

Applied machine learning in game theory

Dmitrijs Rutko

Faculty of Computing
University of Latvia

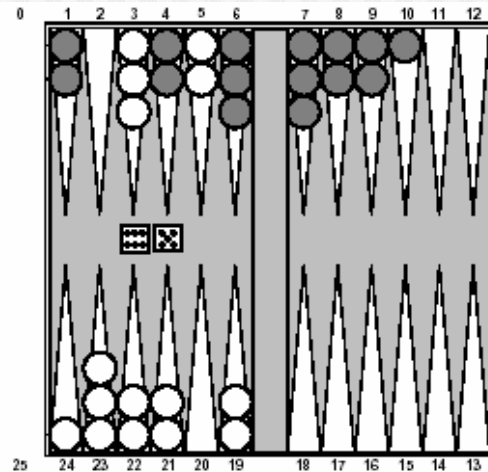
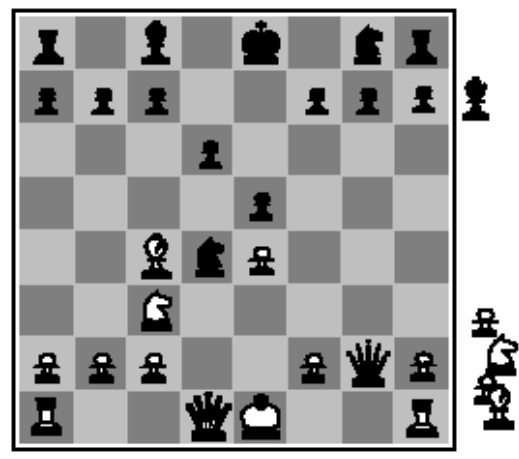
Joint Estonian-Latvian Theory Days at Rakari, 2010

Topic outline

- Game theory
 - Game Tree Search
 - Fuzzy approach
- Machine learning
 - Heuristics
 - Neural networks
 - Adaptive / Reinforcement learning
- Card games

Research overview

- Deterministic / stochastic games
- Perfect / imperfect information games



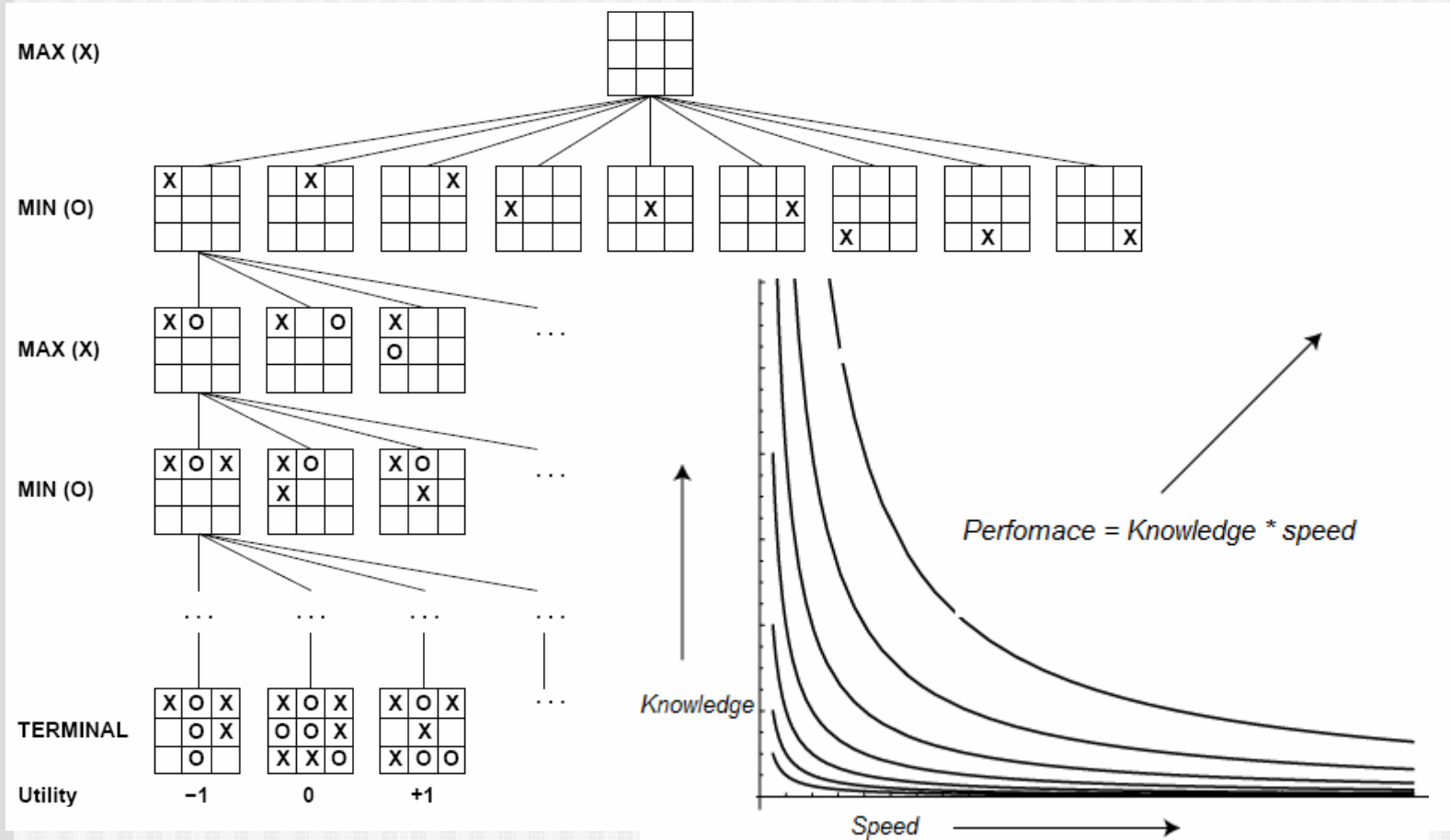
Finite zero-sum games

	deterministic	chance
perfect information	chess, checkers, go, othello	backgammon, monopoly, roulette
imperfect information	battleship, kriegspiel, rock-paper-scissors	bridge, poker, scrabble

