A Virtual Reality Game to Explore Hyperbolic Geometry

Ravi Snellenberg - rsnellenberg@tudelft.nl

Responsible Professor: Martin Skrodzki Supervisor: Rafael Bidarra

TU Delft

1: Background/motivation:

- A different representation of space is used in a Virtual Reality(VR) game instead of the normal Euclidean space we are used to.
- This opens up the possibility of traversing infinitely far in the Virtual space while being confined to a relatively limited space in the real world.
- People will need the ability to traverse this new space in an intuitive or at least competent way.
- One way to help people with navigation in such a space would be to show the person what the fastest way would be to go from point A to point B.

2: Problem Description:

- The environment in which the shortest path needs to be computed is a **combination of a 3x3 grid**, which represents the play area, **and a plane with the Order-5 square tiling**. **See Figure 1 for an example.**
- Getting the shortest path in either the plane or the grid would not be that hard to compute. It is the combination of the two that currently does not have an efficient method.
- The goal is to create an algorithm that can find the shortest path from point A to point B in the virtual world under these conditions as fast as possible.

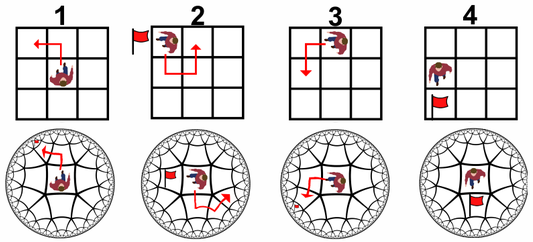

Figure 1: figure shows relation of the real world tiles and the virtual tiles

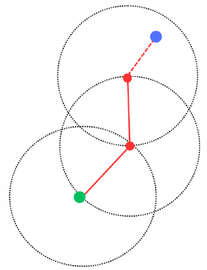

Figure 4: A possible path using pre-computed nodes.

The green dot in this figure is the starting point, the blue dot represents the goal, the dotted circles represent the precomputed area around each node and the redline/nodes represent the walked path.

3: Proposed Solutions:

Since it is possible for the problem to be modelled as a graph, graph traversal algorithms can be a solution. The ones that were used are:

- **Breadth first:** The shortest path can be found by traversing the graph in a breath-first manner. This is used as a baseline
- **A* with scalar:** Dijkstra but it is guided by a heuristic, which in this case is the fastest possible path in the virtual world without taking into account the physical world. How important the heuristic is can be adjusted with a scalar
- **Anytime A*:** Same algorithm as above but it can be stopped at any time and it will return an estimate of how the shortest path would start.
- **Nodes abstraction:** This method precomputes the shortest paths in an area around a point and then models a graph from nodes which each represent an entire area of the graph instead of a single point (See figure 4)
- **Domain-specific rules:** This method uses assumptions that can be made about the problem to compute a path without actually traversing a graph. It currently has the limitation that it can only handle cases when the path starts in the centre tile.

4: Experiment Results:

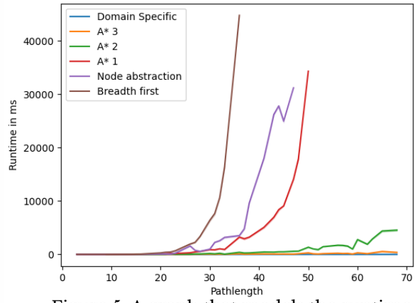

Figure 5: A graph that models the runtime for each algorithm

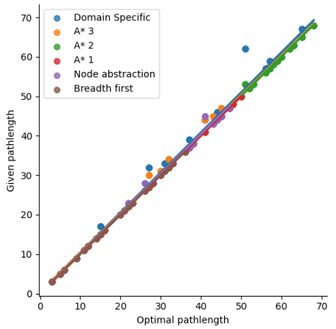

Figure 6: A graph that models the correctness of the returned answers

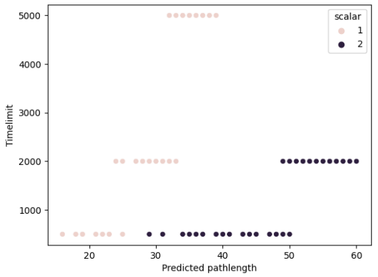

Figure 7: It shows the length of the predicted path the anytime algorithm returns based on the time-limit and scalar
All the predicted paths were optimal

5: Conclusion:

- If the results need to **guarantee the shortest path, A* with scalar 1** should be used, but it is quite slow and thus not viable on goals that are further away
- A* with a scaler larger than 1 will **sacrifice some accuracy for a big speedup** of the answer
- If the goal is truly far away the domain-specific rules or anytime algorithm should be used. The **anytime algorithm does not return a full path** (but this isn't needed in all use cases) and the prediction using **Domain-specific rules is currently the least accurate but fastest algorithm**.

6: Future work:

- The constraint the domain-specific algorithm currently has can still be removed, it might also be possible to have it always return accurate results
- The abstraction nodes method currently just slows things down and makes the answer worse. Both of which could be fixed
- Find the best way to integrate this shortest path into the VR game, to help the player navigate