SFI The Graduate Workshop in Computational Social Science 2024

How Are Collaborative Tasks Conducted ?
Insights from a Simulated Dual-Agent Model

Xiaojie Niu

2024.07.11

Abstract

This research investigates three archetypes of collective intelligence within educational contexts, with a focus on modeling the **Stigmergy** paradigm. A dual-agent cooperative cleaning task is modeled to examine the effects of task difficulty, adaptive competition-cooperation method, localized search algorithms, and simulated annealing techniques on task performance using Genetic Algorithm (GA).

Collective Intelligence in Education

Collective Intelligence (CI): combined capacity of a group to solve shared problems.

Implications for Education

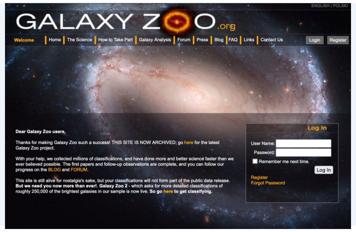
- CI's potential to address complex global issues through cooperation
- Its role in promoting collaborative work across various sectors and educational levels
- The shift from individual-focused to collaborative learning and assessment methods

Collective Intelligence in Education

Three difference Types

Swarm

- Behavioral Convergence and Emergence: Flock of Birds Swarm
- Weak targeting, emphasizing the act of doing
- Large-scale: human-centered computing: the Galaxy Zoo project



Stigmergy

- Optimization based on the original product of the environment
- Generally goal-oriented, emphasizing collective improvement
- Ant foraging pheromone
- Wikipedia; Social annotation, online forum participation

Collaborative problem solving

- Integration of multiple skills and communication processes
- Highly targeted, directed toward problem solving, small scale
- Research Team Operations
- Computer-supported collaborative learning

Stigmergy: Set a Simulated Dual-Agent Model

Why Choose Stigmergy

- **Most Common Form**: swarms tend to be large and loose, Collaborative problem solving requires close cooperation;
- **Emphasis on Task Optimization**: different agents have influencing roles on each other, goal is to imporve task performance, can use genetic algorithms to evolve.

What is the Question

- What factors affect the speed and quality of finding the Stigmergy-type collaborative tasks?
- In the Modeling Context: **how to find a better task solution faster in a double-agent collaborative model based on genetic algorithm?**

Model Settings

Two cleaning robots collaborate to clean a 10x10 grid space.
Each robot is limited to a maximum of 50 moves.

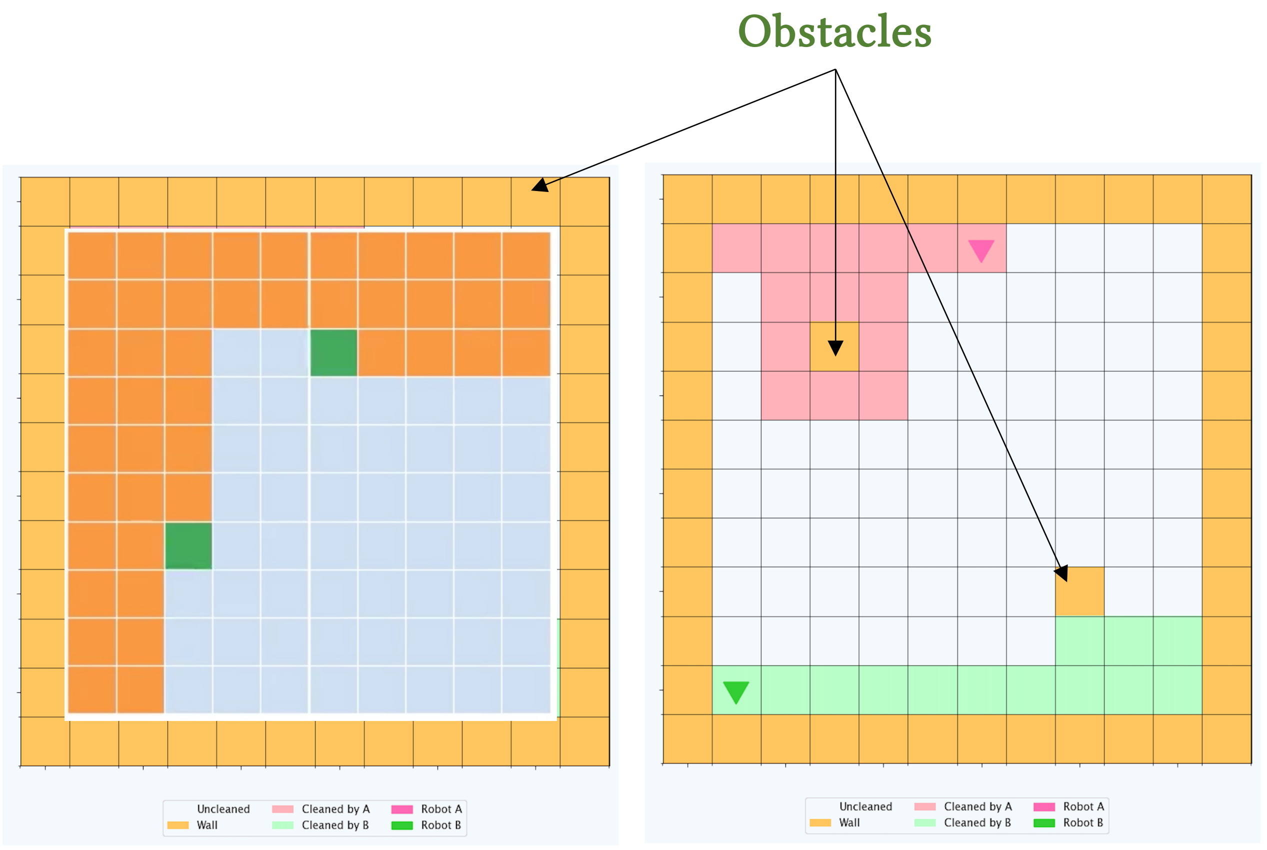
Task Scenario

Environmental Constraints

- Perimeter: Bounded by impassable walls.
- Obstacle Conditions:
  a) No internal obstacles
  b) Internal obstacles present at (3,3) and (8,8)

Robot Dynamics

- Movement: Up/Down/Right/Left.one move per robot per round.
- Cleaning Mechanism: A unit is considered cleaned upon first visit.
- Unit Visitation: Multiple visits allowed, but cleaning credit assigned to the first visitor.