Topic outline

- **Game theory**
 - Game Tree Search
 - Fuzzy approach
- Machine learning
 - Heuristics
 - Neural networks
 - Adaptive / Reinforcement learning
- Card games

Game trees



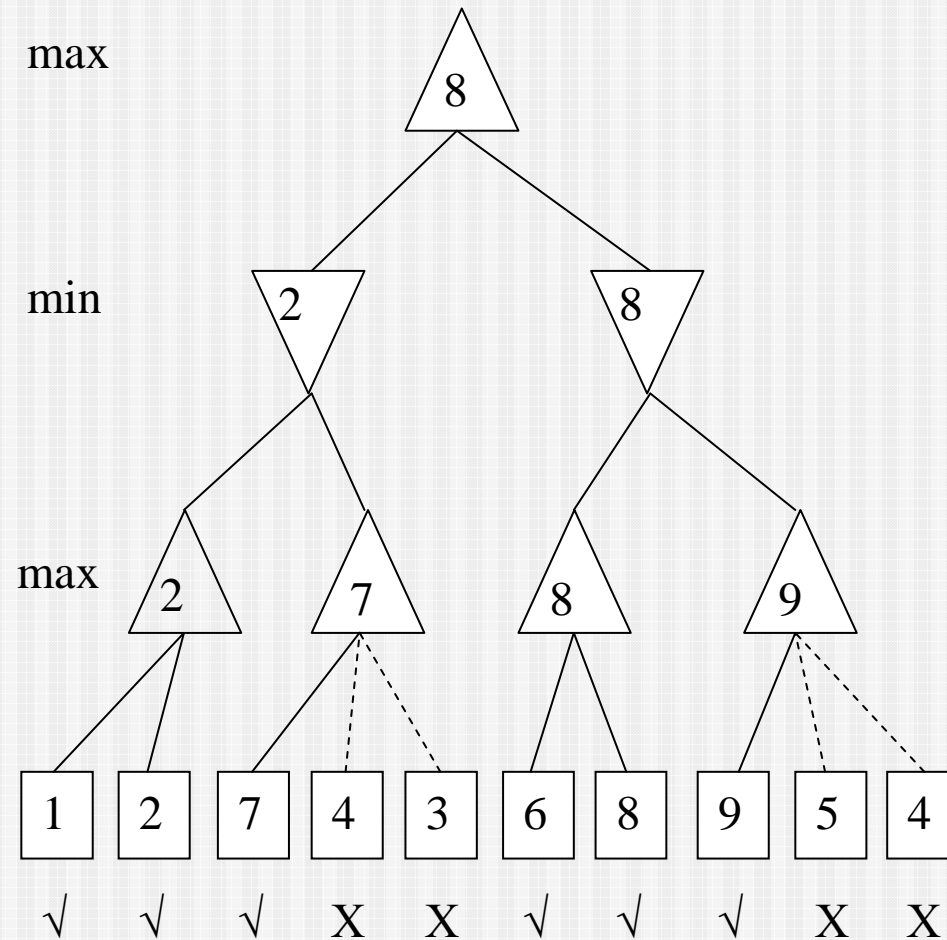
Classical algorithms

- MiniMax

- $O(w^d)$

- Alpha-Beta

- $O(w^{d/2})$

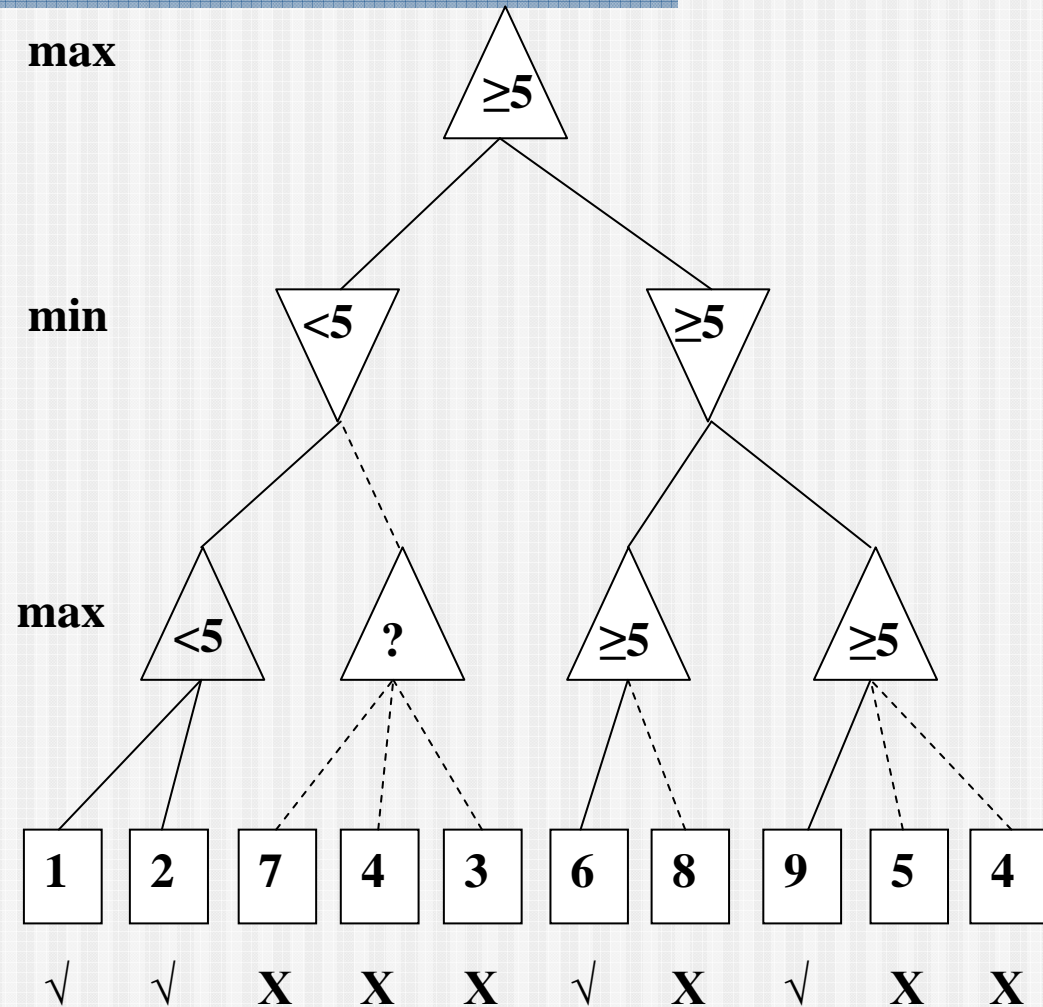


Advanced search techniques

