# Weighted A*

# How Greedy Is Too Greedy?

## Introduction

### How does weighted A* work?

Like A*, weighted A* incorporates a heuristic in the cost function that focuses its node search in the direction of the goal which generally decreases the amount of nodes visited and computing time. If the C-space is particularly constrained such that possible paths are limited the visited node list and time to compute benefits decrease. Also, A* does not guarantee an optimal path and a more constrained C-space decreases the likelihood of it returning an optimal path. Weighted A* simply multiplies a scaling factor (weight) by that heuristic to place a larger emphasis on the heuristic. This amplifies the standard A* advantages and disadvantage……but to what degee?

### Difference between A* and Weighted A* (in yellow):

```
heuristic = compute_distance(new_position, goal_point)

weight = ?

greedy_cost = compute_distance(new_position, [current_node.x, current_node.y, current_node.z]) + current_node.cost + weight*heuristic
```

### Problem Statement:

Using a weighted A* pathfinding algorithm on a quadcopter, what are the effects on computation time and path optimality in increasingly constrained 3D space with varying weights?

## Testing

Testing parameters:

To compare between each trial some parameters must remain constant:

- C-space = 91 X 20 X 20
- Starting Point of = [ 0 , 0 , 0 ]
- Goal Point of = [ 90 , 5 , 2 ]

While the following parameters will be varied:

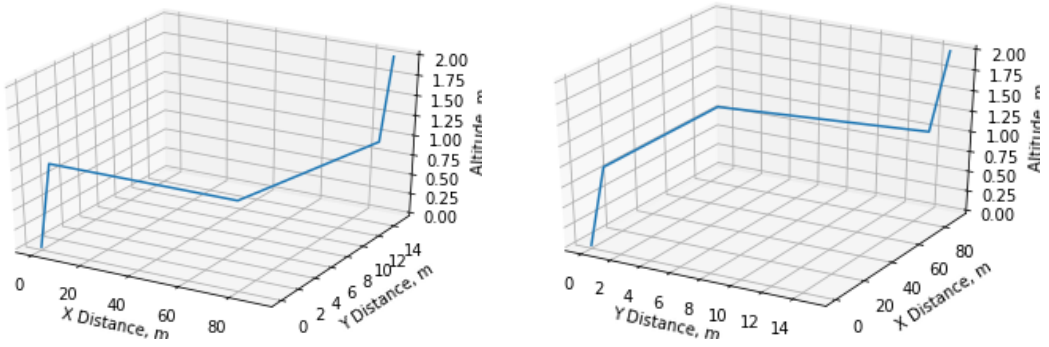
- Path constraints by adding 4 layers of obstacles
- For each layer run trials for weights of 1, 2, 5, 10, and 1000

The code that creates and plots the obstacles changes between the layers thus has varying computing time associate to it. This is why the list of obstacles is created before the computing time starts and computing time is calculated before plotting code runs.

## Results

Top row of graphs show the optimal path (Using Dijkstras) and bottom row shows path of weighted A* where the largest computation time benefit occurred. If weighted A* graphs mimic the optimal path or none of the weighted trials show a computing time benefit then only the optimal path row of graphs is shown. Graph order from left to right, obstacle layer plot with xyz path, xyz path by itself, and yxz path by itself. Trial with lowest time to compute is highlighted yellow.

No layers (Direct Line Of Sight, No Obstacles Present):

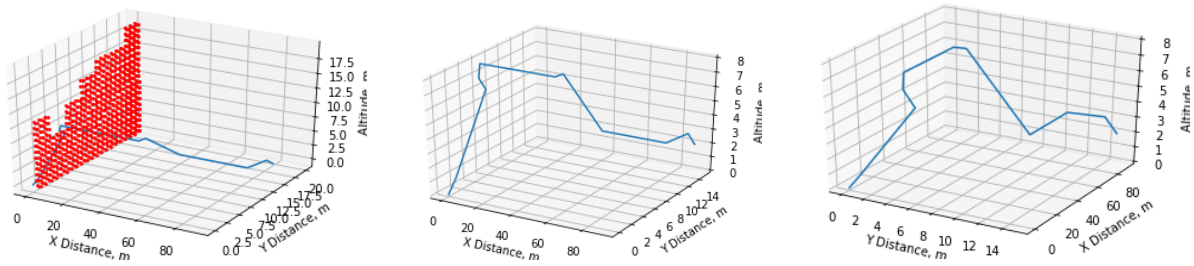


Trial data:

