MECH 6323 - Final Project

Author: Jonas Wagner

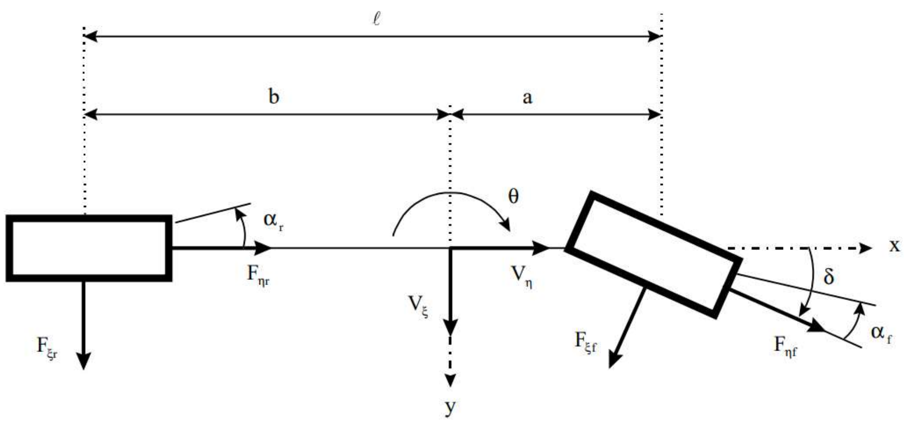
Date: 2022-05-13

Intro

Abstract

In this project a model of dynamics of the Autonomous Vehicle (NOVA) will be examined. The NOVA team aims to eventually develop a more fundemental theory-based control strategy, yet there still exists many uncertain and changing parameters that determine the dynamics of the system. In this project a tool to analyze these dynamics, and specifically how we can look at the nonlinearities of the actuators, uncertain physical parameters, and sensor noise/malicious attack will effect the dynamics of the system.

Nominal Plant Definition



$$
\begin{aligned}
I\ddot{\theta} &= aF_{\eta f}\delta + bF_{\zeta f} - bF_{\zeta r} \\
m(\dot{V}_\zeta + V_\eta\dot{\theta}) &= F_{\eta f}\delta + F_{\zeta f} + F_{\zeta r} \\
m(\dot{V}_\eta + V_\zeta\dot{\theta}) &= F_{\eta f} + F_{\eta r} + F_{\zeta f}\delta \\
\dot{x} &= -V_\zeta \sin(\theta) + V_\eta \cos(\theta) \\
\dot{y} &= V_\eta \cos(\theta) + V_\zeta \sin(\theta)
\end{aligned}
$$

$$\dot{x} = -V_\xi \sin\theta + V_\eta \cos\theta$$

$$\dot{y} = V_\xi \cos\theta + V_\eta \sin\theta$$

$F_{\xi f}$, $F_{\xi r}$ = lateral tire force on front and rear tires

a = vehicle center of gravity location from front axle

b = vehicle center of gravity location from rear axle

m = vehicle mass

I = vehicle yaw inertia

P_f, P_r = longitudinal force on front and rear tires

V_η,V_ξ = longitudinal/lateral velocity of vehicle in body reference frame

x, y = longitudinal/lateral position of vehicle in inertial reference frame

δ = steering angle

η = longitudinal axis in body reference frame

θ = yaw angle

ξ = lateral axis in body reference frame

$$x = [\dot{\theta}, V_\zeta, V_\eta, x, y, \theta]^T$$

$$u = [\delta, F_{\eta f}, F_{\eta r}, F_{\zeta f}, F_{\zeta r}]^T$$

x_sym =

$$\begin{bmatrix} \dot{\theta} \\ V_\zeta \\ V_\eta \\ x \\ y \\ \theta \end{bmatrix}$$

u_sym =

$$\begin{bmatrix} \delta \\ F_{\zeta,f} \\ F_{\zeta,r} \\ F_{\eta,f} \\ F_{\eta,r} \end{bmatrix}$$

$$\dot{x}_1 = \frac{\left(aP_f\delta + bF_{\xi f} - bF_{\xi r}\right)}{I}$$

$$\dot{x}_2 = \frac{\left(P_f\delta + F_{\xi f} + F_{\xi r}\right)}{m} - x_3 x_1$$

$$\dot{x}_3 = \frac{\left(P_f + P_r - F_{\xi f}\delta\right)}{m} - x_2 x_1$$

$$\dot{x}_4 = -x_2 \sin x_6 + x_3 \cos x_6$$

$$\dot{x}_5 = x_2 \cos x_6 + x_3 \cos x_6$$

$$\dot{x}_6 = x_1$$

$x_1 = \dot{\theta}$	yaw rate
$x_2 = V_\xi$	lateral velocity
$x_3 = V_\eta$	longitudinal velocity
$x_4 = x$	longitudinal position with respect to fixed reference
$x_5 = y$	lateral position with respece to fixed reference
$x_6 = \theta$	yaw angle

