Report MPC Programming Excercise

Samuel Nyffenegger and Sebastian Ratz

General Comments

The figures in this report are the figures produced by your script, just combined into subplots. The curves are the same.

Nonlinear model and linearization

1. Structure of Ac and Bc

Ac and Bc are the partial derivatives of system dynamics with respect to the states and inputs in order to linearize the system dynamics.

F = (kF kF kF kF)*(u1 u2 u3 u4)’ (1)
xB’’ = uB’ = 1/m(F*sin(beta)) (2)
yB’’ = vB’ = 1/m(-F*sin(alpha)) (3)
zB’’ = wB’ = 1/m(cos(alpha)*cos(beta)*F - m*g) (4)

Ac(4,8) = d(2)/d(beta) = F/m*cos(beta) = g*cos(beta) = beta small ~ g = 9.81
Ac(5,7) = d(3)/d(alpha) = F/m*cos(alpha) = -g*cos(alpha) = alpha small ~ g = -9.81

Bc(6,1) = d(4)/d(u1) = 1/m*cos(alpha)*cos(beta)*dF/du1 = cos(alpha)*cos(beta)/m*kF = alpha & beta small ~ kF/m = 3.5
Bc(6,2) … Bc(6,4) analog

Bc(10:13,1:4) = jac(M/I) where M = [M_alpha M_beta M_gamma]’

First MPC Controller

2. Tuning Parameters

Q = diag([0.5,10,10,0.5,0.5,0.5,0.5])
R = 0.1*eye(n_inputs)

Q was chosen diagonal for simplicity. The parameters in Q were chosen such that the system output fulfills the specified requirements on z' and alpha, beta. The cost on the states x2 and x3 was chosen more expensive in order to ensure that they are controlled to zero quickly.

R was chosen diagonal as well, with relatively cheap input since there were no restrictions on power consumption or strain on the motors.

The terminal set and terminal cost were chosen corresponding to a LQR controlled system, due to stability guarantees and the relatively simple implementation.

P was thus chosen as the solution of the DARE and the terminal set as the maximum positive invariant set of the autonomous system x(k+1) = (A+BK)x(k).

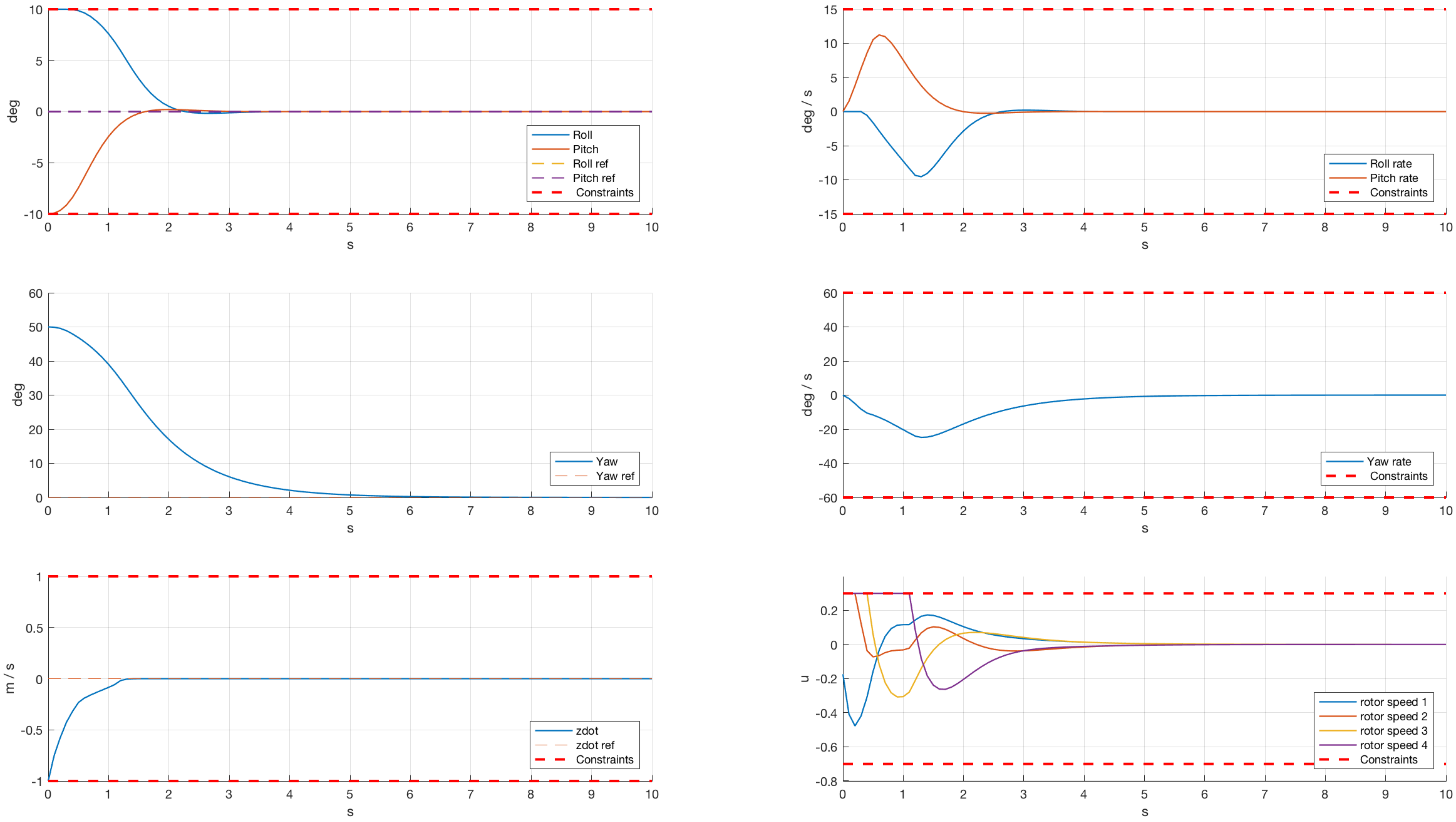
A_x and b_x form the constraints of the polyhedral terminal set A_x * x <= b_x:

```
A_x =
 0   -1.1716         0         0   -0.1124         0         0
         0         0   -1.1716         0         0   -0.1124         0
         0    1.1716         0         0    0.1124         0         0
         0         0    1.1716         0         0    0.1124         0
         0   -1.2245         0         0   -0.0219         0         0
         0         0   -1.2245         0         0   -0.0219         0
         0    1.2245         0         0    0.0219         0         0
         0         0    1.2245         0         0    0.0219         0
         0   -1.1858         0         0    0.1201         0         0
         0         0   -1.1858         0         0    0.1201         0
         0    1.1858         0         0   -0.1201         0         0
         0         0    1.1858         0         0   -0.1201         0
         0         0         0    0.9516         0         0    0.1346
         0   -1.0093         0         0    0.3275         0         0
         0         0   -1.0093         0         0    0.3275         0
         0         0         0   -0.9516         0         0   -0.1346
         0    1.0093         0         0   -0.3275         0         0
         0         0    1.0093         0         0   -0.3275         0
         0         0         0    0.9867         0         0    0.0824
         0   -0.6373         0         0    0.6156         0         0
         0         0   -0.6373         0         0    0.6156         0
         0         0         0   -0.9867         0         0   -0.0824
         0    0.6373         0         0   -0.6156         0         0
         0         0    0.6373         0         0   -0.6156         0
    1.0000         0         0         0         0         0         0
         0         0         0    1.0000         0         0         0
         0         0         0         0    1.0000         0         0
         0         0         0         0         0    1.0000         0
         0         0         0         0         0         0    1.0000
   -1.0000         0         0         0         0         0         0
         0         0         0   -1.0000         0         0         0
         0         0         0         0   -1.0000         0         0
         0         0         0         0         0   -1.0000         0
         0         0         0         0         0         0   -1.0000
```

```
b_x =
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
       10000
```

3. Plots

See next page



deg
s
Roll
Pitch
Roll ref
Pitch ref
Constraints
deg / s
s
Roll rate
Pitch rate
Constraints
deg
s
Yaw
Yaw ref
deg / s
s
Yaw rate
Constraints
m / s
s
zdot
zdot ref
Constraints
u
s
rotor speed 1
rotor speed 2
rotor speed 3
rotor speed 4
Constraints

4. Steady state (xr, ur) as function of r

See matlab code section II.

```
constraints_mpc = [];

%target constraints
constraints_mpc = constraints_mpc + [C*xr == r];
constraints_mpc = constraints_mpc + [sys.A*xr+sys.B*ur == xr];

%delta shifting
constraints_mpc = constraints_mpc + [delta_x(:,1) == xk-xr];
constraints_mpc = constraints_mpc + [delta_u(:,1) == uk-ur];

for i = 2:N
    %system constraints
    constraints_mpc = constraints_mpc + [delta_x(:,i) == sys.A*delta_x(:,i-
1)+sys.B*delta_u(:,i-1)];

    %state constraints
    constraints_mpc =  constraints_mpc +  [-z_dot_max-xr(1) <= delta_x(1,i)
<= z_dot_max-xr(1)];
    constraints_mpc =  constraints_mpc + [-alpha_beta_max-xr(2:3)  <=
delta_x(2:3,i) <= alpha_beta_max-xr(2:3)];
    constraints_mpc = constraints_mpc + [-alpha_beta_dot_max-xr(5:6)  <=
delta_x(5:6,i) <= alpha_beta_dot_max-xr(5:6)];
    constraints_mpc = constraints_mpc + [-gamma_dot_max-xr(7)  <=
delta_x(7,i) <= gamma_dot_max-xr(7)];

    %input constraints

    constraints_mpc = constraints_mpc + [u_min-ur <= delta_u(1:4,i) <=
u_max-ur ];
end
constraints_mpc = constraints_mpc + [u_min-ur <= delta_u(1:4,1) <= u_max-ur
];
```

5. Plots constant reference

See this page + 1 (next page)

