

Machine Learning in Games

The Magic of Research in Microsoft Products

Joaquin Quiñonero Candela, Ralf Herbrich and Thore Graepel
Online Services and Advertising
Microsoft Research Cambridge

Overview

- Why Machine Learning and Games?
- Machine Learning in Video Games
 - Drivatars™
 - Reinforcement Learning
- Machine Learning in Online Games
 - TrueSkill™
 - Halo 3
- The Path of Go
- Conclusions

Why

Test

- Perfect
- measure
- Perfect
- Reduce
- Reduce
- Great



e Al
hes

ent

Games can be very hard!

- Partially observable stochastic games
 - States only partially observed
 - Multiple agents choose actions
 - Stochastic pay-offs and state transitions depend on state and all the other agents' actions
 - Goal: Optimise long term pay-off (reward)
- Just like life: complex, adversarial, uncertain, and we are in it for the long run!

Approximations

From single player's perspective

- Partially Observable Markov Decision Process (POMDP)

Approximate Solutions

- Reinforcement Learning
- Unsupervised Learning
- Supervised Learning

What is the best AI?

- Always takes optimal actions
- Delivers best entertainment value

Overview

- Why Machine Learning and Games?
- Machine Learning in Video Games
 - Drivatars™
 - Reinforcement Learning
- Machine Learning in Online Games
 - TrueSkill™
 - Halo 3
- The Path of Go
- Conclusions

Drivatar™



Adaptive avatar for driving



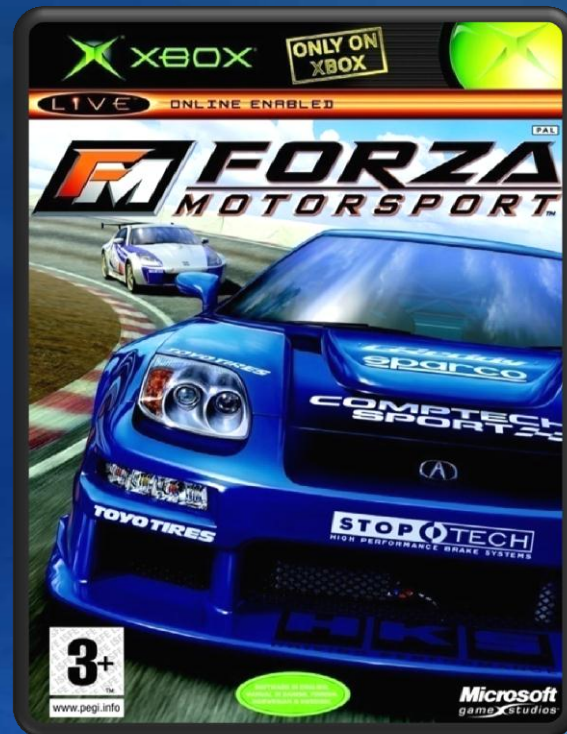
Separate game mode



Basis for all in-game AI



Basis of "dynamic" racing line



Demo: Forza Motorsport

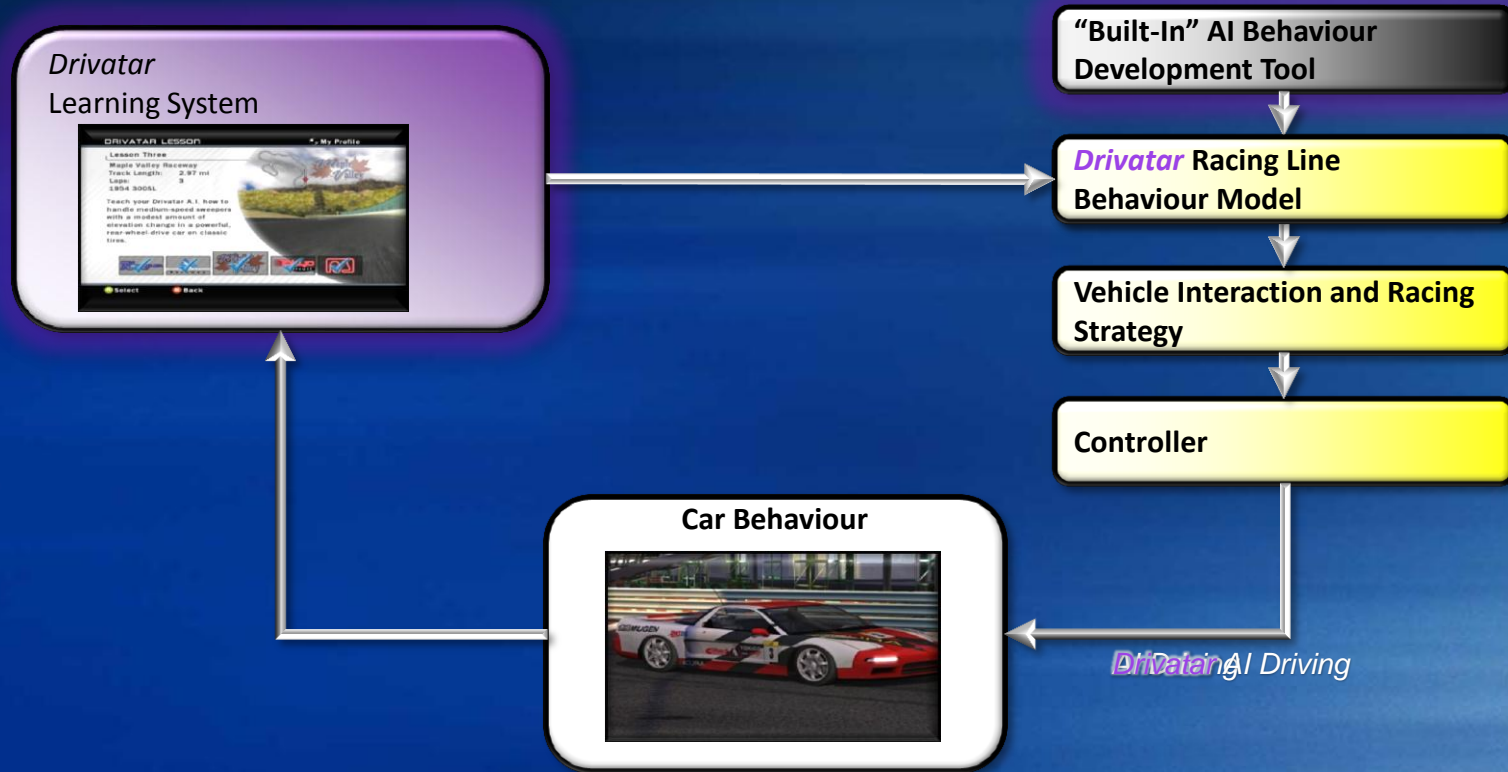


XBOX Game

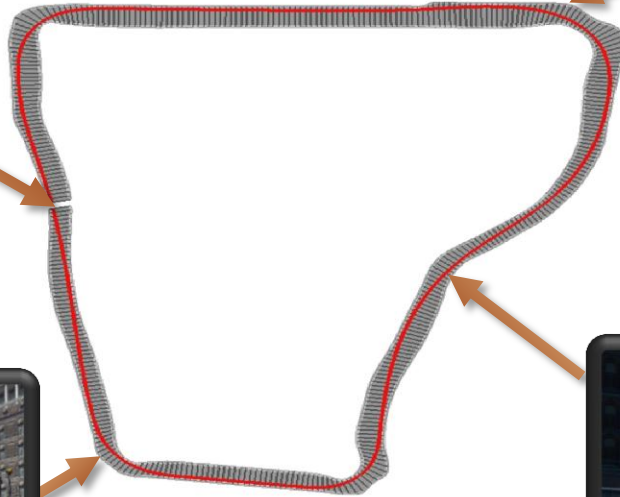
- Dynamic Racing Line
- Learning a Drivatar
- Using a Drivatar