Note that $F_{\eta f} = P_f$ and $F_{\eta r} = P_r$ when in an ideal plant model... effected by slipage

```
bike_nonlin_dot_sym =
```

$$\begin{bmatrix} \dfrac{F_{\eta,f}\, b - F_{\eta,r}\, b + F_{\zeta,f}\, a\, \delta}{I} \\ \dfrac{F_{\eta,f} + F_{\eta,r} + F_{\zeta,f}\, \delta}{m} - V_\eta\, \dot{\theta} \\ \dfrac{F_{\zeta,f} + F_{\zeta,r} - F_{\eta,r}\, \delta}{m} - V_\zeta\, \dot{\theta} \\ V_\eta \cos(\theta) - V_\zeta \sin(\theta) \\ V_\eta \cos(\theta) + V_\zeta \cos(\theta) \\ \dot{\theta} \end{bmatrix}$$

Nominal Model

```
bike_nonlin_dot_nom =
```

$$\begin{bmatrix} 0.0013\, F_{\eta,f} - 0.0013\, F_{\eta,r} + 0.0010\, F_{\zeta,f}\, \delta \\ 6.6667\text{e-}05\, F_{\eta,f} + 6.6667\text{e-}05\, F_{\eta,r} + 6.6667\text{e-}05\, F_{\zeta,f}\, \delta - V_\eta\, \dot{\theta} \\ 6.6667\text{e-}05\, F_{\zeta,f} + 6.6667\text{e-}05\, F_{\zeta,r} - 6.6667\text{e-}05\, F_{\eta,r}\, \delta - V_\zeta\, \dot{\theta} \\ V_\eta \cos(\theta) - V_\zeta \sin(\theta) \\ V_\eta \cos(\theta) + V_\zeta \cos(\theta) \\ \dot{\theta} \end{bmatrix}$$

Linearized Model

This is an unrealistic thing to do outside of single time-steps

```
A_sym =
```

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ -V_\eta & 0 & -\dot{\theta} & 0 & 0 & 0 \\ -V_\zeta & -\dot{\theta} & 0 & 0 & 0 & 0 \\ 0 & -\sin(\theta) & \cos(\theta) & 0 & 0 & -V_\zeta \cos(\theta) - V_\eta \sin(\theta) \\ 0 & \cos(\theta) & \cos(\theta) & 0 & 0 & -V_\eta \sin(\theta) - V_\zeta \sin(\theta) \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

```
B_sym =
```

$$\begin{bmatrix} \frac{F_{\zeta,f}\, a}{\mathrm{I}} & \frac{a\,\delta}{\mathrm{I}} & 0 & \frac{b}{\mathrm{I}} & -\frac{b}{\mathrm{I}} \\ \frac{F_{\zeta,f}}{m} & \frac{\delta}{m} & 0 & \frac{1}{m} & \frac{1}{m} \\ -\frac{F_{\eta,r}}{m} & \frac{1}{m} & \frac{1}{m} & 0 & -\frac{\delta}{m} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Uncertain Plant Dynamics and Disturbances

All the calculations here are dependent on the nominal dynamics.

```
params = struct with fields:
      a: 1.2280
      b: 1.5618
      m: 15000
      I: 1200
    c_f: 0.1200
    c_r: 0.1600
```

Physical Disturbances

Tire Slipage

We have options to model the tire slipage better (actually what alphas are used for) but they are highly nonlinear and are also highly dependent on so it is very difficult to model acurately. Instead we will model tire with a disturbance input and have it be modeled a disturbance within the for each tire. For simplicity we will make it a parameter that is really only dependent on the unchanging parameters since these are what effect this in real life.

Actually... since idk what I am actually talking about for vehicle dynamics, we can going to refer back to what Adi had done for the optimization and use the less specific (just that it is not derived from the parameters directly, but is just done by a single that is tuned per the vehicle operation at the specific weight/configuration/etc.

Wind Resistance

Although the NOVA vehcile will be going slow enough that wind negligence is negligable, including it seemed worth while. The effect of wind resistance itself is modeled as two components.

First, the force due to movement through air proportional to speed is added in the system directly opisite the currect direction of movement. This is inacurrate in general due to car body shape but this could also be modified to be different weighting from different directions but we will instead just provide for this parameter to vary greatly.

$F_{\eta,w} = k_{\eta,w} \cdot V_\eta$

$F_{\zeta,w} = k_{\zeta,w} \cdot V_\zeta$

Secound an additive noise of general stead-state wind either fights against or assists the vehicle. Since wind direction generally changes slowly we will only be looking at lower frequencies.

$F_{ws} = W_{ws} \cdot z_{ws} \cdot (cos(\theta - \theta_{ws})\hat{u}_\eta + sin(\theta - \theta_{ws})\hat{u}_\zeta),\ \ z_{ws} \in [0, 1],\ \ \theta_{ws} \in [0, 2pi]$

$\theta_{ws} = W_{w,\theta_{ws}} z_{w,\theta_{ws}}$

```
W_ws =

     -0.1
  --------
  (s-0.01)

Continuous-time zero/pole/gain model.

W_ws_theta =

  0.062832
  --------
  (s-0.01)

Continuous-time zero/pole/gain model.
```

both a multiplicative noise on the standard are resistance of dependent on the vehicle velocity.

