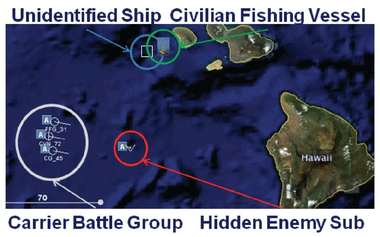
General Purpose Planning Algorithms In Partially Observable Stochastic Games

Bryan McKenney and Wheeler Ruml (Mentor)

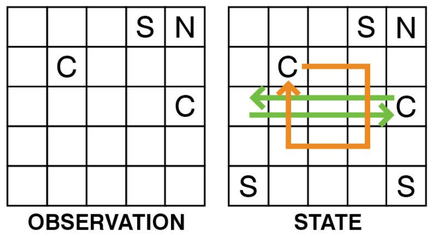
Department of Computer Science, University of New Hampshire, Durham NH 03824

Domains

— Navy Defense —



Tactical Action Officer Sandbox
(Molineaux, Klenk, and Aha 2010)



my Navy Defense domain

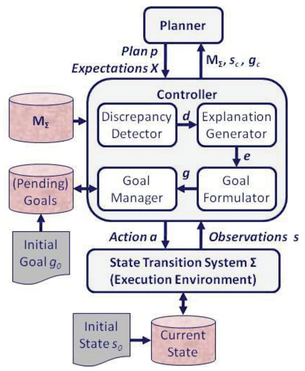
Challenges:

- stochastic (random tie-breaking)
- partially observable (invisible subs)
- open world (unknown # of subs*)
- unpredictable adversaries (inaccurate sub model*)

*my version only

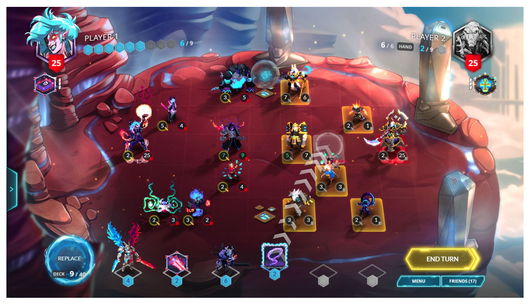
Previous solution:

- Goal-Driven Autonomy (GDA)
  - complex
  - ARTUE: hard-coded goal info



GDA framework
(Molineaux, Klenk, and Aha 2010)

— Duelyst II —



game screen featuring 9x5 game board

minion card

Challenges:

- stochastic (drawing cards, card abilities)
- partially observable (opponent hand + deck hidden)
- unpredictable adversary (human)
- large state and action spaces
- perform several actions in 90-sec turn

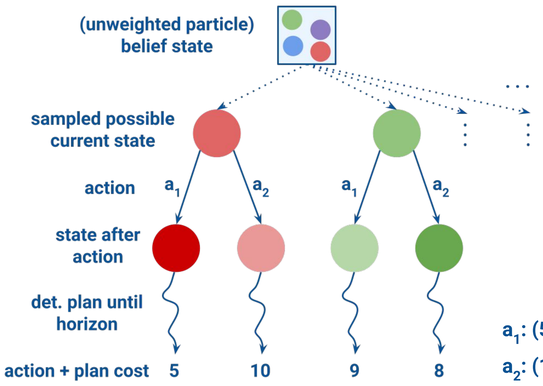
Previous solution:

- expert-rule-based player (Starter AI)
  - hard-coded to use specific cards effectively
  - weak against humans

Duelyst II website

Algorithms

— Hindsight Optimization —

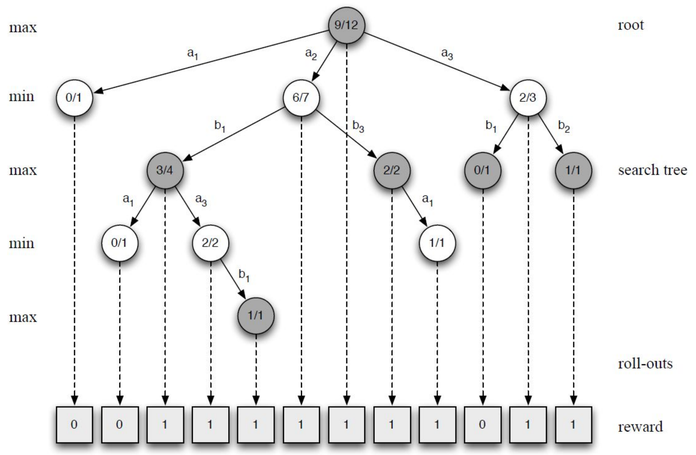


How it works:

1. sample possible worlds from belief
2. plan in each world
3. choose best action across worlds (minimize cost)

a_1: (5 + 9) / 2 = 7 ✅
a_2: (10 + 8) / 2 = 9

— Monte Carlo Tree Search —



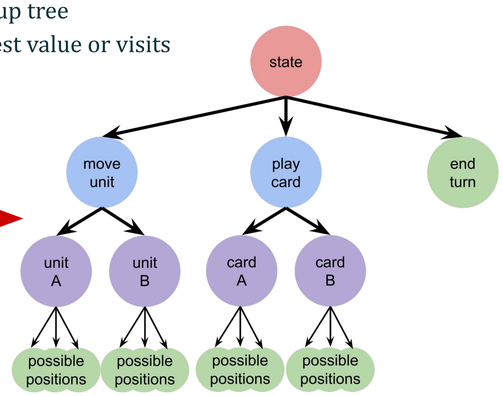
(Gelly et al. 2012)