No Internal obstacles Internal obstacles

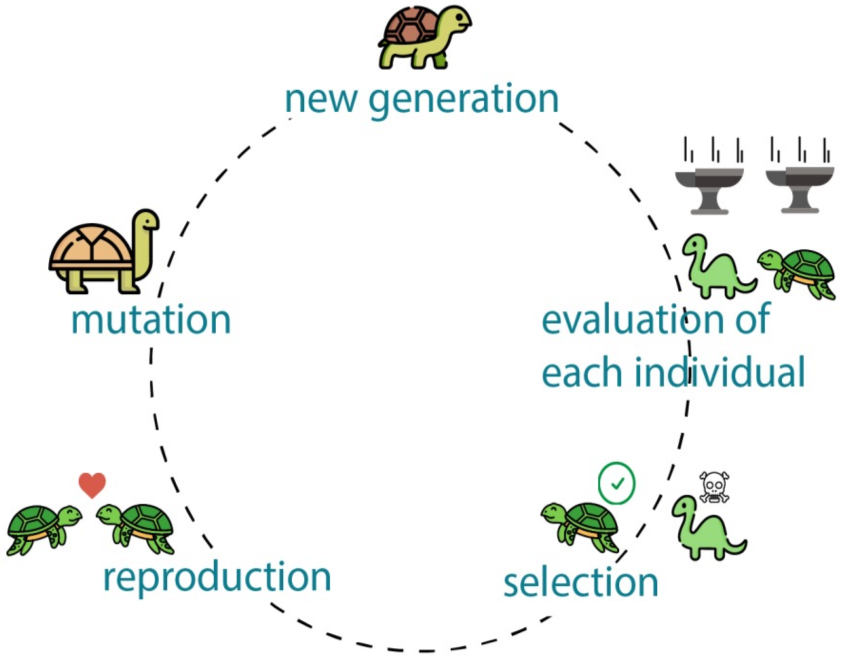
Genetic Algorithm

Population Configuration

- Population size: 500 individuals
- Chromosome length: 50 steps, representing a cleaning strategy
- Initial population: Randomly generated, with each gene encoding an action (0-4 for up/down/left/right/hold still)

Selection Mechanism

- Tournament selection
- Tournament size: 5 individuals
- The fittest individual from each tournament is selected for reproduction



Genetic Algorithm

Crossover Operator

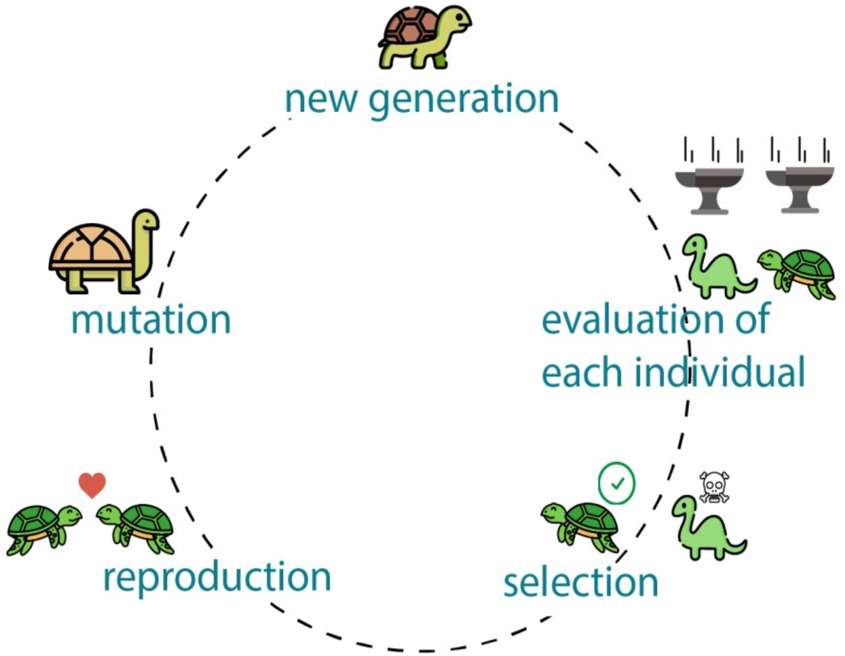
- Single-point crossover
- Crossover point: Randomly selected
- Genetic material beyond the crossover point is exchanged between parents

Mutation Operator

- Mutation rate: 0.05
- Each gene has a 5% probability of being randomly altered to a new value (0-4)

Elitism

- Elite size: 10 individuals
- The top 10 fittest individuals are preserved and directly transferred to the next generation