Currently this is not implimented in the simulation fully. Instead the following more simplistic method is used:

$F_{\eta,r} = -k_{ws,\eta} * V_\eta$

Since wind resistance is an over complication we are eliminating it for this time.

Uncertain/Nonlinear Actuators

Steering and Forward Force Delays

Both a human and autonomous driver takes a while to react, while both the steering mechanism and engine/drivetrain also contain delays. We also assume it is only a front-wheel drive vehicle, so only the front wheel will be providing force to move the vehicle.

$\delta = W_{driver} \cdot u_\delta$

```
W_driver =

    10
  ------
  (s-10)

Continuous-time zero/pole/gain model.
```

$F_{eta,f} = W_{drivetrain} \cdot W_{driver} \cdot u_{gas}$

```
W_drivetrain =

      5
  -------
  (s-0.5)

Continuous-time zero/pole/gain model.
```

Slip angle effects

The front slip angle is calculated as:

$$\alpha_f = \delta - \frac{a\dot{\theta} + V_\zeta}{V_\eta}$$

and the rear slip angle is

$$\alpha_r = \frac{b\dot{\theta} - V_\zeta}{V_\eta}$$

Then we have (ignoring saturation regions... remaining linear for this analysis)

$F_{\zeta,f} = c_f \alpha_f$

and

$F_{\zeta,r} = c_r \alpha_r$

where c_f and c_r are the cornering stifnesses uncertain parameters.

Sensors and Measurement Noise

Gyroscope

The gyroscope will measure the rotational velocity of the system which will allow an observer to measure the current state with fairly high acuracy if you average it (which isn't really that useful when you would need to integrate or differentiate them) but will be noisy at higher frequencies if sampled often.

The gyro measurment output will therefore be:

$y_{gyro} = \dot{\theta} + W_{gyro} \cdot z_{gyro},\ \ z_{gyro} \in [-1, 1]$

```
W_gyro =

  0.0001 (s-2)
  ------------
    (s-5e05)

Continuous-time zero/pole/gain model.
```

Spedometer

The spedometer is just a measurement directly of the Longitudinal Velocity, V_η, and will have a measurement noise that is only really aparent at higher frequencies.

The gyro measurment output will therefore be:

$y_{sped} = V_\eta + W_{sped} \cdot z_{sped},\ \ z_{sped} \in [-1, 1]$

```
W_sped =

  0.1 (s-10)
  ----------
   (s-1000)

Continuous-time zero/pole/gain model.
```

GPS

The GPS system will be used to calculate the global position over a longer period of time for the system. Similar to the Gyroscope, the GPS will have error at low ends and isn't going to ever truely be acurate over-time in a real-world situation by itself without other information used by a Kalmen Filter or similar in congunction with other sensors (such as Lidar) and orther localization techniques (which is a big part of what NOVA has been working on this semester). We will assume that the sensor error output from the gps will be white noise that is filtered out mostly (becouse of the way it is averaged and sampled so higher frequencies become less important) except for a region.

$$y_{gps,x} = x + W_{gps} \cdot z_{gps}, \;\; z_{gps} \in [-1, 1]$$

$$y_{gps,y} = y + W_{gps} \cdot z_{gps}, \;\; z_{gps} \in [-1, 1]$$

```
W_gps =

         10
  -----------------
  s^2 + 100 s + 100

Continuous-time transfer function.
```

Eyeballs (Vison & Loclaization)

The big thing that isn't included in this entire thing is the actual vision that the vehicle has. Generally this is almost a cheat catch-all of everything since although the sensors will have higher accuracy and ultimently a faster response time, it is the vision that will make things consistant. Similarly to the 'driver-delay', we will have a sensing type delay, but all of the information we are able to be awar of is available (although we will have considerably more noise.

$$y_{vision} = y + W_{vision} \cdot z_{vision}, \;\; z_{vision} \in [-1, 1]$$

Parameteric Uncertainty

Parameteric Uncertainty essentially just assumes deviations from potential modeling issues. For the current analysis this is ignored in order to focus on actuator uncertainty.

Actually, I ended up deciding to do just that... but just with a more standard thing.

