

Population Initialization Techniques for RHEA in GVGP

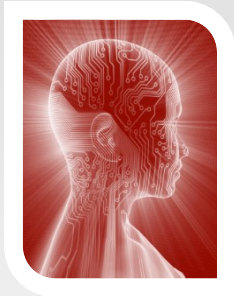
Raluca D. Gaina, Simon M. Lucas, Diego Perez-Liebana



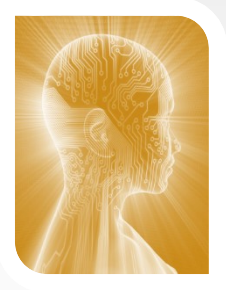
Introduction

- Rolling Horizon Evolutionary Algorithms (**RHEA**) show promise
 - in General Video Game Playing (**GVGP**)
 - as showcased in the General Video Game AI Competition (**GVGAI**).
- Better than **random initialization** for faster evolution?
 - No clear general analysis in previous literature

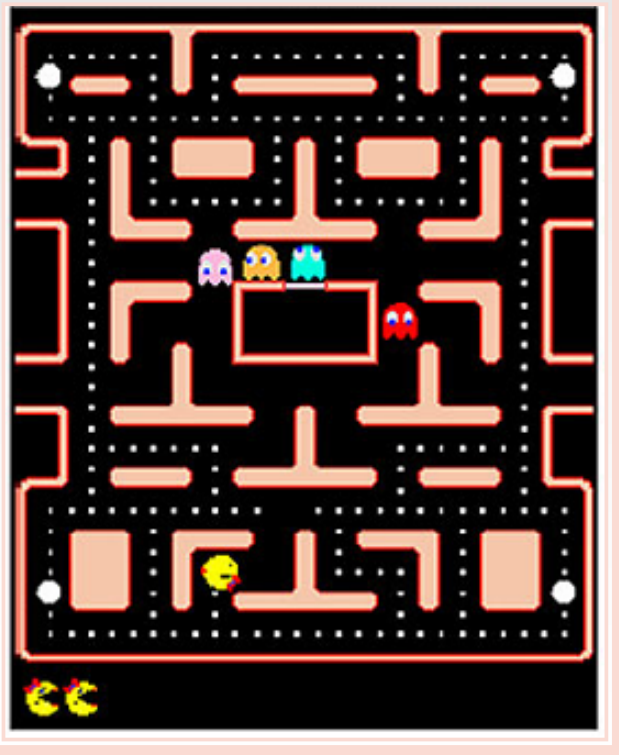
Game AI



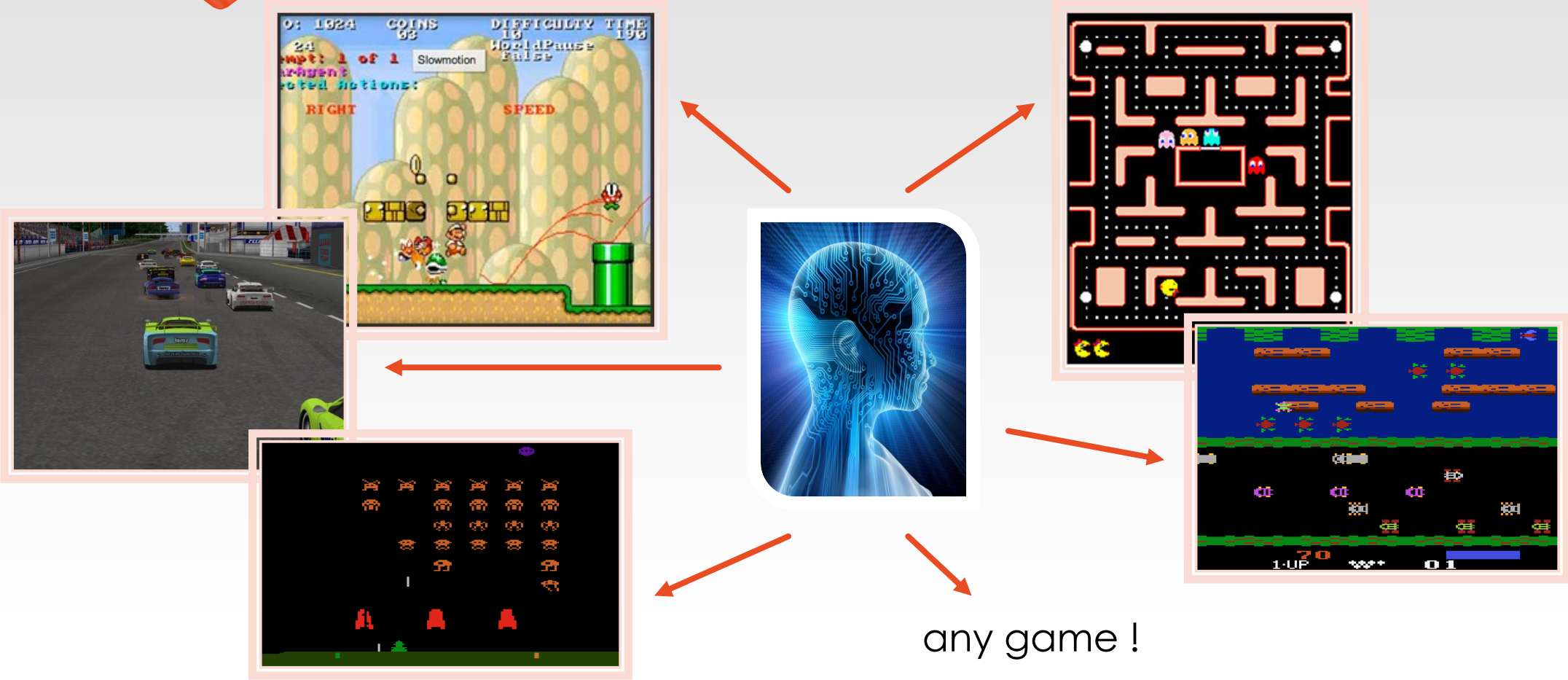
Super Mario AI



Ms. Pacman



General Video Game AI

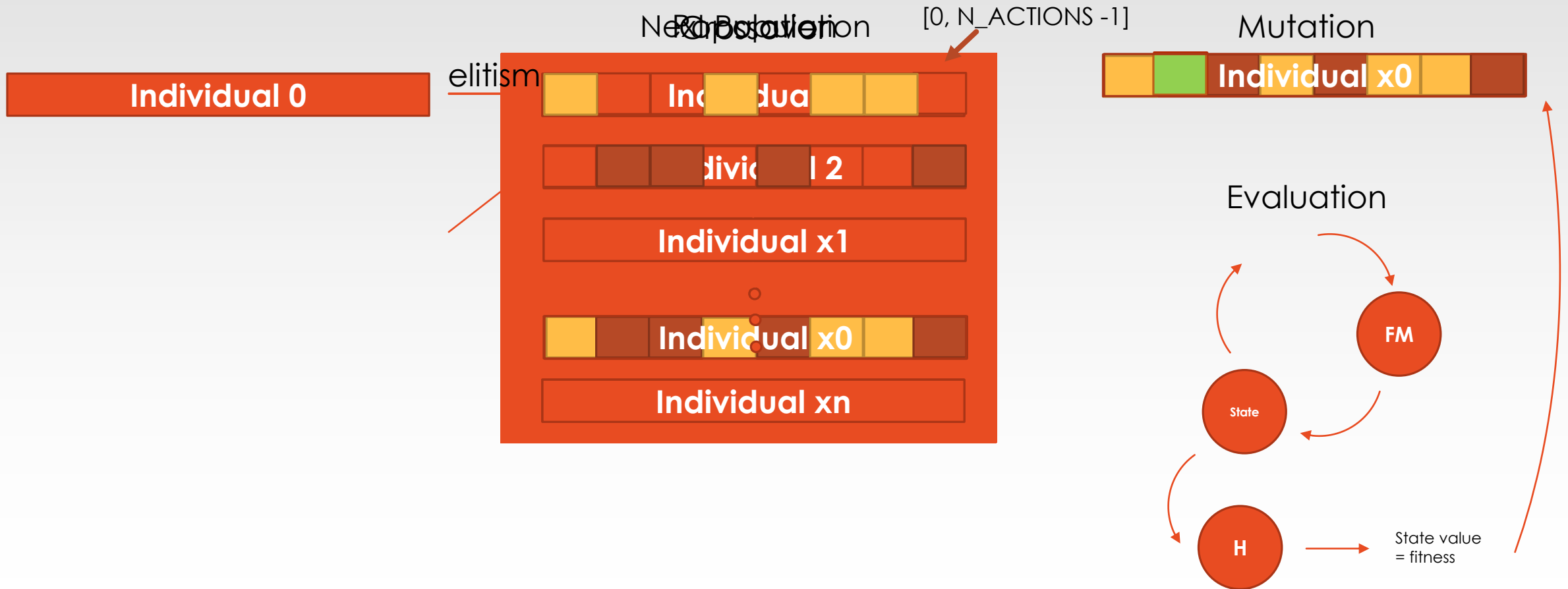


General Video Game AI Competition

- 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.

