FFI Forsvarets forskningsinstitutt
Norwegian Defence Research Establishment

Optimizing flight paths through anti-aircraft gun fire with machine learning

Esben Lund

NMSG-171, 24-25 October 2019

Outline

- Challenge
- Anti-aircraft gun simulation
- Machine learning and reinforcement learning
- Q-learning
- State and action layout
- Rewards
- Results
- Summary

Challenge

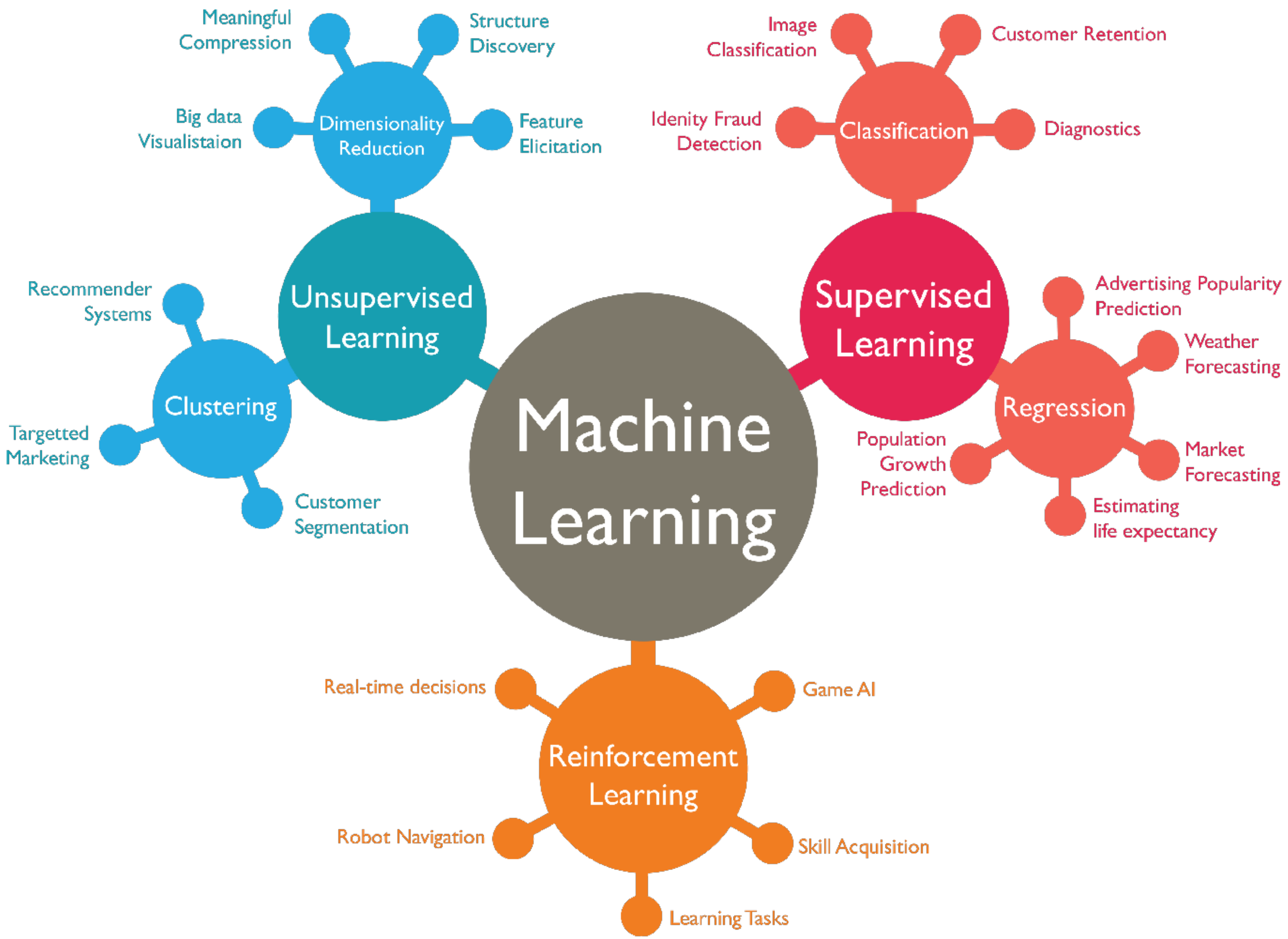
- What is the optimal flight path for avoiding anti-aircraft gun fire?
- Can machine learning find it?

