ECSE/CSDS 376/476
Problem Set 6: Odometry Calibration
Assigned: 3/10/21
Due: 3/17/21

This assignment is a **group** assignment. It is *in addition to* the remote lab assignments.

In PS3, you analyzed GPS records and Canbus steering values to calibrate steering. A result is that, to a good approximation, path curvature is proportional to the published steering value. In this assignment, you should extend your results to computing odometry based on shaft encoder values and steering angle values.

The file slalom_data.csv contains samples of GPS, steering and encoder values. You should find a conversion factor that relates steering angle to path curvature and a second scale factor that relates encoder values to vehicle translation. Starter code, localization_analysis.m, is provided. This loads the logged data and plots out the x-y path according to GPS coordinates.

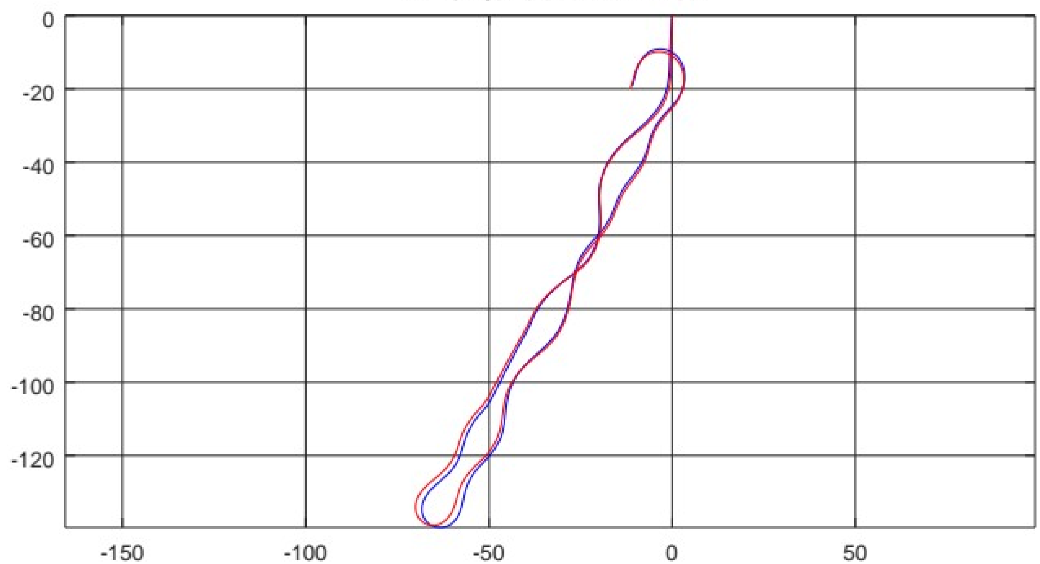
You should compute the corresponding x-y path using odometry (steering and shaft encoder) data only. Plot your odometry results overlayed on the GPS results.

You can initialize the odometry position to be identical to the initial GPS position. You will need to provide an estimate of initial heading.

My own initial attempt is shown below. Perhaps you can improved on it.

In your report, show your result graphically (like the plot below). Specify the scale factors you used. Provide a link to your code. Include a discussion with theory of operation along with any observations you can offer.

x vs y: gps(b) and odom(r)



1) Curvature calibration:
This assignment includes multiple CSV files. These were obtained with an instrumented Ford F-550 using a GPS unit with differential corrections, a shaft-mounted encoder, and readings of steering angle from the vehicles CAN-bus.

The files "circle1.csv" through "circle15.csv" contain logs of latitude (first column), longitude (second column) and a value monotonically related to steering angle (third column). For these files, you can ignore the remaining columns.

Define a coordinate system that has the x-axis pointing East, the y-axis pointing North, and heading defined as 0 rad pointing East with positive rotation pointing "up". (e.g., heading North is +pi/2 radians).

It will be convenient to analyze the data from an initial reference point of (x,y) = (0,0), i.e. by subtracting off the initial lat/lon.

For motions of lattitude, there is (to a good approximation) a constant that relates meters/degree latitude. However, the conversion of meters/degree longitude depends on the the latitude. Assume a latitude of 41.5 deg North. Find the conversion constants.

In Matlab or Octave, you can access this data, e.g., using:

```
data = load("circle15.csv");
lat_start = data(1,1)  %for convenience, compute x,y motions relative to starting location
lon_start = data(1,2)
%convert lat/lon to x,y w/rt starting position
%magic numbers to convert deg lat and deg lon to dy and dx
%NOTE: GPS calls longitude -81deg, not 81deg "West"
%so longitude gets more positive heading east, = x direction
%latitude gets more positive heading north, = y direction
y_vec = (data(:,1)-lat_start)*lat_to_meters;
x_vec = (data(:,2)-lon_start)*lon_to_meters_at_41degN;
```

For the files circle1.csv through circle15.csv, the steering angle was deliberately held approximately constant as the vehicle moved in circles while logging GPS data.

The vehicle's on-board diagnostic (OBD) port provides a digital value related to the steering angle, values of which appear in column 3 of the logs.

From the circular data, construct a function that maps steering value onto path curvature. Plot out your results for this function, curvature as a function of steering-angle value.

2): Lane-Drift Controller
As we discussed in class, analyze the response of a lane-drift controller using a linear control algorithm. Assume you are able to command a curvature, rho(lateral_offset_err, heading_err), and this control algorithm will be formulated as:
rho = -K_offset * d_offset -K_psi*psi

Note that both rho and d_offset are signed. Drifting into the left lane is defined as positive offset, and rotating counterclockwise is considered positive curvature.

The vehicle dynamics is described as:
v_x = speed * cos(psi)
v_y = speed * sin(psi)
omega_z = speed *rho (= d psi/dt)

Choose values for K_psi and K_offset. Choose values for initial offset and heading errors. Simulate (and plot): x(t), y(t), psi(t) and the path x vs y.

Describe influences of initial conditions and recommendations for controller gains.