Plant Model and Controller Design

Nominal Plant

```
sys_nom =

  A =
           x1      x2      x3      x4      x5      x6
   x1       0       0       0       0       0       0
   x2      -1       0     0.1       0       0       0
   x3    -0.1     0.1       0       0       0       0
```

```
 x4        0  -0.7078   0.7064        0        0  -0.7785
 x5        0   0.7064   0.7064        0        0  -0.7786
 x6        1        0        0        0        0        0

B =
            u1          u2          u3          u4          u5
 x1    0.001023   0.0001023           0    0.001302   -0.001302
 x2   6.667e-05   6.667e-06           0   6.667e-05   6.667e-05
```

Uncertain Plant

The uncertain plant with all the dynamics is actually really complicated and not only is dependent on the changing parameters and nonlinearities due to actuators; but is also really complicated with conections to the global coordnate system. We therefore can develop a local plant that incorporates less nonlinearities while being tracked seperatly in the global reference frame. The controller that works for this would not need to worry about that particular nonlinearity.

```
sys_plant_u_local =

  Uncertain continuous-time state-space model with 4 outputs, 5 inputs, 4 states.
  The model uncertainty consists of the following blocks:
    I: Uncertain real, nominal = 1.2e+03, variability = [-1,1]%, 2 occurrences
    a: Uncertain real, nominal = 1.23, variability = [-1,1]%, 1 occurrences
    b: Uncertain real, nominal = 1.56, variability = [-1,1]%, 1 occurrences
    m: Uncertain real, nominal = 1.5e+04, variability = [-1,1]%, 2 occurrences

Type "sys_plant_u_local.NominalValue" to see the nominal value, "get(sys_plant_u_local)" to see al
```

Simulink Modeling

Plant Modeling

(see the end for the simulink model printouts)

Plant Internal Diagram

The plant internal diagram includes the orignal physical dynamics within a standard LTI system block and all of the weighting functions for the actuators and sensors containing the uncertainty.

plant_bike_lin.slx

The plant is then agrigated within another subsystem block that is structured as a generalized plant model to be used for designing the -controller.

fast_test.slx

Becouse of many issues with the complicated model, a simpler system that implimented a less restrictive actuator system (and ignores the wind disturbances which made stabilization nearly impossible) a simpler model (shown below) was created instead.

Generalized Linear Plant Model:

```
sys_plant =

  Uncertain continuous-time state-space model with 22 outputs, 14 inputs, 20 states.
  The model uncertainty consists of the following blocks:
    I: Uncertain real, nominal = 1.2e+03, variability = [-1,1]%, 2 occurrences
    a: Uncertain real, nominal = 1.23, variability = [-1,1]%, 1 occurrences
```

```
    b: Uncertain real, nominal = 1.56, variability = [-1,1]%, 1 occurrences
    m: Uncertain real, nominal = 1.5e+04, variability = [-1,1]%, 2 occurrences

Type "sys_plant.NominalValue" to see the nominal value, "get(sys_plant)" to see all properties, an
```

H-infty Controller Design

The rank of our controllability is needed to ensure that the system is actually controllable.

```
ctrb_rank = 7
```

In many different equalibrium cases the rank of controllability is 1. In other words, the system had many uncontrollable modes that lead to a complete failure to even have a controllable (let alone robust) system. This is what prompted the change to instead

```
State-space model with 5 outputs, 16 inputs, and 20 states.
info =
  hinfINFO with properties:

    gamma: 1.0100
        X: [20×20 double]
        Y: [20×20 double]
       Ku: [5×20 double]
       Kw: [25×20 double]
       Lx: [20×16 double]
       Lu: [5×16 double]
     Preg: [27×30 ss]
       AS: [21×21 ss]
```

A controller was able to be developed, for some very specificic linearizations,

Uncertain Plant Analysis

```
L =

  Uncertain continuous-time state-space model with 5 outputs, 5 inputs, 40 states.
  The model uncertainty consists of the following blocks:
    I: Uncertain real, nominal = 1.2e+03, variability = [-1,1]%, 2 occurrences
    a: Uncertain real, nominal = 1.23, variability = [-1,1]%, 1 occurrences
    b: Uncertain real, nominal = 1.56, variability = [-1,1]%, 1 occurrences

Type "L.NominalValue" to see the nominal value, "get(L)" to see all properties, and "L.Uncertainty
```

It is worth noting that the controller, even in the open loop configuration, was able to eliminate dependence on both m and c_f.

ans = 5×1 struct

Fields	GainMargin	PhaseMargin	DiskMargin	LowerBound	UpperBou
1	[1,1]	[0,0]	0	0	
2	[1,1]	[0,0]	0	0	
3	[1,1]	[0,0]	0	0	

