ME597 Assignment 1

# Motion, Measurement and Estimation

Abdul Qureshi 20388761

ChengLin Yeh 20369072

Oct 26th, 2014

# 

# **Introduction**

The purpose of this lab was to study three of fundamental components of autonomous robot, motion, measurement and estimation. First, a motion model of a three-wheeled omnidirectional model was derived. Next, sensor models for GPS and magnetometer were defined. Finally, Extending Kalman filter and multi-rate Kalman filter was used to combine the measurements from the sensors and measurements from the robot itself to estimate robot’s correct location.

# 1 Motion modeling of omnidirectional wheeled robot

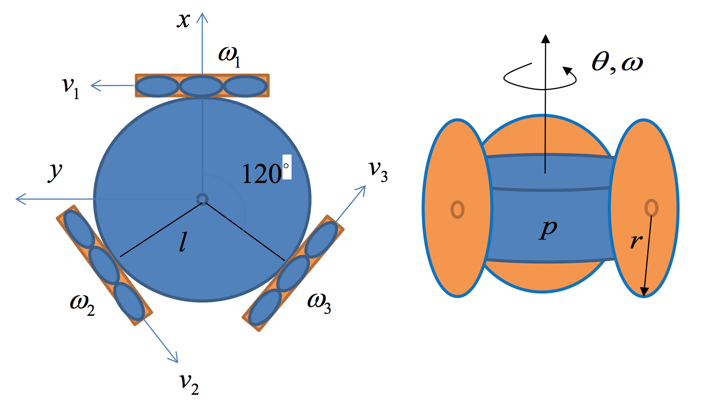


Figure 1 – Omnidirectional wheeled robot

The velocity for the robot can be expressed as:

To derive the motion model for the omnidirectional wheeled robot, we first look at the velocity decomposition of each wheel.

Next, the velocity found above is in the robot’s frame, thus, we need to convert it into the world frame.

# 2 Simulation of the Robot

Given the input , the following figure is what the output looks like over 15s with 10Hz update.