- Transposition tables
- Time efficiency / high cost of space
 - PVS
 - Negascout
 - NegaC*
 - SSS* / DUAL*
 - MTD(f)

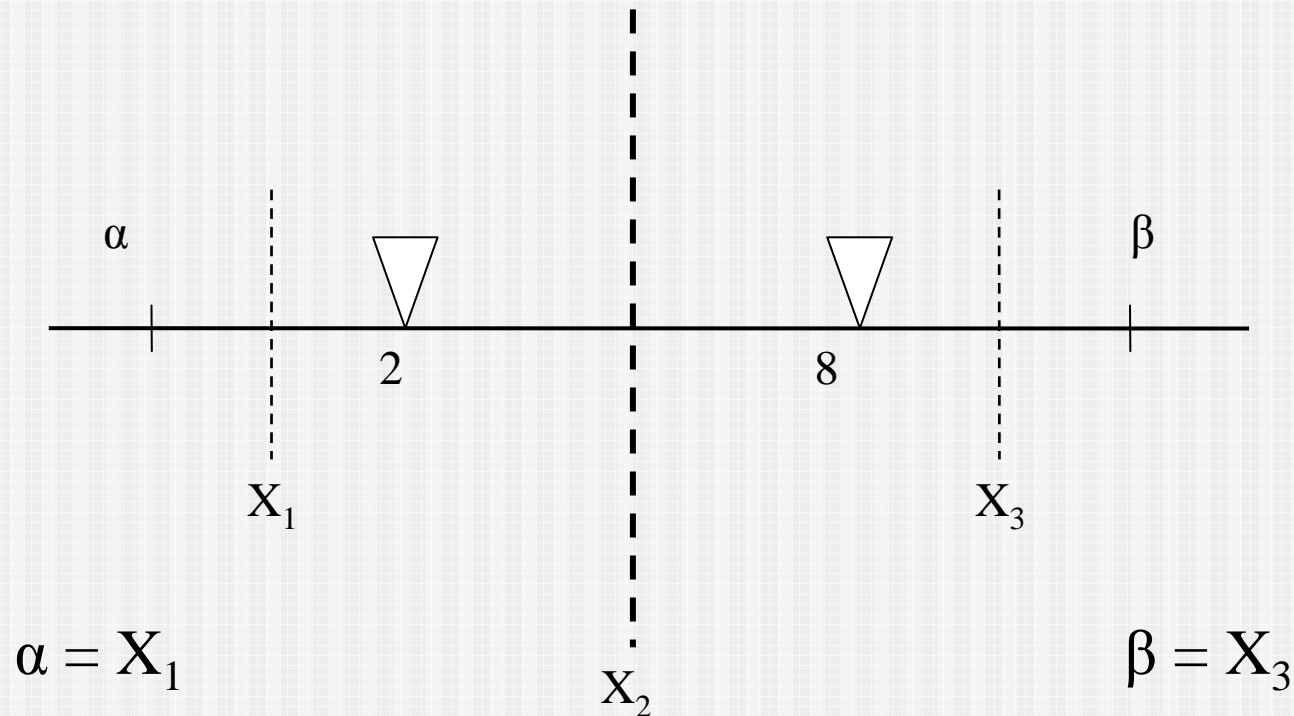
Fuzzy approach

- $O(w^{d/2})$
- More cut-offs



Geometric interpretation

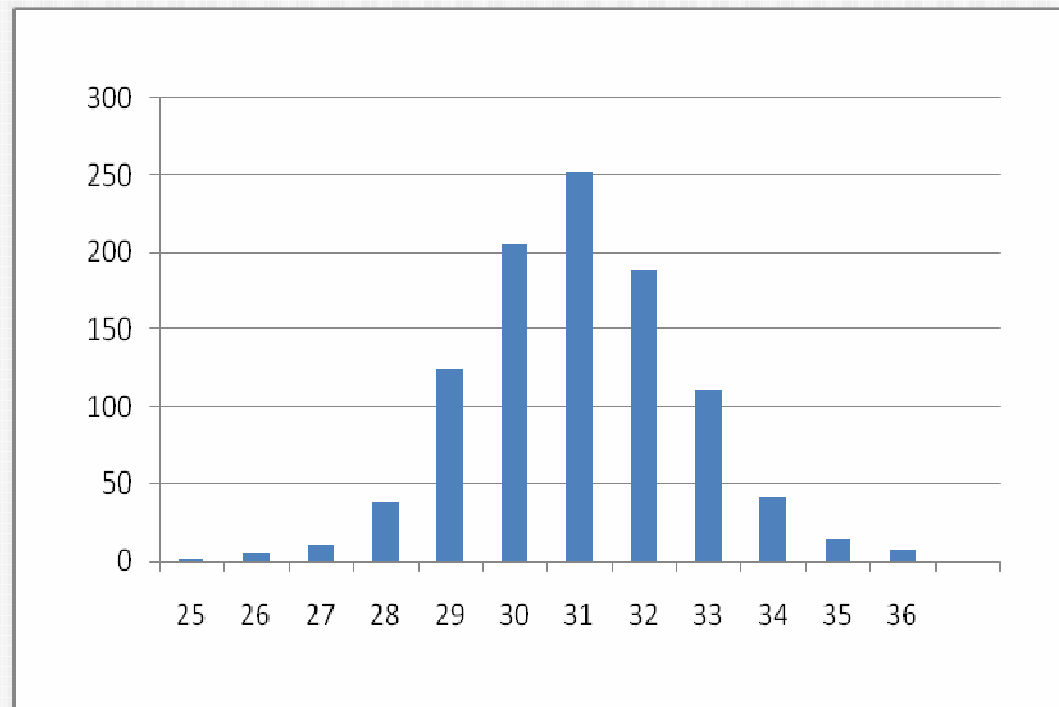
- 1) X_2 - successful separation
- 2) X_1 or X_3 - reduced search window