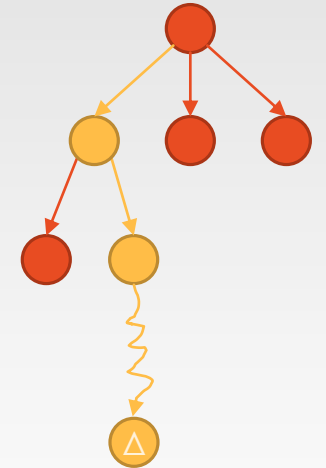
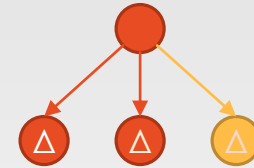
... high-level view of **current game state**
for agents; **real-time** decisions (40ms)

Rolling Horizon Evolution



Methodology

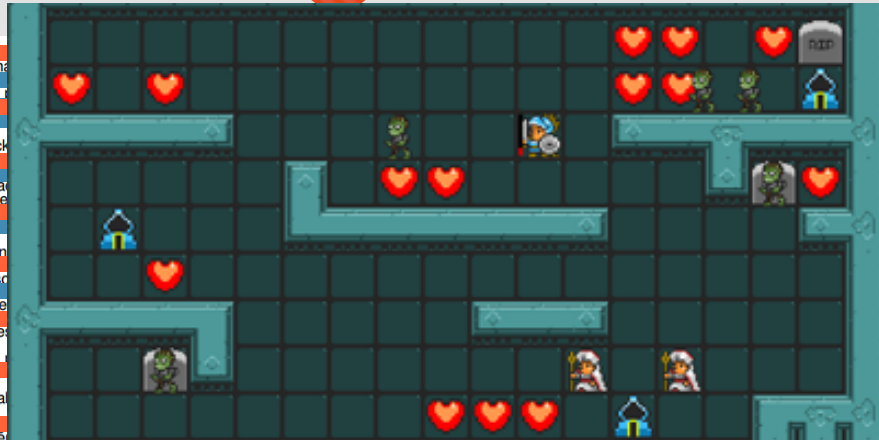
- Try two methods ...
 - One Step Look Ahead (1SLA)
 - Monte Carlo Tree Search (MCTS-S)
- ... on 20 GVGAI games ...
- ... with different core parameter configurations.



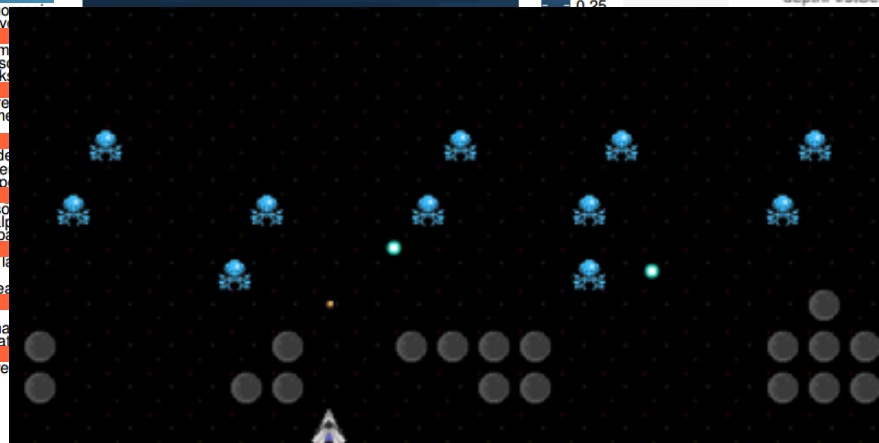
Experiment

- Population size P - Individual length $L = \{1-6, 2-8, 5-10, 10-14, 15-16, 20-20\}$
- All other parameters fixed to default values
- Budget: 900 Forward Model calls
 - L FM calls for 1SLA
 - Half budget for MCTS-S
 - MCTS-S rollout depth = L } => one individual, mutate it to form population
- Validation
 - Comparison with MCTS.