Drivatars Unplugged




















The Racing Line Model



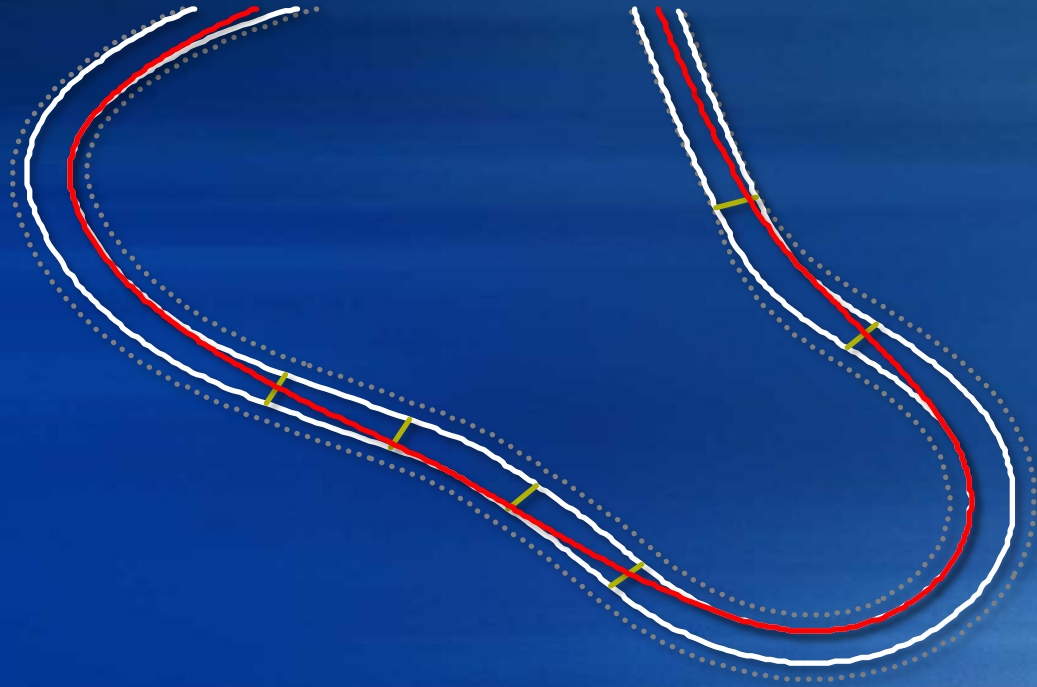
Drivatars: Main Idea

- Two phase process:
 1. Pre-generate possible racing lines prior to the race from a (compressed) racing table.
 2. Switch the lines during the race to add variability.
- Compression reduces the memory needs per racing line segment
- Switching makes smoother racing lines.

Racing Tables

Segments	a_1	a_2	a_3	a_4
				
				
				
				

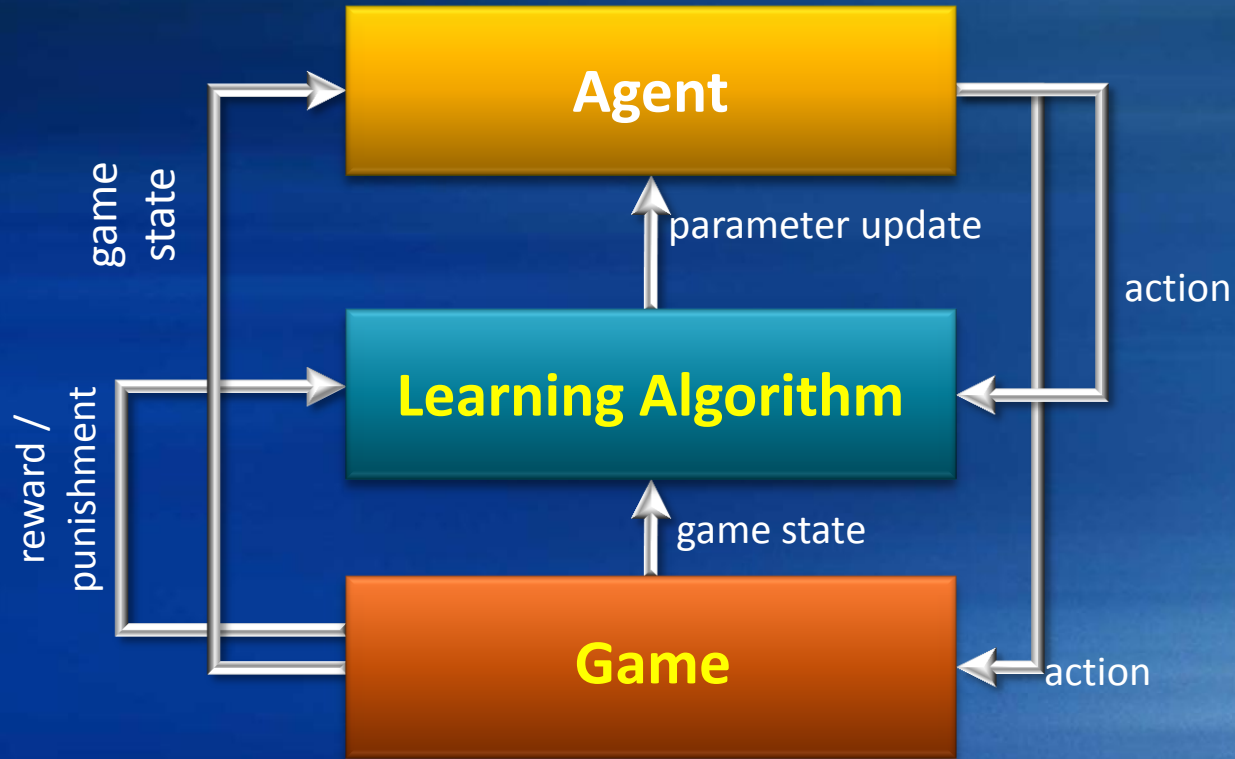
Minimal Curvature Lines



Overview

- Why Machine Learning and Games?
- Machine Learning in Video Games
 - Drivatars™
 - Reinforcement Learning
- Machine Learning in Online Games
 - TrueSkill™
 - Halo 3
- The Path of Go
- Conclusions