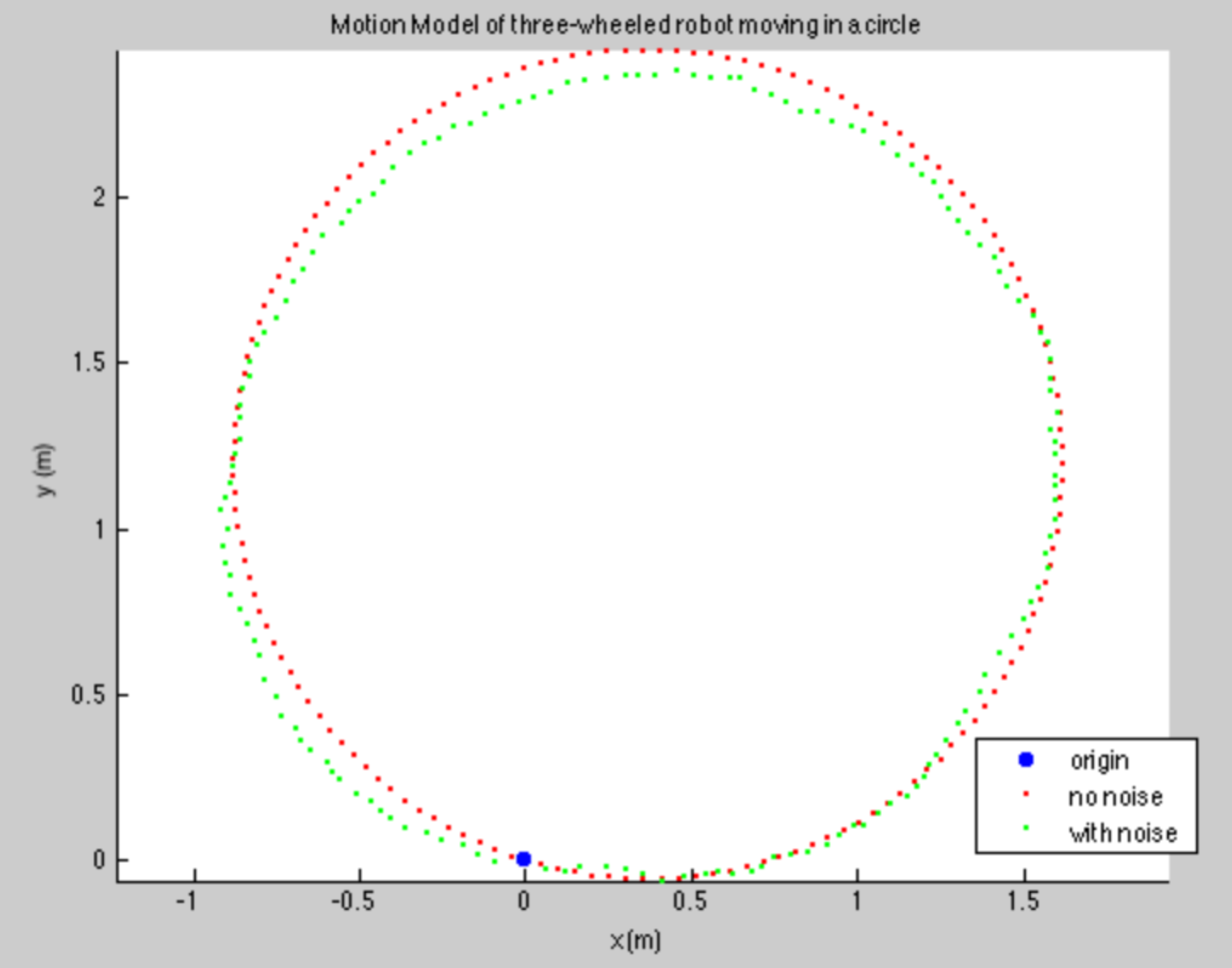


Figure 2 – Omnidirectional wheeled robot simulation given inputs

For the robot to move in a straight line, solve for the following

or

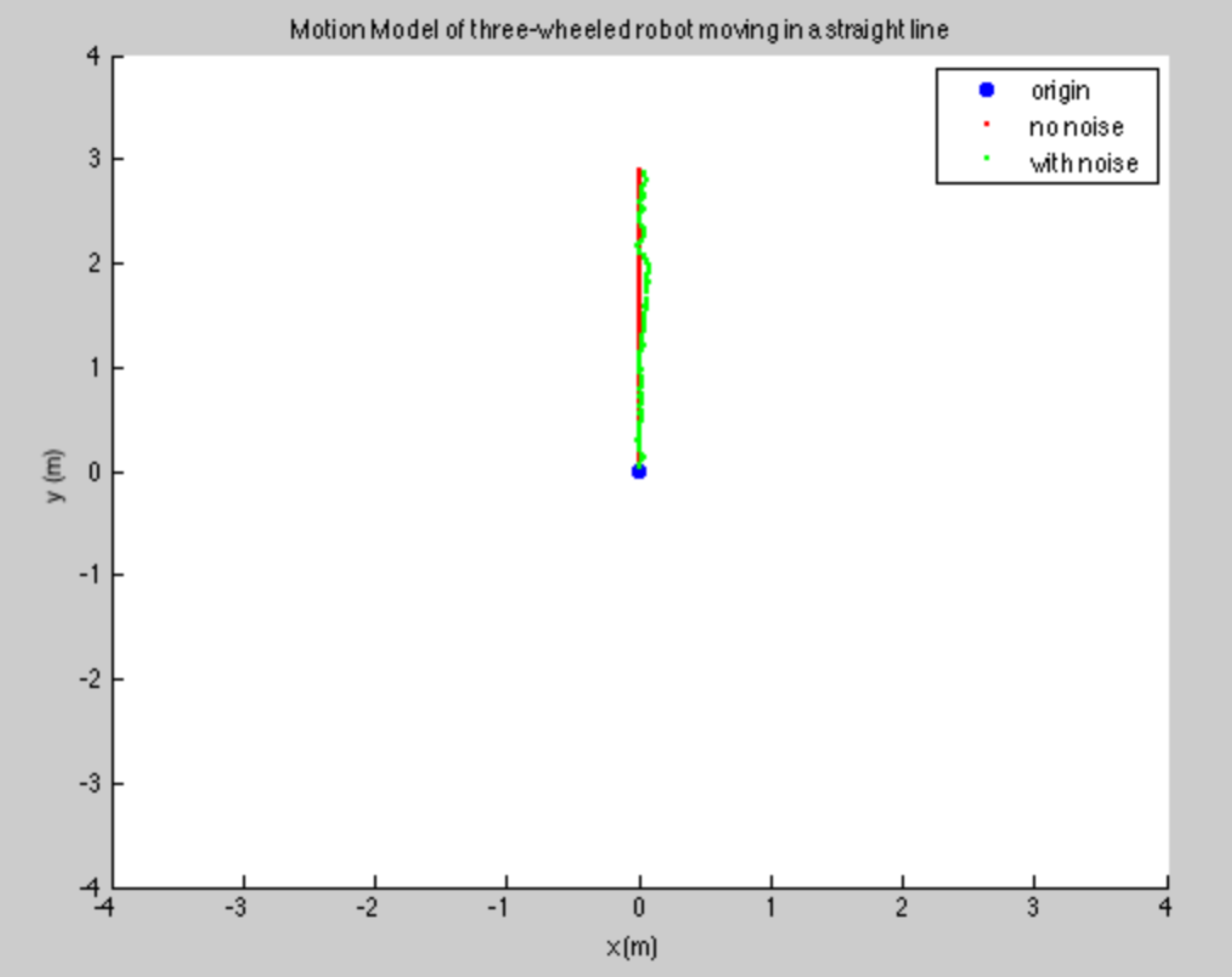
