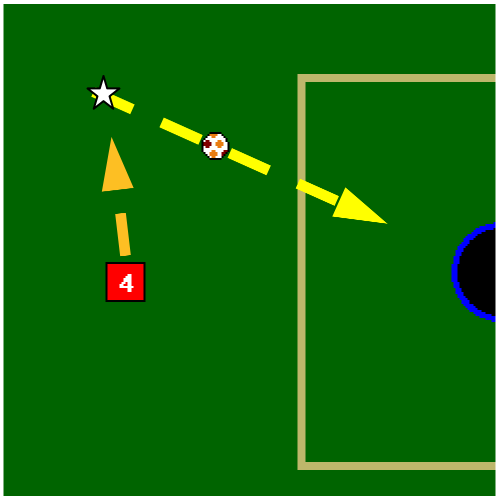
Robot Math



So, you want to hit the ball into the goal?

At a “high level”, we need to line up behind the ball, and then kick it straight in a line toward the goal. Right?

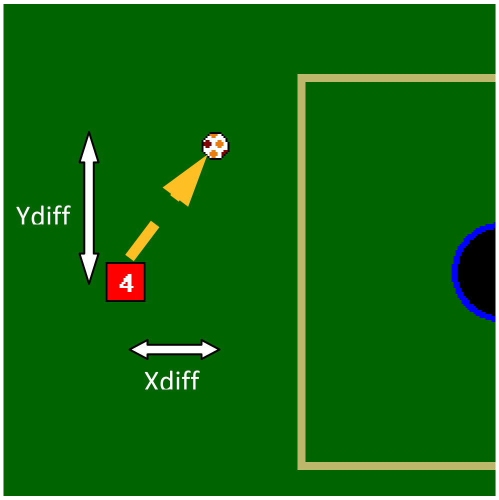
So, we need two things.

1) Where do I need to go before the kick? <u>and</u>

2) What direction do I have to travel to get to a specific point?

You have 6 pieces of information that you can use

- The x location of the ball (X_{ball})
- The y location of the ball (Y_{ball})
- The x location of the goal (X_{goal})
- The y location of the goal (Y_{goal})
- The x location of the robot (X_{robot})
- The y location of the robot (Y_{robot})



“What direction do I have to travel to get to a specific point?”

Ok. You know where the robot is, and you know where you want to go. How do you get there? We’ll since we control the two motors independently. All we have to do is make sure we follow the line. For that, we need to what proportion of our speed is associated with.

1) First, let’s compute the differences in X and Y between us and our goal. (In the picture, the ball’s location is the desired location)

$$x_{diff} = x_{desired} - x_{robot}$$

$$y_{diff} = y_{desired} - y_{robot}$$

2) Recall that we can use the **Pythagorean Theorem** to compute the total distance

$$c^2 = a^2 + b^2$$

so

$$c = \sqrt{a^2 + b^2}$$

Applying our own names for things (and using distance as “c”), we get…

$$distance = \sqrt{(x_{diff})^2 + (y_{diff})^2}$$

BTW. Math.Sqrt() is a good function that might help here.   You can use the caret (^) for exponentiation, but multiplying each item by itself (e.g xdiff * xdiff) works too.

3) Now that we know distance, we can figure out “how much” of the distance is associated with each difference. Let’s call them x_{ratio} and y_{ratio}.

$$x_{ratio} = \frac{x_{diff}}{distance}$$

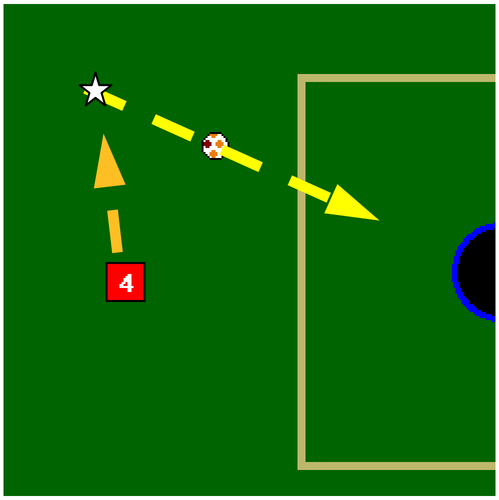
$$y_{ratio} = \frac{y_{diff}}{distance}$$

The two ratios will be in the range of -1 to 1. It’s perfectly fine for them to be negative, indicating a left or up direction.

4) Finally, compute the speed based on the two ratios.   For this, just multiply each by a speed number (for starters, just make speed = 10)

$$x_{speed} = x_{ratio} \cdot speed$$

$$y_{speed} = y_{ratio} \cdot speed$$



"Where do I need to go before the kick?"

1) Recall that the **equation of a line** is…

$$y = mx + b$$

2) First, compute the slope of the line ("m"). If you have two points on a line, you can compute the slope by the following…

$$m = \frac{difference\ of\ y}{difference\ of\ x}$$

So, for us, it's

$$m = \frac{y_{goal} - y_{ball}}{x_{goal} - x_{ball}}$$

3) Next, compute the y-intercept ("b")

$$b = y - mx$$

Since both the ball and the goal are on the same line, we can use either location. So let's use the ball's location.

$$b = y_{ball} - m\, x_{ball}$$

4) Now we can use "m" and "b" to compute a third location that is also on the same line (let's call this the desired location [aka where we want the robot to go]).

We will choose the x location a little arbitrarily by choosing an offset from the ball of 25 units away from the ball. If we are travelling to the right, we want to choose a value <u>less</u> than the ball. If we are travelling to the left, then we choose a value <u>more</u> than the ball. So it's ± 25. (An "if" statement would be good to make this decision).

$$x_{desired} = x_{ball} \pm 25$$

Next, compute the y location to be on the line with the right slope and y-intercept.

$$y_{desired} = m\, x_{desired} + b$$

That's it. You can now send your robot to that location. To make your robot hit the ball toward the goal once you are on the line, just change the desired location to the goal and run towards it. (Note: this is less simply put when the ball is in motion).