1) Could I obtain a list of the entrances and exits? x,y,z coordinates and any other information you have for them?  
2) Are the paths in the sample I have complete enough to infer entrances and exits?  
3) If not, would it be possible to obtain a larger sample with enough information to infer entrances and exits, i.e. complete paths?  
4) Is there a larger sample which could be made available?

Answer:

1) Attached is the yaml, file by which you can infer the co-ordinates and names of the regions. I suppose the file would be self explanatory (Please have a look at it and let me know if you need other details).

2) Since the paths are probabilistic, I'm not completely sure. However almost all of the entrances should be covered. Regarding the exits, it depends on the strength of the agents.

3) Find the sample trajectories for another time period i.e 17:00 hrs [here](https://drive.google.com/open?id=0B678QrRx9jO-VkRFZ2xNeDRUNm8). This data should cover better regions than the previous.

4) Unfortunately, I don't have access to the lab systems (from India) and cannot run another simulation. The only other person (Thanasis) who could run a simulation has also graduated and is working in California. So, I'm not sure if we can provide you new data.

Answer:

You can use [Sublime](https://www.sublimetext.com/3) (Text Editor) to open the yaml files.

The entire building is made up of rectangles (boxes) - for the walls and triangles - for the surface. The type in each block (**collision\_geometry**) tells you the object under consideration.

The **config** block tells you the **position (x, y, z)**and orientation (quaternion) of the objects (In this case mostly walls).

The entrances/exits can be identified using the name. The format is "**\*\_Entrance\_\*".**Ex: The entrance for the subway has the name "Subway\_Entrance\_2\_12714".

**Question:**

Sorry to bother you yet again. I am confused because a number of the Entrances seem to have the same position coordinates. For example, the entrance below is apparently as x,y,z = [0,0,0]. Are they relative to something else?

In my experiments last week I was assuming the x,y,z coordinates from the thread data were global. From the 3 D graphing that seemed to make sense for the agents.

Linda

name: 41st\_Street\_Entrance\_North\_East\_10509

        collision\_geometry:

          type: polygon

          triangles: [

                    76.6,-11.75, 6.0, 60.55,-8.14, 6.0, 60.61,-11.75, 6.0,

                    100.6,-11.75, 6.0, 105.32,-11.75, 6.0, 132.91,-8.32, 6.0,

                    105.32,-11.75, 6.0, 107.24,-11.75, 6.0, 132.91,-8.32, 6.0,

                    107.24,-11.75, 6.0, 123.69,-11.75, 6.0, 132.91,-8.32, 6.0,

                    60.55,-8.14, 6.0, 76.6,-11.75, 6.0, 85.04,-11.75, 6.0,

                    89.54,-11.75, 6.0, 60.55,-8.14, 6.0, 85.04,-11.75, 6.0,

                    92.88,-11.75, 6.0, 60.55,-8.14, 6.0, 89.54,-11.75, 6.0,

                    100.6,-11.75, 6.0, 60.55,-8.14, 6.0, 92.88,-11.75, 6.0,

                    100.6,-11.75, 6.0, 132.91,-8.32, 6.0, 60.55,-8.14, 6.0,

                    ]

          material: off\_white

        visualization\_geometry:

          type: box

          dims: [0.0000001, 0.0000001, 0.0000001]

          material: off\_white

        config:

          position: [0, 0, 0]

          orientation: [0, 0, 0, 1]

        isElevator: false

Answer:

Your’re right.

Firstly, the building is made up of two kinds of objects **box**and **polygon.**Each object has two kinds of geometries **collision geometry** and **visualization geometry.**

In this specific case the main entities of the building, are **floors** and **walls.** In the case of walls you don't have any visualization geometry ( since both the geometries are treated alike). So the walls are boxes with dimensions (**dims)**given by the [length, width and height].

Each **floor** is represented by a set of **triangles. (**Polygon's are triangulated for efficiency). Each triangle in-turn is represented by its**center** (centroid).

In the above example,

 triangles: [

**76.6 (x),-11.75 (y), 6.0(z), 60.55(x),-8.14(y), 6.0(z), 60.61(x),-11.75(y), 6.0(z),**

**100.6(x),-11.75(y), 6.0(z),etc**

                    ]

I suppose you can use the names of the regions as identifiers to extract the required data.

Unfortunately there is not an accurate way to determine the location of an entrance/exit. We map regions by one of the triangles (given in the yaml) and pick the one that's closest to the agent i.e the distance between the agent and his destination coordinates should be atmost 0.2 meters.

However I suppose you can use a random triangle in the set of triangles (of the region) as the xyz coordinates of the entrance/exit.