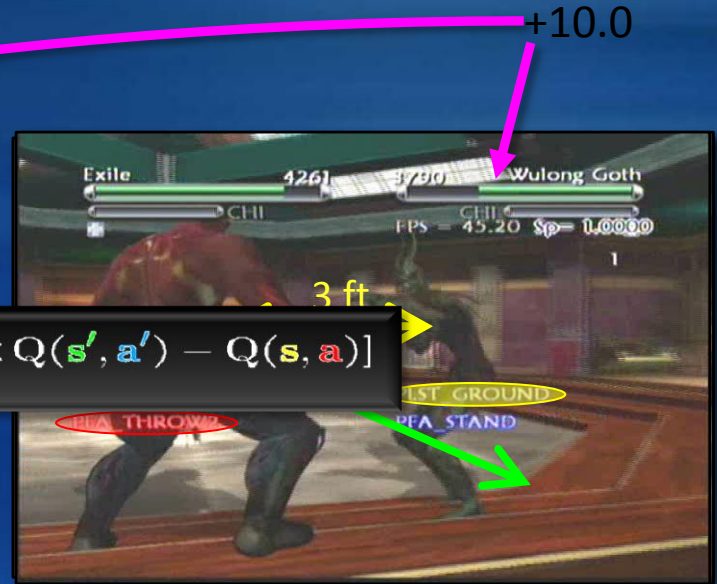
Reinforcement Learning



Tabular Q-Learning

Q-Table		actions		
		THROW	KICK	STAND
game states	1ft / GROUND			
	2ft / GROUND			
	3ft / GROUND	13.2	10.2	-1.3
	4ft / GROUND			
	5ft / GROUND			
	6ft / GROUND			
	1ft / KNOCKED			
	2ft / KNOCKED			
	3ft / KNOCKED			
	4ft / KNOCKED			
5ft / KNOCKED	3.2	6.0	4.0	
6ft / KNOCKED				

$$Q(s, a) \leftarrow Q(s, a) + \alpha [r + \gamma \max_{a'} Q(s', a') - Q(s, a)]$$



Results

• Game state features

- Separation (5 binned ranges)
- Last action (6 categories)
- Mode (ground, air, knocked)
- Proximity to obstacle

• Available Actions

- 19 aggressive (kick, punch)
- 10 defensive (block, lunge)
- 8 neutral (run)

• Q-Function Representation

- One layer neural net (tanh)

Reinforcement Learner



In-Game AI Code



Learning Aggressive Fighting

Reward for decrease in Wulong Goth's health

Early in the learning process ...



... after 15 minutes of learning

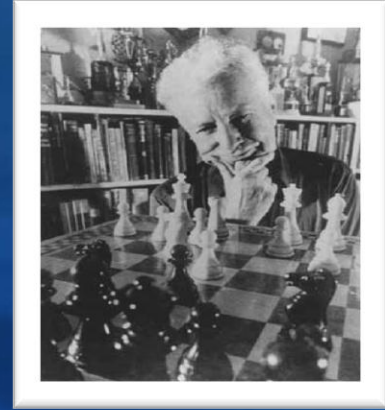


Overview

- Why Machine Learning and Games?
- Machine Learning in Video Games
 - Drivatars™
 - Reinforcement Learning
- Machine Learning in Online Games
 - TrueSkill™
 - Halo 3
- The Path of Go
- Conclusions

Motivation

- Competition is central to our lives
 - Innate biological trait
 - Driving principle of many sports
- Chess Rating for fair competition
 - ELO: Developed in 1960 by Árpád Imre Élő
 - Matchmaking system for tournaments
- Challenges of online gaming
 - Learn from few match outcomes efficiently
 - Support multiple teams and multiple players per team



The Skill Rating Problem

- Given:

- Match outcomes: Orderings among k teams consisting of n_1, n_2, \dots, n_k players, respectively

- Questions:

- Skill s_i for each player

- Global

- Fairness

The image displays several overlapping data visualizations from a game:

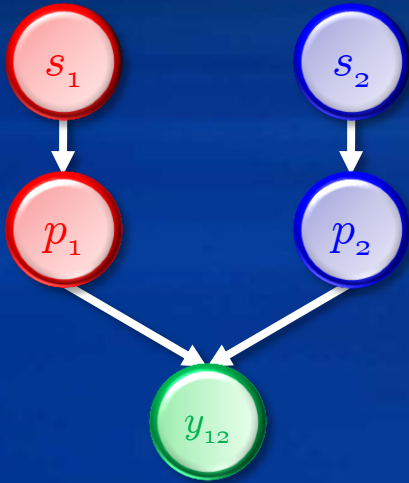
- Team Score:** A table showing 'Red Team' with a score of 50.
- Player Performance Table:** A table with columns: Level, Gamertag, Avg. Life, Best Spree, and Score.

Rank	Level	Gamertag	Avg. Life	Best Spree	Score
1st	N/A	SniperEye	N/A	N/A	25
2nd	N/A	xXxHALOxXx	N/A	N/A	24
3rd	N/A	AjaySandhu	N/A	N/A	15
3rd	N/A	AjaySandhu(G)	N/A	N/A	15
5th	N/A	Robert115	N/A	N/A	11
5th	N/A	TurboNegro84(G)	N/A	N/A	11
7th	N/A	TurboNegro84	N/A	N/A	5
8th	N/A	SniperEye(G)	N/A	N/A	1
- Ranked Player List:** A list of 17 players with their scores.

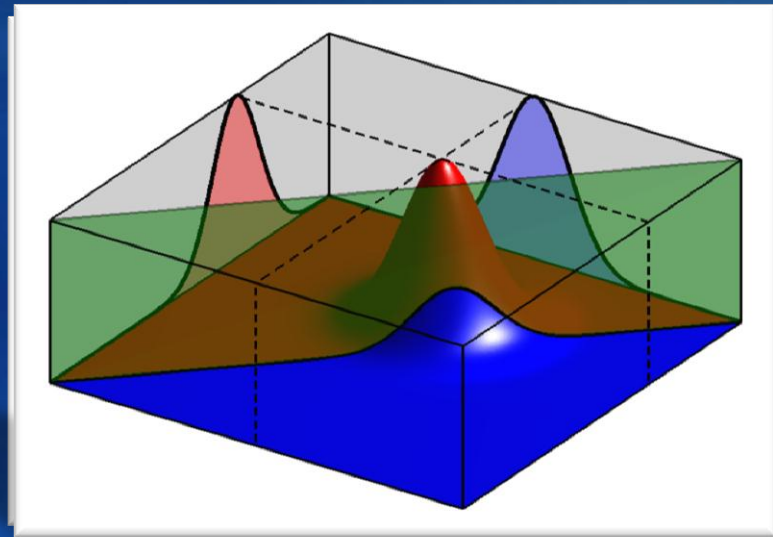
Rank	Score	Player Name
1	27	SEWICSYDE OWNS
2	26	FATAL REVENGE
3	25	Paranoia 1
4	25	Paulk
5	25	lxX OMG Xxl
6	25	BittyTom
7	24	brian 2007
8	24	SEXY MOZES
9	24	droplates
10	24	jaCKdaSaMuRai
11	24	ll Me ll
12	24	iamNightMare
13	24	a retarded007
14	24	Perfected Brit
15	24	THE MUFFIN MANx
16	23	TheVunit
17	23	Mr Sushi87