How it works:

1. repeatedly for some time or number of simulations:
   a. search tree until new node found
   b. rollout until end of game from new node to get value
   c. backpropagate value up tree
2. choose action with highest value or visits

My version:

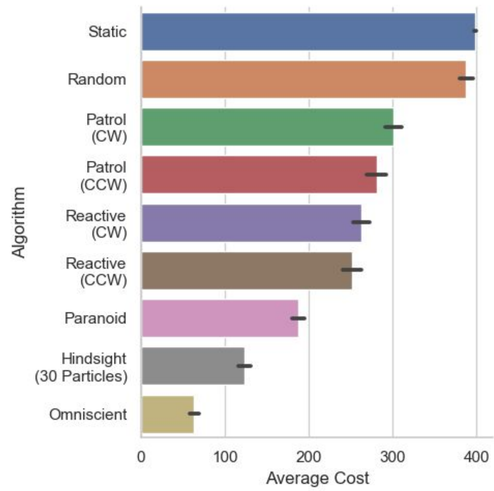
- rollout until end of *turn*
- simple static evaluator
- action trees



Static evaluator:

- win/loss
- board strength

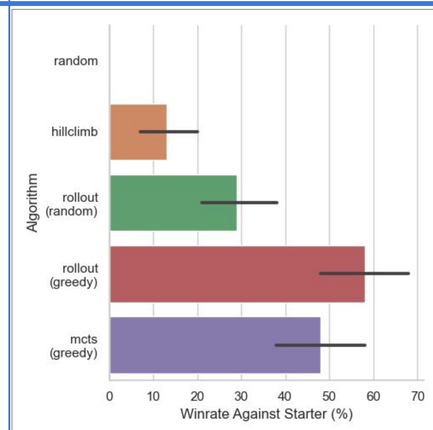
Results

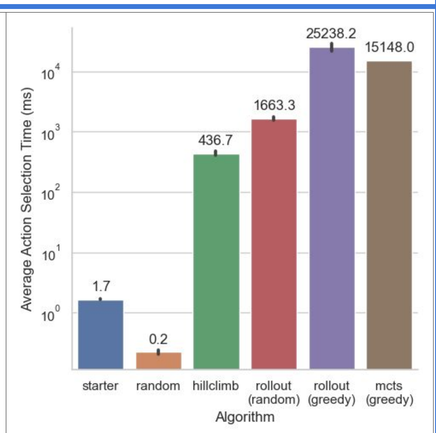


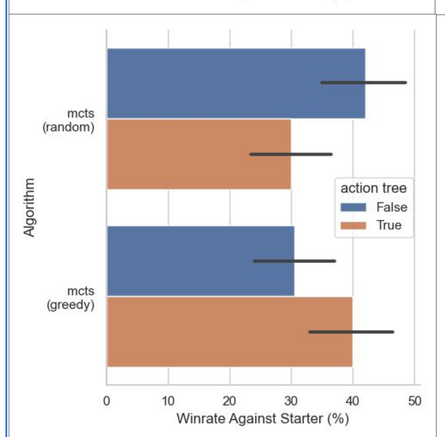
Hindsight optimization shows:

- strong performance
- emergent patrolling behavior
- implicit goal reasoning

Hindsight optimization is a simpler and more general alternative to GDA!







- MCTS has decent performance
- greedy rollout algorithm has strong performance
  - static evaluator is effective
- action trees help when rollouts are slow (greedy) and hurt when they are fast (random)

A simple general planner can defeat an expert-rule-based AI!

Acknowledgements

Many thanks to:

- Professor Wheeler Ruml, for being a great mentor.
- The UNH Hamel Center for Undergraduate Research, for research and AAAI presentation funding.
- The UNH CS Department, for poster and AAAI presentation funding.
- AAAI, for accepting my Bridge paper and giving me presentation funding.
- Dream Sloth Games, for resurrecting Duelyst and letting me work on it.
- Chuck Olson, creator of the Starter AI, for his invaluable AI vs AI experimentation framework.

References

- Gelly, S.; Kocsis, L.; Schoenauer, M.; Sebag, M.; Silver, D.; Szepesvári, C.; and Teytaud, O. 2012. The Grand Challenge of Computer Go: Monte Carlo Tree Search and Extensions. *Communications of the ACM*.
- Molineaux, M.; Klenk, M.; and Aha, D. 2010. Goal-Driven Autonomy in a Navy Strategy Simulation. In *Proceedings of AAAI-10*.
- Paredes, A.; and Ruml, W. 2017. Goal Reasoning as Multi-level Planning. In *Proceedings of the ICAPS-17 IntEx Workshop*.