the result of solving these equation is

, to move in the x direction

or

, to move in the y direction

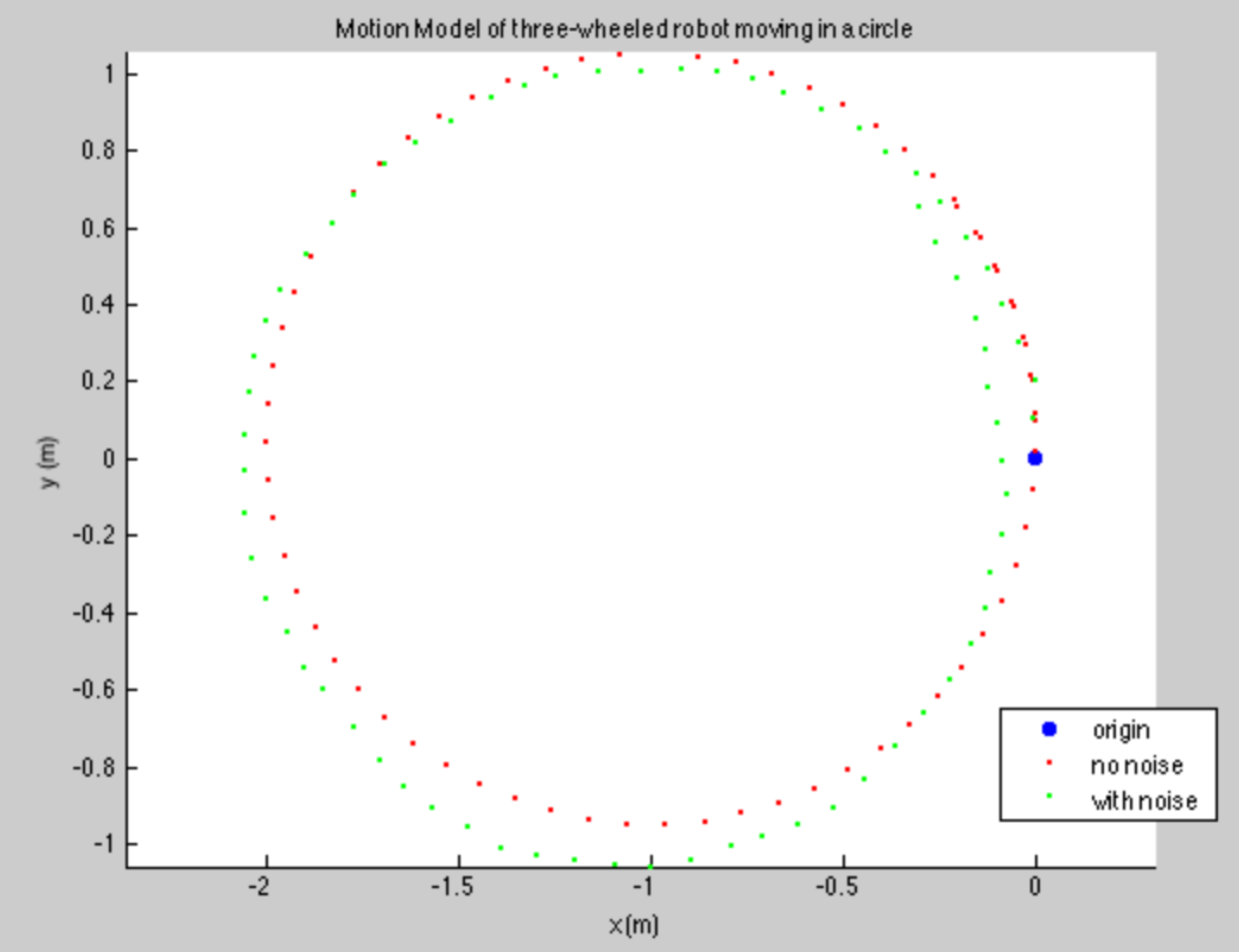
The result of this calculation is shown in the below figure



