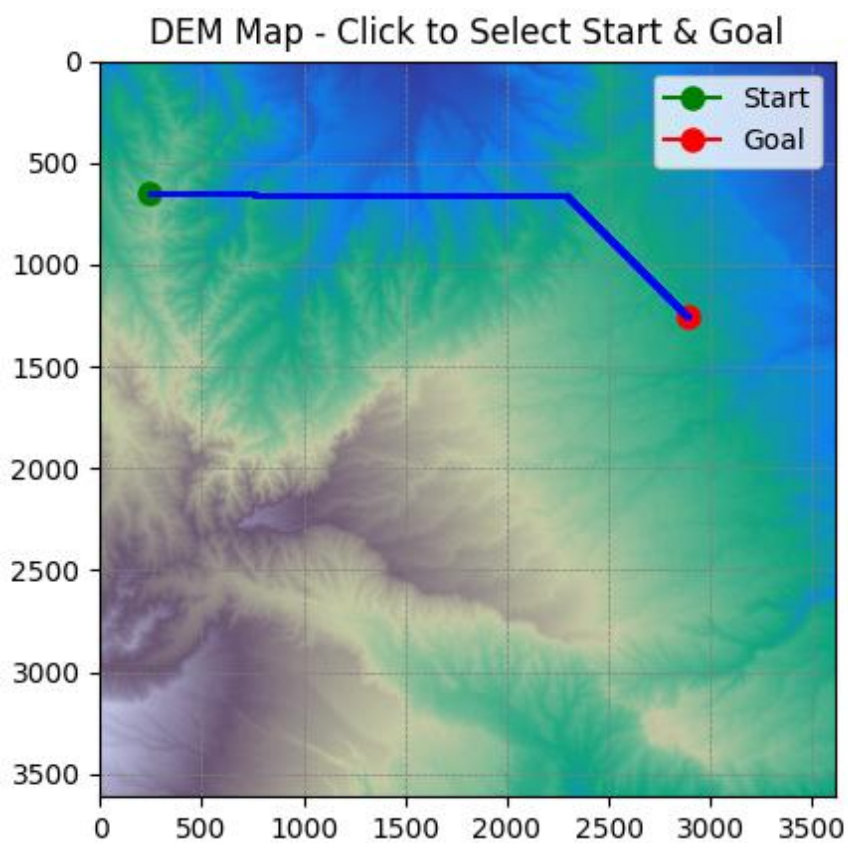
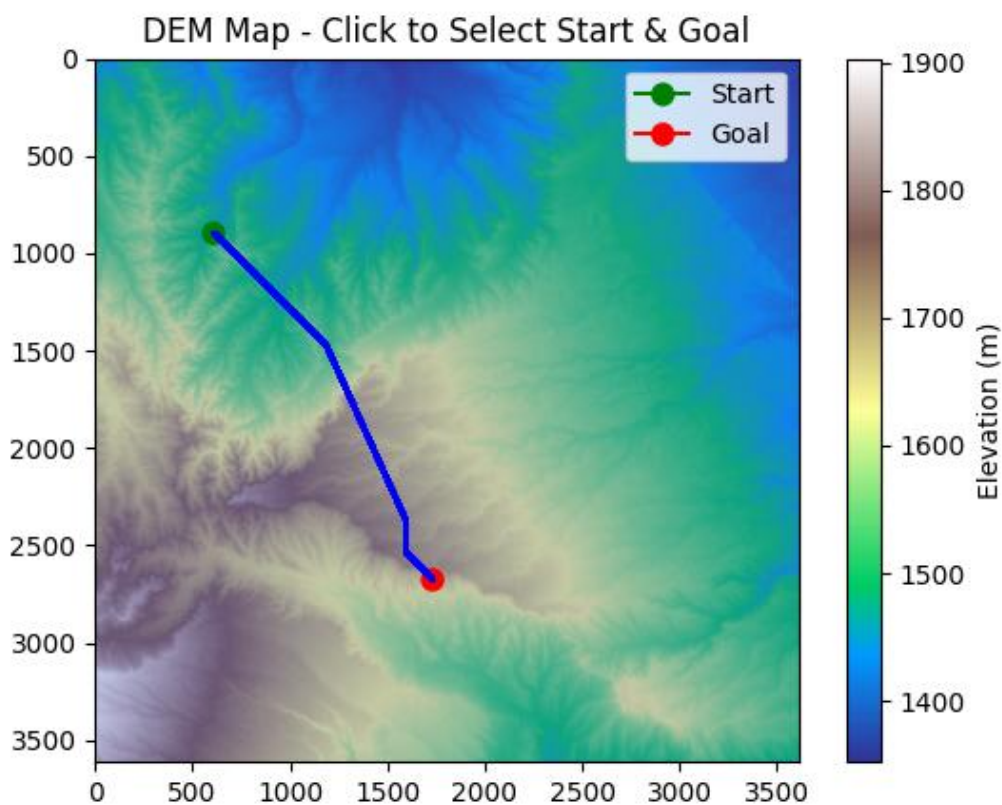