Anti-Aircraft Gun Simulation

- Simulated by an in-house developed tool
- Typical high rate of fire 30 mm close-in weapons system (CIWS), with an effective range of 2.5 km
- Mechanical properties and limitations are considered
- Bullet ballistics are simulated
- Includes simulation of a fire control radar with a flight path predictor (Kalman filter)

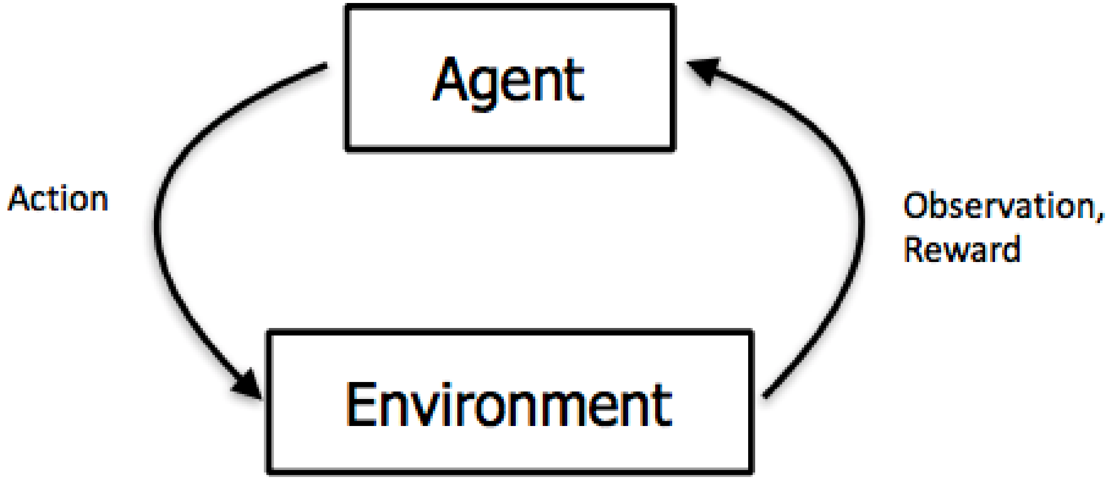
Machine Learning

- A lot of research into new applications in many fields
- Supervised image classification, and reinforcement learning game AI are probably the most well known applications
- Often referred to as «Deep Learning» when multi-layered neural networks are involved



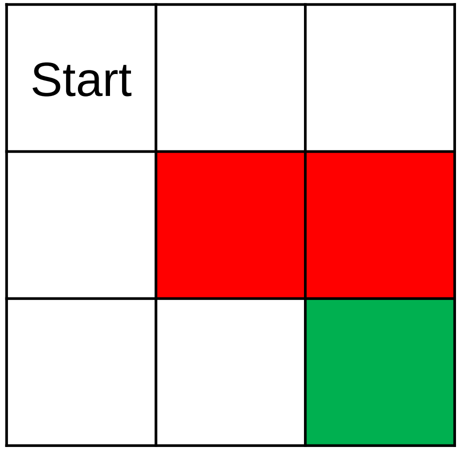
Reinforcement Learning

- The flying object is the Agent
- The AA gun simulation serves as the environment
- The agent moves through the flight corridor taking actions according to the feed back (rewards) it gets from the environment
- The most famous applications of RL are AlphaGo and AlphaZero (Chess) by Google Deepmind



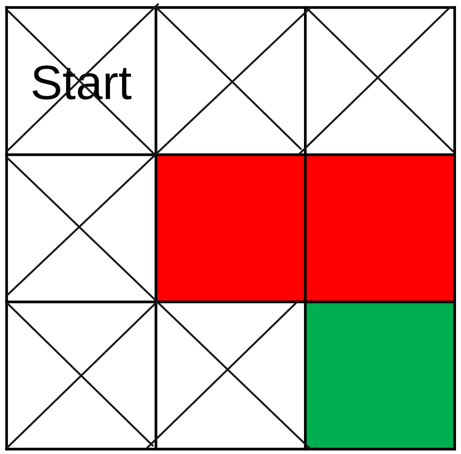
Simple Q-learning Illustration

- How do you get from the starting point to the green square, while avoiding the red squares, most efficiently?