To move in a circle with radius of r,

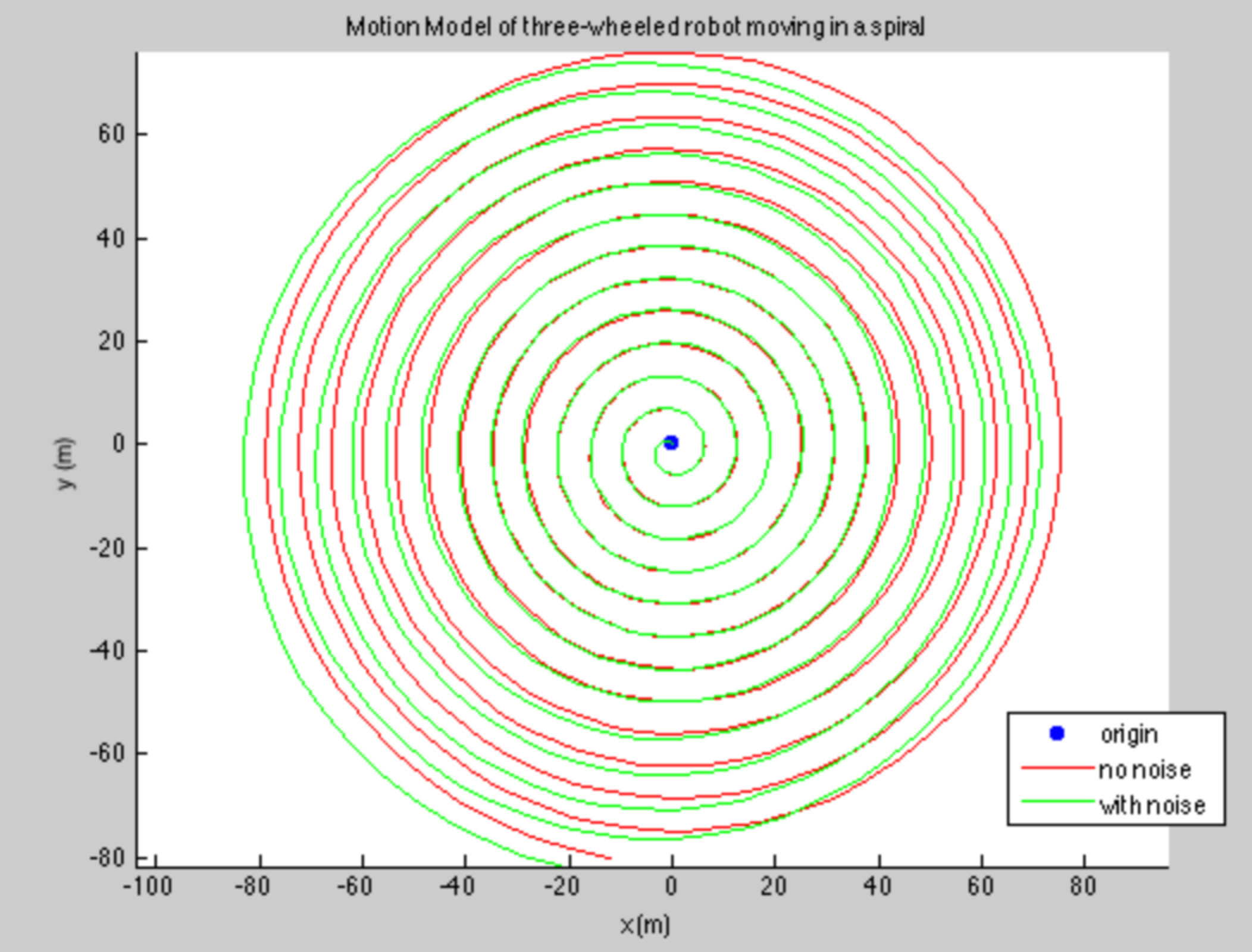
For this specific question r = 1, for the sake of simplicity, set y = 0

