RHEA Enhancements for GVGP

Raluca D. Gaina, Simon M. Lucas, Diego Perez-Liebana

IEEE CIG 2017

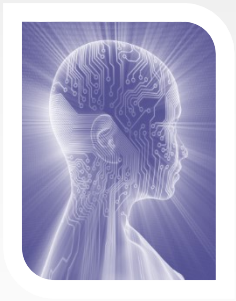
University of Essex

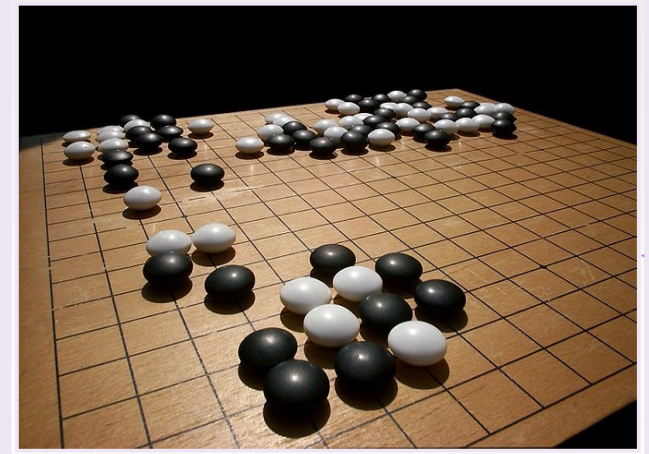
iGGi

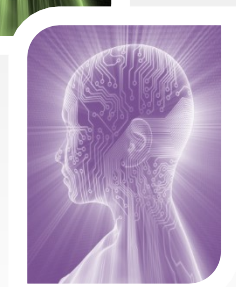
Introduction

- Rolling Horizon Evolutionary Algorithms (RHEA) show promise
  - in General Video Game Playing (GVGP)
  - as showcased in the General Video Game AI Competition (GVGAI).
- Several improvements in literature in various contexts
  - do they work in GVGP?
  - do they work together?

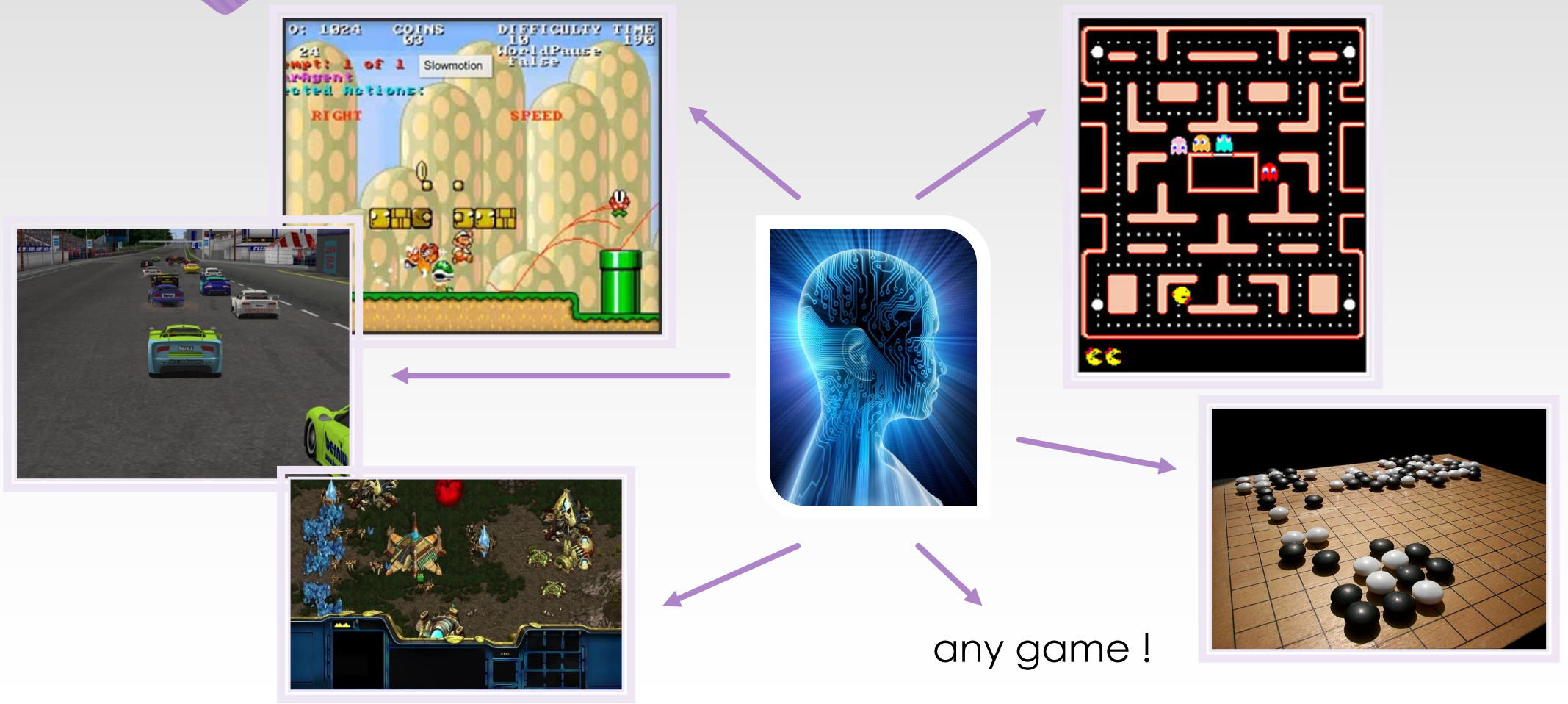
Game-Playing AI







General Video Game AI



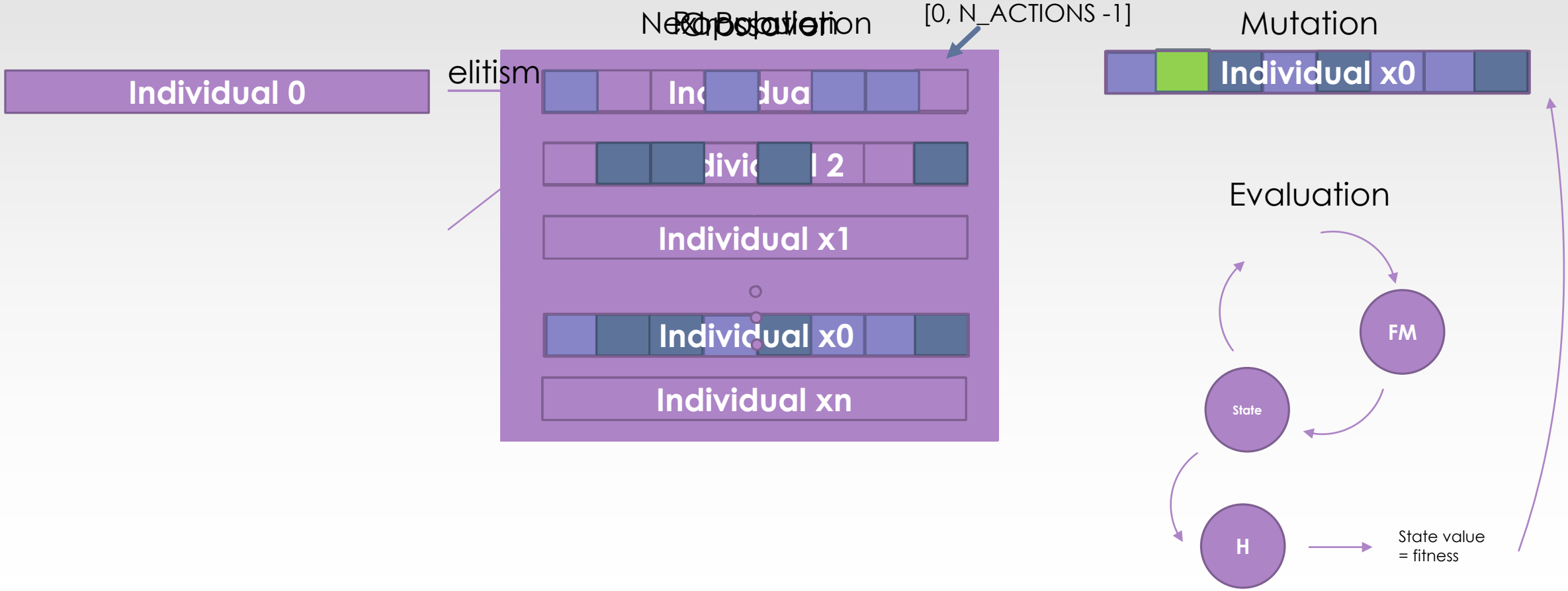
any game !

General Video Game AI Competition

- 2D grid-physics games
- Arcade, puzzles, shooters, adventure.
  - Ways to interact with the environment
  - Ways to win
  - Elements in a game
  - Scoring systems
  - Single and two player, cooperative and competitive.

... high-level view of current game state for agents; real-time decisions (40ms)

Rolling Horizon Evolution

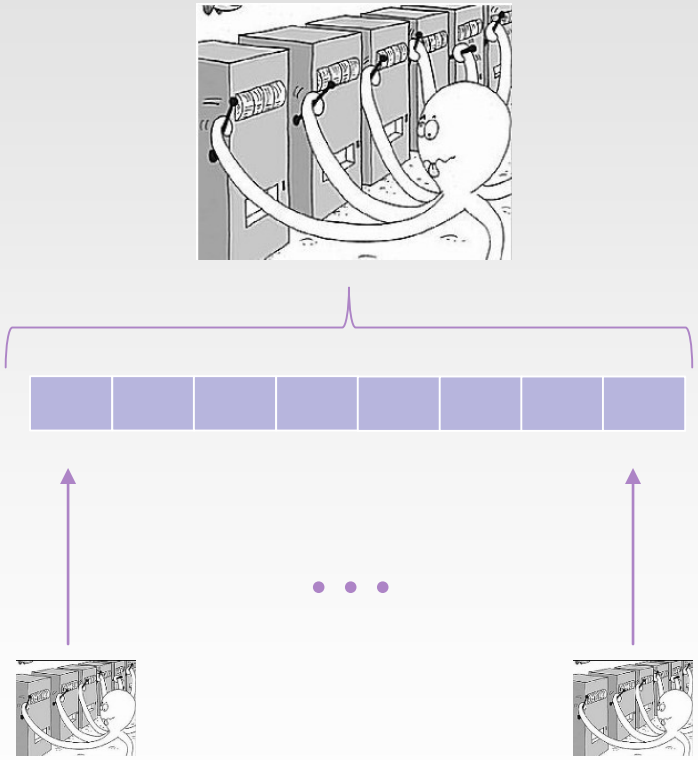


Methodology

- Look at 4 parts in the evolution process …
  - Mutation operator
  - Population management
  - Action recommendation policy
  - Individual evaluation
- … in isolation and combined … → hybrids
- … split into 2-part experiment …
- … on 20 GVGAI games …
- … with different core parameter configurations.

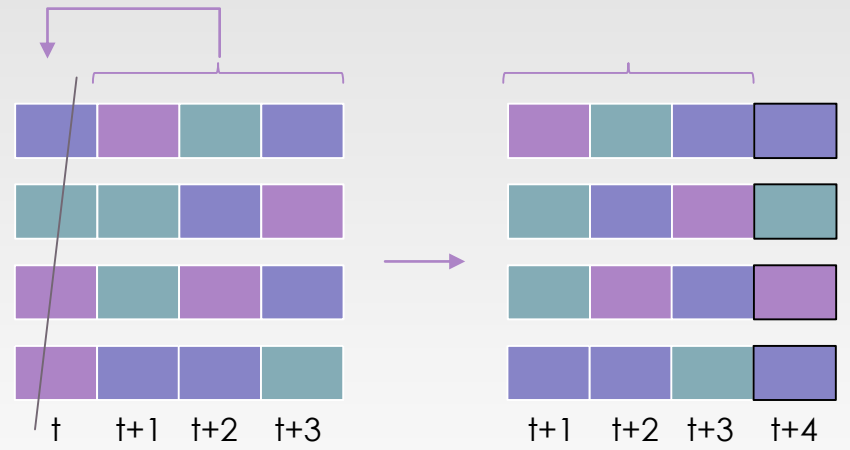
Mutation operator

- Bandit-based mutation (EA-bandit)
- $UCB1 = \underset{a \in A(s)}{argmax} \left\{ Q(s,a) + C\sqrt{\frac{lnN(s)}{N(s,a)}} \right\}$
- 2-layer UCB
  - Individual level: which gene?
  - Gene level: which value?



Population management

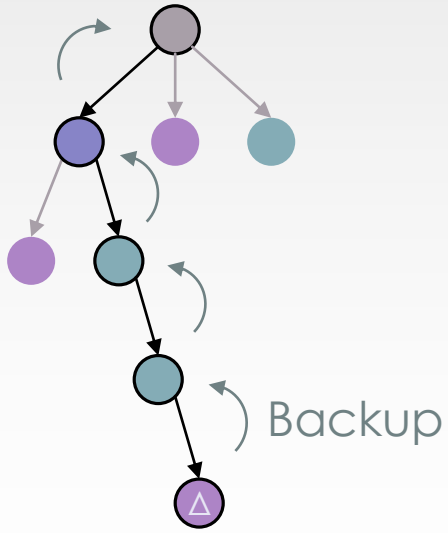
- Shift buffer (EA-shift)
- Keep population between game ticks, no resetting
- Shift population to the left at next game tick
- Add new random action at the end



Game step t Game step t+1

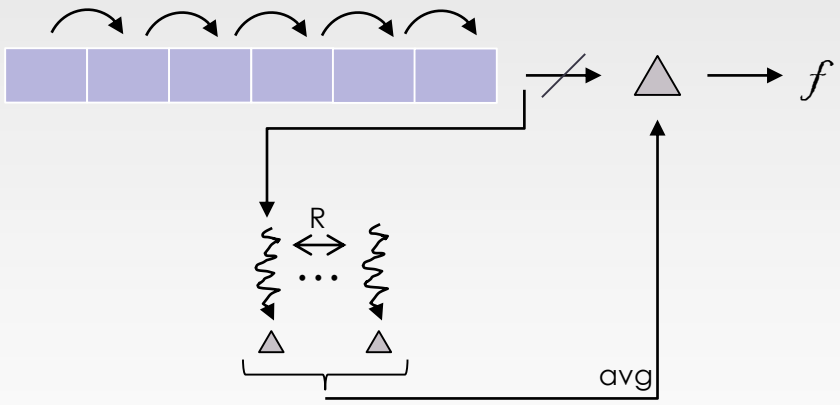
Action recommendation policy

- Statistical tree (EA-tree)
- During evaluation, keep action statistics in a tree structure
- Similar to Monte Carlo Tree Search …
- … but tree only used to recommend action
- Final action: most visited node at top level



Individual evaluation

- Monte Carlo rollouts (EA-roll)
- At the end of individual evaluation
  - Monte Carlo simulation
  - Length L/2.
- Repeat R times
  - Use average value as individual fitness



Experiment

- Population size P - Individual length L = {1-6, 2-8, 5-10, 10-14}
- All other parameters fixed to default values
- Budget: 900 Forward Model calls
- First part:
  - EA-bandit, EA-tree and EA-shift (plus hybrids)
- Second part:
  - EA-tree, EA-shift and EA-roll (plus hybrids)
- Validation
  - Comparison with MCTS.