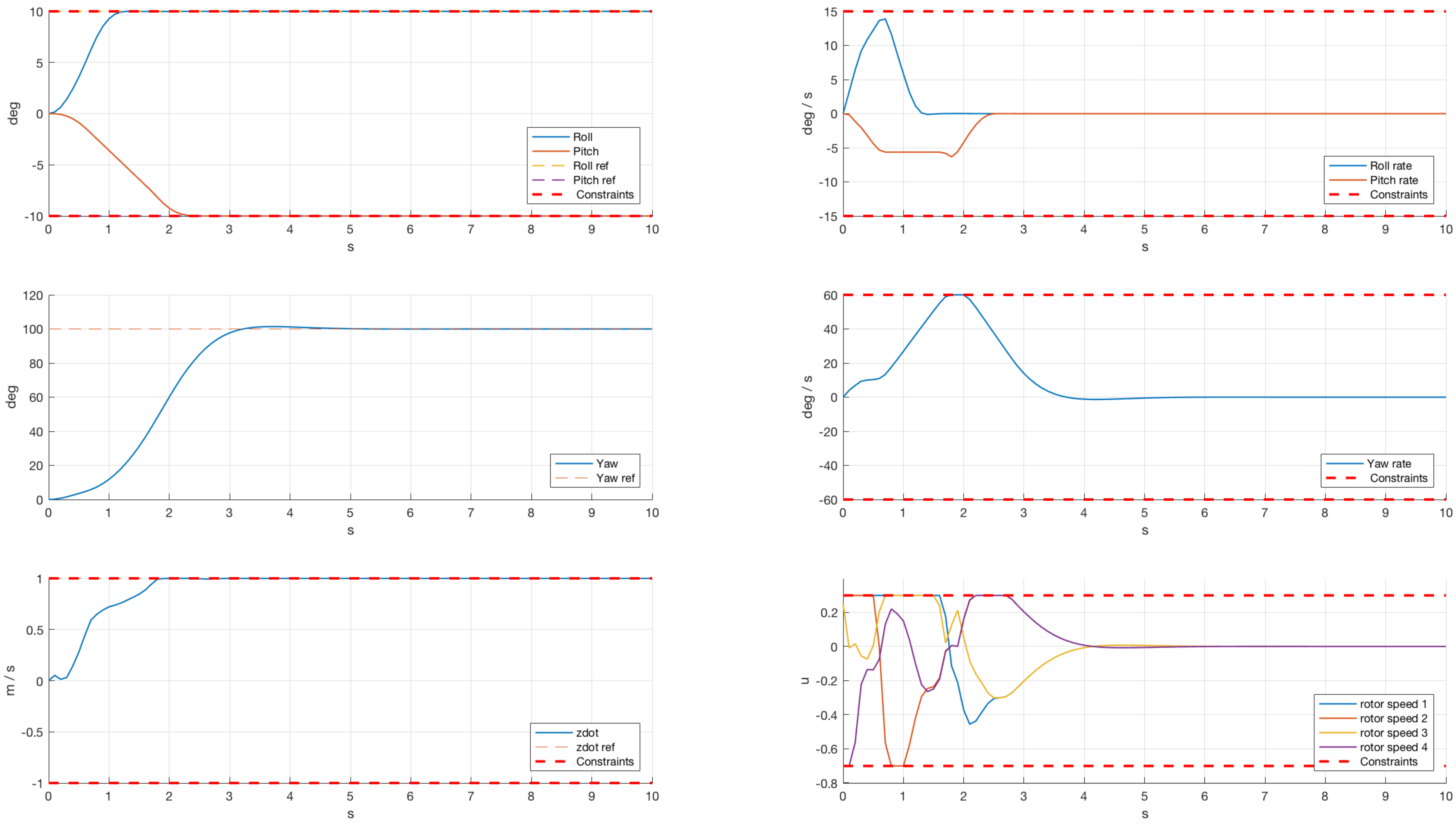
6. Plots of varying reference signal

See this page + 2

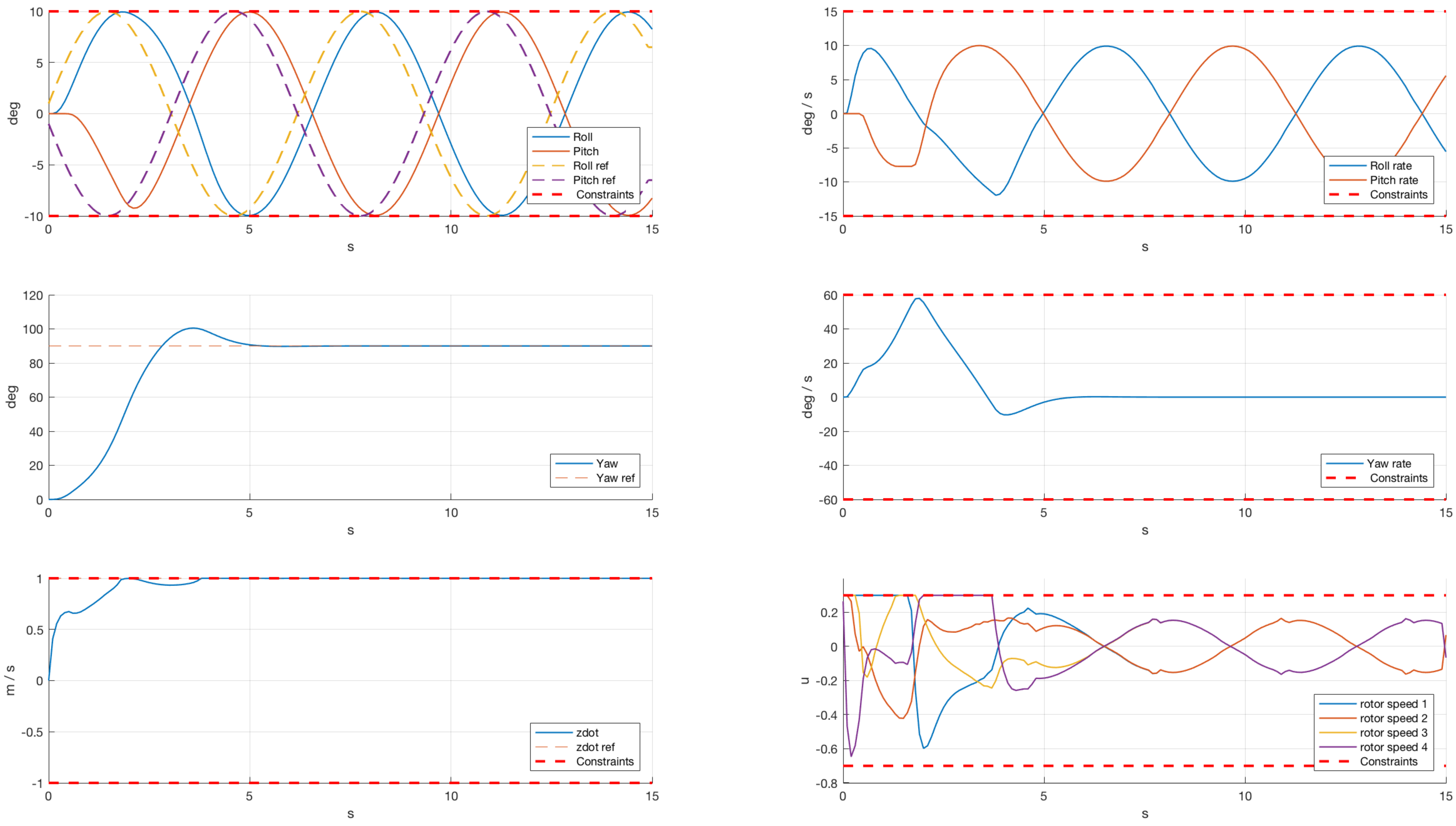
First simulation of nonlinear model

7. Plots

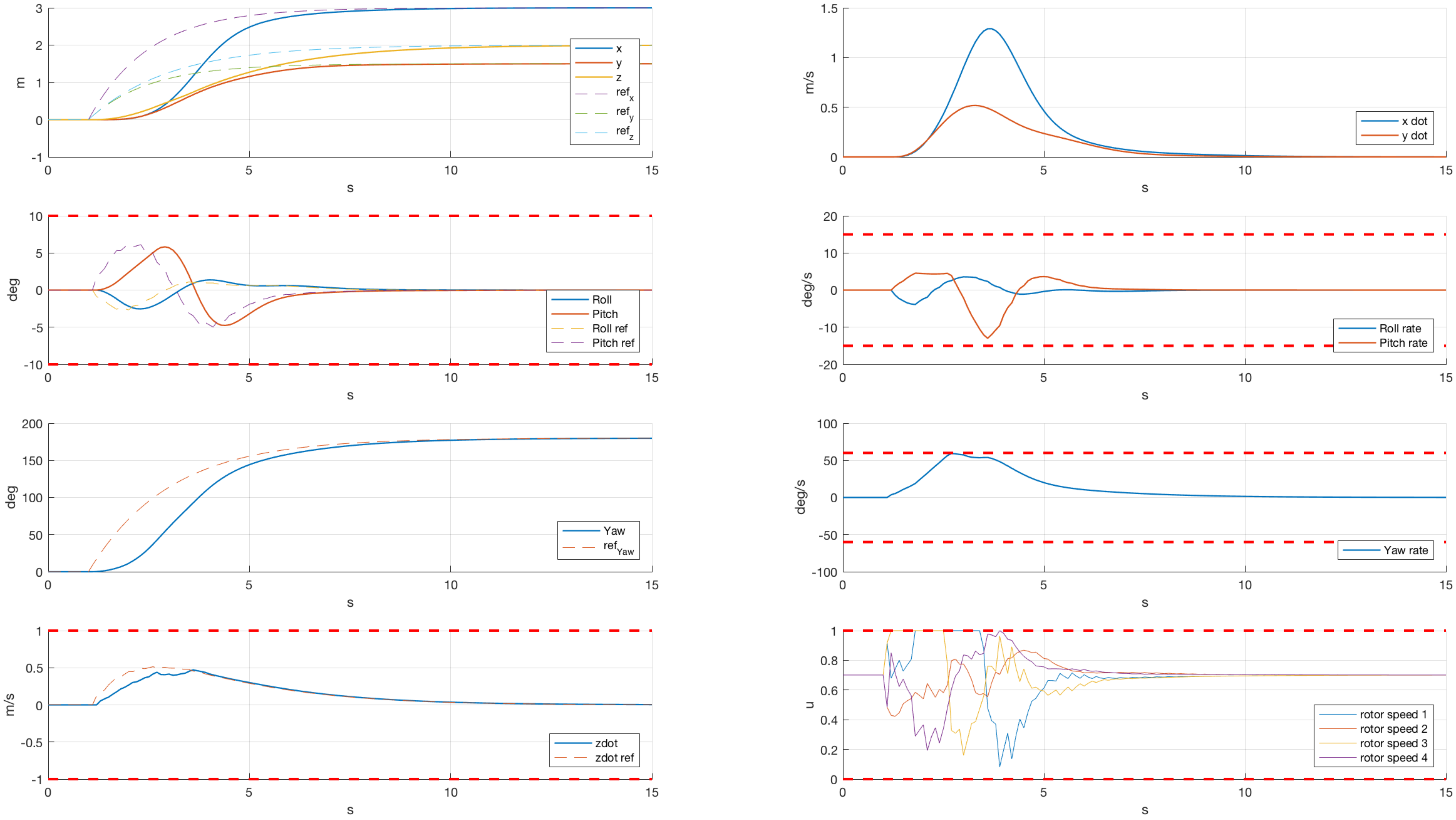
See this page + 3 & 4



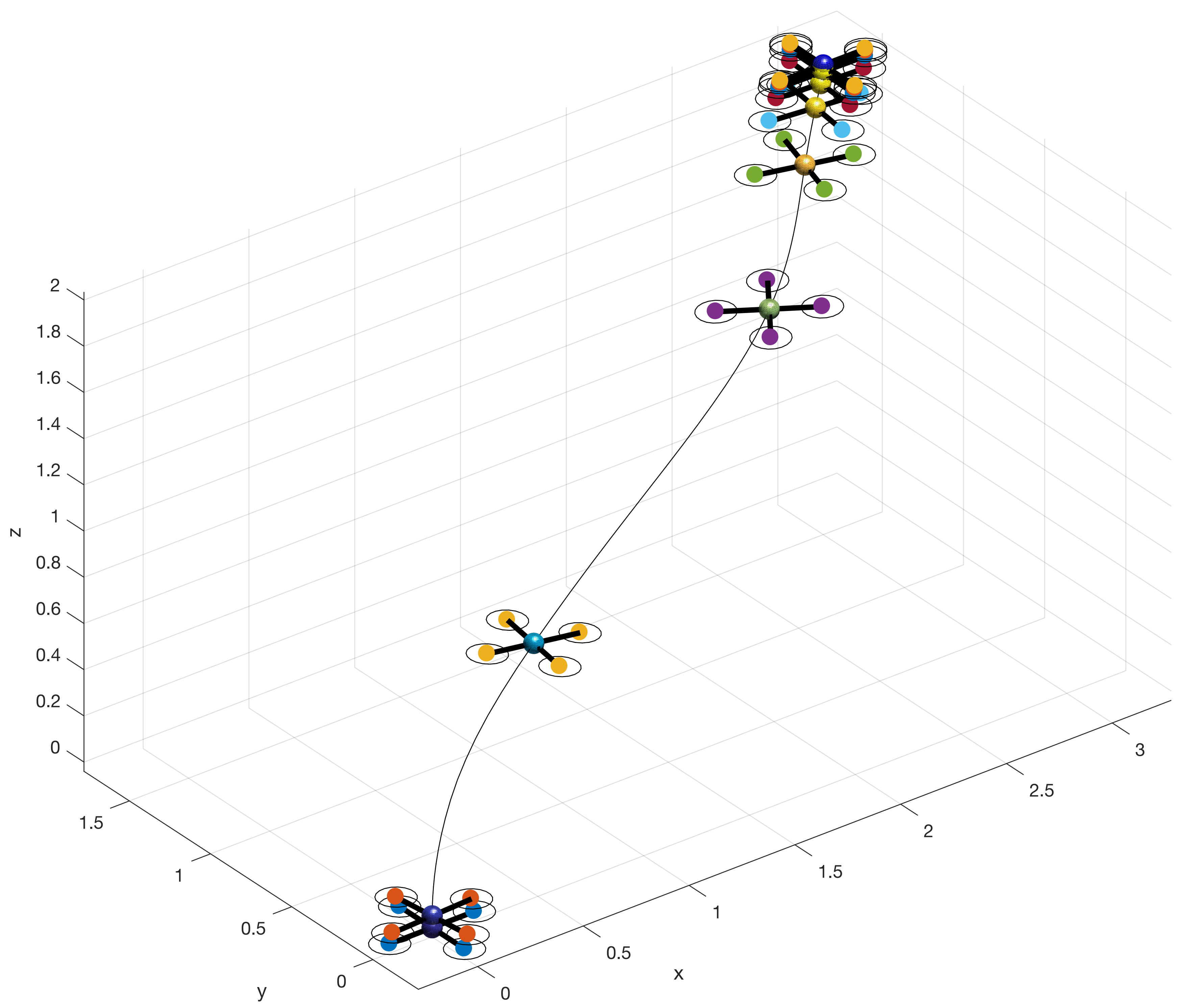
deg
s
Roll
Pitch
Roll ref
Pitch ref
Constraints
deg / s
Roll rate
Pitch rate
Constraints
Yaw
Yaw ref
Yaw rate
m / s
zdot
zdot ref
u
rotor speed 1
rotor speed 2
rotor speed 3
rotor speed 4



deg
s
Roll
Pitch
Roll ref
Pitch ref
Constraints
deg / s
Roll rate
Pitch rate
Constraints
Yaw
Yaw ref
Yaw rate
Constraints
m / s
zdot
zdot ref
Constraints