Simple A* over DEM without any Constraints



1. Raster Grid Constraints

- The DEM is a discrete grid, not a continuous surface.
- A* must move from pixel to pixel — either horizontally, vertically, or diagonally.
- So even the "straightest" diagonal will be an approximation (like a stair-step pattern).

2. 8-Connected Grid Search

- The directions allowed are limited to:
 - 4 cardinal (N, S, E, W)
 - 4 diagonal (NE, NW, SE, SW)
- There's no interpolation between directions, so the path is forced to switch between these options to approximate diagonality.

3. Tie-Breaking and Priority Queue Behavior

- If two paths have equal cost, the order in which they are inserted into and popped from the priority queue (heap) can influence which one is chosen.
- This may cause slightly uneven step patterns, even though distances are equal.

4. Floating Point Precision

- Even small floating-point rounding differences in g_score or f_score can nudge the algorithm to favor one neighbor over another.
- It does not consider slope or terrain type—only elevation limits.
- There's no terrain cost weighting, just geometric (Euclidean) distance.

DEM-Based Path Planning GUI with Slope-Constrained A* Algorithm

This application is a Python-based graphical user interface (GUI) for terrain-aware path planning using a Digital Elevation Model (DEM) and slope data. It allows users to interactively select start and goal points on a terrain map and computes an optimal path using the A* algorithm, while respecting constraints such as slope and elevation. It is particularly suited for applications like military or autonomous vehicle navigation over rugged terrain.

Data Inputs and Preprocessing:

- A DEM file that contains elevation values.
- A slope raster that defines the slope angle (in degrees) for each pixel.

After loading, the system asks for a minimum and maximum elevation value. It then generates two masks:

- A GO area mask where the elevation lies within the specified range.
- A valid mask that further restricts movement to areas where slope is below a defined threshold (45°).

A* Pathfinding Logic:

The system implements the A* algorithm with the following considerations:

- It supports movement in 8 directions: up, down, left, right, and diagonals.
- The movement cost between points is computed using Euclidean distance.
- The algorithm maintains a score (g) for the cost from the start to each node and a heuristic (h) estimating the distance to the goal.
- The sum of cost and heuristic ($f = g + h$) is used to prioritize exploration.
- Only nodes within the valid mask (elevation and slope constraints) are considered for traversal.

Once the goal is reached, the system reconstructs the path by backtracking from the goal to the start.

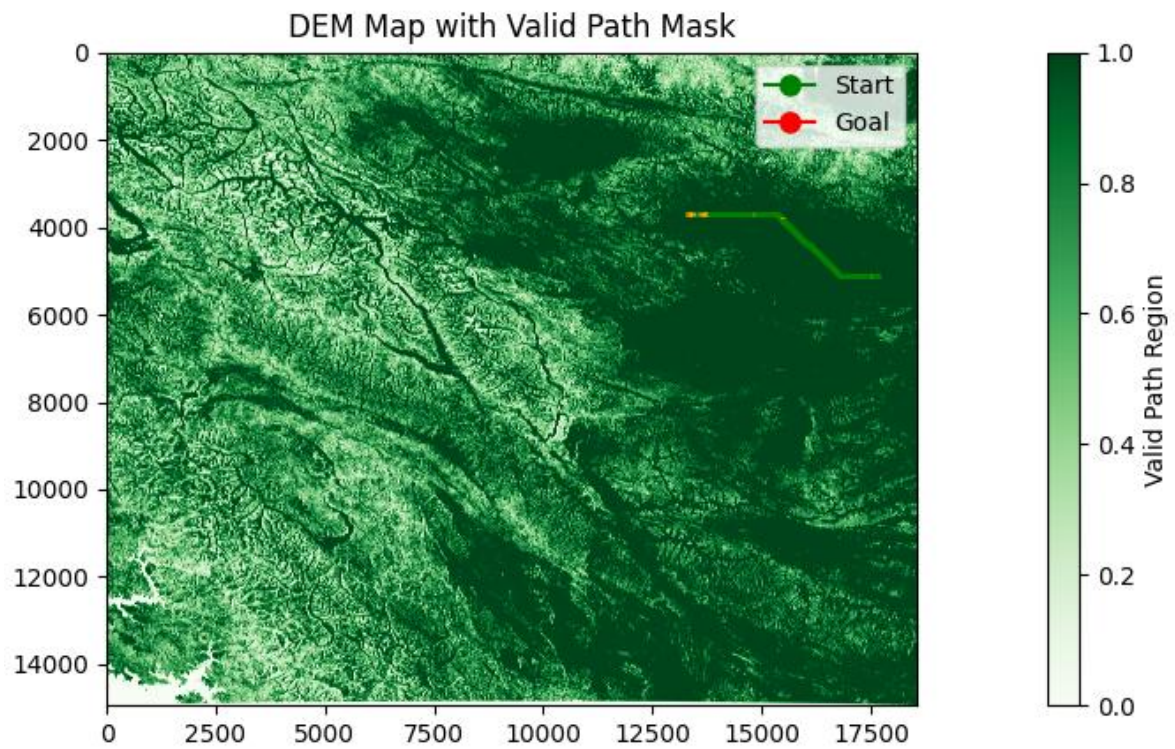
Visualization and Output:

