

AI: OPTIMIZING TO SURVIVE

R e n H e

OVERVIEW

- Motivation, background
- Cases
- An algorithm
- A potential application

MOTIVATIONS

- Survival is an important topic for living things
- What do we need to build a lasting AI?

POTENTIAL PROBLEMS

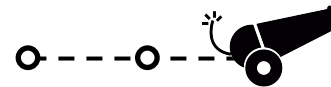
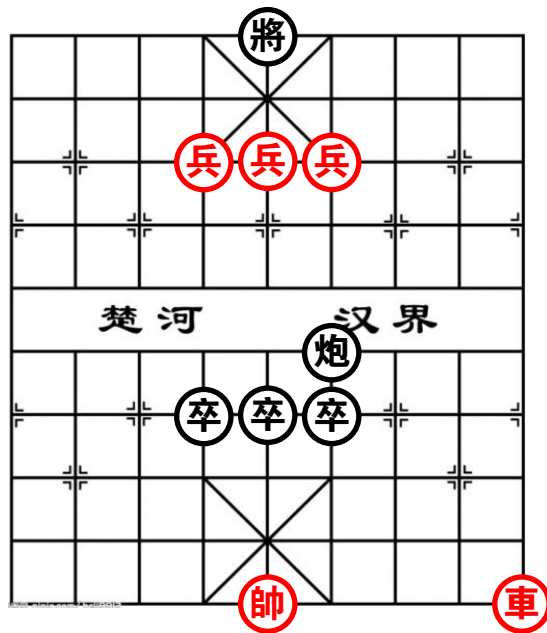
- The objective is non-ending
- Physical restrictions

CASE 1

- No unavoidable danger:
 - Individual threats are evadable
 - Possibility to permanently remove threats:

CASE 1

- No unavoidable danger:
 - Individual threats are evadable
 - Possibility to permanently remove threats:

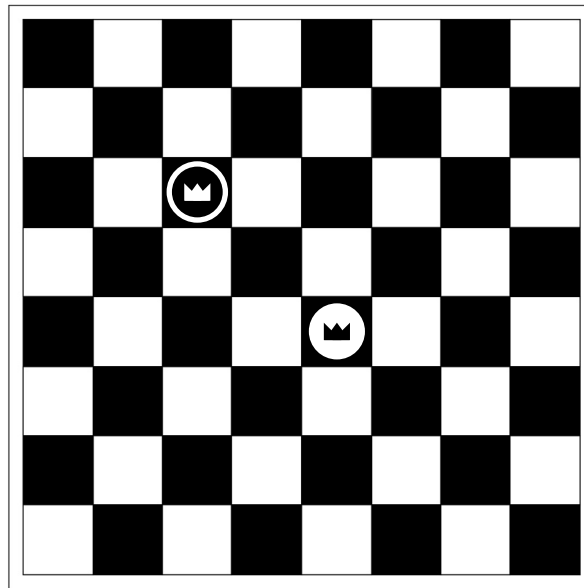


CASE 1

- No unavoidable danger:
 - Individual threats are evadable
 - Possibility to permanently remove threats:
- Can be solved with basic logics:
 - If spotted signal of danger, dodge
 - Remove the threat if possible

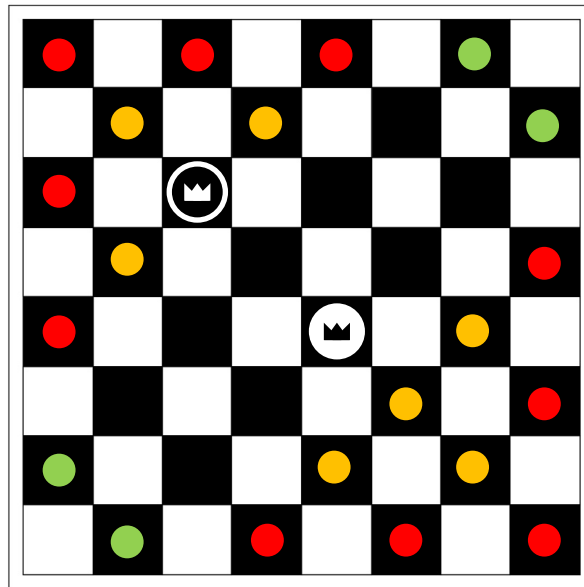
CASE 2

- Chance of dead ends
- Unable or no guaranteed way to permanently remove threat



CASE 2

- Chance of dead ends
- Unable or no guaranteed way to permanently remove threat



CASE 2

- Chance of dead ends
- Unable or no guaranteed way to permanently remove threat
- We need to consider:
 - Avoid dead ends
 - Find a safe shelter or infinite loops
 - Remove the threat if possible