The result is shown in the below figure



To move in a spiral, constant velocity for the wheels will not suffice. To solve for a spiral, fix angular velocity at one and keep y = 0 but set x to a function z = x\*t

The result is in the following figure



# 3 Measurement Model

The measurement model for the robot will use GPS and a magnetometer on a 2D plane. The GPS output will have error in the form of an additive gaussian distribution with standard deviation of 0.50m in both the north and east direction. The final measurement, yg will be as follows:

Yg = CXg + dg

The matrix C represents the covariance between the different measurements, in this case just an identity matrix, and dg = N(0, 0.5).

The magnetometer also has additive gaussian noise of 10 degrees north, with the model presented as follows:

Ym = CXm + dm, dm = N(0, )

Represented as one model, the measurements can be represented as:

,

# 4 Extended Kalman Filter

The extended Kalman filter is simply the non-linear version of the Kalman filter where the measurement and motion model are linearized. The states vector that will be used to model the robot is as follows:

The linearized motion model is represented as follows:

where the motion equations are broken down by the direction in the inertial frame, x,y and with respect to the angular velocities of the wheels, Since the state vector was picked to have and the position can be represented using this information as well as the previous state, the current position can be achieved without any further linearization. The prediction update equations for the mean and covariance are then represented by:

Since the measurement model is already linear for the specific problem, it can be represented as:

The measurement update equations can then be written as:

The update equations were then used with the rotation inputs of w1 = -1.5 rad/s, w2 = 2.0 rad/s and w3 = 1.0 rad/s over a 15 second simulation. The measured and estimated positions of the robot in the x, y directions are illustrated in the figures below, with the estimate shown in red and the measurements in blue. The x and y position correct themselves after slight divergences at the start and the theta positon is pretty accurately predicted.

|  |  |  |
| --- | --- | --- |
|  |  |  |

The measured and estimated motion of the robot is illustrated in the figures directly below. As can be seen, the angular velocities of all three wheels converge back to the measured values over time.

|  |  |  |
| --- | --- | --- |
|  |  |  |

The error ellipses were screen captured at multiple points while the simulation was running over a 50 second interval. The illustrations are shown chronologically below. The prediction was set to a value far off of the actual position to start but quickly corrected itself. Immediately after this, in the third image, the prediction began to diverge a bit but quickly converged back to the actual measurements and after multiple circles, as seen in the last image, the prediction followed actual measurements fairly closely.

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |