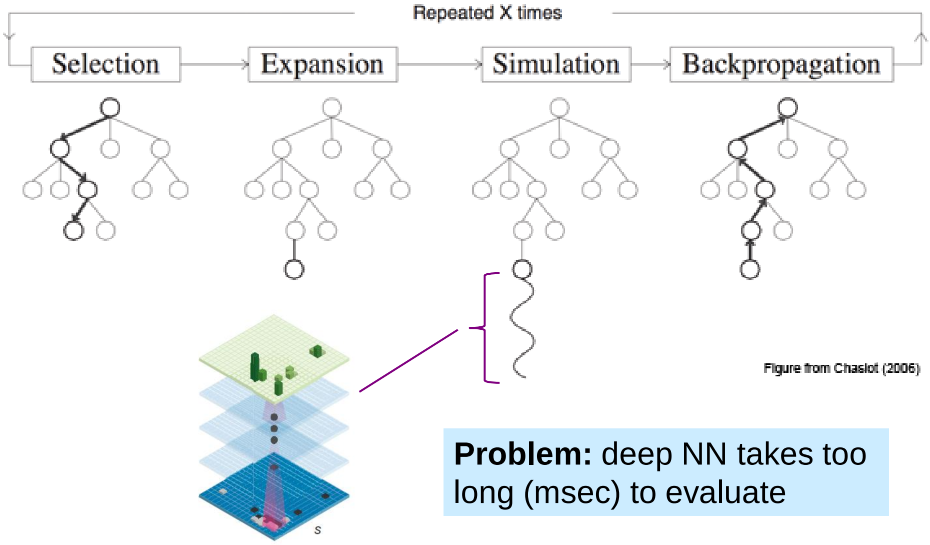


Figure from Chaslot (2006)

Problem: deep NN takes too long (msec) to evaluate

Monte Carlo Tree Search

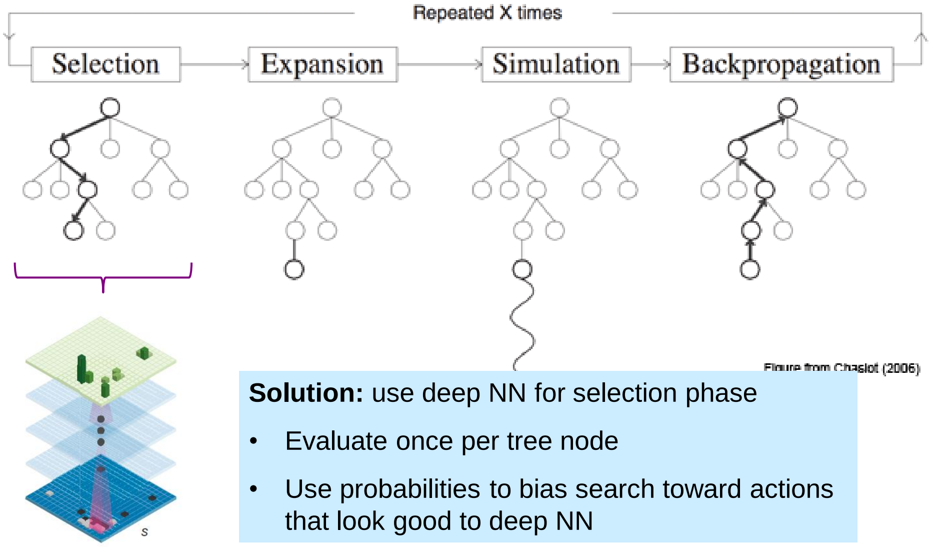


Figure from Chaslot (2006)

Solution: use deep NN for selection phase

- Evaluate once per tree node
- Use probabilities to bias search toward actions that look good to deep NN

Monte Carlo Tree Search

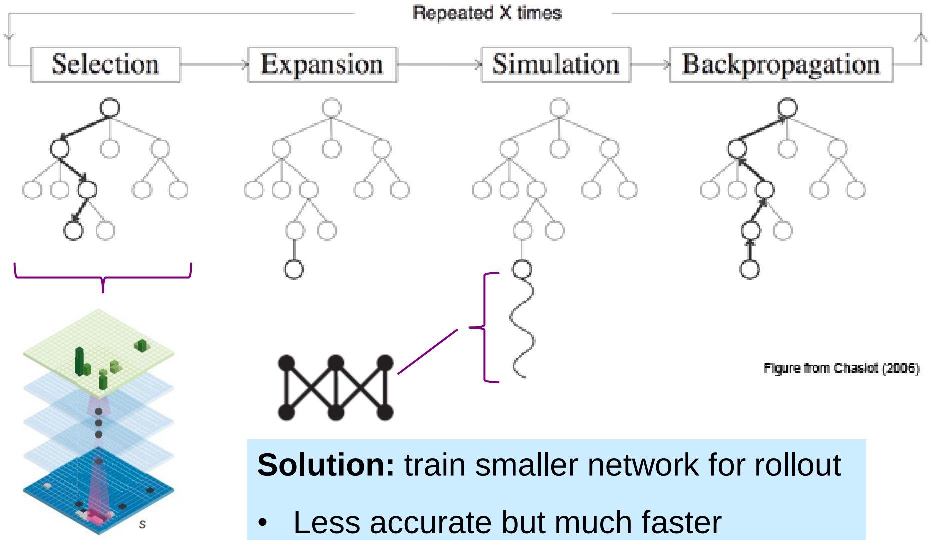


Figure from Chaslot (2006)

Solution: train smaller network for rollout

- Less accurate but much faster

AlphaGo

- Deep Learning + Monte Carlo Tree Search + HPC
- Learn from 30 million expert moves and self play
- Highly parallel search implementation
- 48 CPUs, 8 GPUs (scaling to 1,202 CPUs, 176 GPUs)

Image: AP/ Lee Jin-man

2015 :
AlphaGo beats European Champ (5-0)

lots of self play

March 2016 :
AlphaGo beats Lee Sedol (4-1)

Computers are good at Go now – So What?

- The idea of combining search with learning is very general and widely applicable
- Deep Networks are leading to advances in many areas of AI now
  - Computer Vision
  - Speech Processing
  - Natural Language Processing
  - Bioinformatics
  - Robotics
- It is a very exciting time to be working in AI