BNS enhancement through self-training

■ Traditional statistical approach

Minimax value	Tree count
25	1
26	5
27	11
28	38
29	124
30	206
31	252
32	189
33	111
34	42
35	14
36	7
1000	

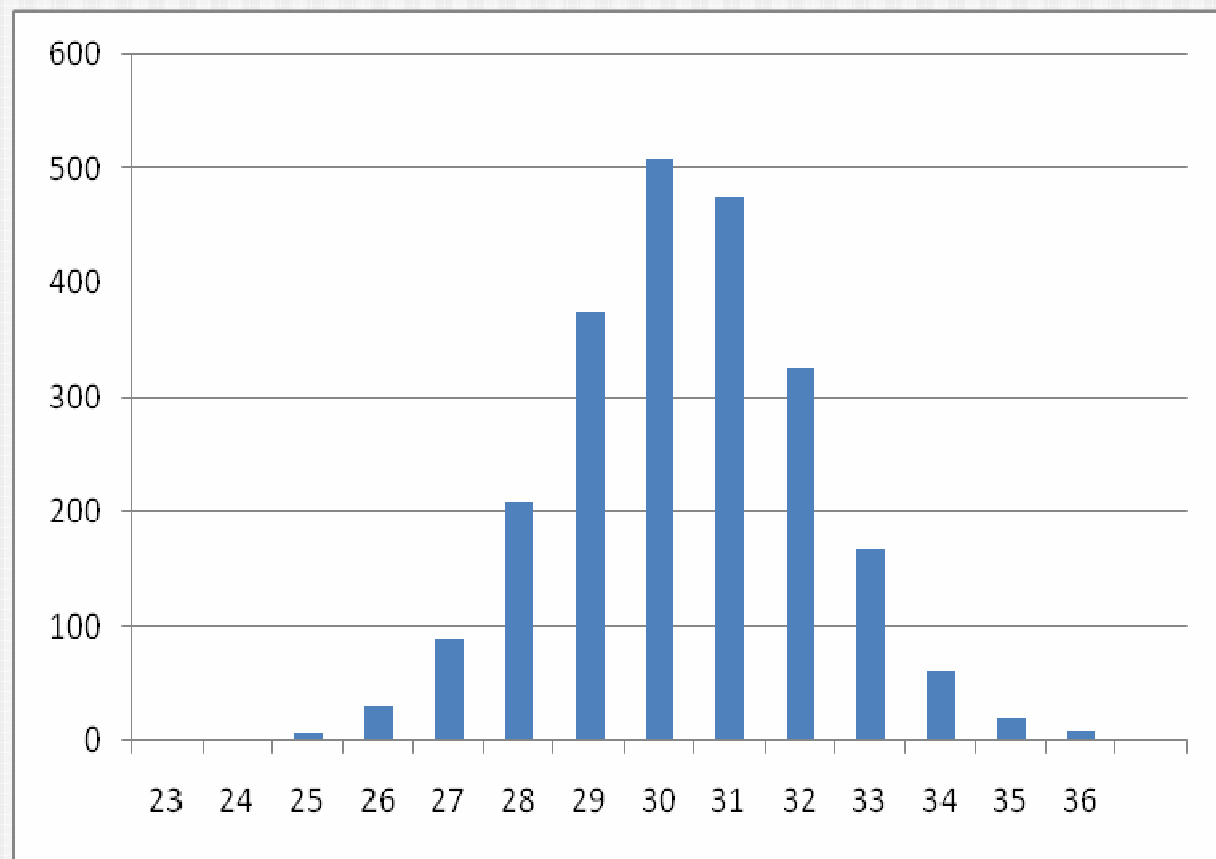


Two dimensional game sub-tree distribution

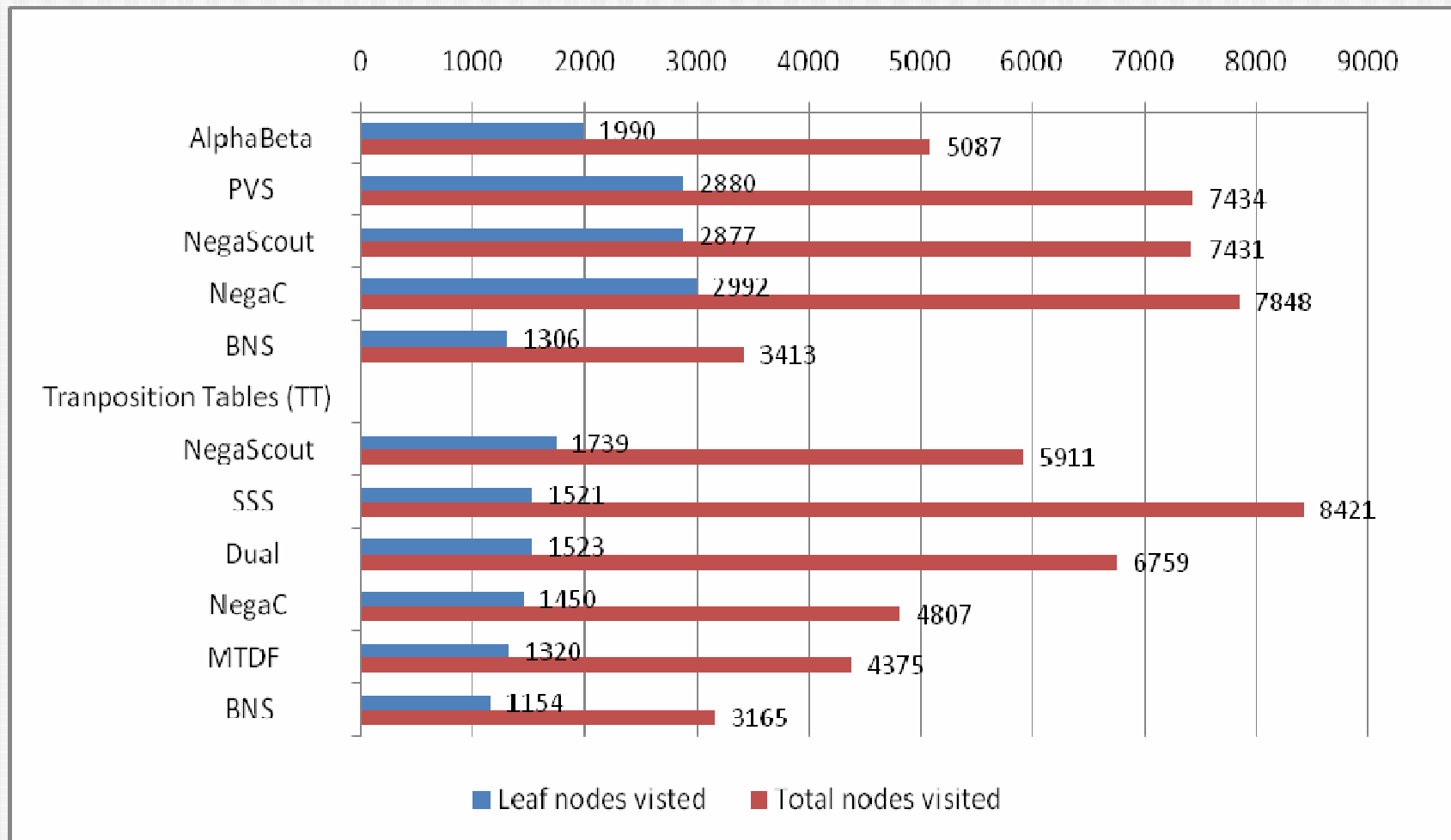
	23	24	25	26	27	28	29	30	31	32	33	34	35	36	Tree count
23	0														0
24	0	0													0
25	0	1	0												1
26	0	0	2	3											5
27	0	0	5	3	3										11