The computed path is visualized over the DEM with colour coding to indicate terrain difficulty:

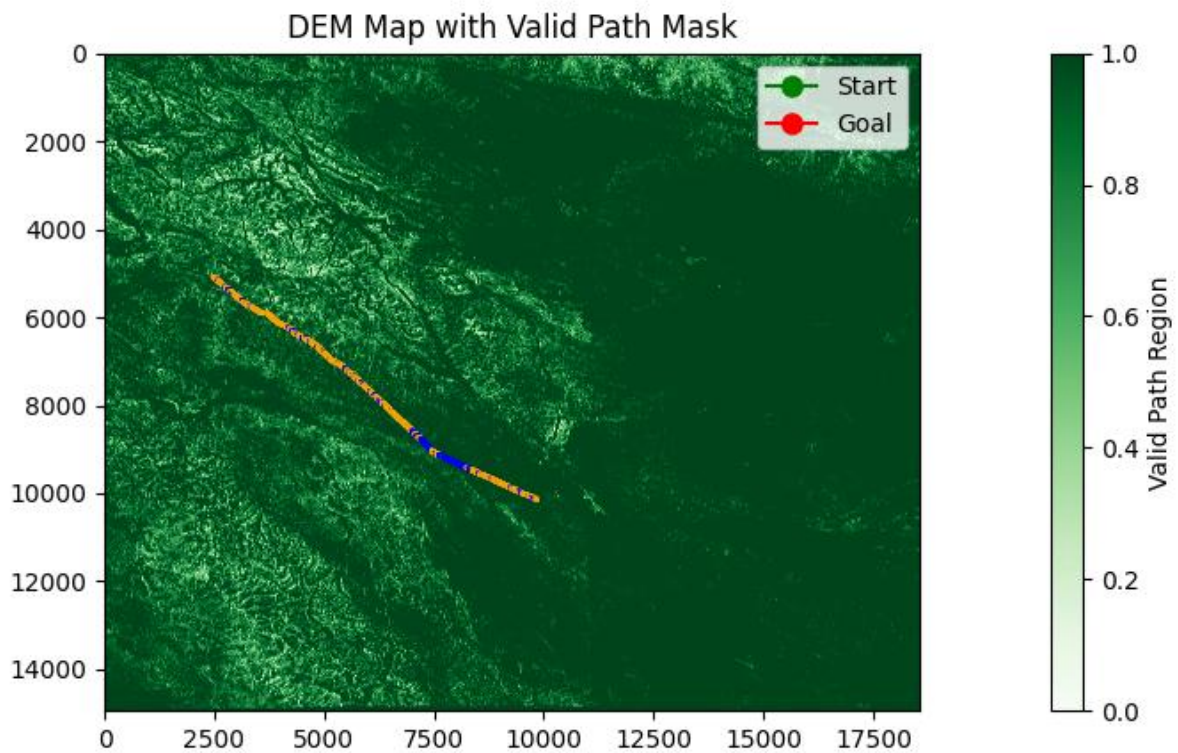
- Blue for low-slope segments ($\leq 15^\circ$)
- Orange for moderate slope (15° to 45°)
- Red for higher slopes (if allowed)

Output:

Slope Threshold was set to 35° (Elevation Min: 1500m; Max: 7500m)



Slope Threshold was set to 45° (Elevation Min: 100m; Max: 8500m)



Elevation Profile Along Path