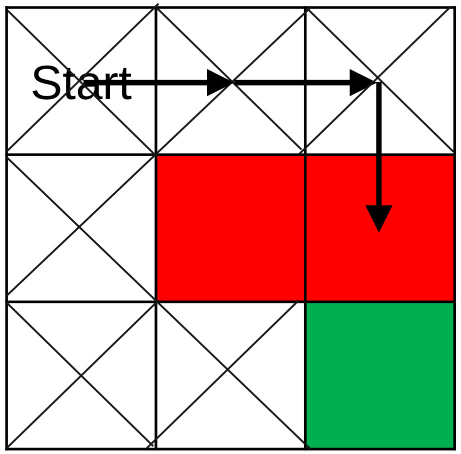
Simple Q-learning Illustration

- Each square (state) has four possible actions;
  - Move up, right, down, or left
- Assign a quality value (Q-value) to each action in every square (state), reflecting the maximum future rewards available to this particular action
- The table containing all states and their respective Q-values is called the Q-table



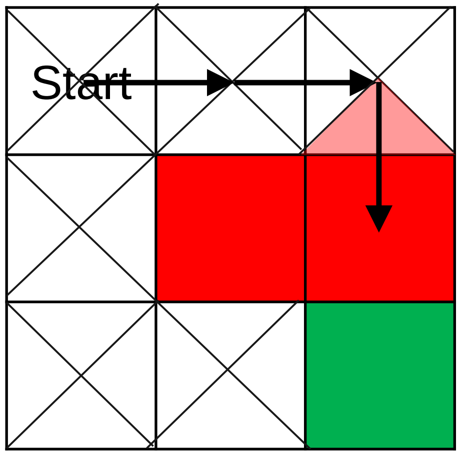
Simple Q-learning Illustration

- Fill Q-values into the Q-table by generating semi-random paths in the grid
- Every path (episode) starts in the top left square, and ends when it hits a red or green square