Two Player Match Outcome Model

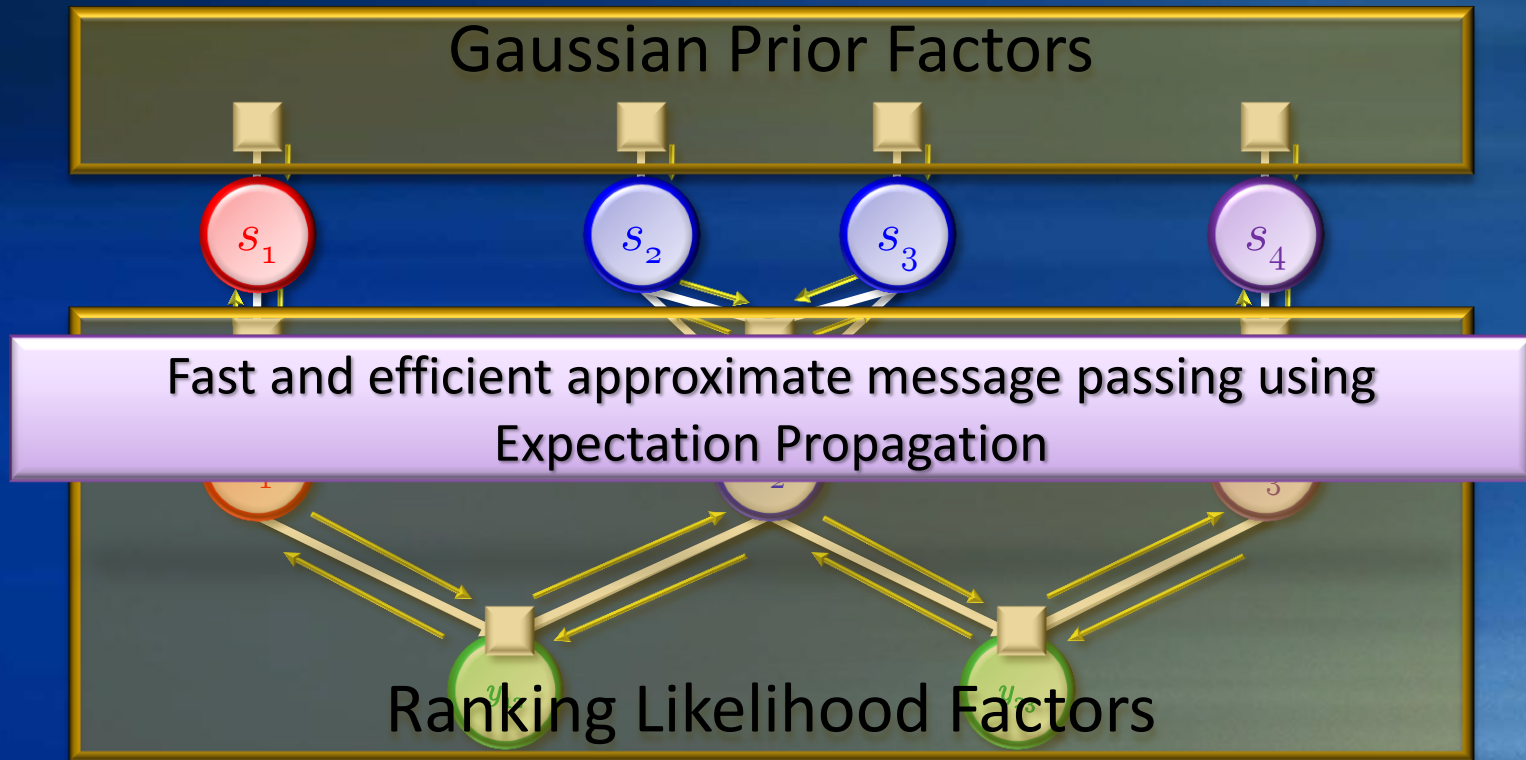
- Latent Gaussian performance model for fixed skills
- Possible outcomes: Player 1 wins over 2 (and vice versa)



$$\mathbf{P}(y_{12} = (1, 2) | p_1, p_2) = \mathbb{I}(p_1 > p_2)$$



Efficient Approximate Inference



Applications to Online Gaming

- **Leaderboard**

- Global ranking of all players

- **Matchmaking**

- For gamers: Most uncertain outcome
- For inference: Most informative
- Both a

$$\mu_i - 3 \cdot \sigma_i$$

	Level	Gamertag	Avg. Life	Best Spree	Score
1st	10	BlueBot	00:00:49	6	15
1st	7	SniperEye	00:00:41	4	14
1st	9	ProThepirate	00:01:07	3	13
1st	10	dazdemon	00:00:59	3	8
2nd	10	WardHarry	00:00:41	2	12
2nd	3	Asuka	00:00:47	2	10
2nd	9	Antidote4Losing	00:00:41	2	9
2nd	2	BlackKnight	00:00:48	3	10

$P(p_i \approx p_j | \mu_i = \mu_j, \sigma_i^2 + \sigma_j^2)$
 $P(p_i \approx p_j | \mu_i - \mu_j = 0, \sigma_i^2 + \sigma_j^2 = 0)$

1	27	SEWICSYDE OWNS
2	26	FATAL REVENGE
3	25	Paranoia 1
4	25	Paulk
5	25	IxX OMG Xxl
6	25	BittyTom
7	24	brian 2007
8	24	SEXY MOZES
9	24	droplates
10	24	jaCKdaSaMuRai
11	24	ll Me ll
12	24	iamNightMare
13	24	a retarded007
14	24	Perfected Brit
15	24	THE MUFFIN MANx
16	23	TheVunit
17	23	Mr Sushi87