28	0	1	0	12	12	13									38
29	0	0	2	10	35	43	34								124
30	1	2	6	9	26	58	71	33							206
31	0	0	6	10	27	41	78	57	33						252
32	0	1	3	13	17	30	32	41	38	14					189
33	0	0	1	2	8	12	26	28	21	11	2				111
34	0	0	0	1	3	5	13	8	6	2	2	2			42
35	0	0	0	0	0	2	4	3	2	3	0	0	0		14
36	0	0	0	0	0	0	1	2	2	1	1	0	0	0	7

Statistical sub-tree separation

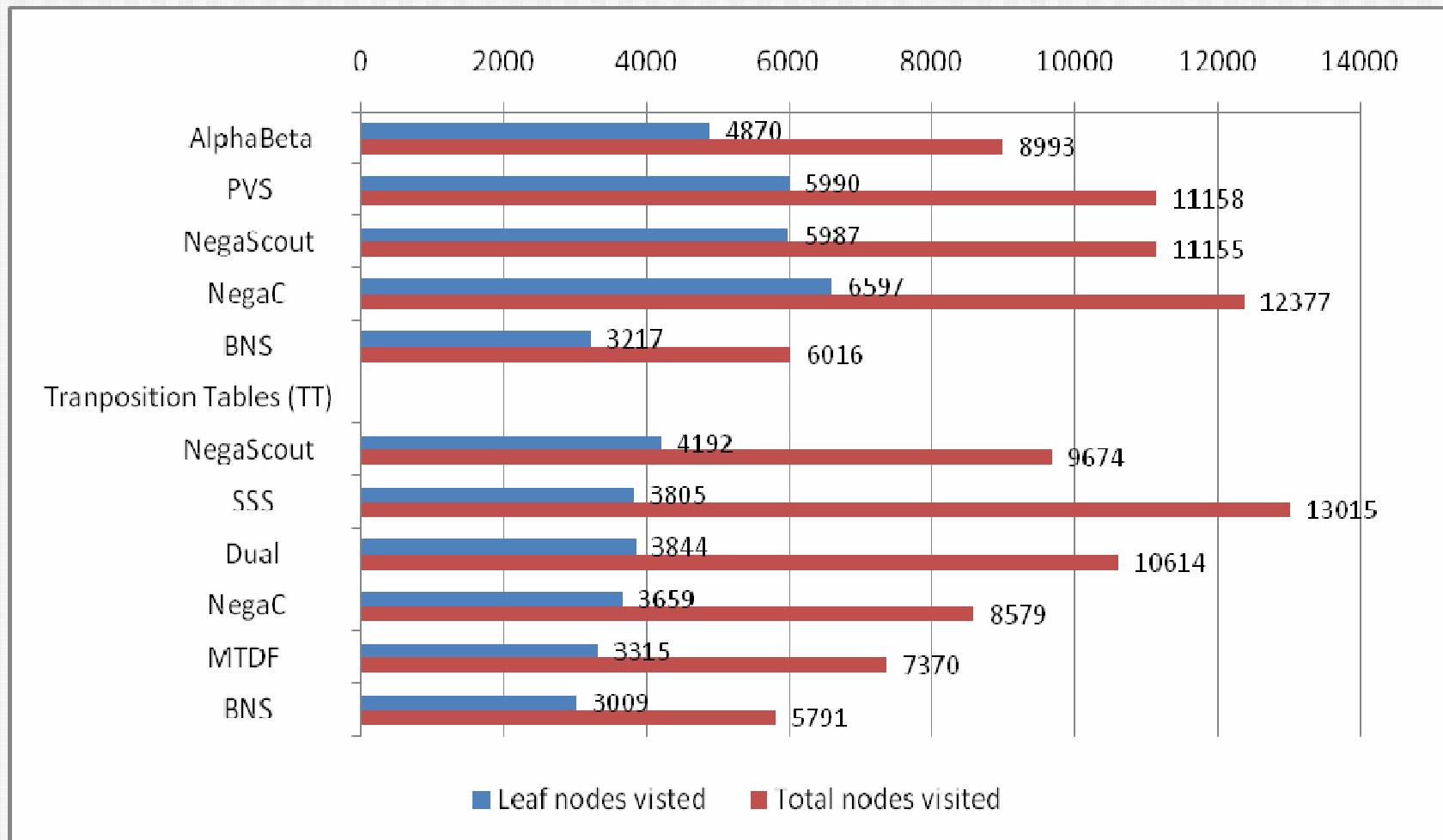
Separation value	Tree count
23	0
24	1
25	6
26	30
27	88
28	208
29	374
30	509
31	475
32	325
33	167
34	61
35	21
36	7
	2272