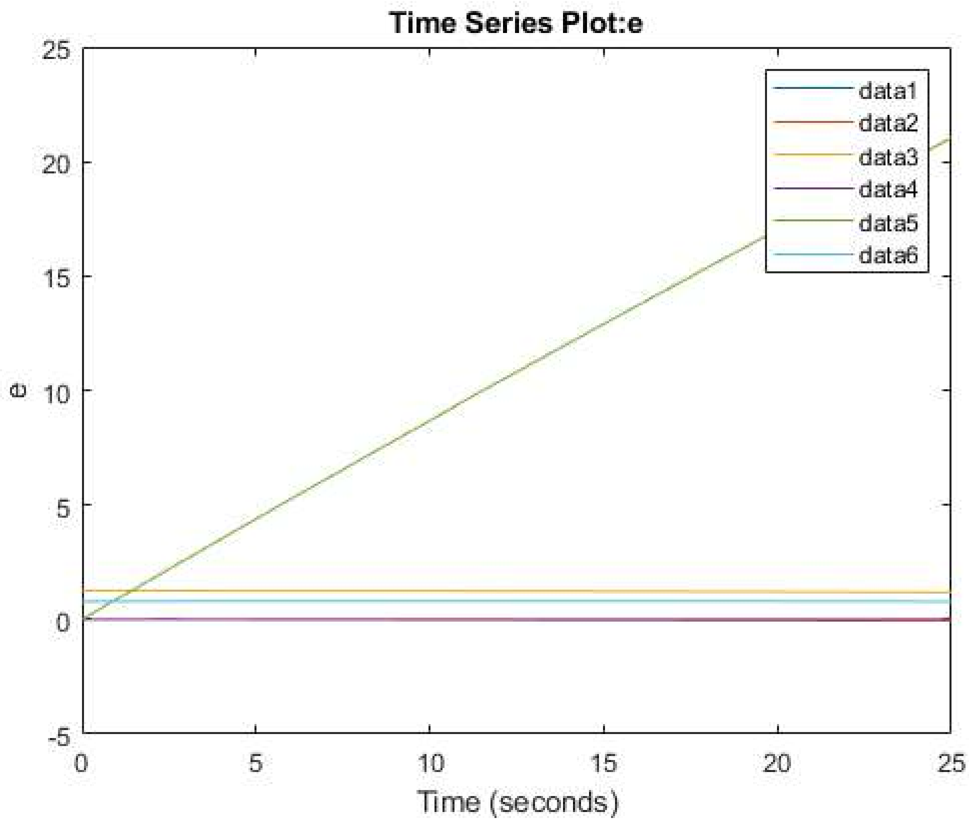
Fields	GainMargin	PhaseMargin	DiskMargin	LowerBound	UpperBou
4	[1,1]	[0,0]	0	0	
5	[1,1]	[0,0]	0	0	

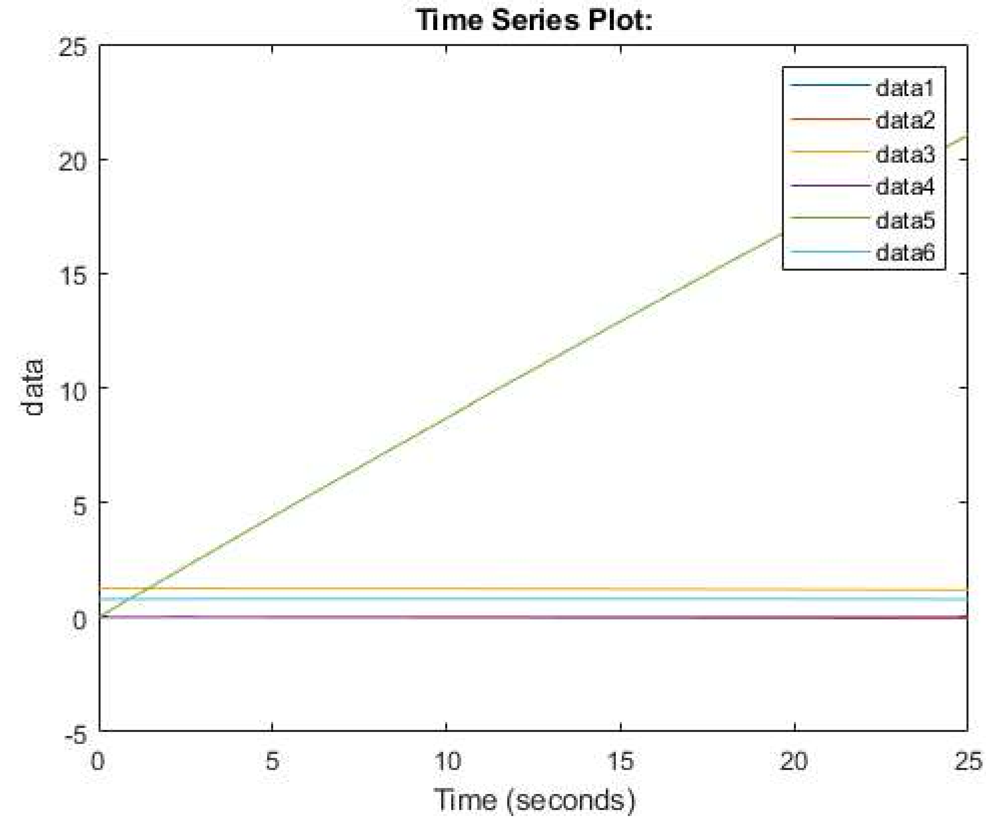
This result shows us that the system is robust at all

