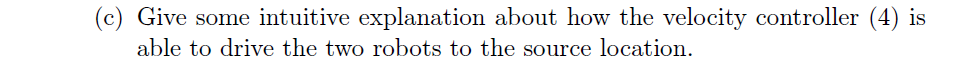
The change in direction is perpendicular to the line segment connecting the robots locations. Assume the vector representing the line between robots is as given above. The magnitude is non linear so we must linearize it around the starting value of one.

In order to get the direction vector for the robots we will represent v =

Where

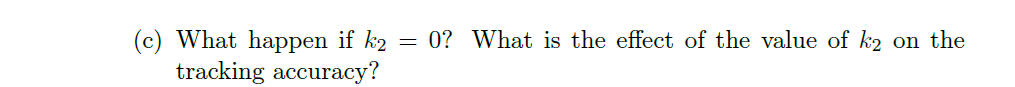
Simplified

And

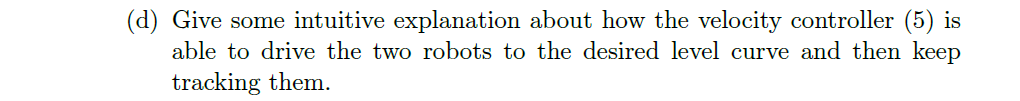


The controller works by treating the two robots similarly to the front tires of a car with differential steering. While the two robots are not at the target, the controller tells the one furthest away to move faster than the other. The end result of that command is the two robots moving in unison along a near circular pattern and pointing more towards the target. This eventually leads to the robots moving towards the target and arriving.

**2**

****

If then the robots would stop moving if they ever actually reach the level curve will cause the tracking to drift slightly during the initial approach, but also ensures that the robots track the level curve and don't get stuck.

****

The first part of the control equation drives the robots towards the level curve by moving the robots furthest away at a higher velocity than the closer one thereby causing the robots to "face" the level curve. The second term in the equation is overshadowed by the first at a large distance, but causes the robots to move tangent to the level curve when close and facing the level curve. When moving tangent moves the robots off the level curve, then the first term will move the robots closer proportional to the distance they are away from the curve.