u
rotor speed 1
rotor speed 2
rotor speed 3
rotor speed 4
Constraints



m
s
x
y
z
ref_x
ref_y
ref_z
m/s
x dot
y dot
deg
Roll
Pitch
Roll ref
Pitch ref
deg/s
Roll rate
Pitch rate
Yaw
ref_Yaw
Yaw rate
zdot
zdot ref
u
rotor speed 1
rotor speed 2
rotor speed 3
rotor speed 4



z
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
y
0
0.5
1
1.5
x
0
0.5
1
1.5
2
2.5
3

Offset free MPC

8. L Matrix

L = [Lx; Ld] = [eye(7); eye(7)]

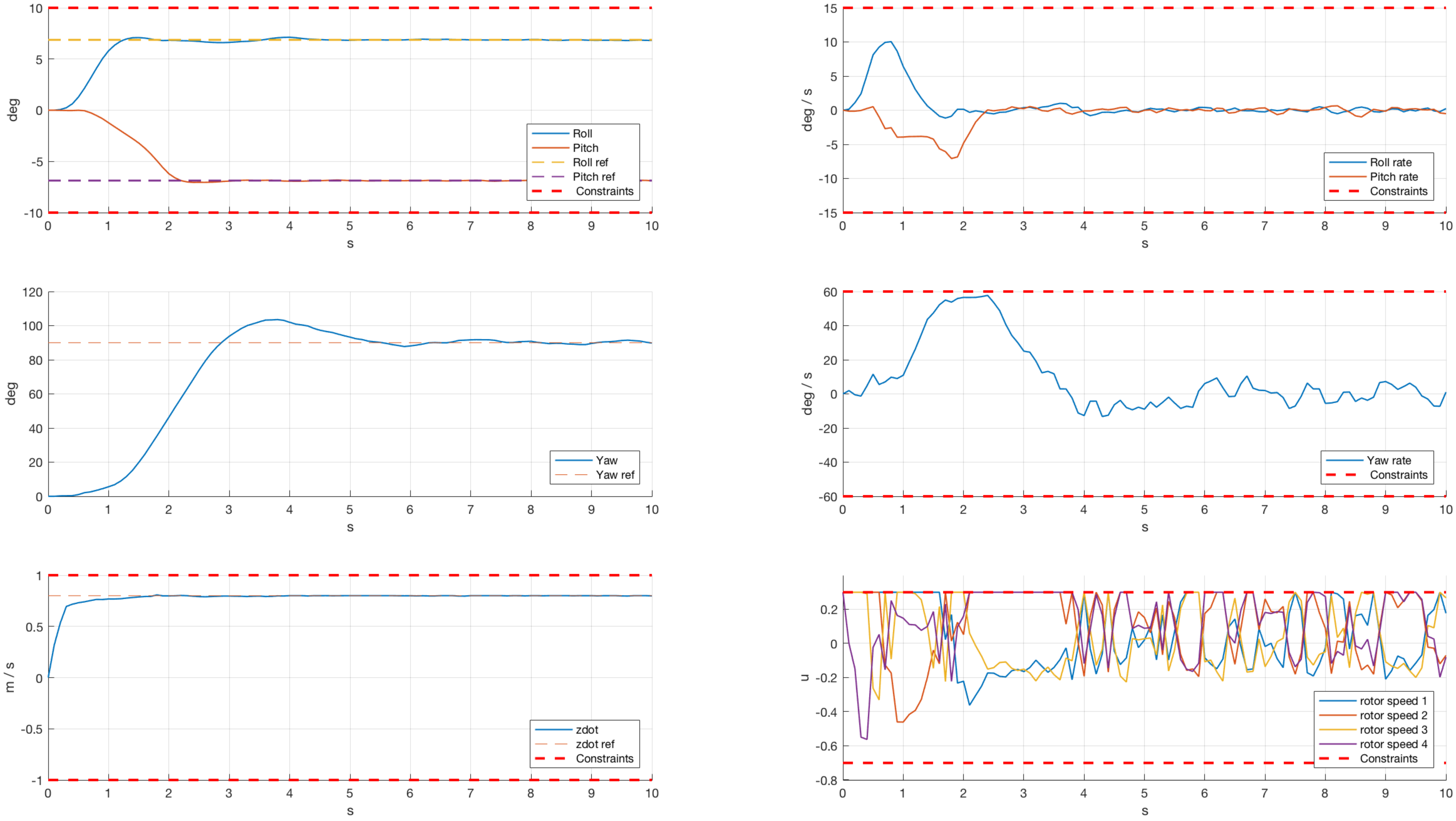
Due to the independence of the disturbances, Lx and Ld were chosen diagonal. Their influence on the states are thus estimated directly and without bias or scaling.