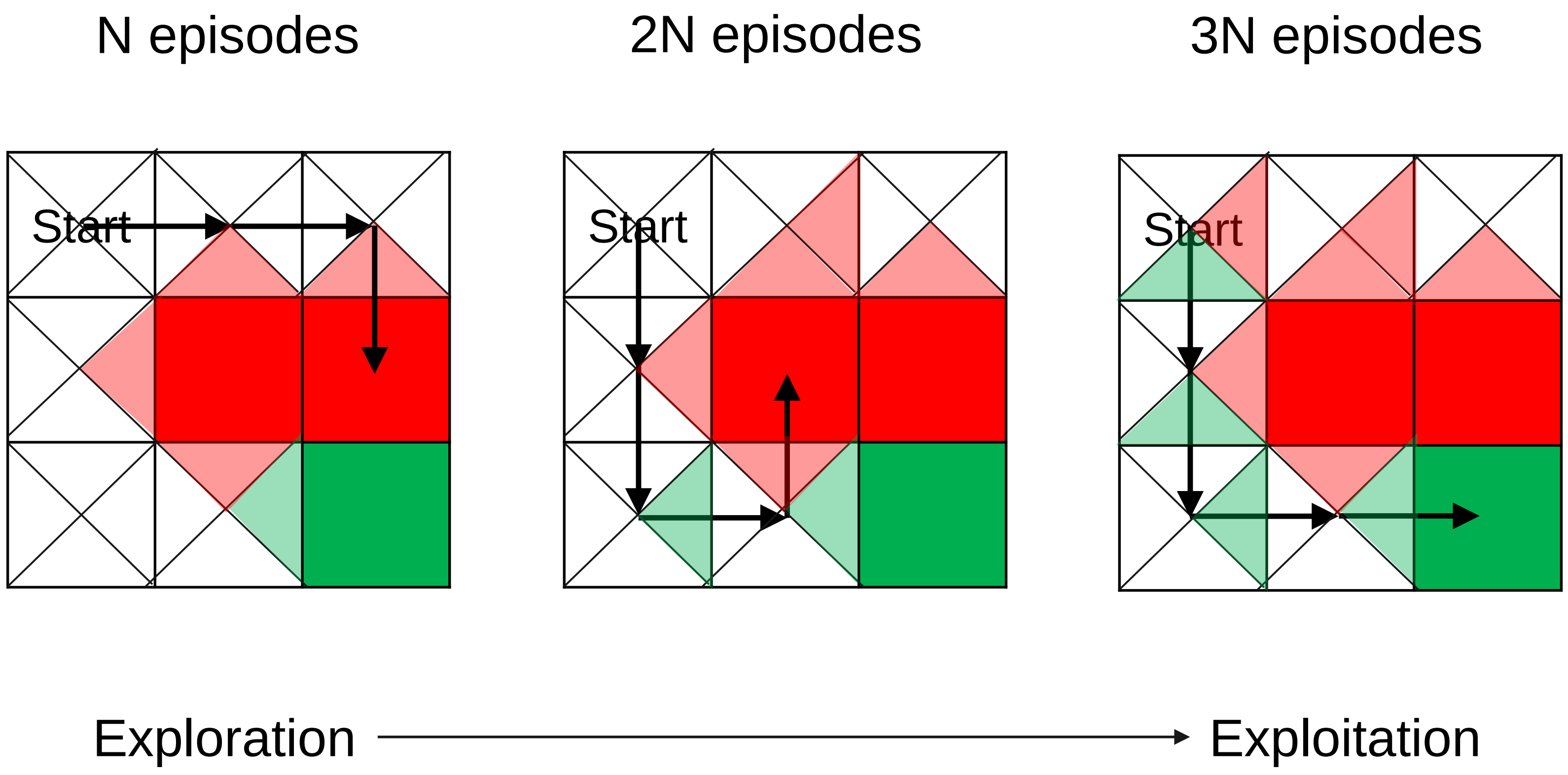
Simple Q-learning Illustration

- After each episode is finished, the Q-values visited along the track are updated
- The updated Q-values are a mix of the old Q-values, the reward given by the environment for taking the action, and the maximum Q-value available in the next state (square)