20 Games from GVGAI corpus

Survive Zombies

Aliens

Sea Quest

Results overview (part 1)

- Shift buffer best, Bandit mutation worst
- Performance proportional to parameter values, but algorithm ranking not stable
- Shift buffer matches and surpasses best vanilla performance even with small parameter values

Win percentage for config 5-10. Color bar: in how many unique games row was significantly better than column.

A	Vanilla
B	EA-Shift
C	EA-Tree
D	EA-Tree-Shift
E	EA-Bandit
F	EA-Bandit-Shift
G	EA-Bandit-Tree
H	EA-Bandit-Tree-Shift

EA-bandit (part 1)

- One of worst variants (Vanilla RHEA better)
- 1-6: worst configuration

- Most beneficial in large configs.

EA-tree (part 1)

- Mid-table, better than Vanilla and EA-Bandit
- Worst hybrids: +bandit mutation

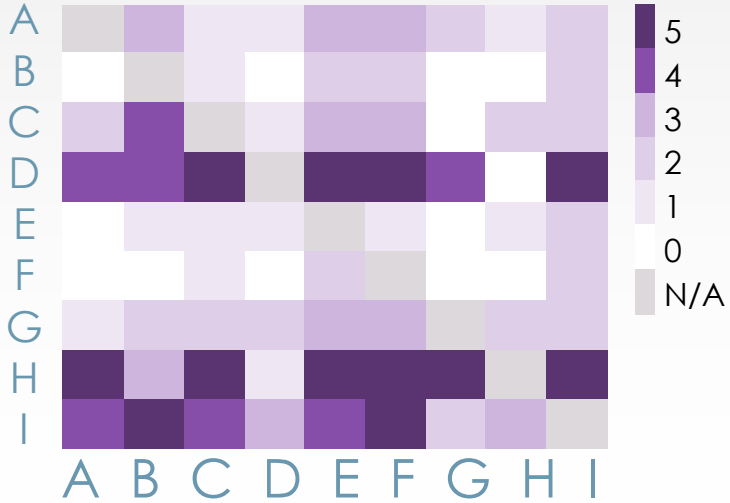
- Most beneficial in low configs.

EA-shift (part 1)

- Best variant
- Higher scores in most games
- 1-6: tree hybrids better
- Worst hybrids: +bandit mutation

EA-roll (part 2)

- Best: EA-Shift-Roll (10-14, R=5), matches MCTS
- Rollouts most advantageous in low configs
- All variants with rollouts better than without
- Tree hybrids worse



Win percentage for config 10-14, R=5. Color bar: in how many unique games row was significantly better than column.

A	Vanilla
B	EA-Roll
C	EA-Shift
D	EA-Shift-Roll
E	EA-Tree
F	EA-Tree-Roll
G	EA-Tree-Shift
H	EA-Tree-Shift-Roll
I	MCTS

Results - MCTS Validation

- EA-Shift-Roll matches generality of MCTS, but higher win percentage

#	Algorithm	F1 Points	Avg. Wins
1	EA-Shift-Roll	430	42.05 (2.48)
2	MCTS	430	41.30 (1.76)

  - EA-Tree-Roll worse than MCTS
  - Still better in puzzle games

#	Algorithm	F1 Points	Avg. Wins
1	MCTS	451	41.30 (1.76)
2	EA-Tree-Roll	409	35.90 (2.27)

Conclusions

- Uni-variate bandit system does **not** work when individual = sequence of actions (epistasis)
- Stats tree more beneficial in small configs
- Shift buffer led to a significant increase in score gain (win rates in small configs)
- Shift buffer + rollout saw increase inversely proportional to individual length
- Best: EA-Shift-Roll (10-14, R=5) matches MCTS generality

Future Work

- Meta-heuristic: which variant is best for this problem?
- Improved bandit mutation might work better
- More games to better judge significance

Questions?