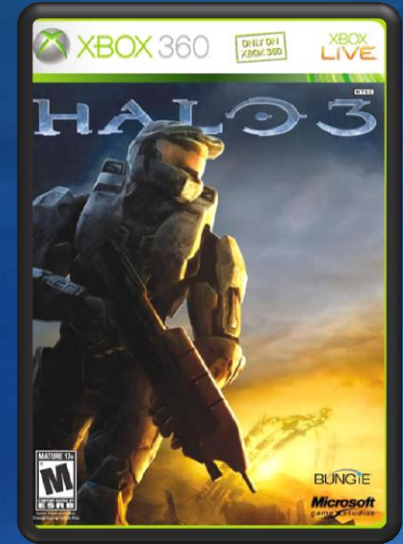
Xbox 360 & Halo 3

● Xbox 360 Live

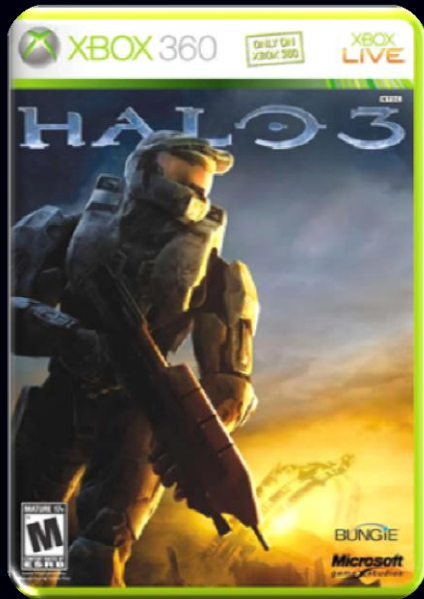
- Launched in September 2005
- Every game uses TrueSkill™ to match players
- > 35 million players
- > 4 million matches per day
- > 2 billion hours of gameplay / month

● Halo 3

- Launched on 25th September 2007
- Largest entertainment launch in history
- > 200,000 player concurrently (peak: 1,000,000)



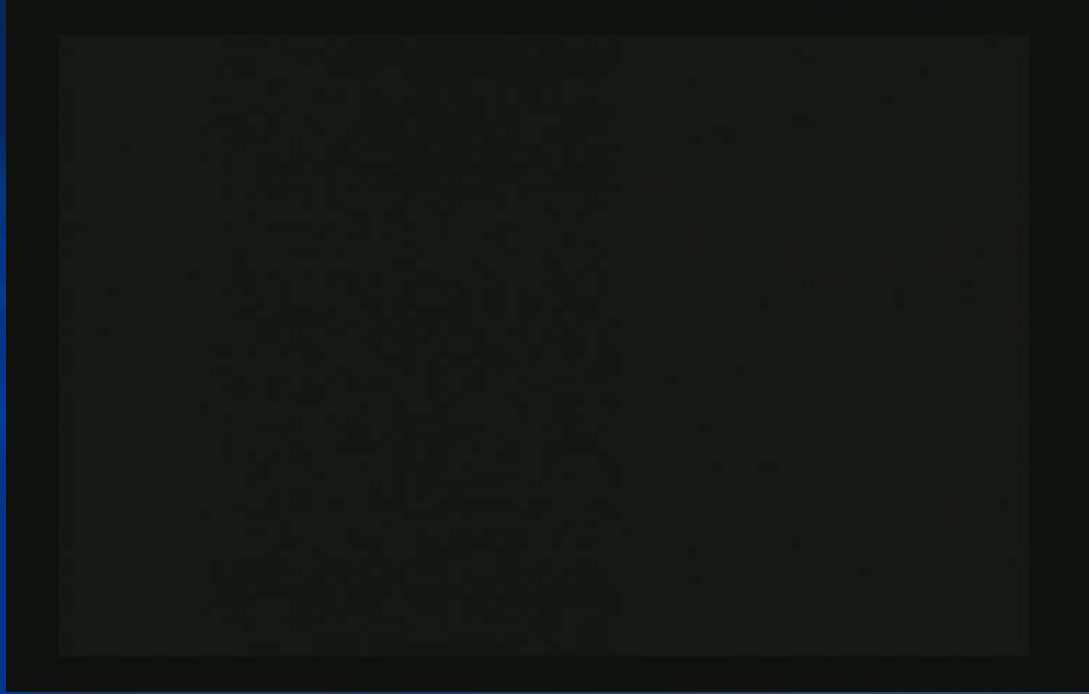
Demo: Halo 3



Halo 3 Game

- Matchmaking
- Skill Stats
- Tight Matches

Halo 3 in Action



Overview

- Why Machine Learning and Games?
- Machine Learning in Video Games
 - Drivatars™
 - Reinforcement Learning
- Machine Learning in Online Games
 - TrueSkill™
 - Halo 3
- **The Path of Go**
- Conclusions

Learning to Play Go



The Game of Go

- Started about 4000 years ago in ancient China.
- About 60 million players worldwide.
- 2 Players: Black and White.
- Board: 19×19 grid.
- Rules:
 - Turn: stone placed on vertex.
 - Capture.
- Aim: Gather territory



Computer Go

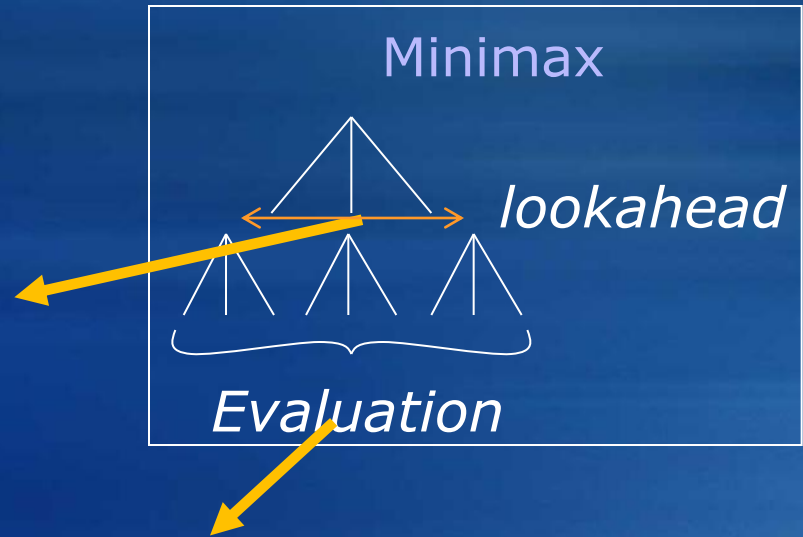
- 5th November 1997:
Gary Kasparov beaten by Deep Blue.



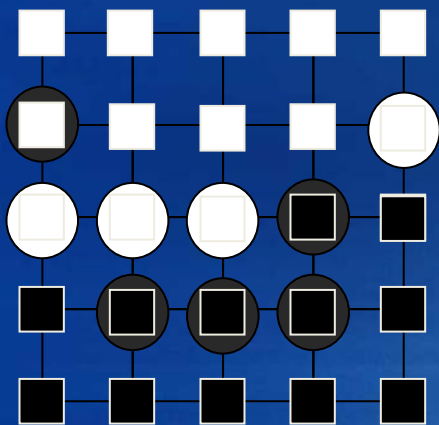
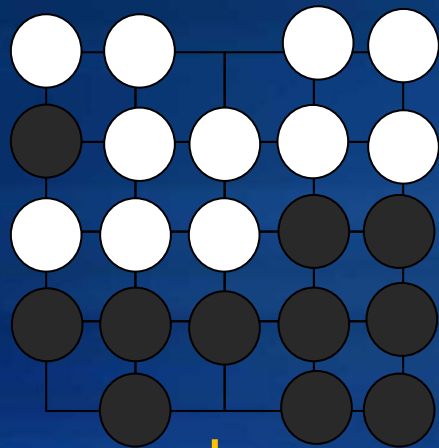
- Best Go programs cannot beat amateurs.