Experimental results. 2-width trees



Experimental results. 3-width trees



Future research directions in game tree search

- Multi-dimensional self-training
- Wider trees
- Real domain games

Topic outline

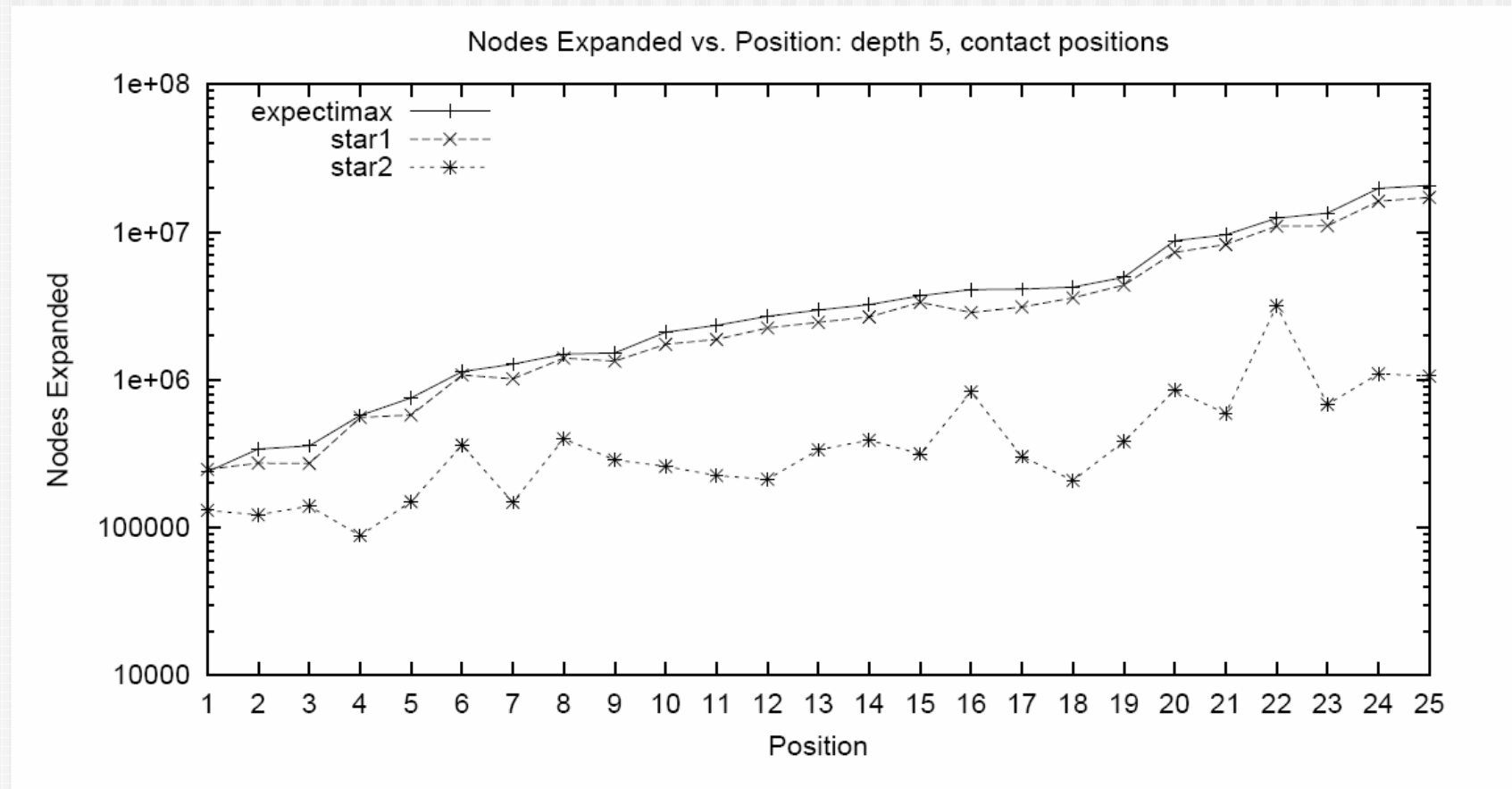
- Game theory
 - Game Tree Search
 - Fuzzy approach
- **Machine learning**
 - Heuristics
 - Neural networks
 - Adaptive / Reinforcement learning
- Card games

Expectiminimax algorithm

- $Expectiminimax(n) =$
 - $Utility(n)$
 - If n is a terminal state
 - $Max s \in Successors(n) Expectiminimax(s)$
 - if n is a max node
 - $Min s \in Successors(n) Expectiminimax(s)$
 - if n is a min node
 - $\sum s \in Successors(n) P(s) * Expectiminimax(s)$
 - if n is a chance node

- $O(w^d c^d)$

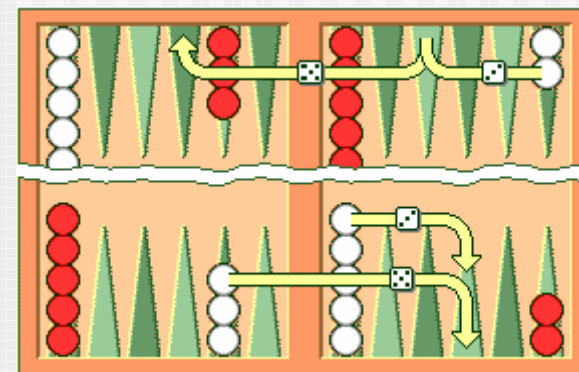
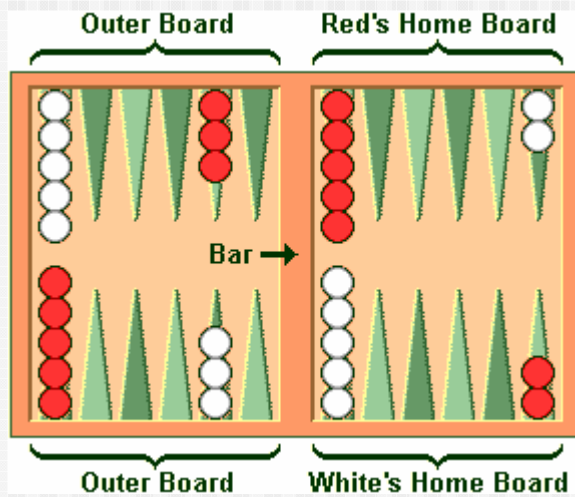
Performance in Backgammon



*-Minimax Performance in Backgammon,
Thomas Hauk, Michael Buro, and Jonathan Schaeer

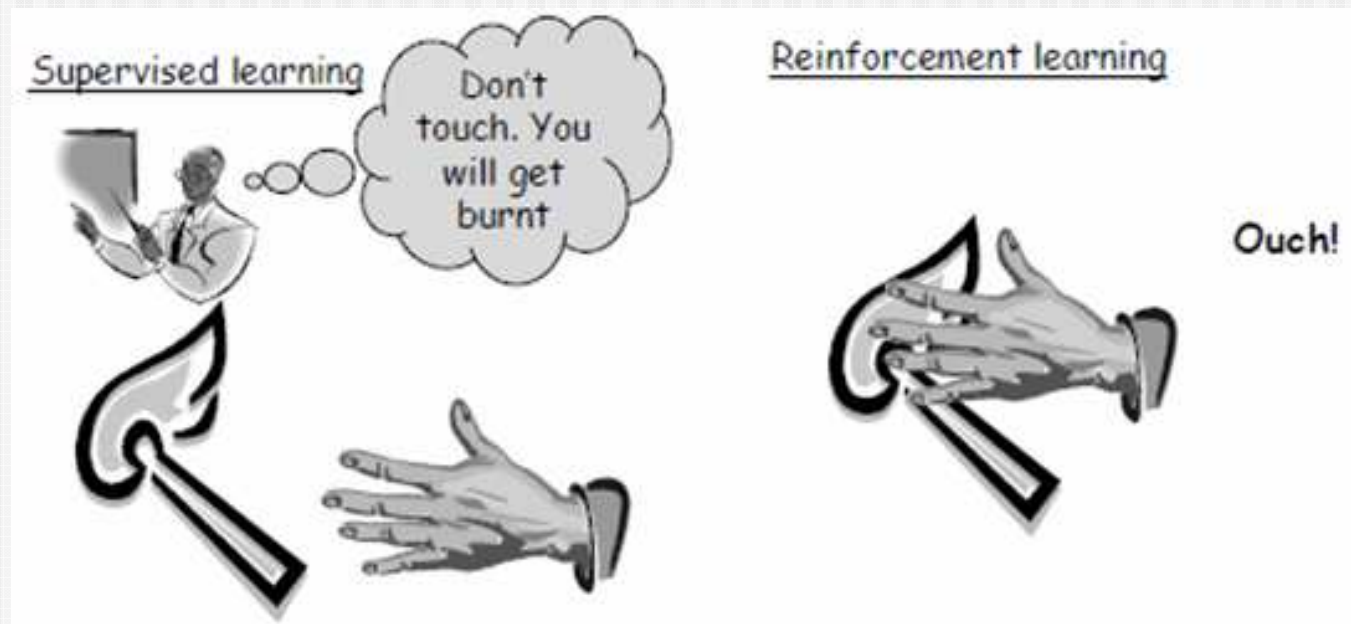
Backgammon

- Evaluation methods
 - Static – pip count
 - Heuristic – key points
 - Neural Networks