20 Games from GVGAI corpus

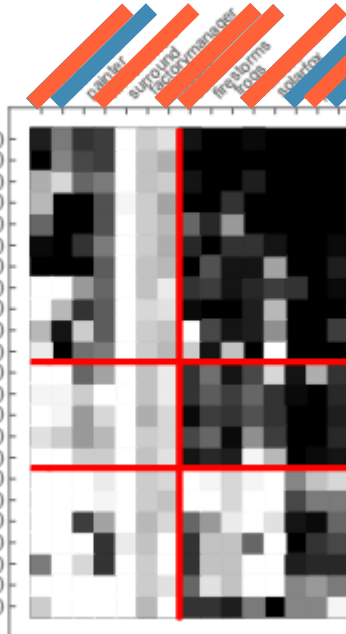


Survive Zombies



Aliens

Classifications by M...
 advanced set: 10 sta...
 difficulty and game



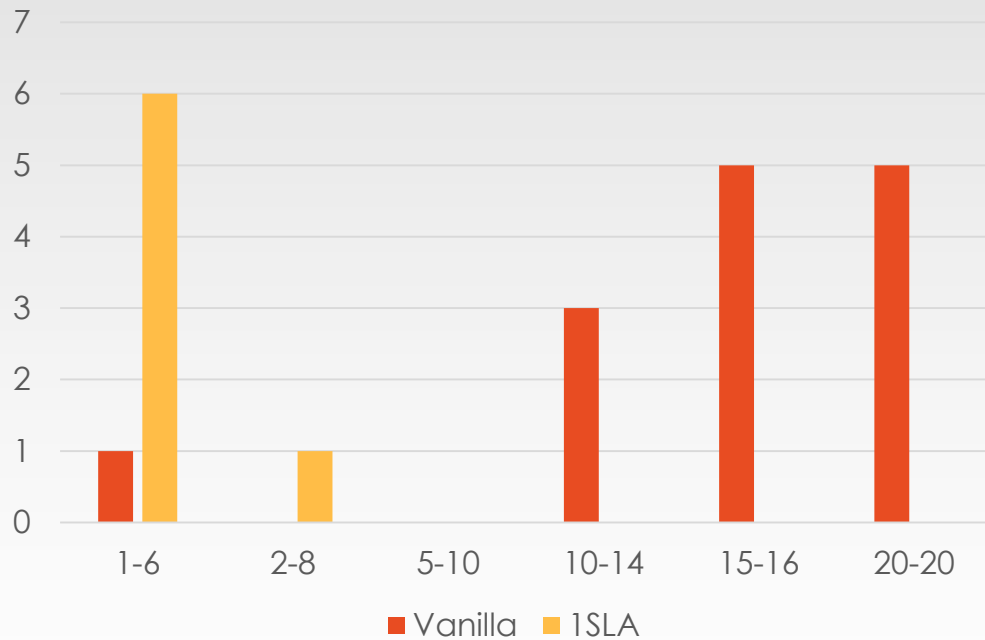
Sea Quest

Results Overview

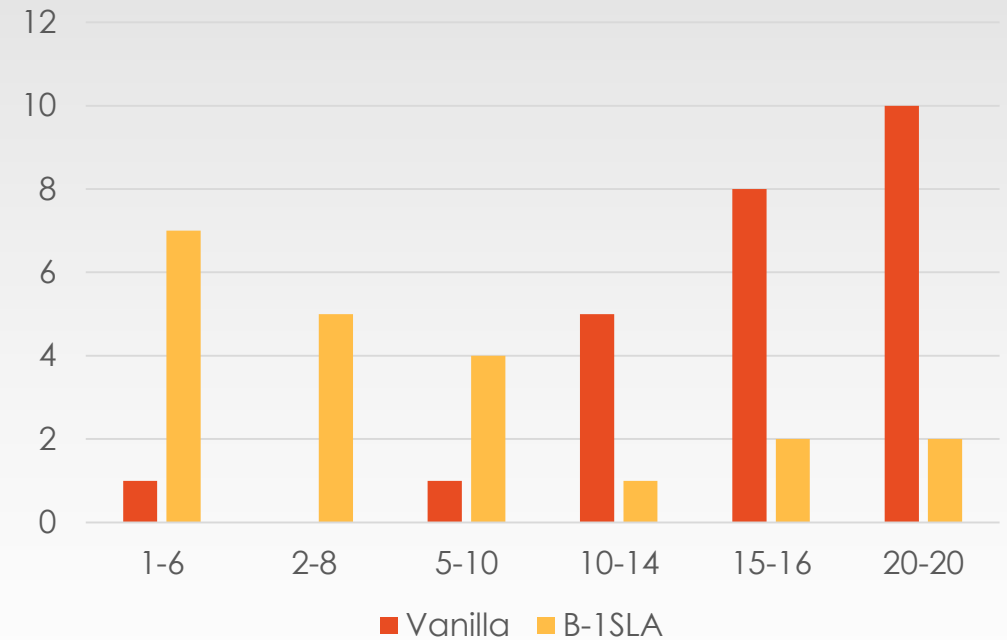
- Improvement much **bigger** when **small** pop size
- **MCTS seeding** significantly better
- **3** games in which MCTS seeding consistently **bad**: puzzles / long term reward
- Some games remain at **0%** win rate