Computer Go

- Minimax search defeated.
- **High Branching Factor.**
 - Go: ~200
 - Chess: ~35
- **Complex Position Evaluation.**
 - Stone's value derived from configuration of surrounding stones.

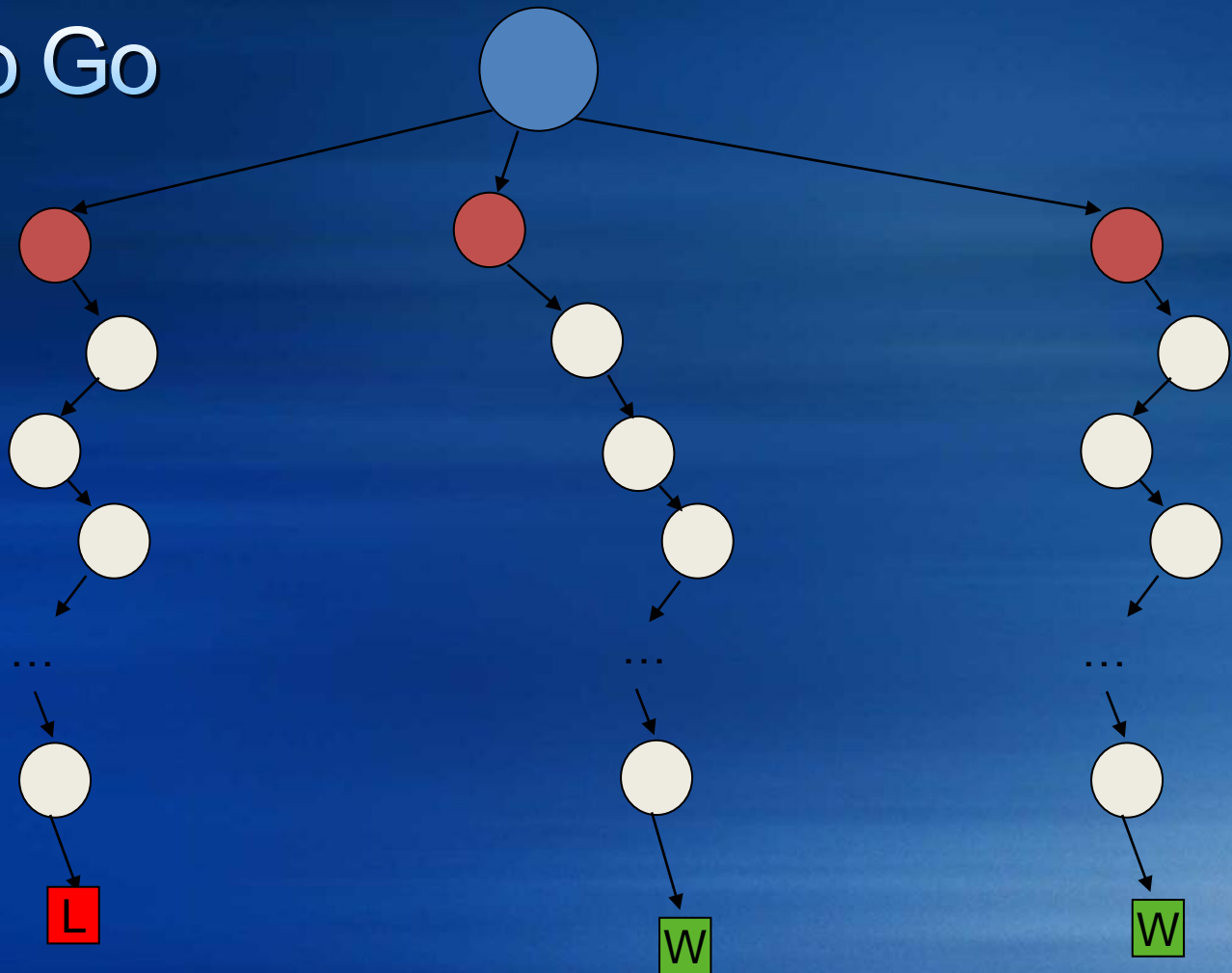


Monte Carlo Go

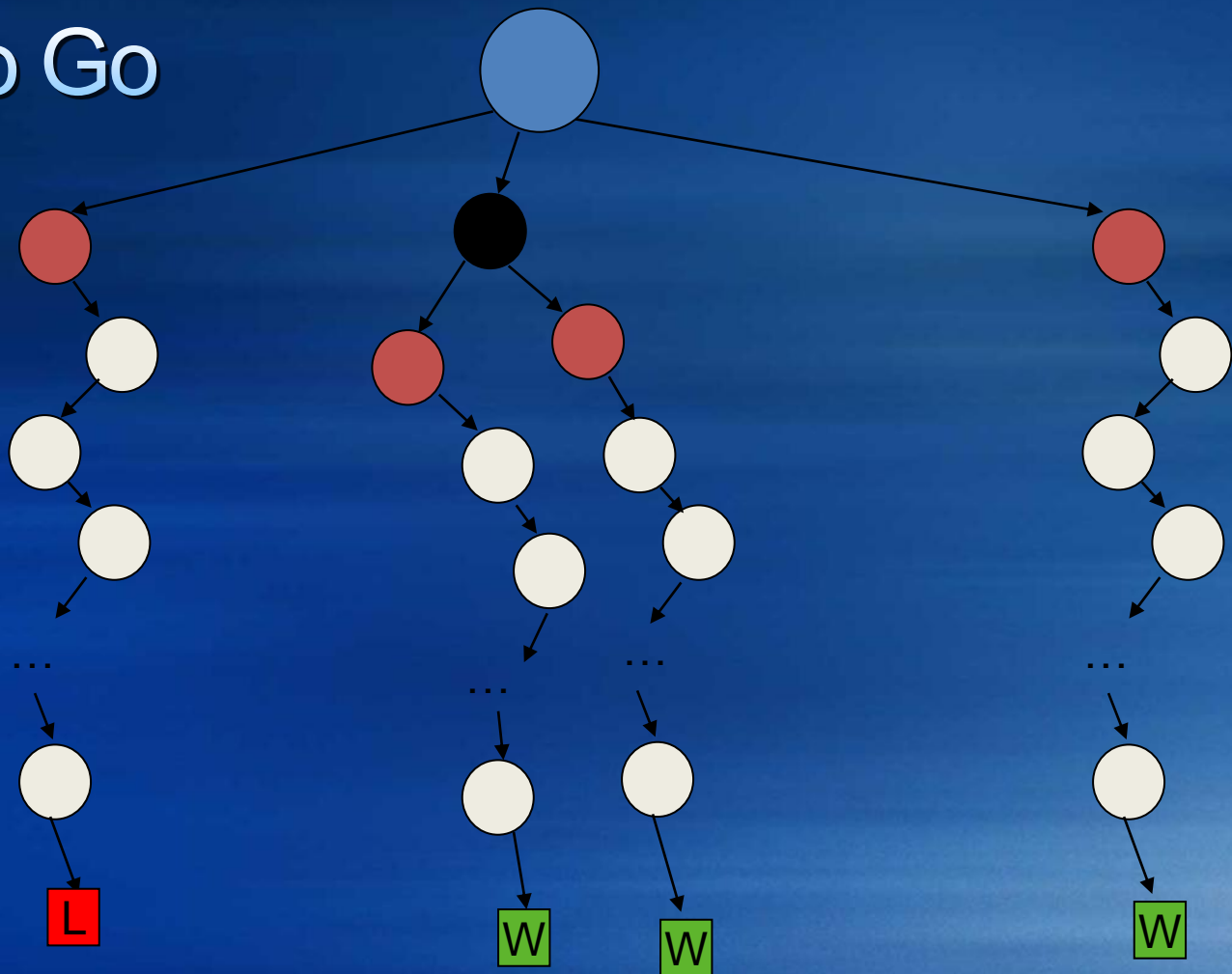


Territory Hypothesis

Monte Carlo Go

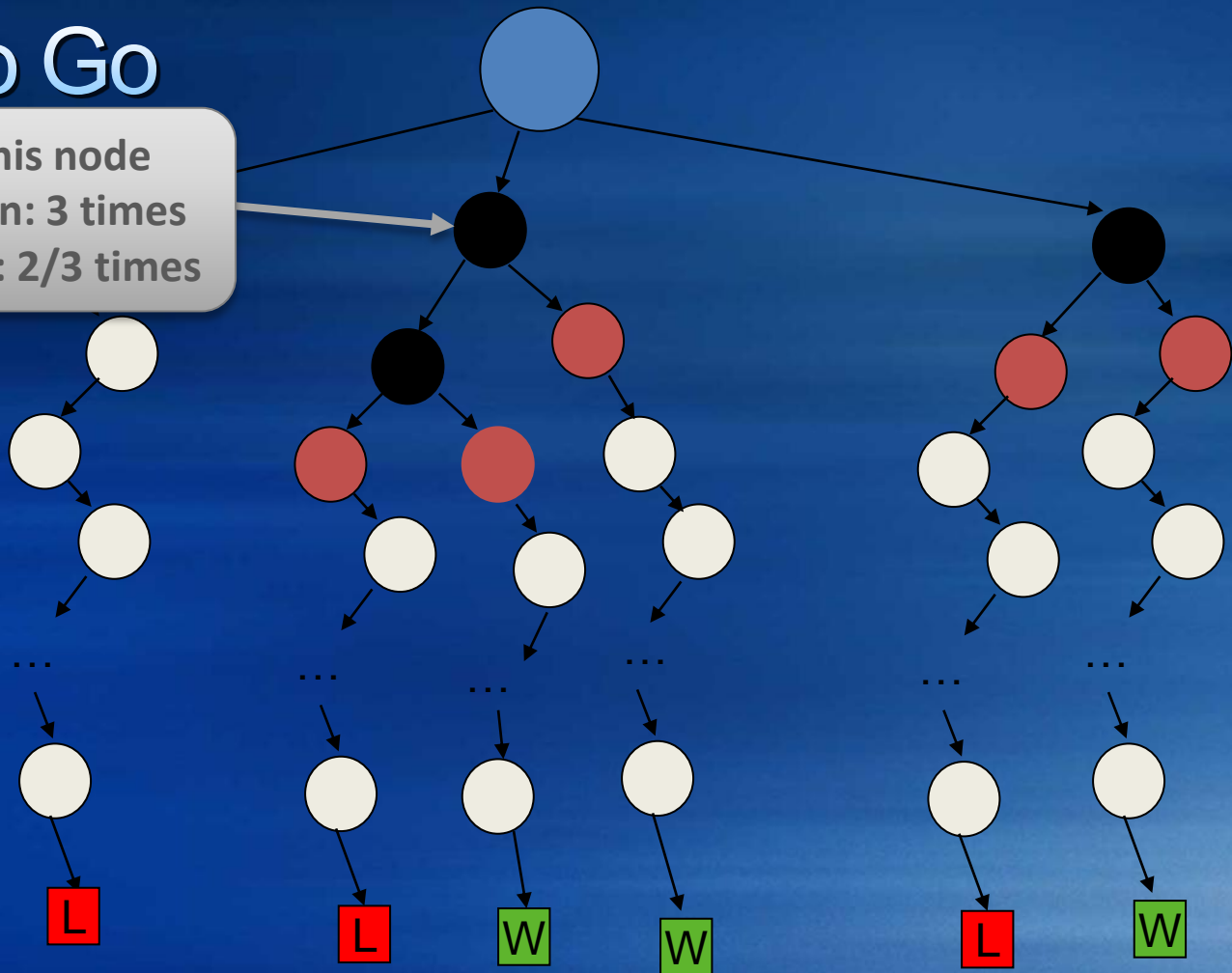


Monte Carlo Go

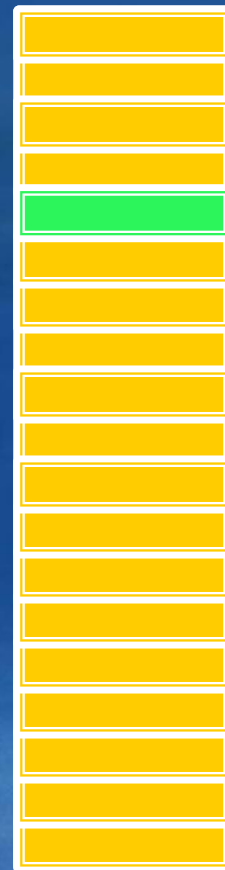


Monte Carlo Go

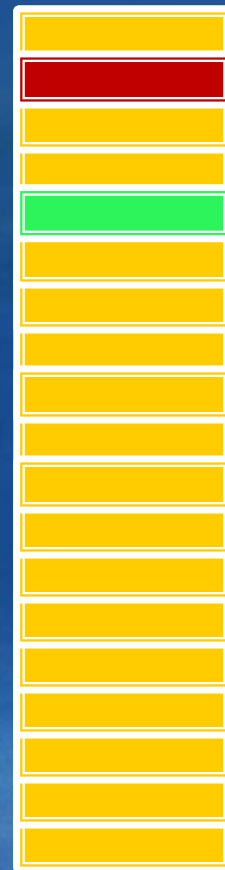
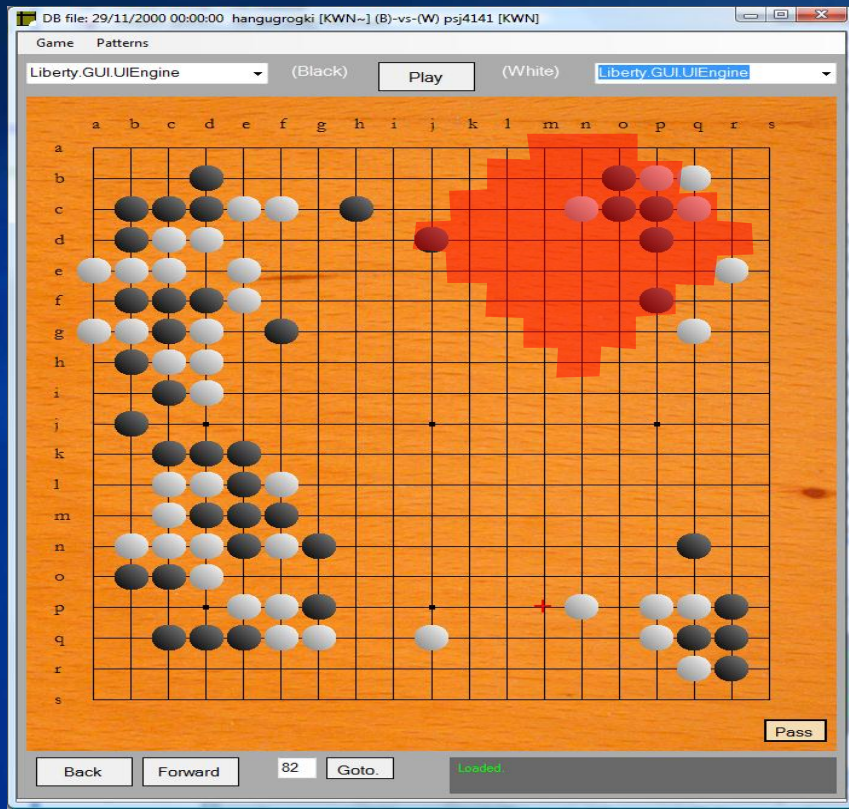
This node
Seen: 3 times
Win: 2/3 times



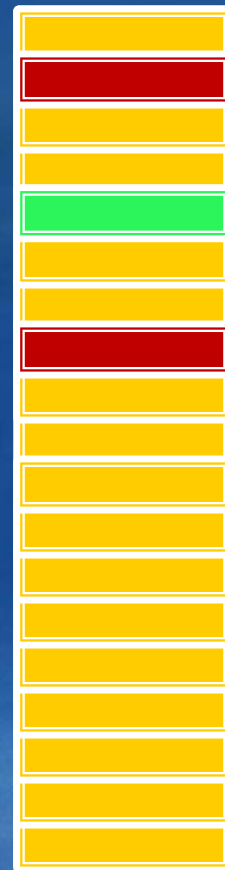
Learning From the Experts with TrueSkill



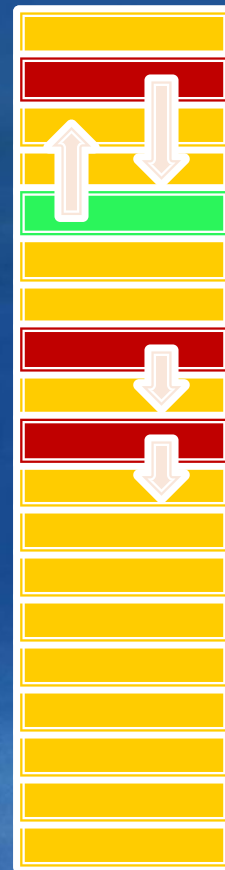
Learning From the Experts with TrueSkill



Learning From the Experts with TrueSkill

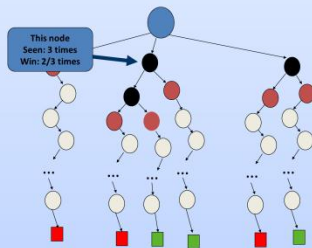
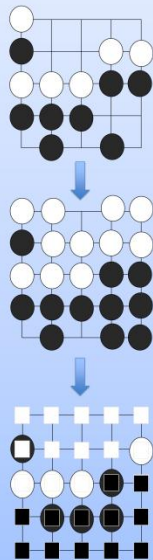