9. Plots of constant reference

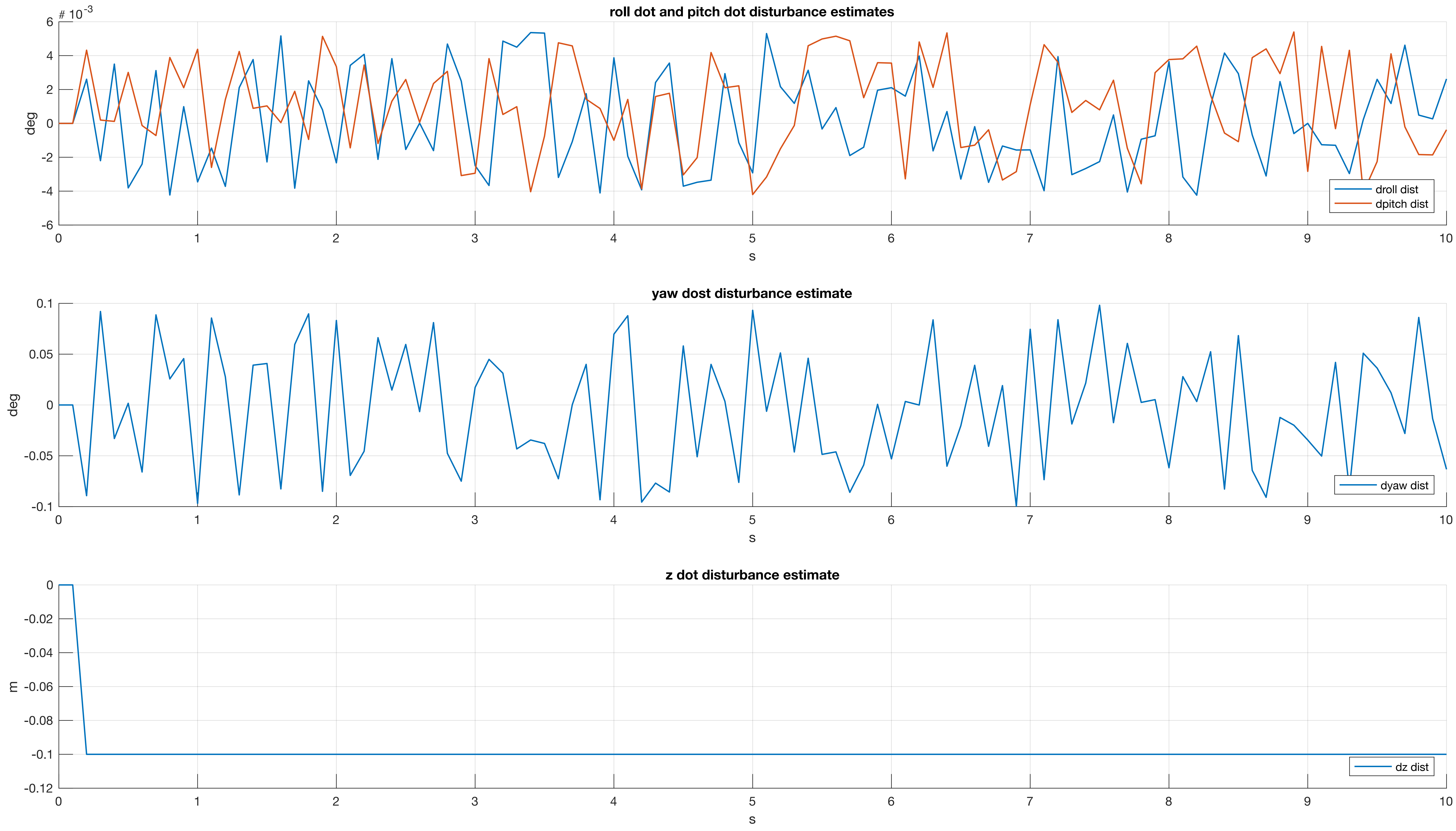
See next two pages

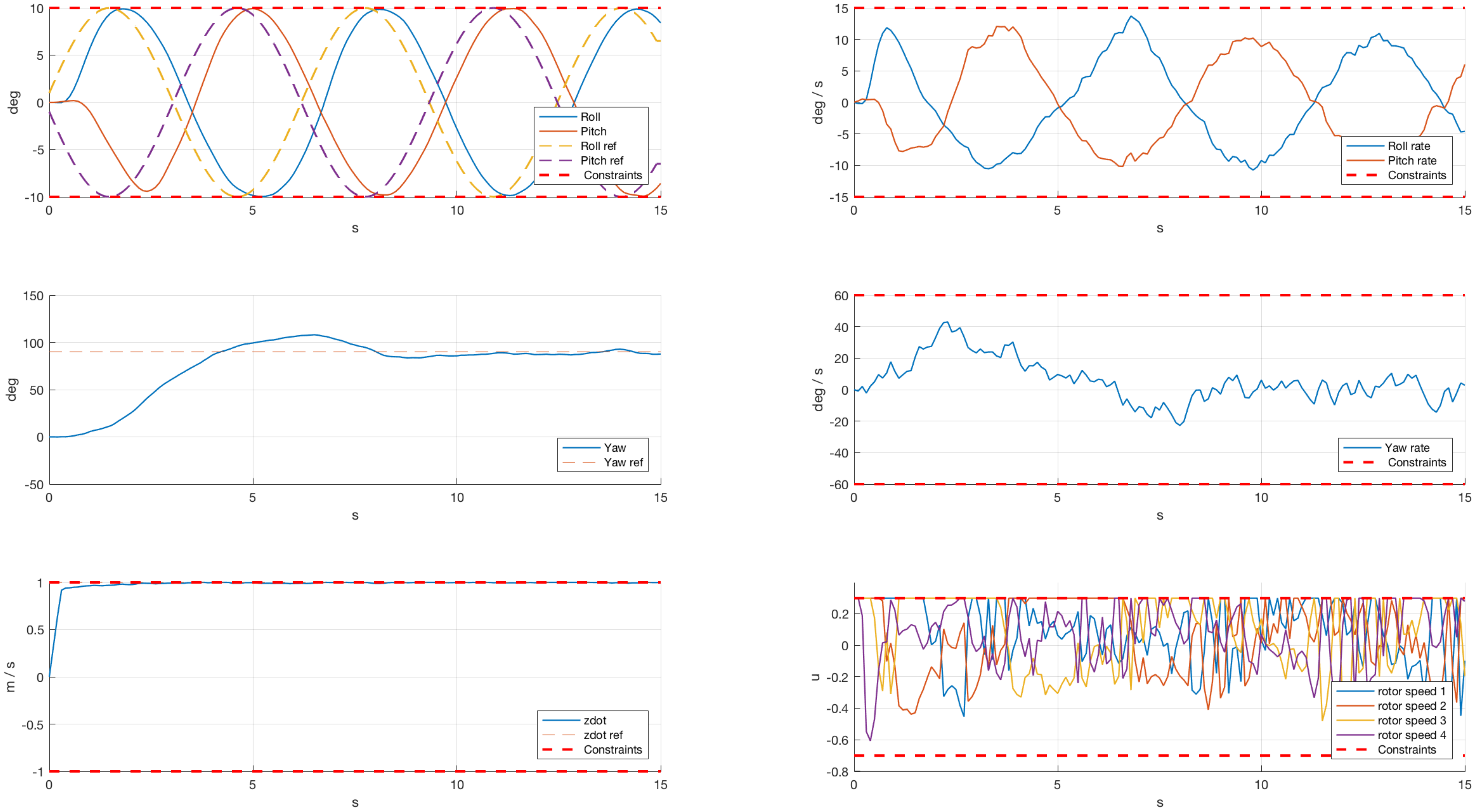
10. Plots of varying reference

See this page + 3 & 4

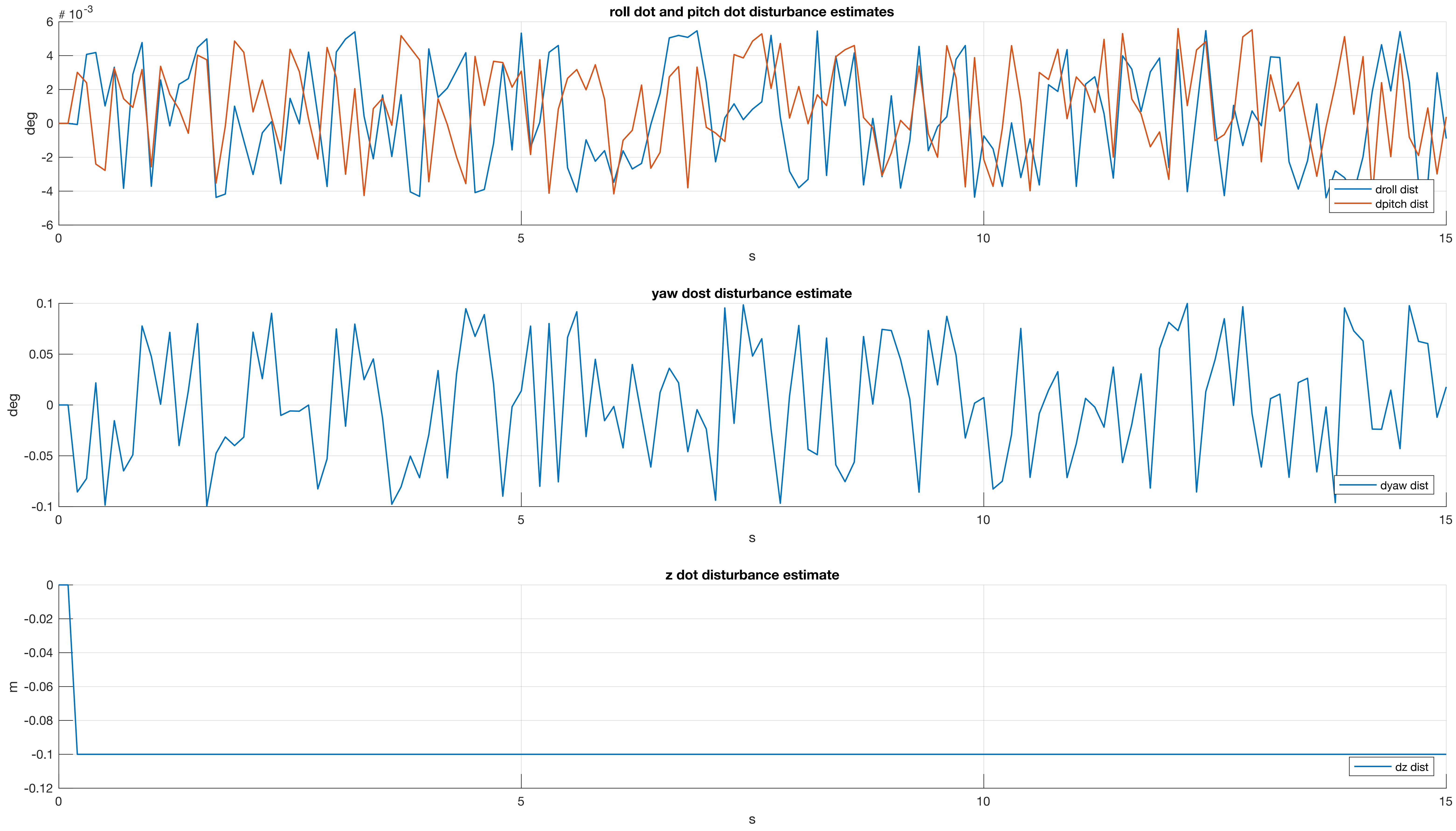


deg
s
Roll
Pitch
Roll ref
Pitch ref
Constraints
deg / s
s
Roll rate
Pitch rate
Constraints
deg
s
Yaw
Yaw ref
deg / s
s
Yaw rate
Constraints
m / s
s
zdot
zdot ref
Constraints
u
s
rotor speed 1
rotor speed 2
rotor speed 3
rotor speed 4
Constraints


roll dot and pitch dot disturbance estimates
10^-3
deg
s
droll dist
dpitch dist
yaw dost disturbance estimate
deg
s
dyaw dist
z dot disturbance estimate
m
s
dz dist



deg
s
Roll
Pitch
Roll ref
Pitch ref
Constraints
deg / s
Roll rate
Pitch rate
Constraints
Yaw
Yaw ref
Yaw rate
m / s
zdot
zdot ref
u
rotor speed 1
rotor speed 2
rotor speed 3
rotor speed 4


roll dot and pitch dot disturbance estimates
10^-3
deg
s
droll dist
dpitch dist
yaw dost disturbance estimate
deg
s
dyaw dist
z dot disturbance estimate
m
s
dz dist