Testing Controller

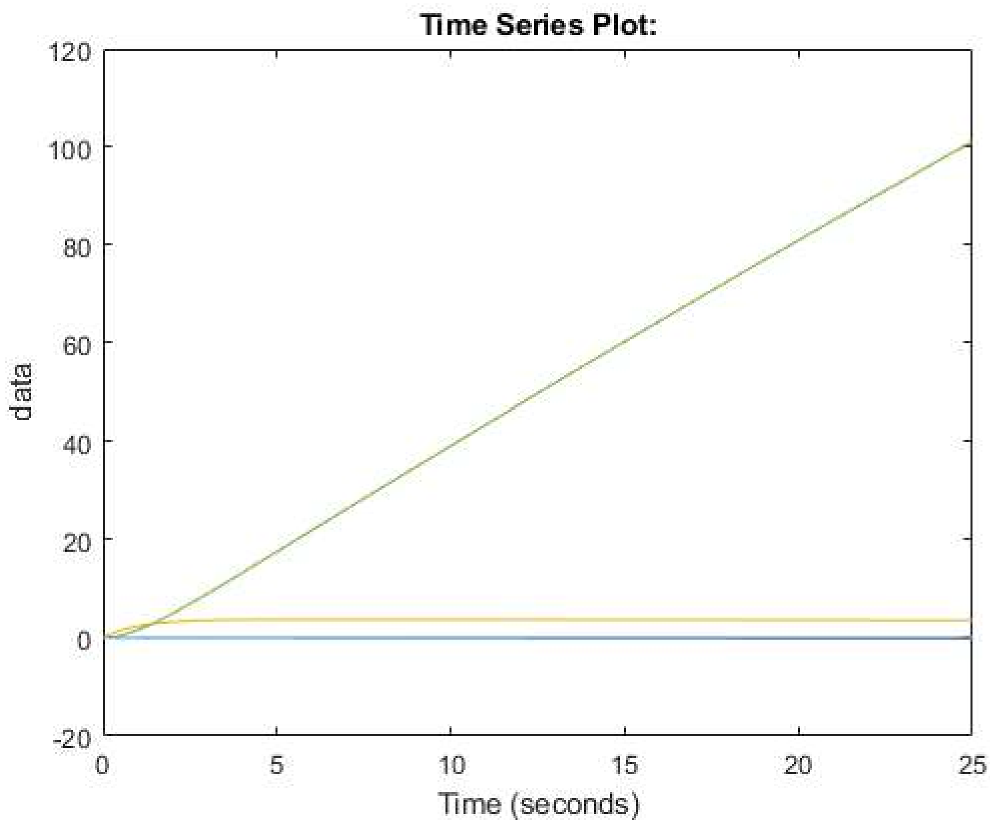
The contntroller can be

Ploting Things



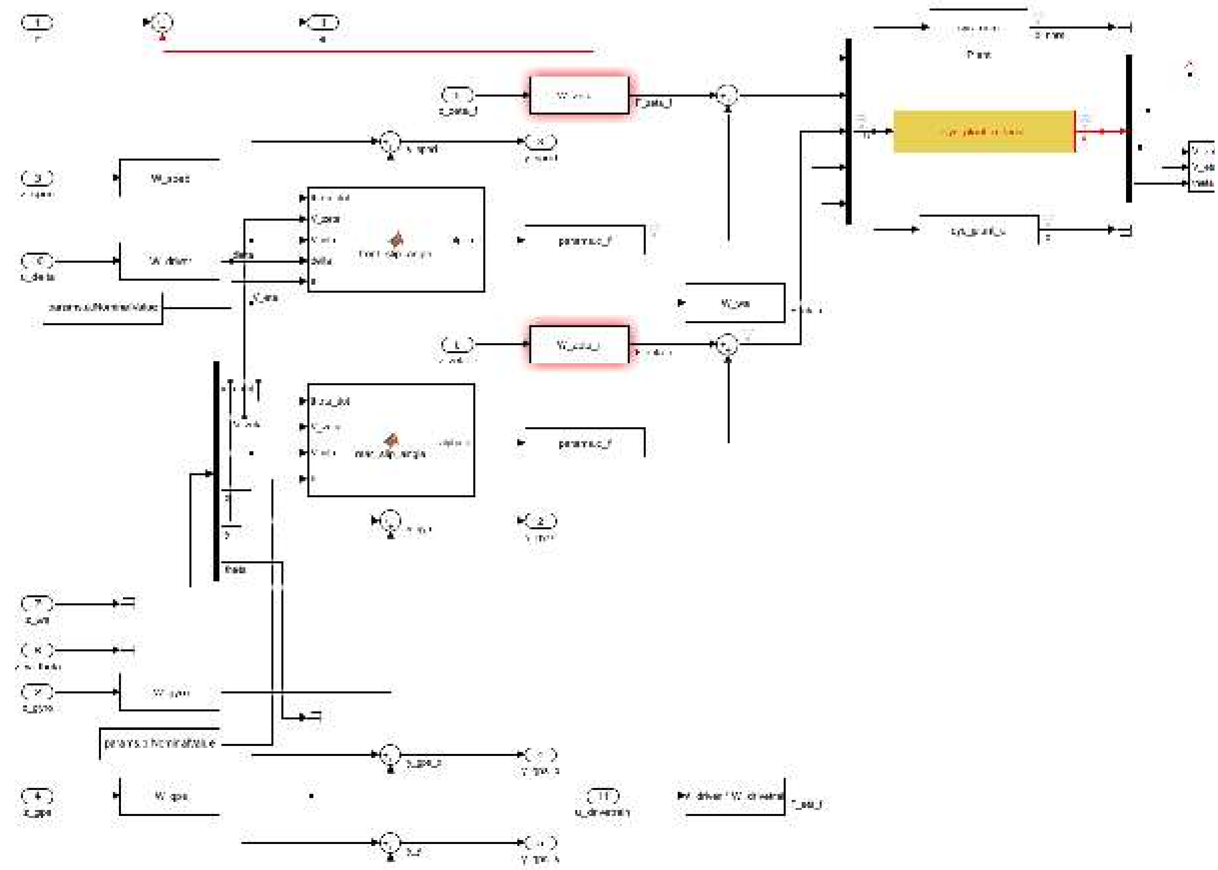