- Game **Chopper**: 26% => **100%** win rate (1-6)
 - Big improvement in low config shows promise of RHEA with improved evolution

Results – Vanilla vs 1SLA

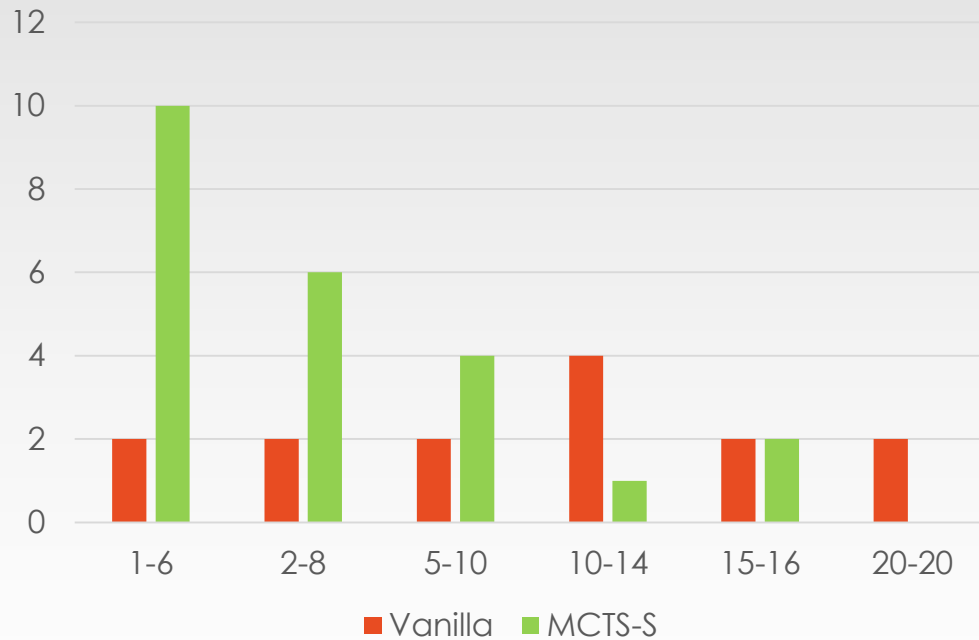


Win rate, no. games significantly better
Overall: 8, 6

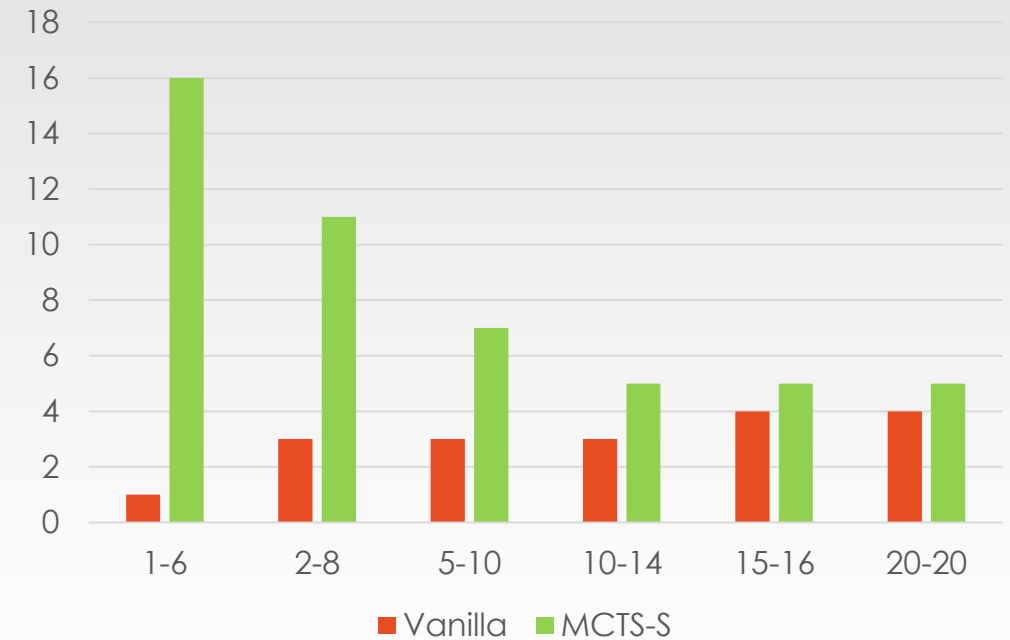


Score, no. games significantly better
Overall: 11, 8

Results – Vanilla vs MCTS-S

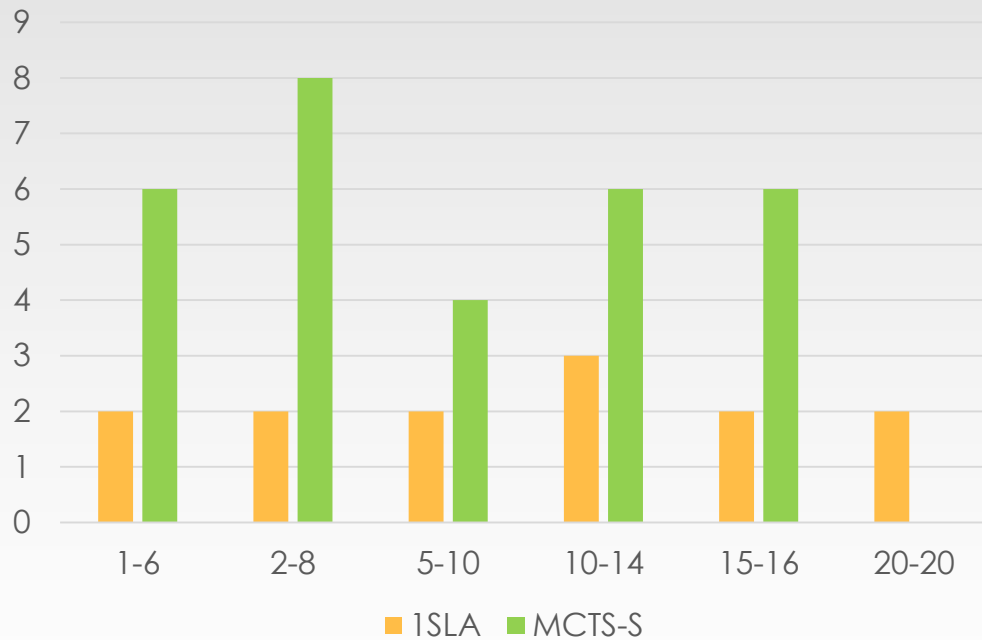


Win rate, no. games significantly better
Overall: 4, 12

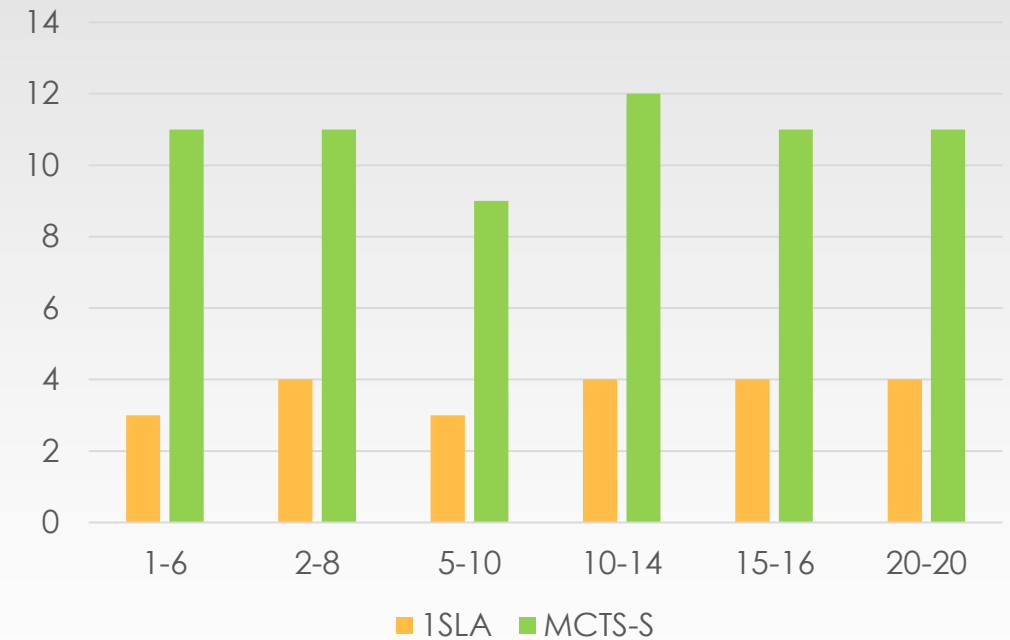


Score, no. games significantly better
Overall: 5, 16