Genetic Algorithm

Fitness Evaluation

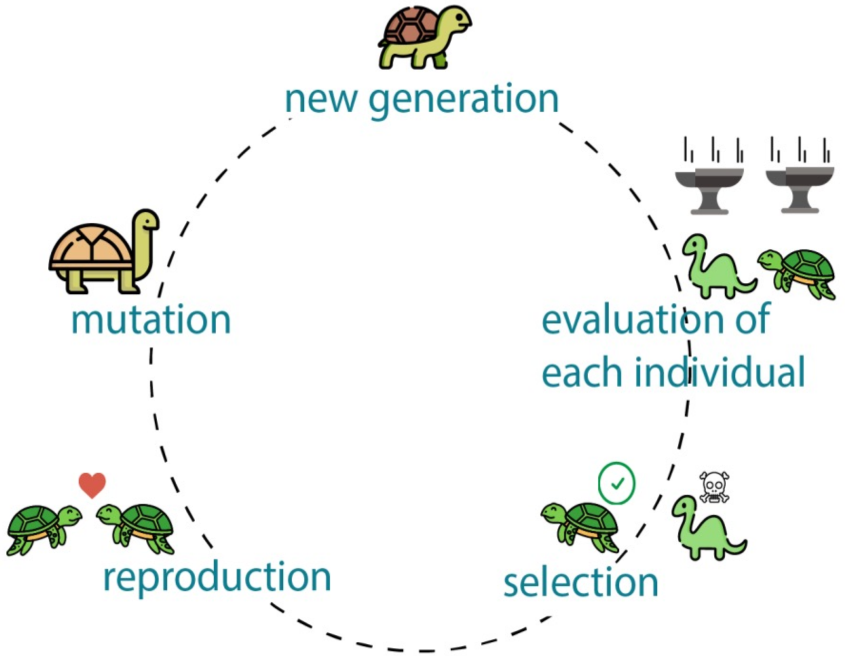
- Base score: Total number of cleaned units
- Penalty factors: Wall hit * 0.5 /Repeated cleaning attempts * 0.3
- Fitness = Base score - Penalties

Termination Criteria

- Maximum generations: 2000
- Or 200 consecutive generations without improvement

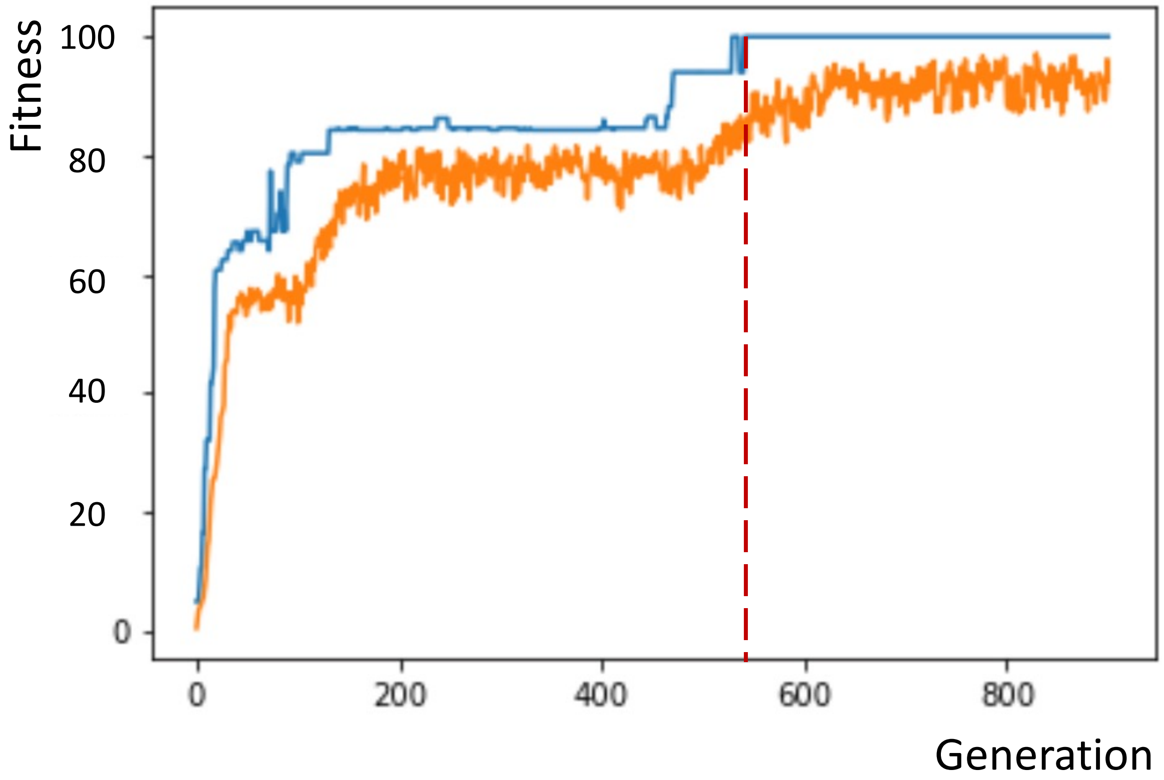
Problem-Specific Parameters

- Dual-robot cleaning scenario (Robot A and Robot B)
- Grid dimensions: 12x12, with a 10x10 interior cleanable area
- No internal obstacles / Two internal obstacles present



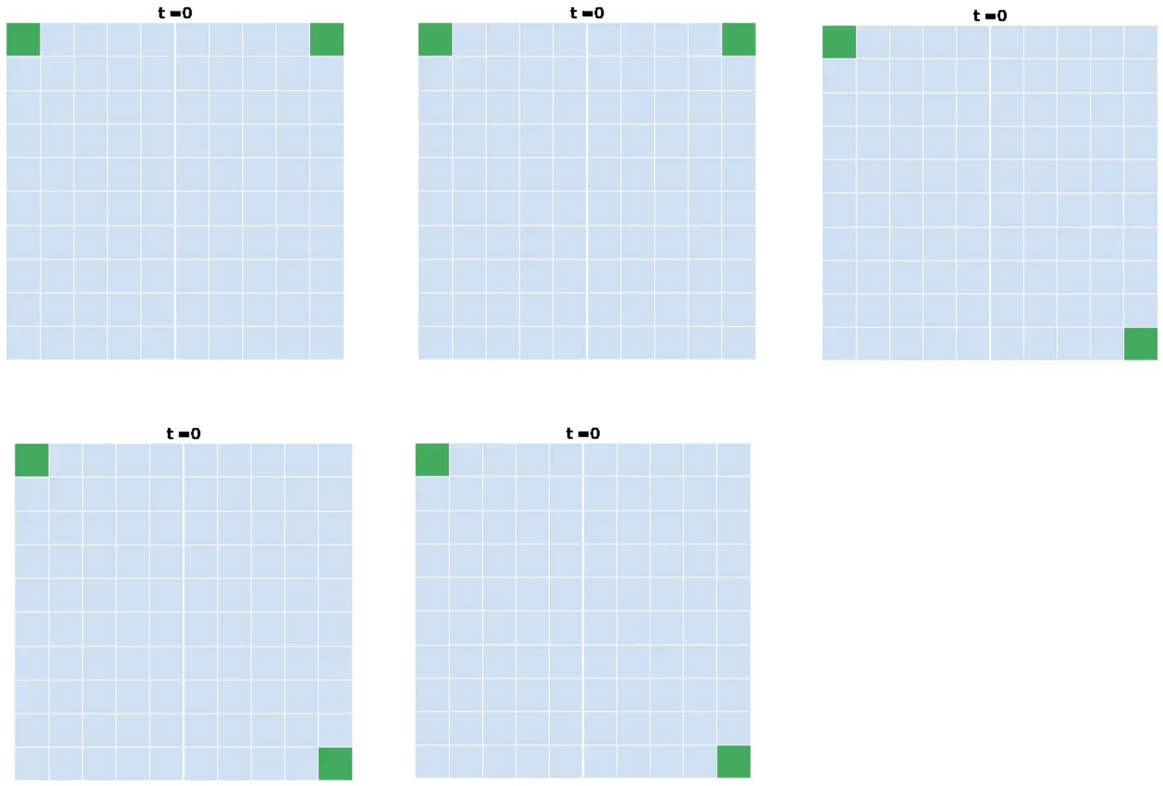
Result: Start From No Internal obstacles

Dual-agent cooperative coevolution



Fitness Evolution Plot

Converges around the 500th generation
Identifying five distinct strategies



Consider how to discover strategies more rapidly?

Consider how to discover strategies more rapidly?

Introduce **a cooperation mechanism based on competition**

Phase 1: Individual Evolution

Agents A and B evolve separately.

Fitness is determined by individual performance.

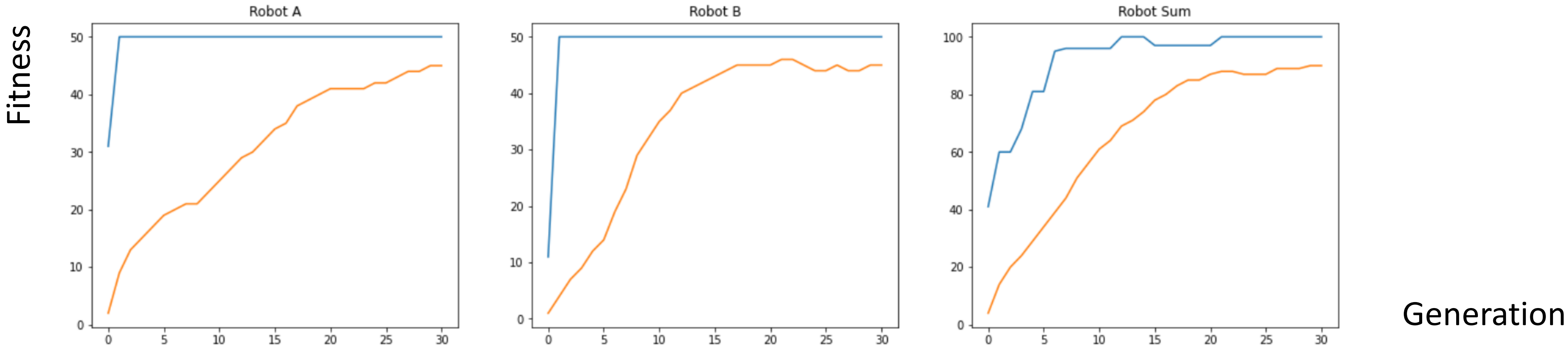
Phase 2: Combined Evolution

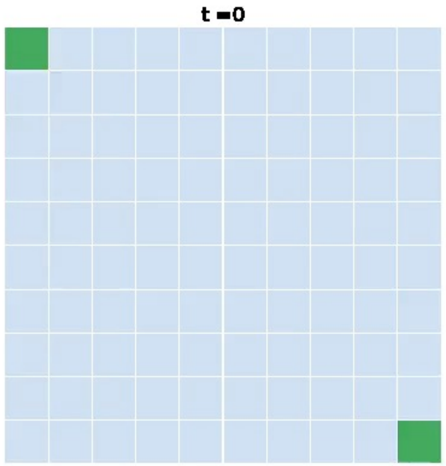
High-performing subpopulations of A and B are selected and paired.