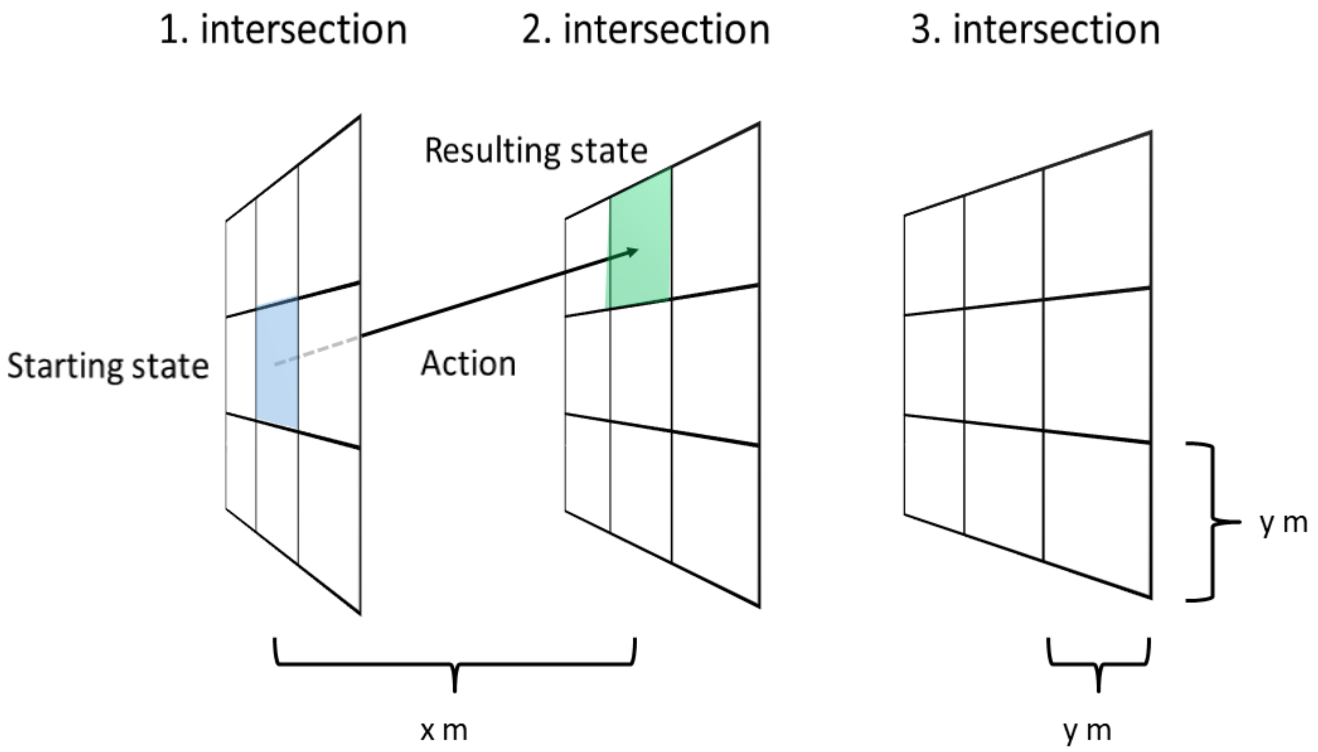
Simple Q-learning Illustration

- As the Q-values are gathered and propagated back through the Q-table, the randomness of the path is reduced by increasingly choosing the action with the highest Q-value in each step
- This is called going from exploration to exploitation



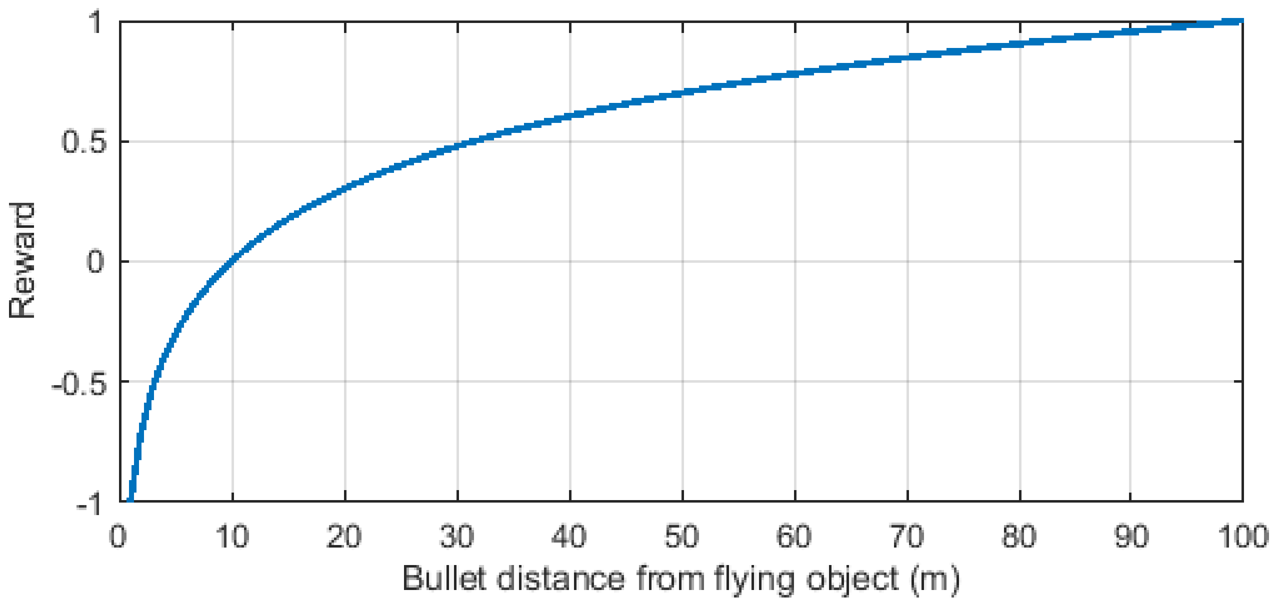
State and Action Layout

- The flight corridor is divided into intersections with fixed separations
- Each intersection contains a 51x51 grid
- Every transition from one intersection (state) to the next has 9 possible actions
- This gives a relatively small Q-table, which does not need to be approximated by a neural network