The optimal path cost is: 96.8488 m

| | | |
|---|---|---|
| Weight = 1 (NO Weight) | Path cost = 96.8488 | Time to Compute = 10.5379 seconds |
| Weight = 2 | Path cost = 96.8488 | Time to Compute = 10.7860 seconds |
| Weight = 5 | Path cost = 96.8488 | Time to Compute = 9.9759 seconds |
| Weight = 10 | Path cost = 96.8488 | Time to Compute = 10.5829 seconds |
| Weight = 1000 | Path cost = 96.8488 | Time to Compute = 10.3139seconds |

## 1 Layer Of Obstacles:

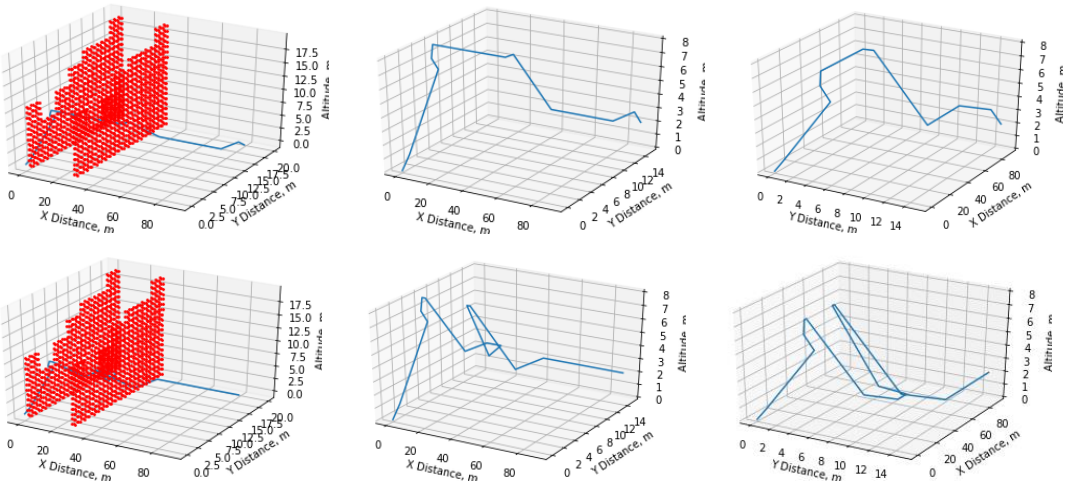



Trial Data:

The optimal path cost is: 105.6807 m

| | | |
|---|---|---|
| Weight = 1 (NO Weight) | Path cost = 105.6807 | Time to Compute = 22.0799 seconds |
| Weight = 2 | Path cost = 105.6807 | Time to Compute = 19.6059 seconds |
| Weight = 5 | Path cost = 105.6807 | Time to Compute = 23.4110 seconds |
| Weight = 10 | Path cost = 105.6807 | Time to Compute = 23.7709 seconds |
| Weight = 1000 | Path cost = 105.6807 | Time to Compute = 23.3079 seconds |