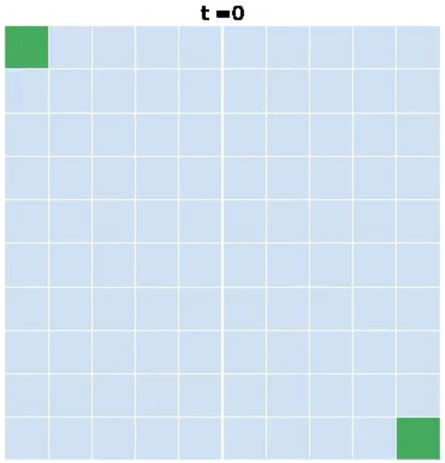
Fitness is assessed based on collaborative task performance.

- **Selection**: Elite subsets $S\_A' \subset S\_A$ and $S\_B' \subset S\_B$ are selected based on individual fitness scores.
- **Genetic Operations**: Crossover and mutation are applied within S_A' and S_B' respectively.
- **Fitness Evaluation**: $F(s\_a, s\_b) = g(s\_a, s\_b)$, where $s\_a \in S\_A'$, $s\_b \in S\_B'$, and g evaluates the cooperative task performance.

Result With A cooperation mechanism based on competition







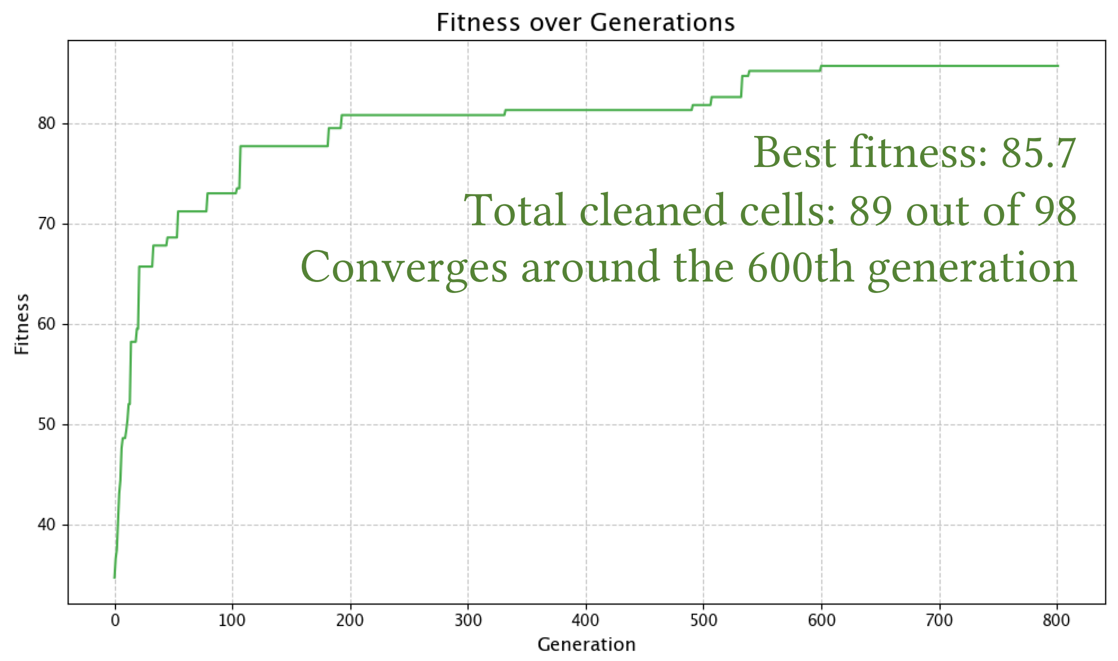
Task-completion strategies were discovered **rapidly** (within 30 generations), but only two distinct strategies emerged.

Competition accelerated task completion while reducing strategic diversity.

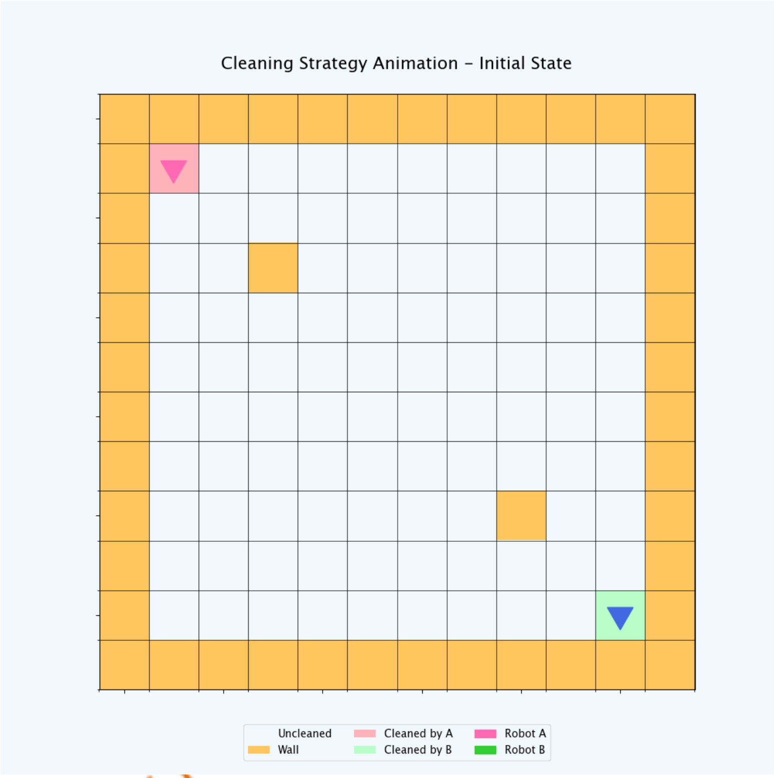
What about the task with internal obstacles?

* For model with internal obstacles, no strategy was found that could cover all uncleaned units within 50 steps. Therefore, strategies that cleaned more units within 50 steps were considered better.