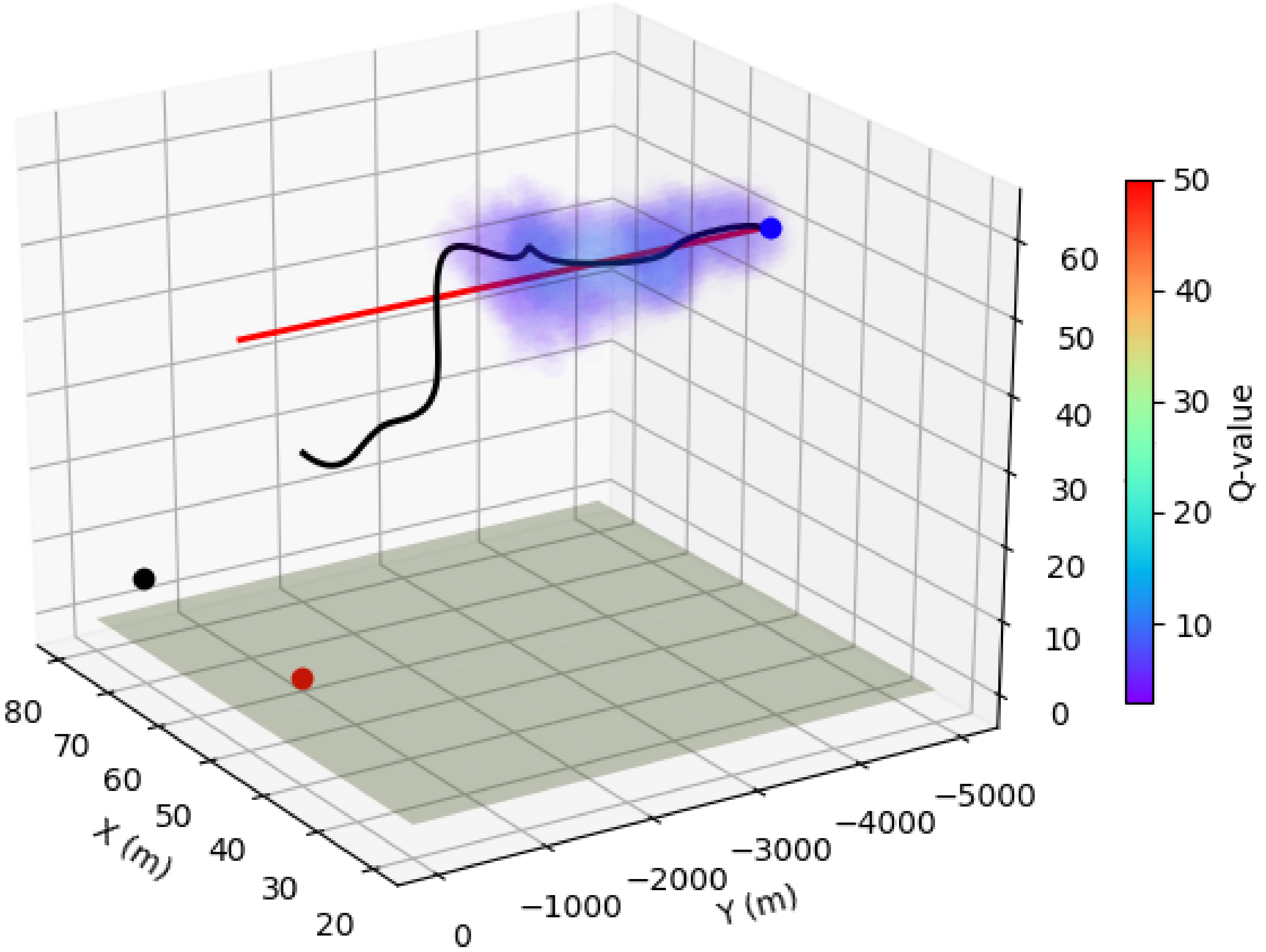
Reward Calculation

- The reward is found by taking the logarithm of the minimum separation between the AA-bullets and the flying object divided by 10 (limited to the range 1-100 m)
- This gives rewards ranging from -1 at 1 m separation, to 1 at 100 m range
- This keeps the average reward close to zero (avoiding divergence in the Q-values), and reduces the gradient of the rewards at large separations where the benefit of increasing the separation is smaller
- The reward for reaching the target is fixed at 500