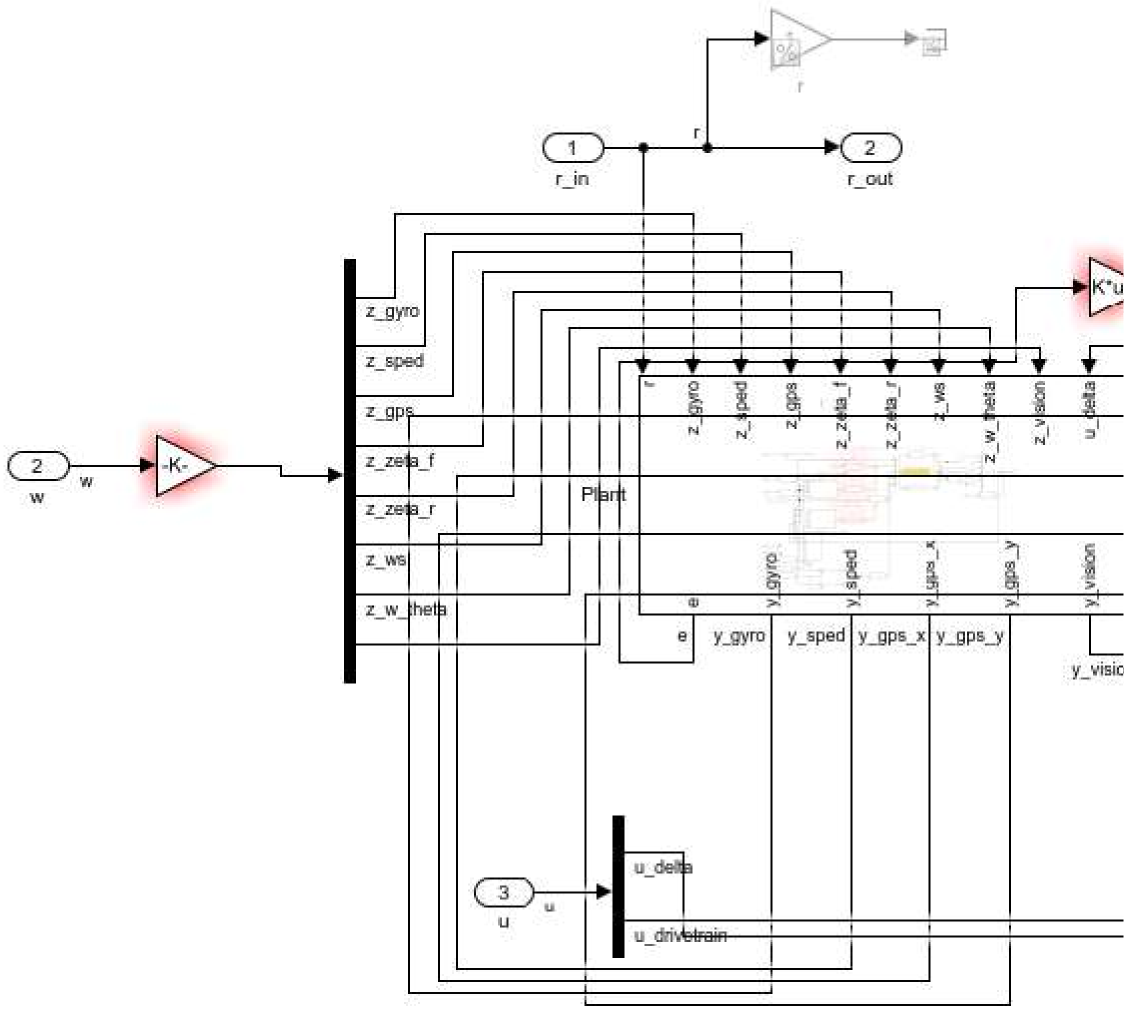
Time Series Plot:
data1
data2
data3
data4
data5
data6
data
Time (seconds)