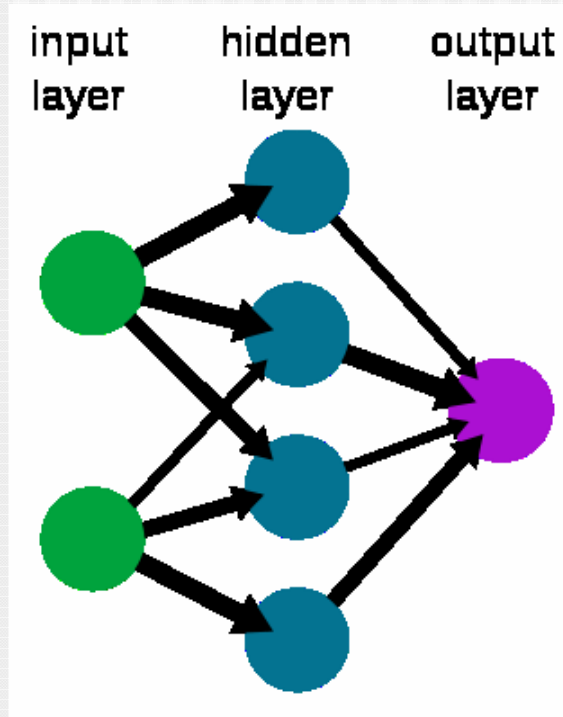
Temporal difference (TD) learning

- Reinforcement learning
- Prediction method

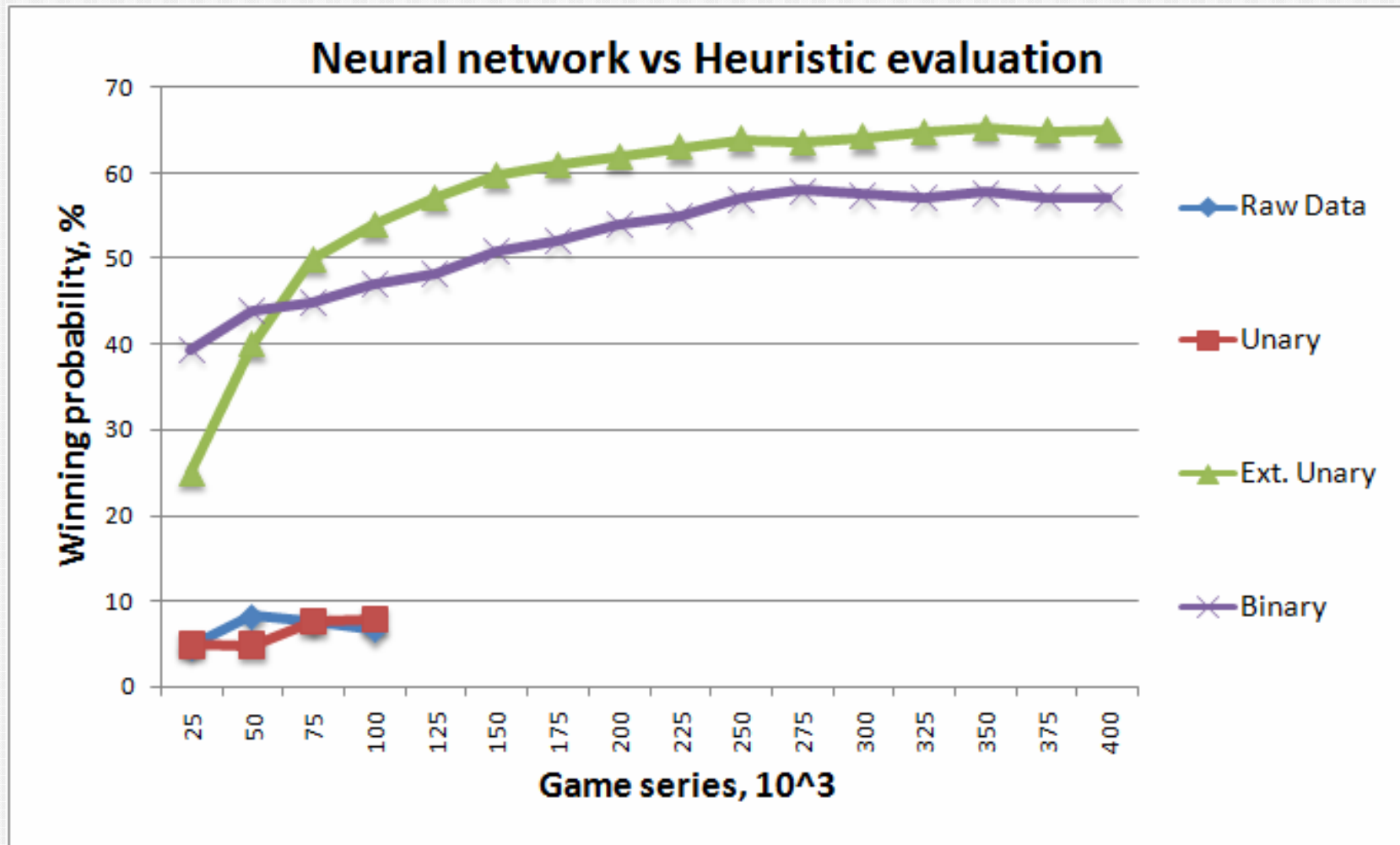


Experimental setup

- Multi-layer perceptron
- Representation encoding
 - Raw data (27 inputs)
 - Unary (157 inputs)
 - Extended unary (201 inputs)
 - Binary (201 input)
- Training game series – 400 000 games



Learning results



Program “DM Backgammon”

The screenshot displays the DM Backgammon program interface. The game board is a 14-point backgammon board with points numbered 1 to 14 from right to left. The board is divided into two halves by a central bar. The top half (points 1-7) is currently controlled by Player 2, and the bottom half (points 8-14) is controlled by Player 1. The board shows the following piece distribution:

- Player 1 (Black): 3 pieces on point 12, 3 pieces on point 13, 3 pieces on point 14, 1 piece on point 17, 1 piece on point 18, 4 pieces on point 19, 3 pieces on point 22, 1 piece on point 23, and 2 pieces on point 24.
- Player 2 (Red): 2 pieces on point 7, 2 pieces on point 8, 1 piece on point 4, 2 pieces on point 5, 1 piece on point 6, 1 piece on point 2, 1 piece on point 1, 3 pieces on point 13, 3 pieces on point 14, 1 piece on point 17, and 2 pieces on point 24.

The central bar is empty. The interface includes a menu bar (File, Options, Help), a status bar (Player1, Moves: 0 of 2, Searched: 1, Elapsed: 0, EVAL: 0,479), and two move lists:

Player1	Player2
1: 21 1/3 3/4	1: 12 6/4* 8/7
2: 13 bar/1 1/4*	2: 11 bar/24 6/5 6/5 8/7
3: 51 1/2 12/17	3: 15 13/8 24/23
4: 55 12/17 17/22 17/22	4: 24 6/2* 6/4*
17/22	5:
5: 24	

Topic outline

- Game theory
 - Game Tree Search
 - Fuzzy approach
- Machine learning
 - Heuristics
 - Neural networks
 - Adaptive / Reinforcement learning
- Card games

Artificial Intelligence and Poker*

AI Problems	Poker problems
Imperfect information	Hidden cards
Multiple agents	Multiple human players
Risk management	Bet strategy and outcome
Agent modeling	Opponent(s) modeling
Misleading information	Bluffing
Unreliable information	Taking bluffing into account

* Joint work with Annija Rupeneite

Questions ?

dim_rut@inbox.lv