Bachelor in Computer Science and Engineering

Artificial Intelligence 2024-2025
Grupo 89

Final practice

"Heuristic Search in Radars"

Jorge Adrian Saghin Dudulea – 100522257
Denis Loren Moldovan – 100522240
Ignacio Cortina de Antonio – 100522372

Índice

Parte I

Introduction

uwu

Parte II

Explanation of the system

1. Mathematical modeling of the search problem

1.0.1. State Space:

Let the state space S be defined as:

$$S = \{(y,x) \in \mathbb{Z}^2 | 0 \leq y < H, 0 \leq x < W, \Psi^*(y,x) \leq \tau\}$$

where: - $H \times W$: Dimensions of the grid - $\Psi^* : \mathbb{Z}^2 \to [\varepsilon, 1]$: Scaled detection probability function - $\tau \in (0,1]$: Detection tolerance threshold

1.0.2. Operator Set:

The action set A defines valid transitions between states:

$$A = \{Up, Down, Left, Right\}$$

Each action $a \in A$ maps to a movement vector:

$$\delta(a) = \begin{cases} (-1,0) & \text{if } a = Up \\ (1,0) & \text{if } a = Down \\ (0,-1) & \text{if } a = Left \\ (0,1) & \text{if } a = Right \end{cases}$$

1.0.3. Transaction Function:

$$T : S \times A \to \text{S} \cup \{\emptyset\}$$

$$T((y,x),a) = \begin{cases} (y',x') & \text{if } (y',x') \in S \text{ where } (y',x') = (y,x) + \delta(a) \\ \emptyset & \text{otherwise} \end{cases}$$

1.0.4. Cost function:

$$c((y_1,x_1),(y_2,x_2) = \Psi^*(y_2,x_2)$$

2. Initial and Goal States

2.0.1. Initial State:

$$s_0 = (y_0, x_0) \text{ where } (lat_0, lon_0) \mapsto (y_0, x_0)$$

2.0.2. Goal State:

For a sequence of POIs $\{p_1, \ldots, p_k\}$:

$$s_{goal} = \{(y_i, x_i)\}_{i=1}^{k} \text{ where each } p_i \mapsto (y_i, x_i)$$

2.0.3. Solution Characteristics:

A solution is a path $\pi = [s_0, s_1, \ldots, s_n]$ where:

1. $s_n \in s_{goal}$
2. $\forall i, \exists a \in A$ such that $s_{i+1} = T(s_i, a)$
3. Total costs $C(\pi) = \sum_{i=0}^{n-1} c(s_i, s_{i+1})$ is minimized

3. Heuristics designed

3.1. Euclidean distance:

$$h_1((y,x),(y_{goal},x_{goal})) = \sqrt{(y-y_{goal})^2 + (x-x_{goal})^2} \cdot \varepsilon$$

3.1.1. Admissibility Proof:

1. Actual path cost between adjacent cells $\geq \varepsilon$
2. Straight-line distance is the minimum possible path length
3. Thus $h_1 \leq$ actual cost (underestimates)

3.2. Manhattan distance:

$$h_2((y,x),(y_{goal},x_{goal})) = (|y-y_{goal}| + |x-x_{goal}|) \cdot \varepsilon$$

3.2.1. Admissibility Proof:

1. Manhattan distance $\geq$ Euclidean distance
2. Each move costs $\geq \varepsilon$
3. Therefore $h_2 \leq$ actual cost (underestimates)

4. Map and searh space generation

4.1. Implementation of the map

Let the rectangular area delimited by geodetic coordinates ($\text{lat}_0, \text{long}_0$) ($\text{lat}_1, \text{long}_1$) be devided into a grid $H \times W$. Each cell represents a unique point in where - The latitude and longitude intervals are calculated as

$$\Delta\, lat = \tfrac{lat_0 - lat_1}{H-1} \quad \Delta\, long = \tfrac{long_0 - long_1}{W-1}$$

- Each cell (i, j) is mapped to a geographic coordinates:

$$lat_i = lat_1 + i \cdot \Delta lat, lon_j = long_0 + j \cdot \Delta lon$$

For each radar $r_k \in \{1, 2, ..., N_r\}$, located in the coordinates (lat_k, lon_k), we compute:

- The maximum detection range (R_{max}) using the provided rada equation.

$$R_{max} = \frac{P_t G^2 \lambda \sigma}{(4\pi)^3 P_{min} L}$$

- The euclidean distance (d) from each grid cell to the radar using the conversion factor $K = 111{,}000$ (meters per degree)

$$d = K \cdot \sqrt{(lat_i - lat_k)^2 + (lon_1 - lon_k)^2}$$

- If $d \leq R$, compute the detection possibility using the 2D Gaussian function

$$\Psi_k^*(i,j) = \frac{e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)}}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}}$$

  Where

  - $x = (lat_i, lon_j)$
  - $\mu = (lat_k, lon_k)$
  - Σ is the covariance matrix, typically diagonal (can be fixed for simplification)

- if $d > R_{max}$, set $\Psi_k^*(i,j) = 0$

- Then for each cell, compute the maximum detection possibility from all radars:

$$\Psi^*(i,j) = max_k \Psi_k^*(i,j)$$

The raw detection possibilities $\Psi^*(i,j)$ are scaled to the interval $[\varepsilon,1]$ using the modified Min-Max normalization:

$$\Psi^*_{\text{scaled}}(i,j) = \left(\frac{\Psi^*(i,j) - \Psi^*_{\text{mín}}}{\Psi^*_{\text{máx}} - \Psi^*_{\text{mín}}}\right) \cdot (1-\varepsilon) + \varepsilon$$

where: - $\Psi_{\text{mín}}$ and $\Psi_{\text{máx}}$ are the minimum and maximum non-zero detection values in the map. - ε is a small positive constant to ensure no cell has cost 0 (e.g., $\varepsilon = 0{,}001$)

The resulting detection map is stored in memory as a matrix:

$$M \in \mathbb{R}^{H\times W}, \quad M[i][j] = \Psi^*_{\text{scaled}}(i,j)$$

5. Search graph generation

Given the detection map M, we construct a directed weighted graph $G = (V,E)$, where:

5.1. 2.1 Vertices

Each vertex $v_{i,j} \in V$ corresponds to a cell (i,j) such that:

$$M[i][j] \leq \text{detection threshold}$$

Only these vertices are considered traversable by the spy plane.

5.2. 2.2 Edges

An edge exists between vertex $v_{i,j}$ and its valid neighbors $v_{i',j'}$ if:

$$(i',j') \in \{(i+1,j),(i-1,j),(i,j+1),(i,j-1)\}$$

and both cells satisfy:

$$M[i][j] \leq \text{threshold}, \quad M[i'][j'] \leq \text{threshold}$$

For each such edge:

$$e_{(i,j)\to(i',j')} \in E \quad \text{with weight} \quad w = M[i'][j']$$

That is, the cost of moving from a cell to a neighbor is the detection cost of the **destination** cell, reflecting the risk of exposure.

The graph is stored in memory as an adjacency list or adjacency matrix, depending on the implementation.

Parte III

Experiments

uwu

Parte IV

Use of AI

uwu

Parte V

Conclusion

uwu