## 2 Layers Of Obstacles:

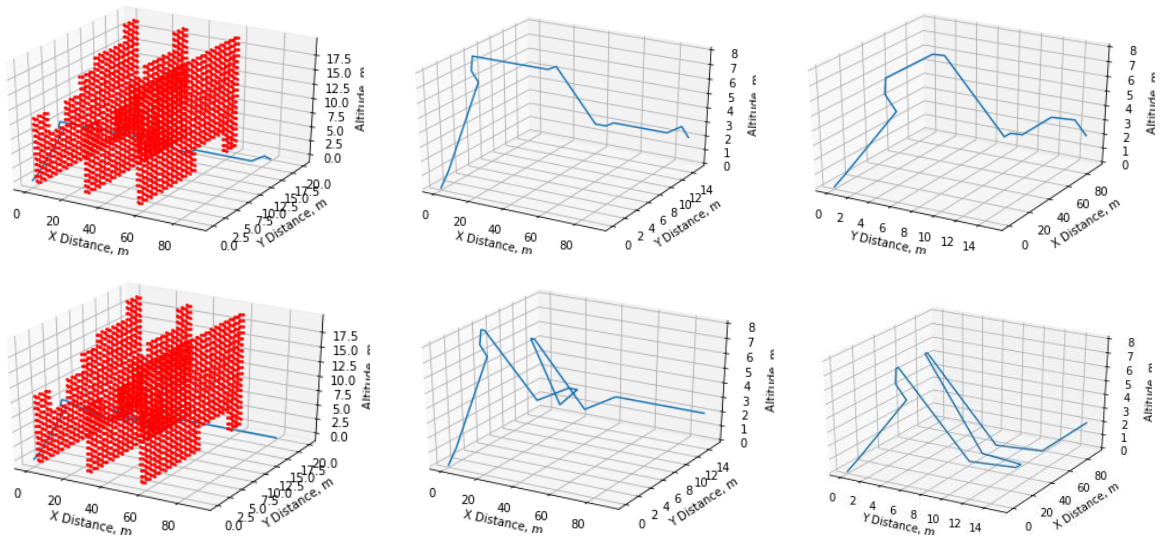


Trial Data:

The optimal path cost is:  105.6807 m

| | | |
|---|---|---|
| Weight = 1 (NO Weight) | Path cost = 112.5582 | Time to Compute = 29.7999 seconds |
| Weight = 2 | Path cost = 114.6580 | Time to Compute = 30.9847 seconds |
| Weight = 5 | Path cost = 115.4864 | Time to Compute = 29.2729 seconds |
| Weight = 10 | Path cost = 116.9505 | Time to Compute = 29.5680 seconds |
| Weight = 1000 | Path cost = 117.7790 | Time to Compute = 31.2670 seconds |

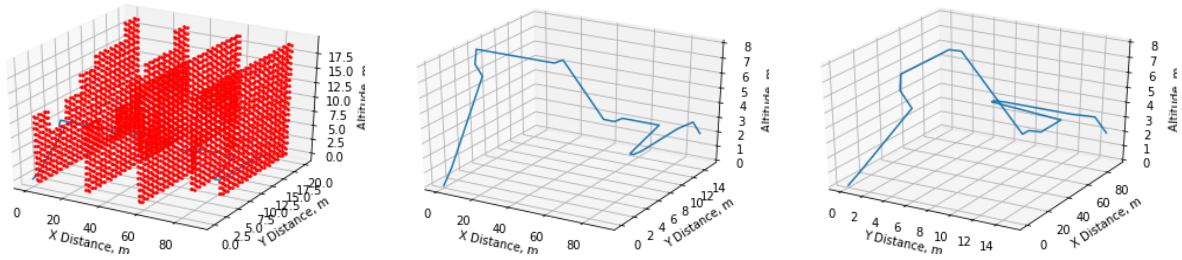
## 3 Layers Of Obstacles:




The optimal path cost is:  105.6807 m

| | | |
|---|---|---|
| Weight = 1 (NO Weight) | Path cost = 112.5582 | Time to Compute = 41.3920 seconds |
| Weight = 2 | Path cost = 114.6580 | Time to Compute = 43.3239 seconds |
| Weight = 5 | Path cost = 115.4864 | Time to Compute = 41.9139 seconds |
| Weight = 10 | Path cost = 116.9505 | Time to Compute = 50.5340 seconds |
| Weight = 1000 | Path cost = 117.7790 | Time to Compute = 40.7965 seconds |

## 4 Layers Of Obstacles:



The optimal path cost is: 114.9944 m

| | | |
|---|---|---|
| Weight = 1 (NO Weight) | Path cost = 123.5288 | Time to Compute = 51.1589 seconds |
| Weight = 2 | Path cost = 126.4570 | Time to Compute = 51.4984 seconds |
| Weight = 5 | Path cost = 127.2854 | Time to Compute = 53.4080 seconds |
| Weight = 10 | Path cost = 128.7495 | Time to Compute = 53.4882 seconds |
| Weight = 1000 | Path cost = 129.5780 | Time to Compute = 54.4770 seconds |

# Analysis Of Results

## No layers (Direct Line Of Sight, No Obstacles Present):

A* and weighted A* path costs were not affected.