Result of Only Cooperation



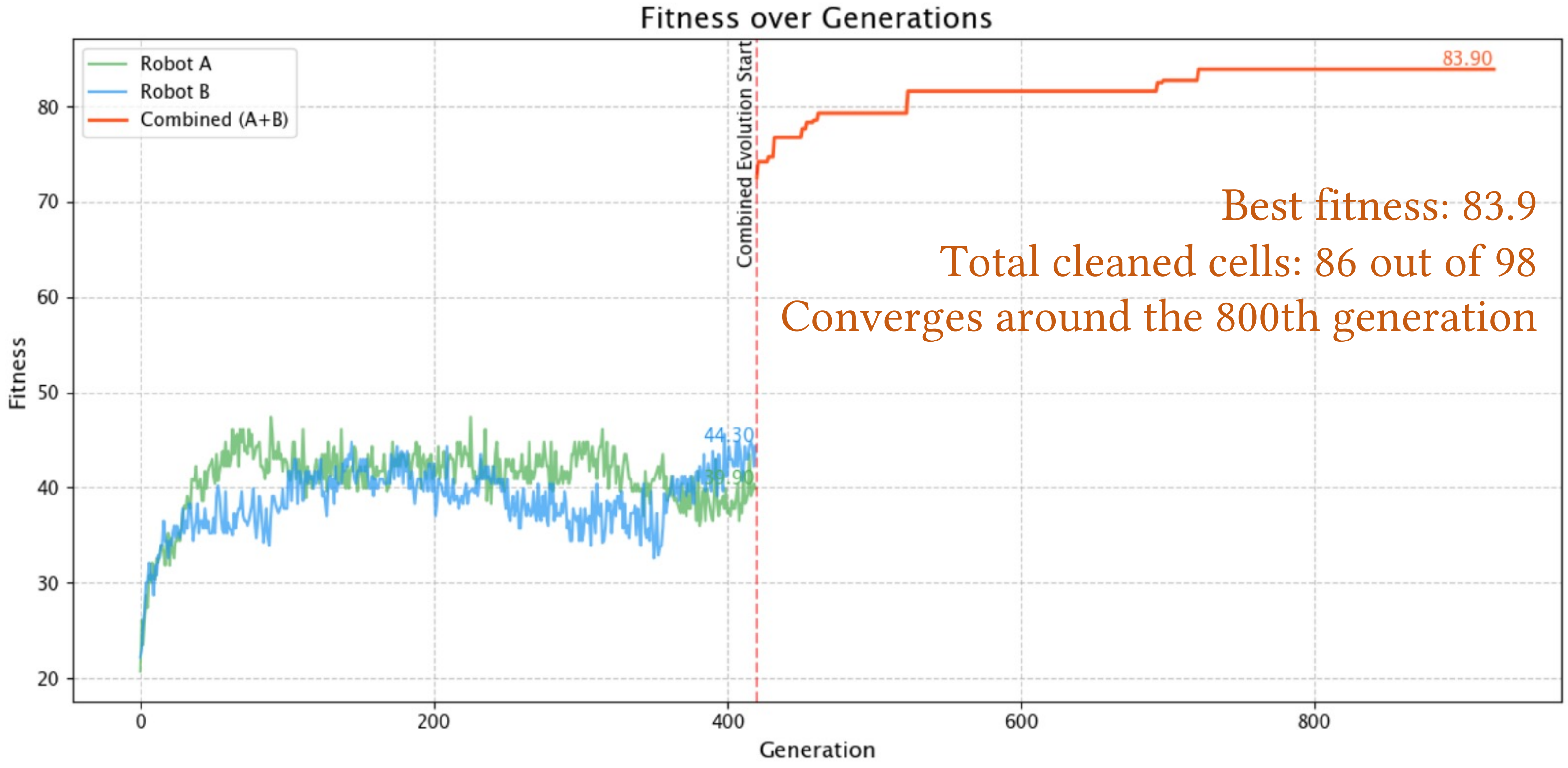
Best fitness: 85.7
Total cleaned cells: 89 out of 98
Converges around the 600th generation



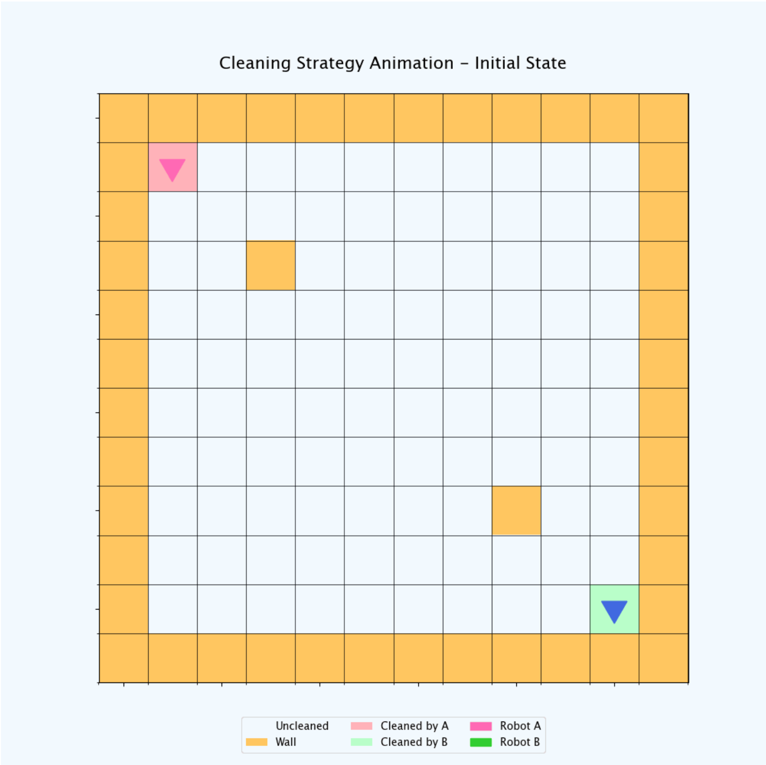
Then try the cooperation mechanism based on competition

Result of Cooperation based Competition

Longer generation and no better task performance ☹



Best fitness: 83.9
Total cleaned cells: 86 out of 98
Converges around the 800th generation



Any other ways to improve this model ?