Simulation of the nonlinear model

11. Plots of step signal

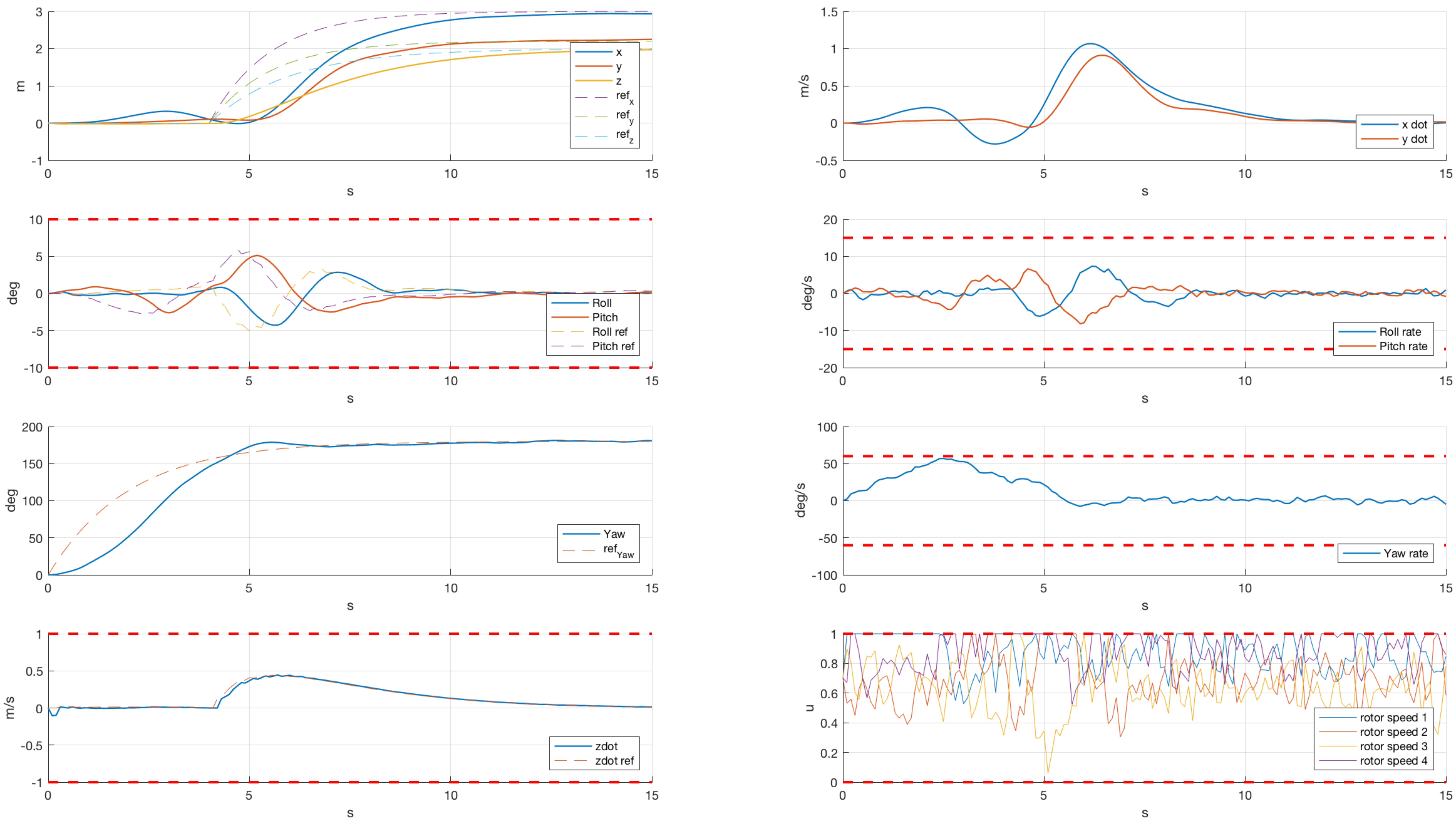
See next 3 pages

12. Plots of hexagon reference

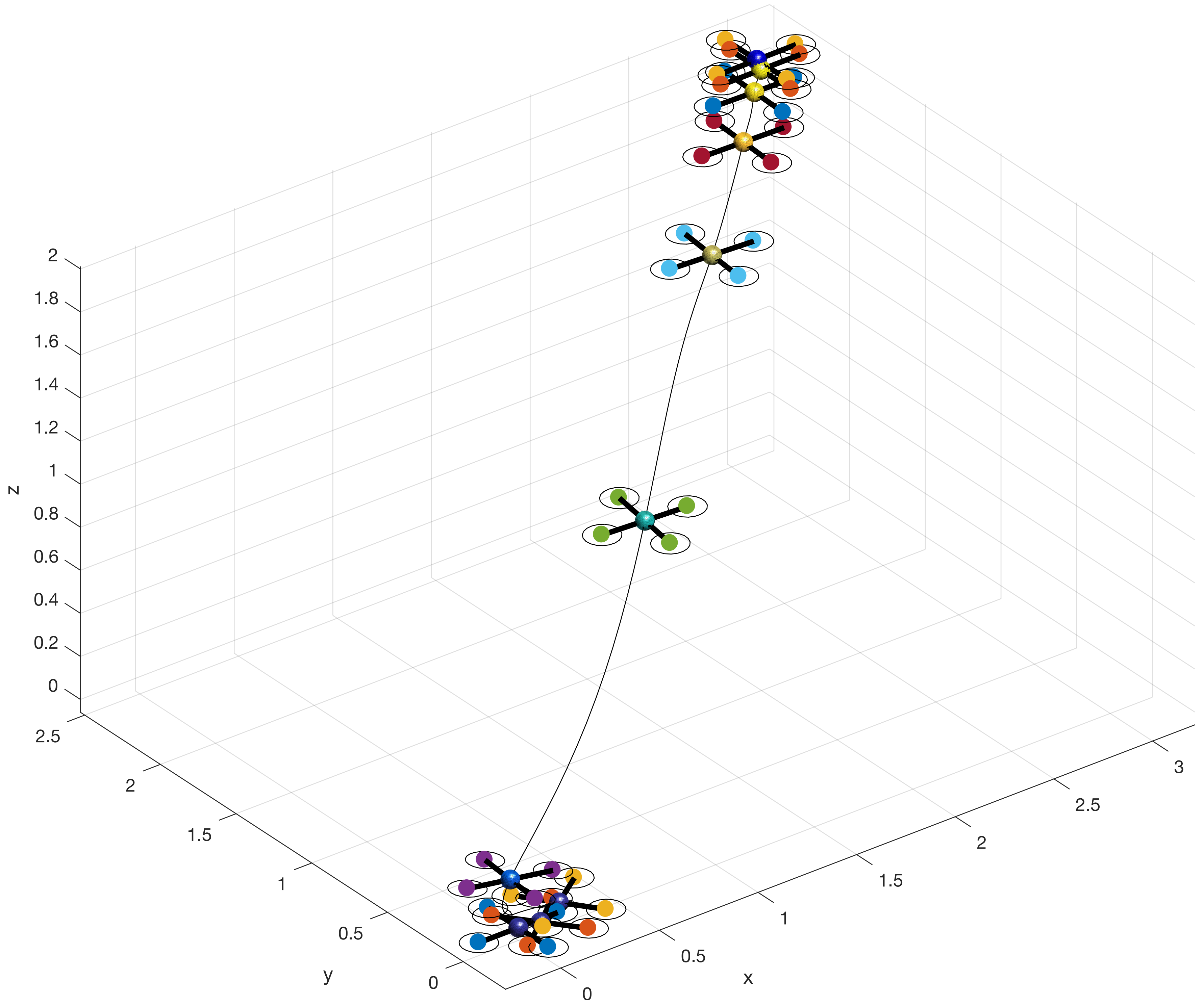
See this page + 4 to 6

13. Plots infinity shaped reference

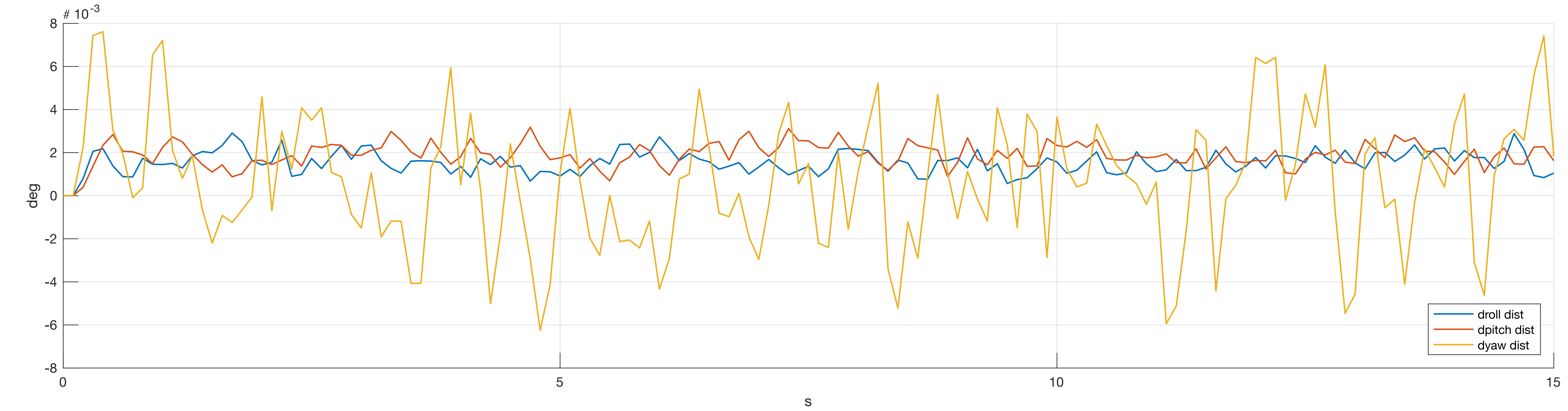
See this page + 7 to 9



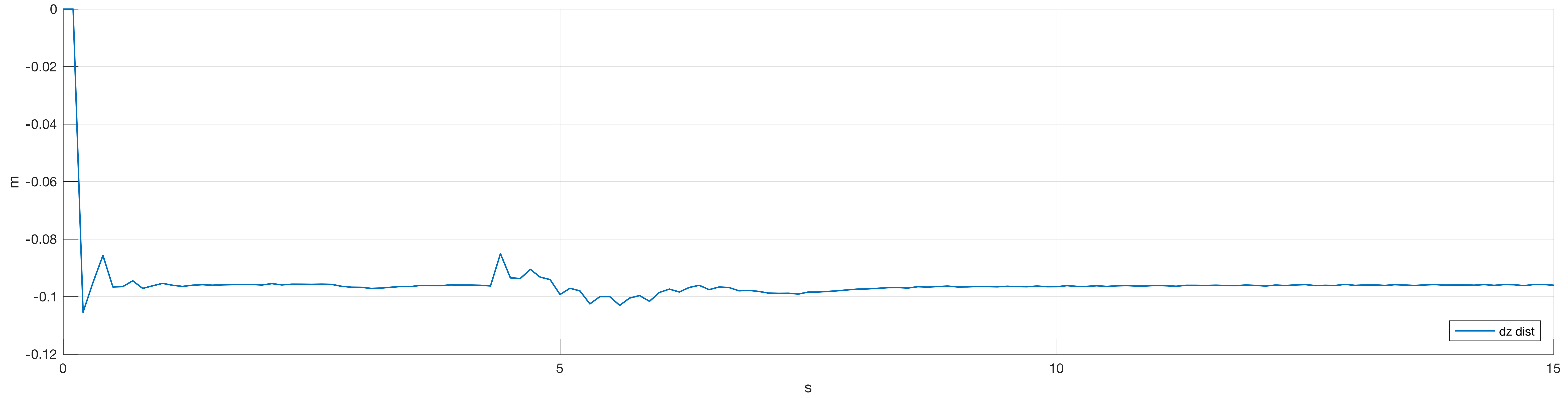
m
s
x
y
z
ref_x
ref_y
ref_z
m/s