All trials returned the optimal path of 96.8 meters.

The best time to compute trial was using a weight of 5 with a time savings of .562 seconds

## 1 Layers Of Obstacles:

A* and weighted A* path costs were not affected.

All trials returned the optimal path of 105.7 meters.

The best time to compute trial was using a weight of 2 with a time savings of 2.474 seconds

## 2 Layers Of Obstacles:

A* and weighted A* path costs were affected.

the optimal path cost is 105.7 meters.

A* returned a path cost of 112.6 meters.

All weighted A* trials returned increasingly higher path costs with a cost differential of 5.2 meters using a weight of 1000.

The best time to compute trial was using a weight of 5 with a time savings of .527 seconds

## 3 Layers Of Obstacles:

A* and weighted A* path costs were affected.

the optimal path cost is 115.0 meters.

A* returned a path cost of 112.6 meters.

All weighted A* trials returned increasingly higher path costs with a cost differential of 5.2 meters using a weight of 1000.

The best time to compute trial was using a weight of 1000 with a time savings of .5955 seconds

## 4 Layers Of Obstacles:

A* and weighted A* path costs were affected.

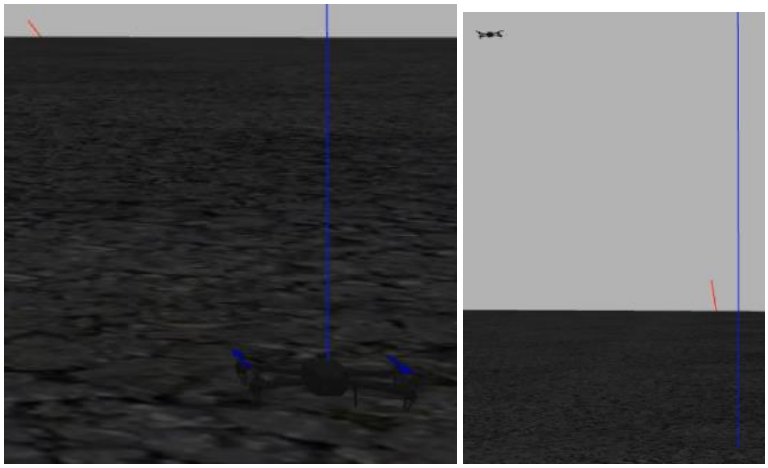
the optimal path cost is 105.7 meters.

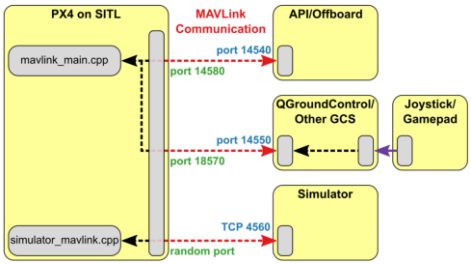
A* returned a path cost of 123.5 meters.

All weighted A* trials returned increasingly higher path costs with a cost differential of 6.0 meters using a weight of 1000.

No weighted A* trials returned an better time to compute.

## Weighted A* and Quadcopter Simulation Integration






ROS was used with PX4 and the Gazebo simulator To try out my 3D weighted A* algorithm. It uses the MAVROS MAVLink node to communicate with PX4. Install was difficult but integration of my weighted A* algorithm was relatively easy.

## Conclusion

Increasing obstructions had a substantial effect on path cost with an increasingly more constrained C-space. All layer sets had showed evidence that there is an optimal weight except for layer for configuration. This could be because a more thorough analysis needed to be done on layer 4 configuration, exploring more and/or broader range of weights. It's also possible that Layer 4 could has reached a threshold of constraints such that adding a weight would have a negative effect. This why If I were to continue with this research I would run a much larger set of trials, weights between 1 and 100,000 using a step size of .1, and add 20+ layers of complexity(If my system can handle it). I also would like to add a heuristic at adjusts for higher drone velocities. Looking at path cost logged from the simulator.