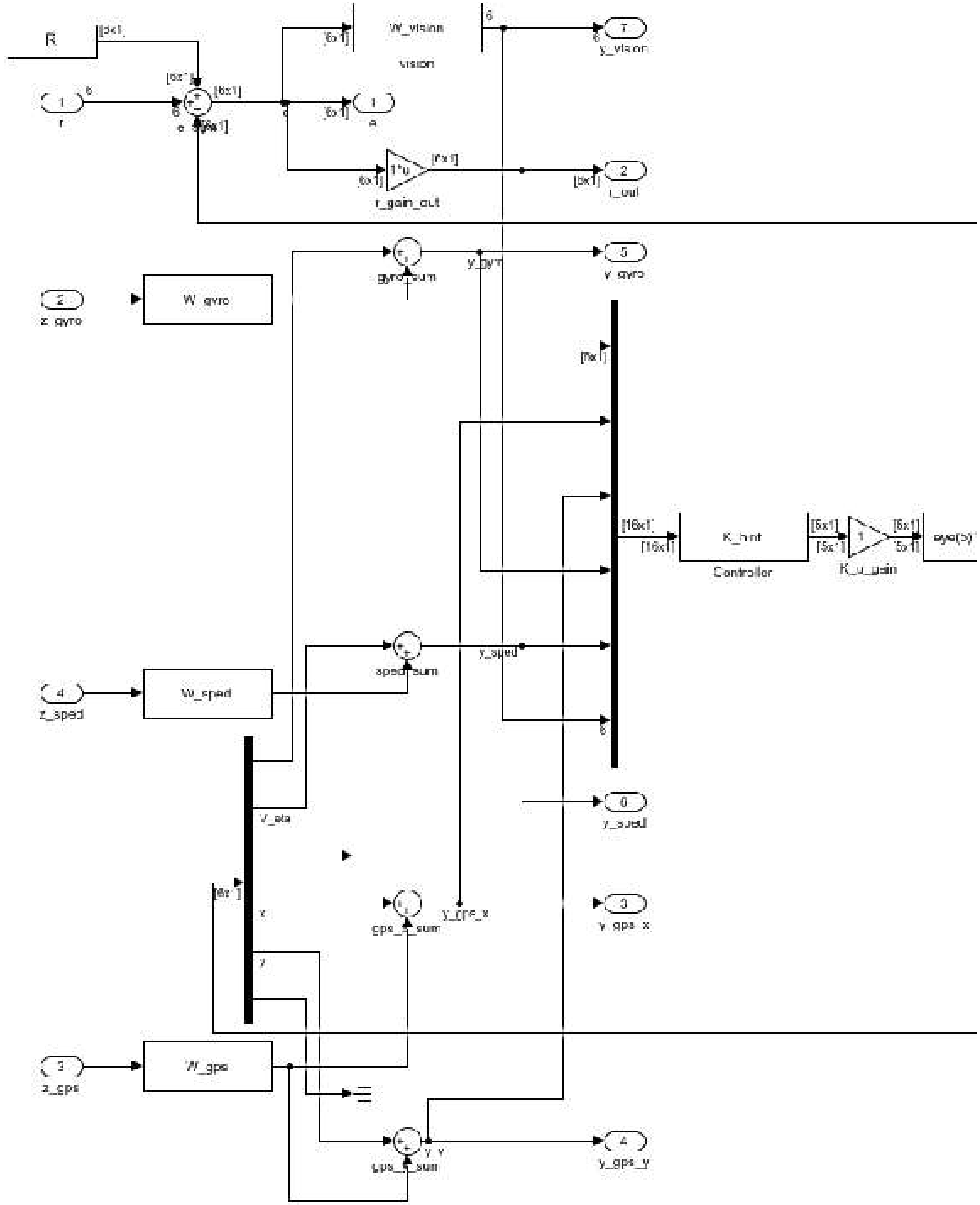
Time Series Plot:
data
Time (seconds)

Print outs


params.u.NominalValue
theta
u_delta
y_gps_x
y_gps_y
u_drivetrain



1
r_in
r
2
r_out
r
2
w
w
-K-
K*u
z_gyro
z_sped
z_gps
z_zeta_f
z_zeta_r
z_ws
z_w_theta
Plant
r
z_gyro
z_sped
z_gps
z_zeta_f
z_zeta_r
z_ws
z_w_theta
z_vision
u_delta
e
y_gyro
y_sped
y_gps_x
y_gps_y
y_vision
e
y_gyro
y_sped
y_gps_x
y_gps_y
3
u
u
u_delta
u_drivetrain


R
W_vision
vision
y_vision
r
e
r_gain_out
r_out
gyro_sum
y_gyro
z_gyro
W_gyro
[16x1]
K_hinf
Controller
[5x1]
K_u_gain
sped_sum
y_sped
W_sped
z_sped
V_eta
y_gps_x
W_gps
z_gps
y_gps_y