x dot
y dot
deg
Roll
Pitch
Roll ref
Pitch ref
deg/s
Roll rate
Pitch rate
Yaw
ref_Yaw
Yaw rate
zdot
zdot ref
u
rotor speed 1
rotor speed 2
rotor speed 3
rotor speed 4



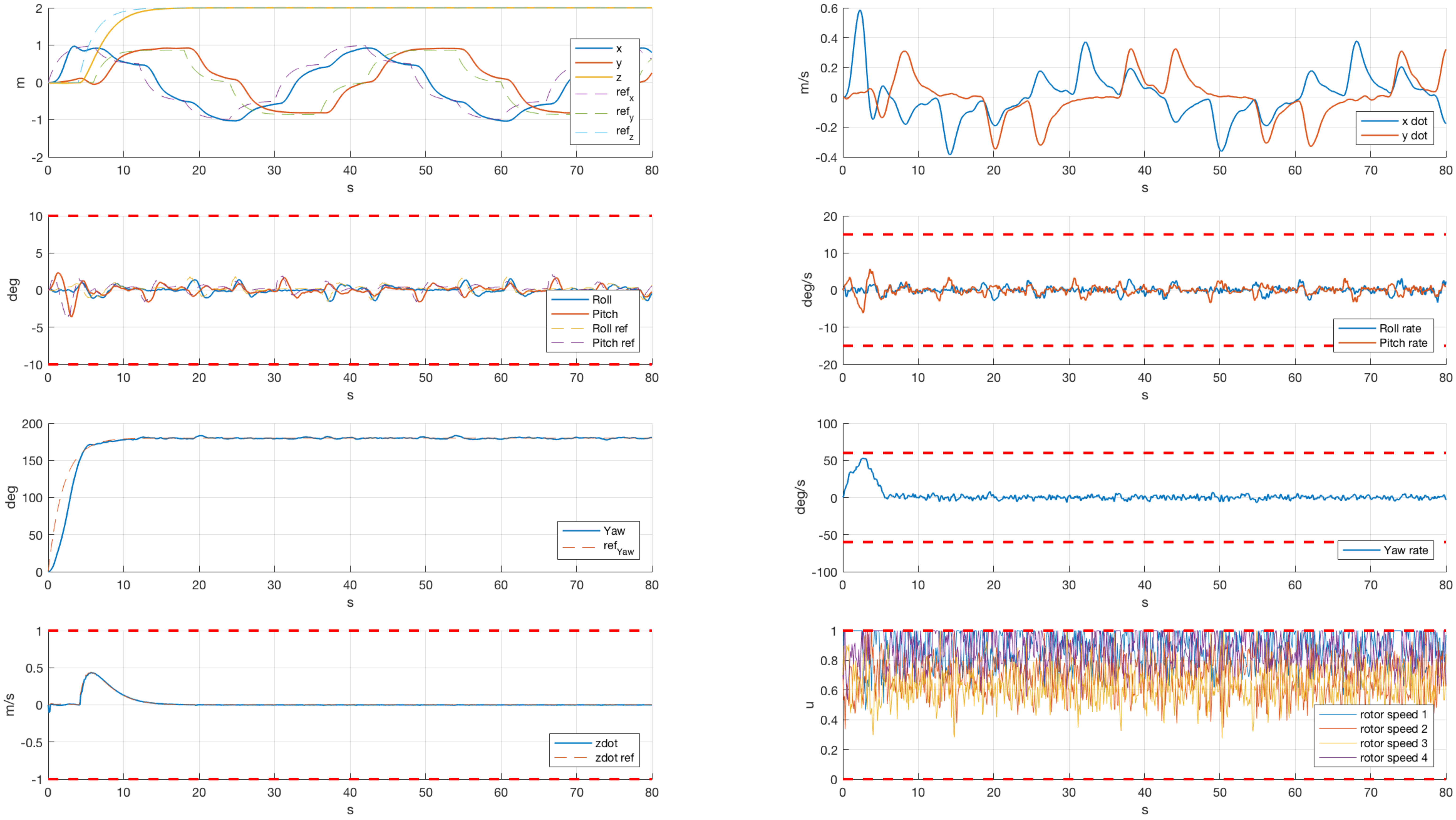
z
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
y
2.5
2
1.5
1
0.5
0
x
0
0.5
1
1.5
2
2.5
3



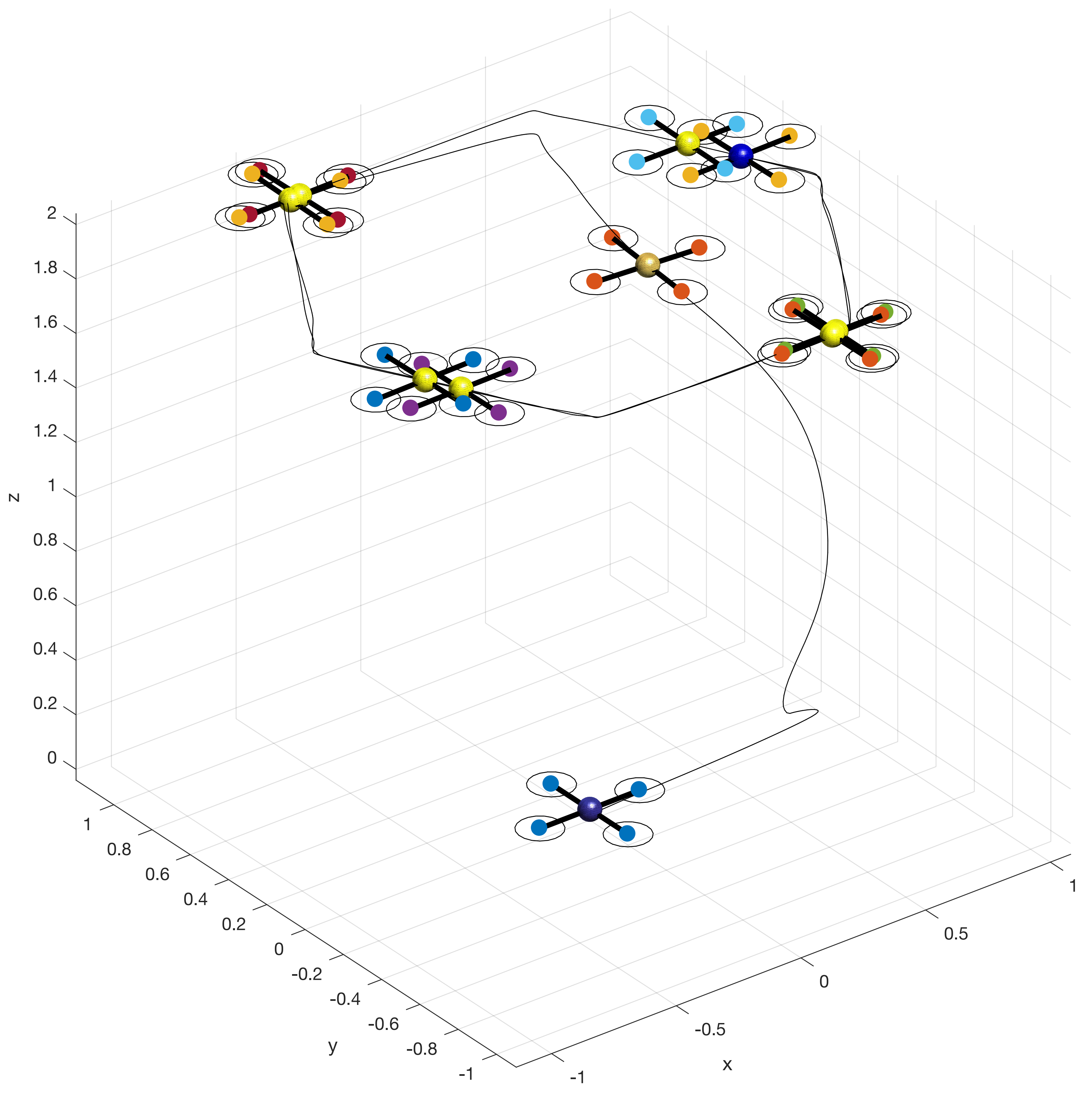
10-3
deg
s
droll dist
dpitch dist
dyaw dist