Heuristic Strategy in move_robot

- Primary attempt: Execute the specified action (0-3 representing up, down, left, right).
- Secondary heuristic: If primary action is invalid (go into wall or repeat movement), employ a directional priority system:

  For upward movement (0): [3, 1, 2, 0]

  For downward movement (1): [2, 0, 3, 1]

  For leftward movement (2): [0, 3, 1, 2]

  For rightward movement (3): [1, 2, 0, 3]
- Tertiary fallback: If all directions are invalid, select a previously visited non-wall cell.

Any other ways to improve this model?

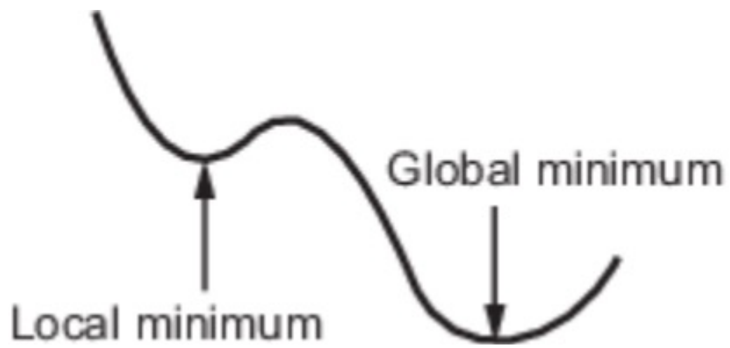
Memory Mechanism

- Data structure: Two sets, memory_A and memory_B, for robots A and B respectively.
- Functionality: Track cleaned cell coordinates (x, y).
- Implementation: In evaluate_fitness function, update sets as robots clean new cells.
- Application: Used in move_robot to avoid revisiting cleaned cells, enhancing exploration efficiency.
- Performance impact: Reduces redundant cleaning, optimizing overall coverage.

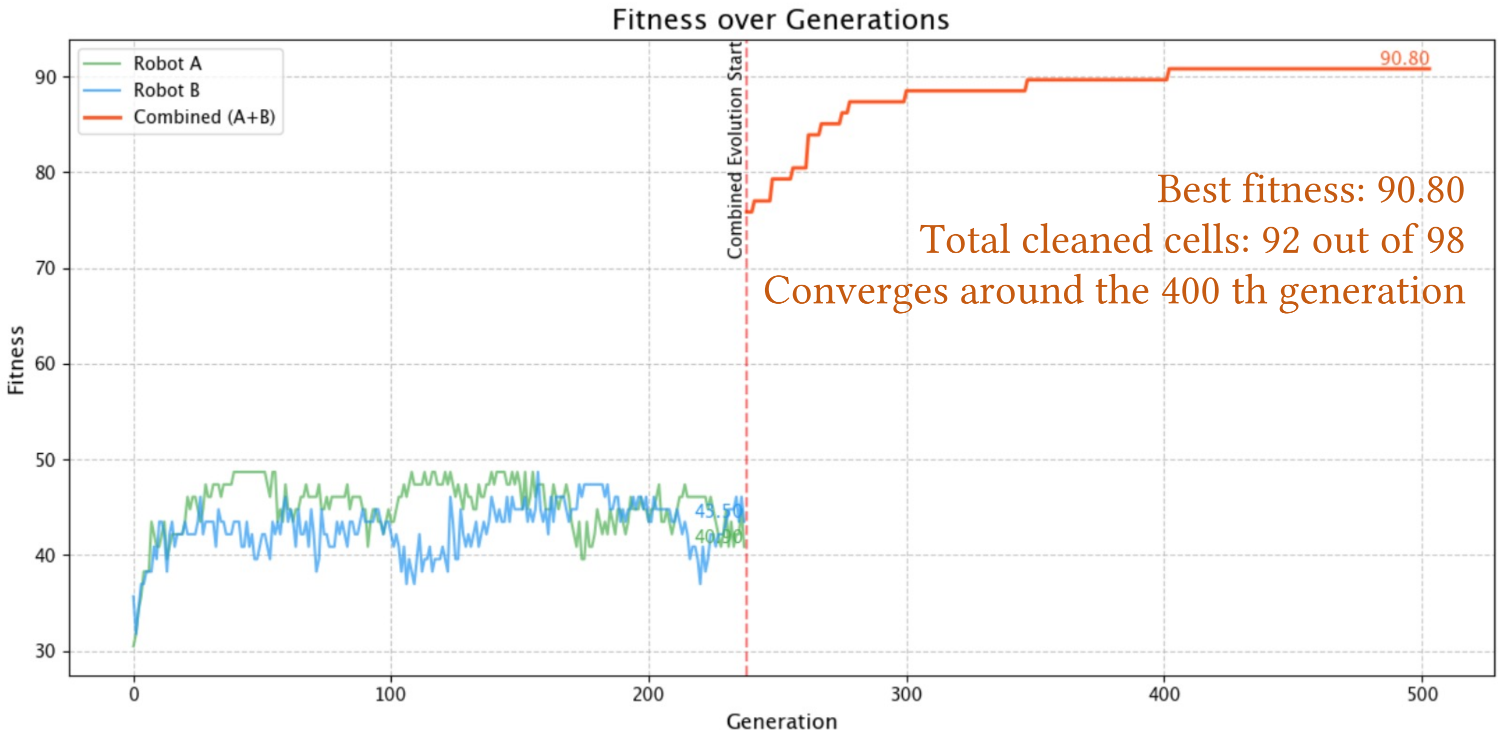
Any other ways to improve this model?