CASE 2

- Algorithmically, in this case
 - Iterate
 - Find neighboring safe position
 - Repeat
 - If back to a visited point, there is a loop

MORE REALISTICALLY

- Continuous time
- Continuous harm that is possible to recover
- We do optimization

SCENARIO

- Example: a student working on a project
 - A student has a list of project throughout time
 - Resting with unfinished project results in guilt
 - A past due project causes immense guilt
 - Too much guilt accumulated causes mental failure
 - Finishing a project removes any guilt caused by it
 - Work efficiency decreases without rests
 - Work efficiency restores when there is no work
 - Each individual project is manageable in given ddl
 - Each rest has a minimum length

CASE 3

- The list has certain pattern
- Assume the list is finite
 - Solvable with an optimization model
- Assume the list is infinite
 - Manually input cycle information if possible
 - Or find pattern with the machine
 - Past data analyzing + optimization for each cycle

CASE 4

- Schedule cannot be known from advance
 - Only able to analyze history of project solving
 - Work efficiently until pattern is found
- Obviously, dangerous

BUT

- We might be demanding too much
- Everything has some pattern
- In real life, every currently existing creature has past information of some sort

CASE 4 (Modified)

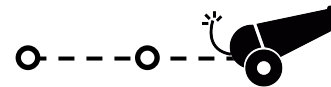
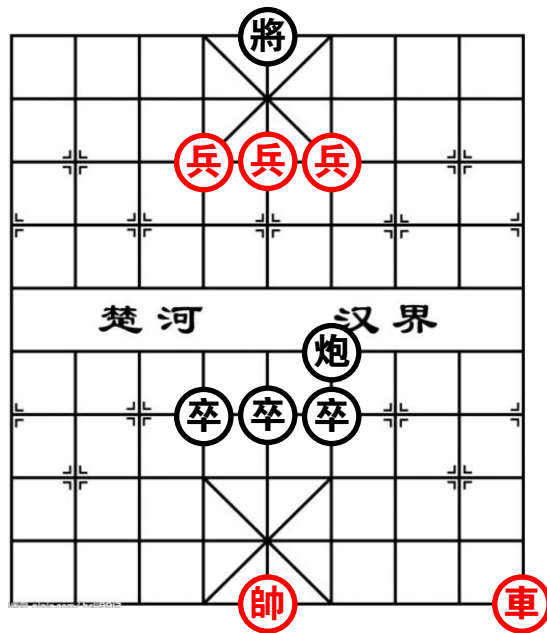
- Cycle pattern known
 - Predict future trend
 - Analyze past experiences
 - Based on those, optimize immediate strategy
- May face difficulty, but unavoidable

A GENERAL ALGORITHM

- Set up a core evaluation parameter, we can call it "hedon"
- Every parameter has a hedon function
 - Manually input a prototype
 - Self-modifies overtime with learning algorithms
- Maximize immediate hedon level at every instant

A GENERAL ALGORITHM

- Example: Discrete



A GENERAL ALGORITHM

- Examples: Continuous
 - Functions
 - Learning algorithms
 - “Events”: points to modify d^2y/dx_i^2
 - “Cycle parameters” and “emergency parameters”
 - Apply learning algorithms

REAL LIFE

- Consider a wild animal, e.g. an antelope
 - Finds food
 - Avoids predators
 - Faces bad weather conditions
 - Finds a good place to live
 - If possible, finds better place to live, migrates

REAL LIFE

- To build a robot that can survive in the wild?
 - Physical requirements
 - “Instincts” that encourage good behaviors
 - Would be more helpful if there is an “instructor”
 - Apply the algorithm

REAL LIFE

- Potential “instincts”:
 - Go for the sun
 - Avoid dangerous animals
 - Avoid extreme temperatures
 - Avoid water
 - Exploring and reproducing

IDEAS

- Does it mean that we can create some kind of life?
- Efficiency comparisons?
- Is it ethical, if possible to create a specie that assists human beings?
- Consequences of that?

CONCLUSION AND,
QUESTIONS?

THANK YOU.