Assuming positions are stored on the car, not the transponders.

Assumes angle returned is between 0 and 360.

Import numpy as np;

Transponders

= [[id, pos x, pos y], [id, pos x, pos y], [id, pos x, pos y], …];

inRange = [];

Intersections = [];

myPos = [0,0];

got\_Response(id, angle)

{

inRange.add(pos);

findPos();

}

findPos();

{

Int i = 0;

While (i < inRange.length - 1)

{

Int j = i + 1;

While (j < inRange.length - 1)

{

If (inRange[i].angle != inRange[j].angle && inRange[i].angle !=

inRange[j].angle + 180) //Make sure their not parallel/antiparallel

{

a1 = tan^(-1)(inRange[i].angle);

a2 = tan^(-1)(inRange[j].angle);

b1 = inRange[i].pos\_y - a1\*inRange[i].pos\_x;

b2 = inRange[j].pos\_y - a2\*inRange[j].pos\_x;

A = np.matrix([a1, -1], [a2, -1]);

B = np.matrix([-1\*b1], [-1\*b2]);

X = A.getI()\*B

intersections.add(X);

}

}

}

x\_sum;

y\_sum;

//This is just averaging the intersections found, another method may be better but this works for //now.

Foreach point in intersections

{

x\_sum += point.item(0);

y\_sum += point.item(1);

}

myPos[0] = x\_sum / point.length();

myPos[0] = y\_sum / point.length();

}