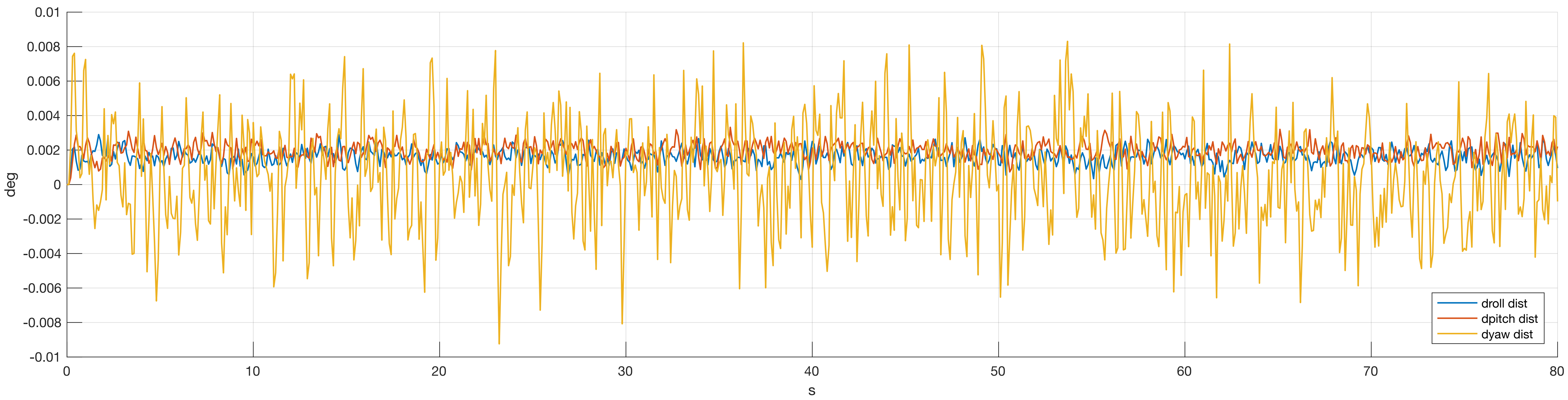


m
s
dz dist

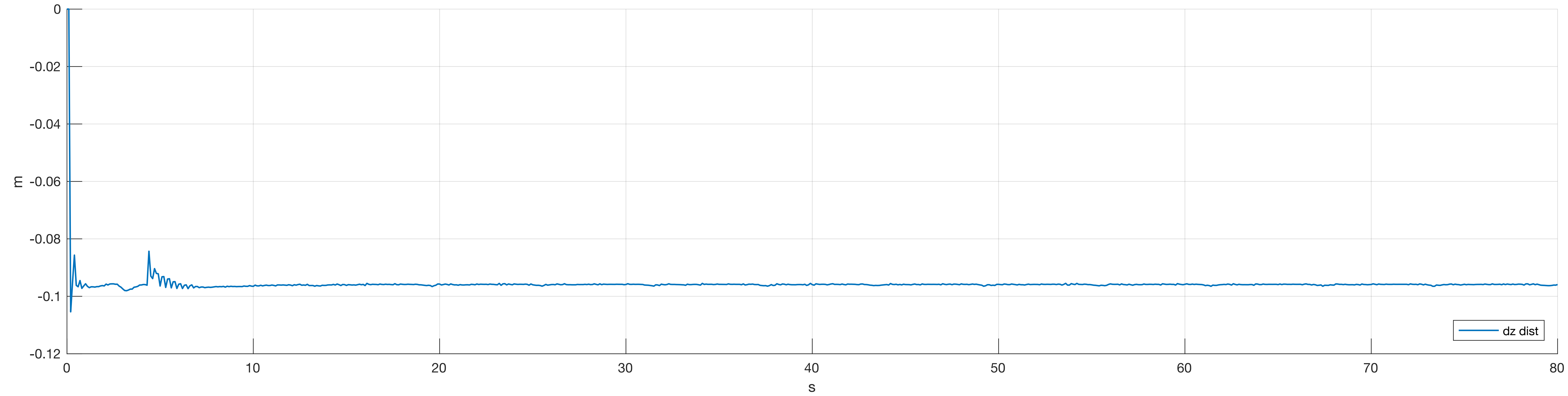


m
s
x
y
z
ref_x
ref_y
ref_z
m/s
x dot
y dot
deg
Roll
Pitch
Roll ref
Pitch ref
deg/s
Roll rate
Pitch rate
Yaw
ref_Yaw
Yaw rate
zdot
zdot ref
u
rotor speed 1
rotor speed 2
rotor speed 3
rotor speed 4

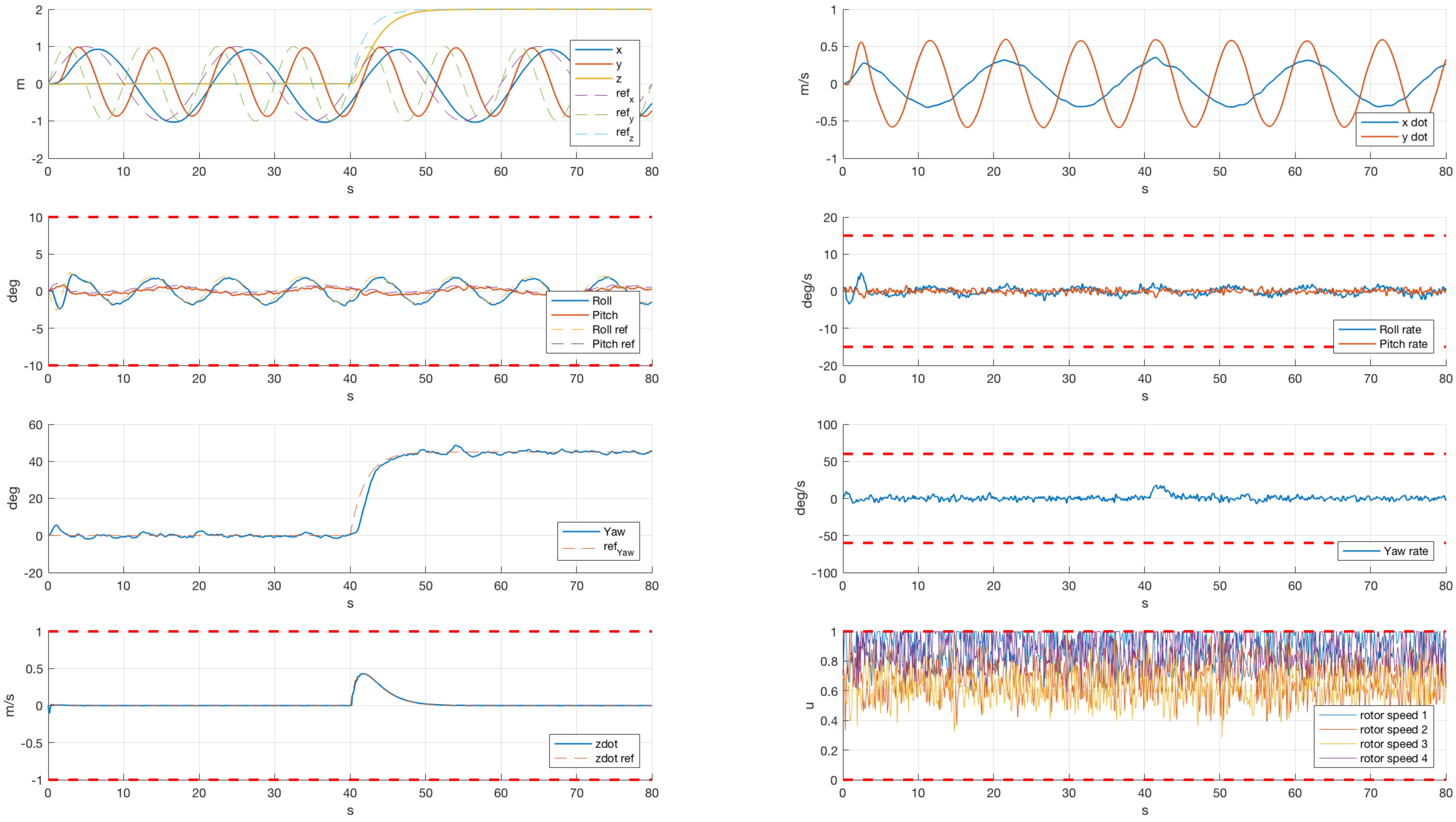

z
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
y
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
x
-1
-0.5
0
0.5
1

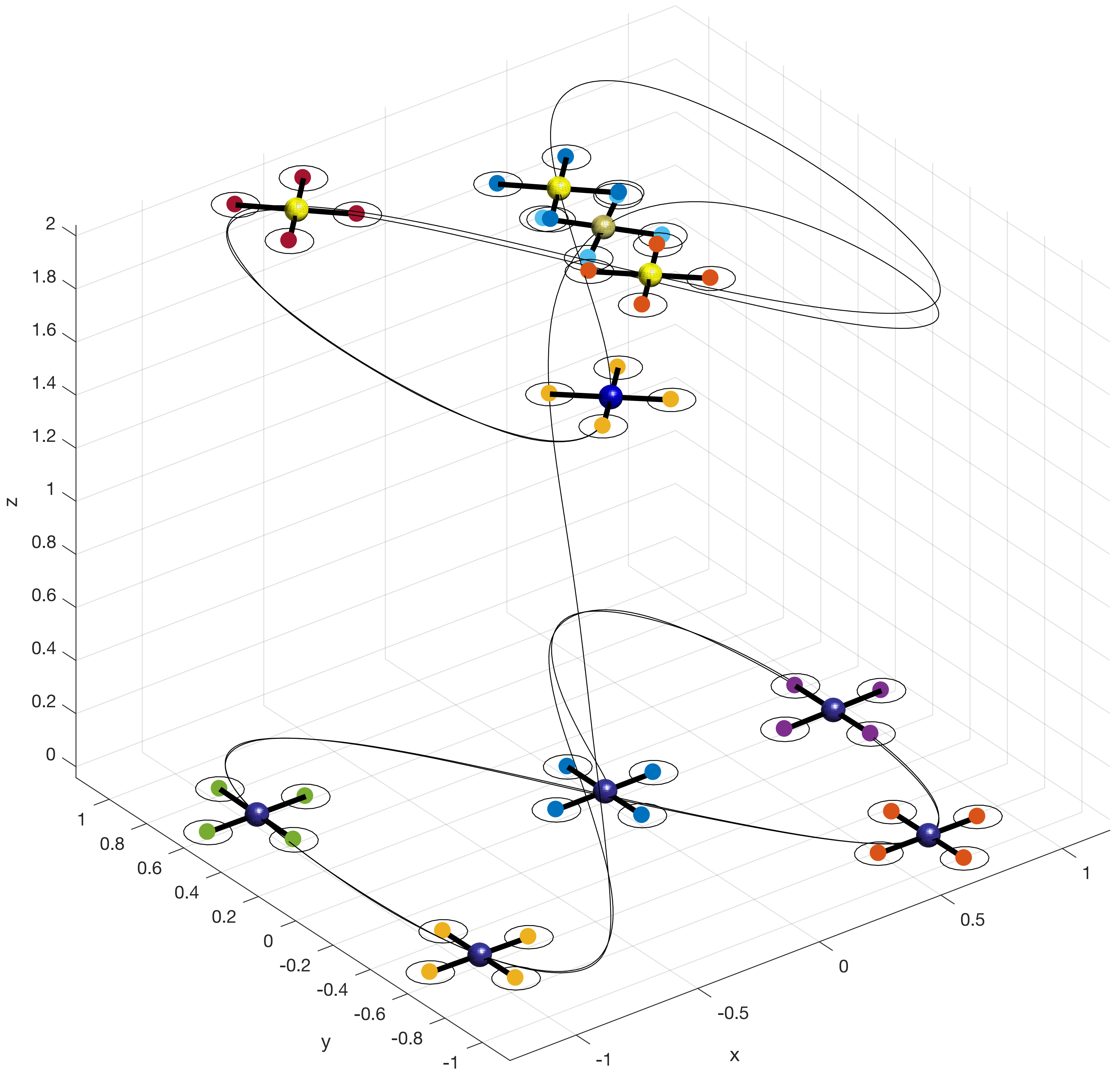


0.01
0.008
0.006
0.004
0.002
0
-0.002
-0.004
-0.006
-0.008
-0.01
deg
0
10
20
30
40
50
60
70
80
s
droll dist
dpitch dist
dyaw dist