Results – 1SLA vs MCTS-S

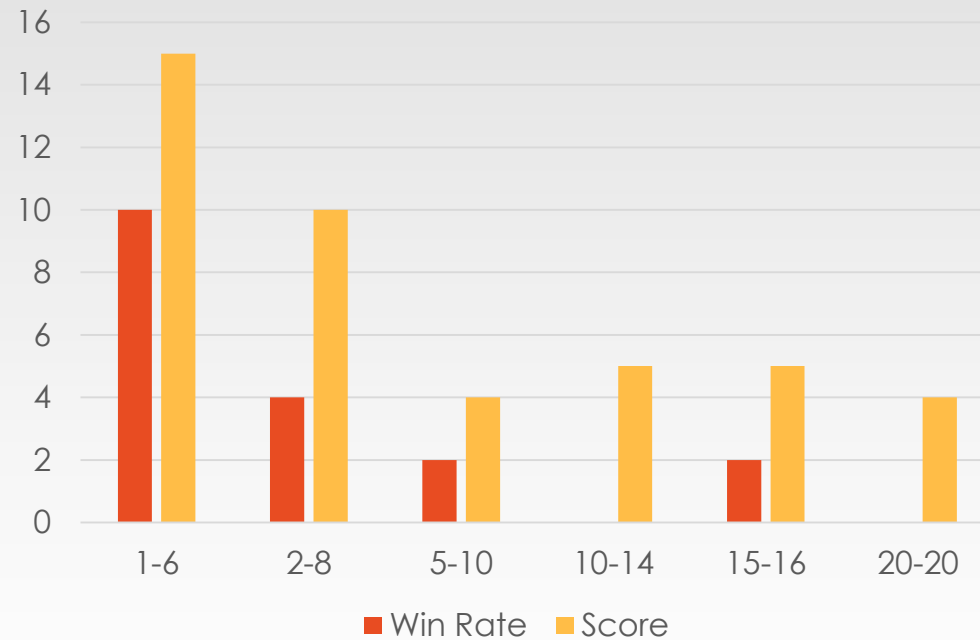


Win rate, no. games significantly better
Overall: 3, 10



Score, no. games significantly better
Overall: 5, 13

Results - MCTS Validation



No. games MCTS-S better than Vanilla, but not MCTS
Overall: 10, 15

Summary

- Analysis of One Step Look Ahead (**1SLA**) and Monte Carlo Tree Search (**MCTS-S**) seeding for vanilla Rolling Horizon Evolutionary Algorithm (**RHEA**)
- Multiple RHEA parameter configurations
- Win rate and score measured on **20** GVGAI games
- Overall and pairwise comparison
- Validation against **MCTS**

Conclusions

- Seeding improves performance if population size is **small**
- **MCTS seeding** significantly better (performance drops if rollout depth too large)
- MCTS seeding **worse** in puzzle games / longer lookaheads
- **Limited exploration**, search too narrow
- MCTS seeding **not** worse than simply MCTS

Future Work

- *Meta-heuristic*: which seeding method is best?
- Better *exploration* of search space & use of solution provided by seeding
- Better *evolution* paired with powerful seeding method
- *More* games to better judge significance

Questions?