Learning From the Experts with TrueSkill



Machine Learning Assisted Monte Carlo Go

Monte Carlo Go



- Play random games ('rollouts').
- Each game gives a sample win/loss.
- Average estimates position value.
- Store game tree in memory.
- Bootstrap rollout policy.

Pattern Ranking System

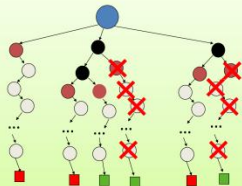


Pattern Table

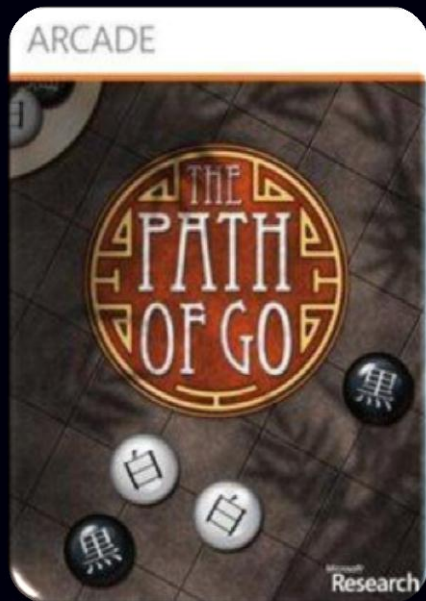
- Learn pattern rankings using TrueSkill.
- Moves chosen over other moves by experts are inferred to have higher value.
- Training Data: 200,000 Expert Go Games.

Pattern Pruning

- Too many possible moves to evaluate.
- Pattern system estimates move quality.
- Prune bad moves from the game tree.



Demo: The Path of Go



The Path of Go

- MSRC Go AI (written in F#)
- TrueSkill Match Making
- XNA Game Studio

Overview

- Why Machine Learning and Games?
- Machine Learning in Video Games
 - Drivatars™
 - Reinforcement Learning
- Machine Learning in Online Games
 - TrueSkill™
 - Halo 3
- The Path of Go
- **Conclusions**

Conclusions

- Computer games can be used as test beds for research.
- Machine learning can be used to improve the user experience in computer games.
- Both research and applications are in their infancy and there are many open questions.
- XNA framework exists to plug in machine learning algorithms.
- For more question, please drop us a line

Joaquin Quiñonero Candela, Ralf Herbrich, Thore Graepel
Online Services and Advertising Group

joaquinc @ microsoft.com

Microsoft[®]

Your potential. Our passion.[™]

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries. The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED, OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.