Simulated Annealing Integration

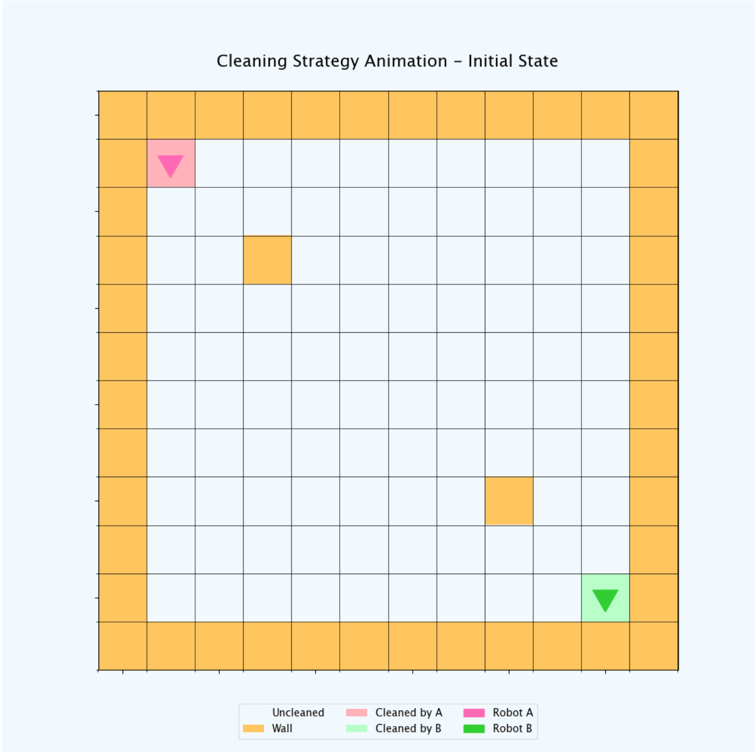
- Temperature function: $T(g) = 1 / (g + 1)$, where g is the current generation.
- Acceptance probability: $P(\Delta E, T) = \exp(-1 / T)$
- Implementation: Applied post-genetic operations in each generation.
- Mechanism: Probabilistically accepts inferior solutions based on current temperature.
- Purpose: **Facilitates escape from local optima, enhancing global search capabilities**.



Then what is the result?



Best fitness: 90.80
Total cleaned cells: 92 out of 98
Converges around the 400 th generation



Findings

Model characteristics for simple tasks

- Simple tasks are defined as those without internal obstacles.
- Competition-based cooperation significantly enhances the speed of finding optimal strategies.

Impact of increased task difficulty

- As tasks become more complex, finding optimal strategies becomes more challenging.
- The beneficial effect of competition becomes less apparent in these harder scenarios.

Evolution of individual optimal strategies

- Entities A and B each evolve their own optimal strategies independently.
- Once they reach this stage, further cooperation between A and B:
- a) Requires more time to evolve
- b) Does not lead to significant performance improvements

Additional strategies for improvement

- Implementing specific walking strategies can enhance overall strategy quality.
- Employing methods to avoid local optima can:

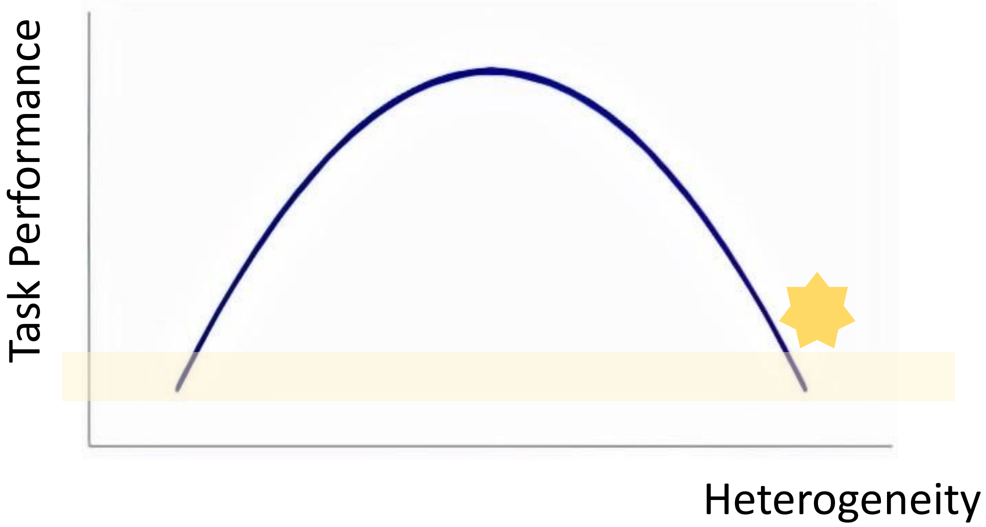
  a) Improve the quality of strategies

  b) Increase the efficiency of the strategy search process

Discussion

- Agent A and B each evolve their own optimal strategies independently.
- Cooperation between A and B actually requires more time to evolve and does not bring about significant performance improvements.

In group tasks, is it beneficial or detrimental to cooperation when members have their own specialized areas?



Too little differences in cognitive styles lead to increased difficulty in collaboration; too little difference is not conducive to a comprehensive understanding of the task? (Aggarwal & Woolley, 2013)

More agents, more task types, and more agent style....

Thank You !

Xiaojie Niu

2024.07.11