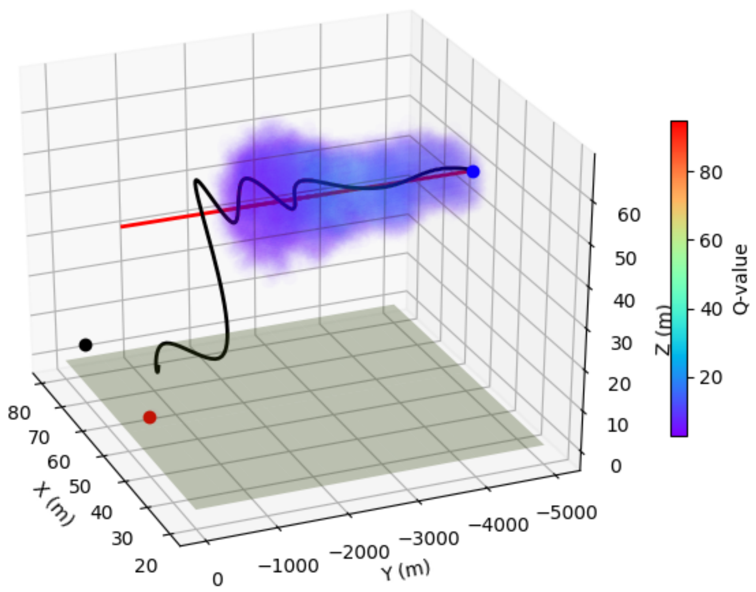
Result After 10 000 Episodes

- Maximum Q-value in each state (only Q-values > 3)
- The blue dot is the starting point, the black dot shows the location of the AA gun, and the red dot indicates the target position (reward = 500)
- The red line shows the initial flight path
- The black line is smoothed, and shows the path of optimal total reward as found by the Q-learning

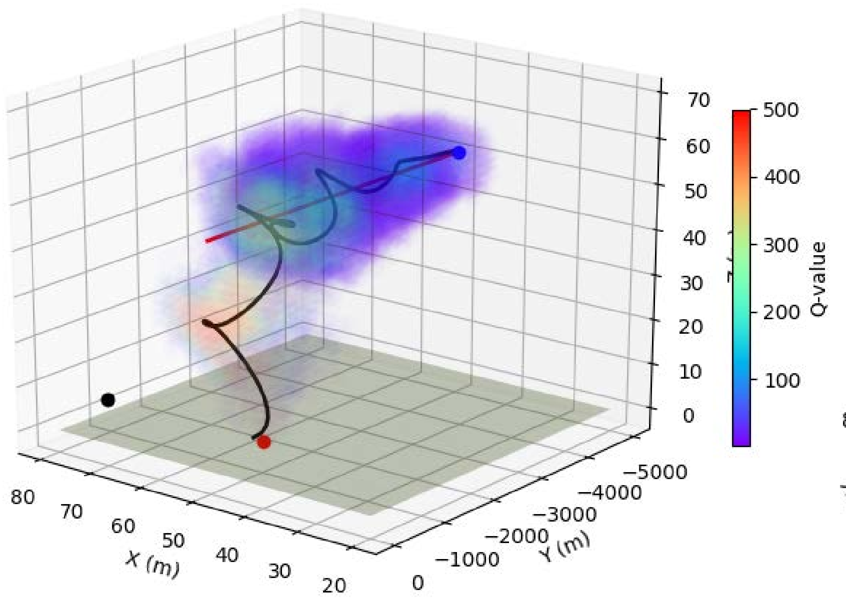


Results After Further Episodes

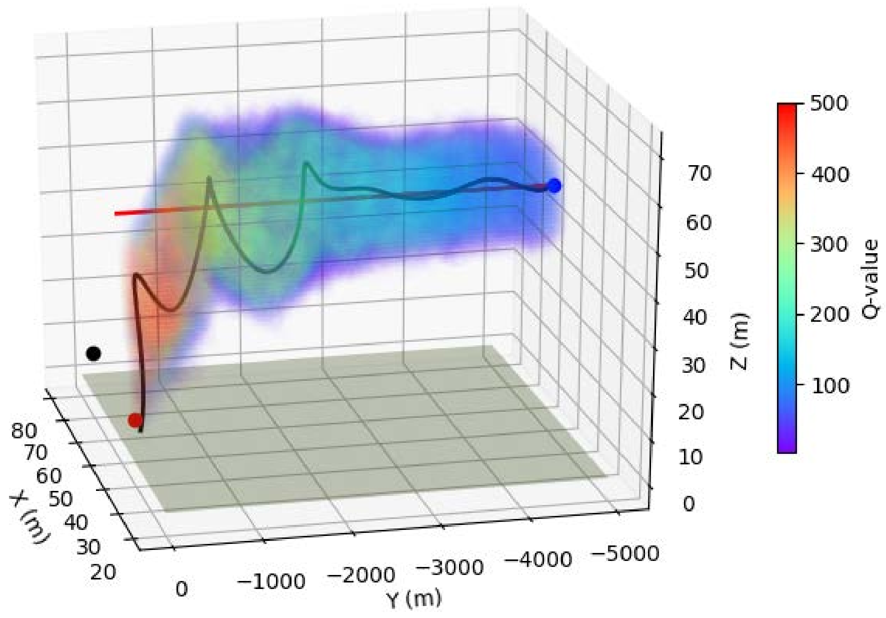
50 000



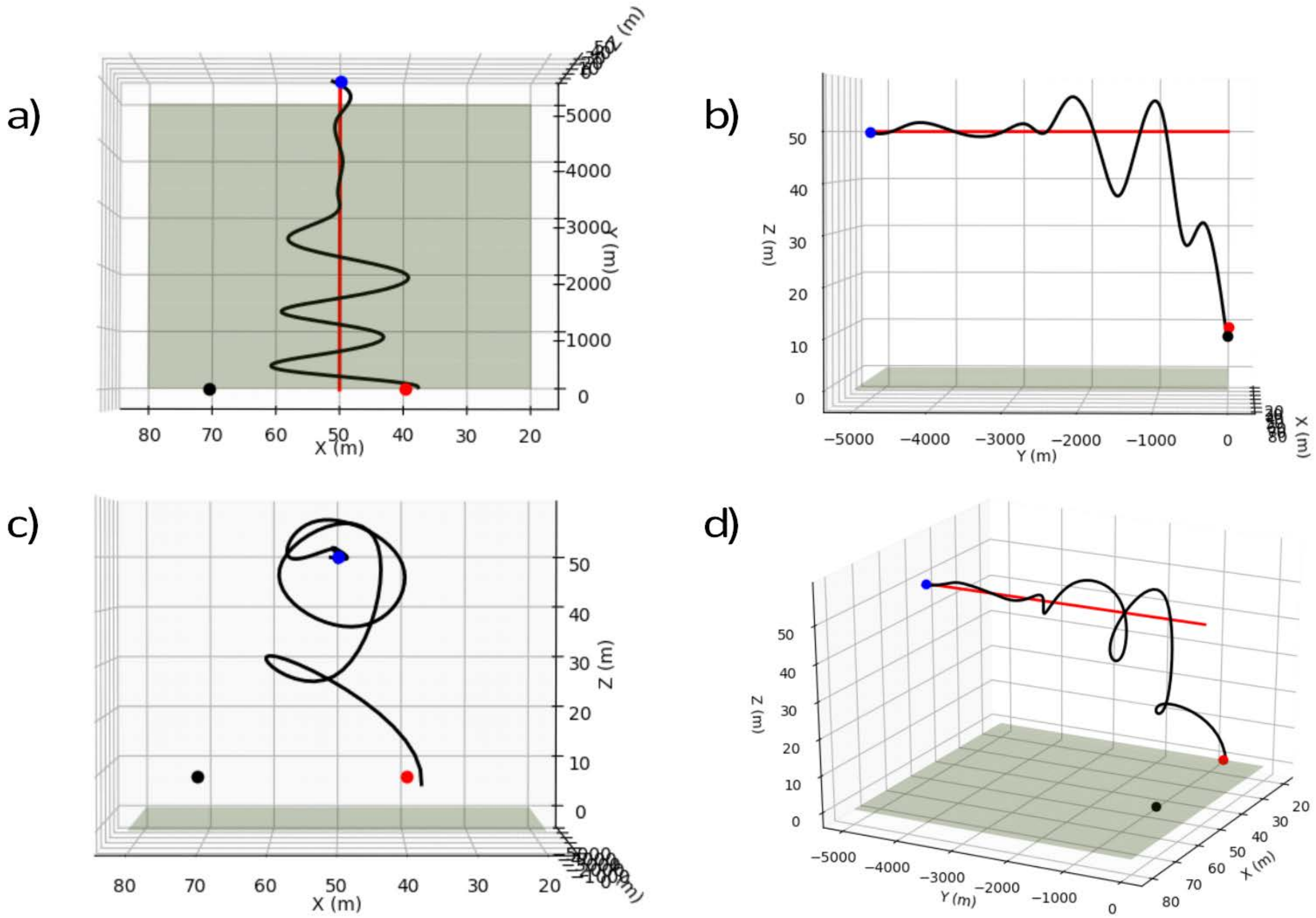
100 000



500 000



Result After 500 000 Episodes



Summary

- Machine learning can produce non-trivial flight paths for avoiding AA guns
- The results presented here are specific to one particular setup, but the method is very general
- Extra AA guns (or other threats) can be added to the environment without much computing cost