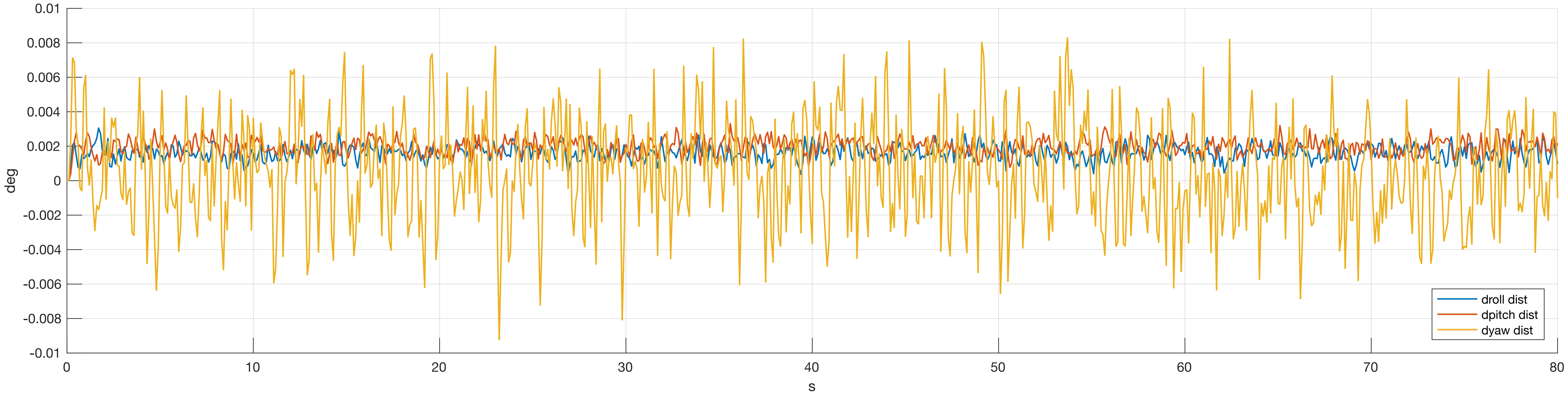


0
-0.02
-0.04
-0.06
-0.08
-0.1
-0.12
m
0
10
20
30
40
50
60
70
80
s
dz dist

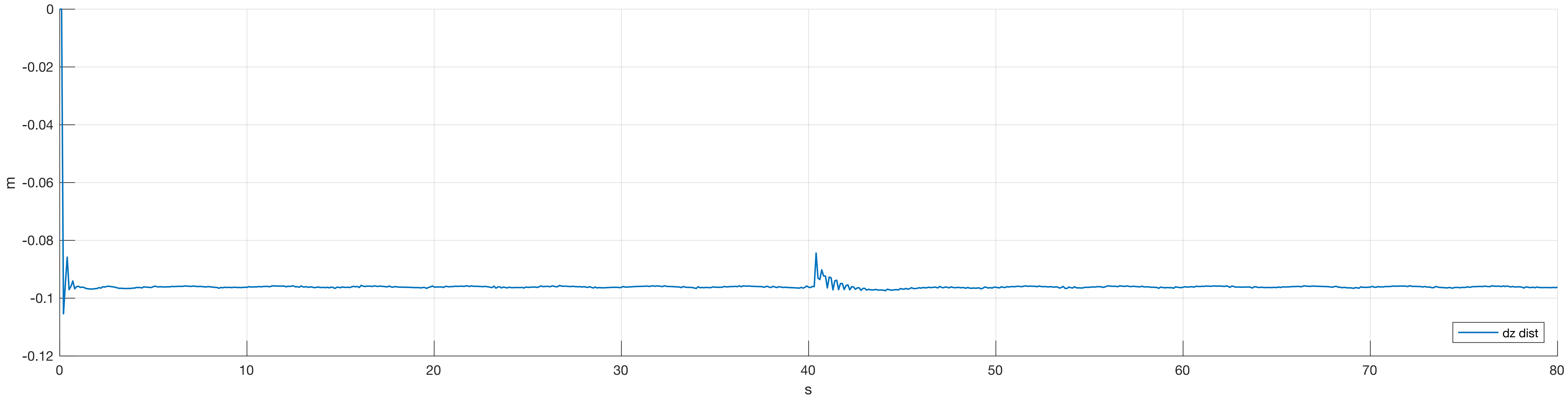

m
s
x
y
z
ref_x
ref_y
ref_z
m/s
x dot
y dot
deg
Roll
Pitch
Roll ref
Pitch ref
deg/s
Roll rate
Pitch rate
Yaw
ref_Yaw
Yaw rate
zdot
zdot ref
u
rotor speed 1
rotor speed 2
rotor speed 3
rotor speed 4



z
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
y
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
x
-1
-0.5
0
0.5
1



0.01
0.008
0.006
0.004
0.002
0
-0.002
-0.004
-0.006
-0.008
-0.01
deg
0
10
20
30
40
50
60
70
80
s
droll dist
dpitch dist
dyaw dist



0
-0.02
-0.04
-0.06
-0.08
-0.1
-0.12
m
0
10
20
30
40
50
60
70
80
s
dz dist

Slew rate constraints

14. Plots and delta value

Plot see the next two pages.

Delta = 0.19 was chosen in order to avoid performance deterioration or infeasibility problems.

Soft constraints

15. Cost function

The cost function for the soft constraints was chosen as:

```
delta_x = sdpvar(n_states, N,'full');
delta_u = sdpvar(n_inputs, N,'full');

delta = 0.1;

v = 1*[1;1;1;1]';
s = 0.1*[1;1;1;1]';

objective_mpc = 0;
for i = 1:N
   objective_mpc = objective_mpc + delta_x(:,i)' * Q * delta_x(:,i) + delta_u(:,i)' * R *
delta_u(:,i);
   objective_mpc = objective_mpc + v*epsilon(:,i) + s*epsilon(:,i).^2;
end
```

16. Plots of constant reference

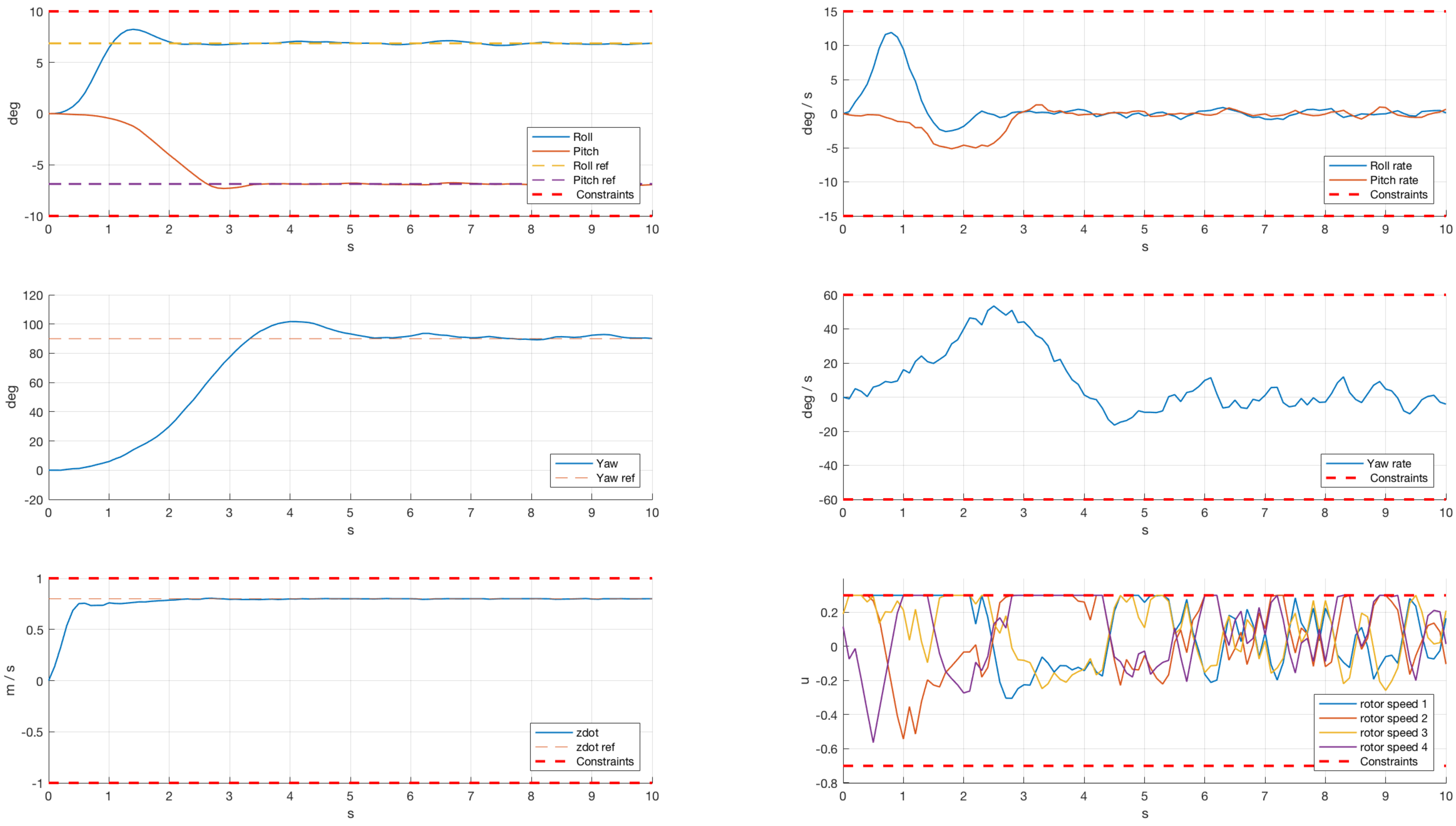
See this page + 3 to 4

Forces Pro

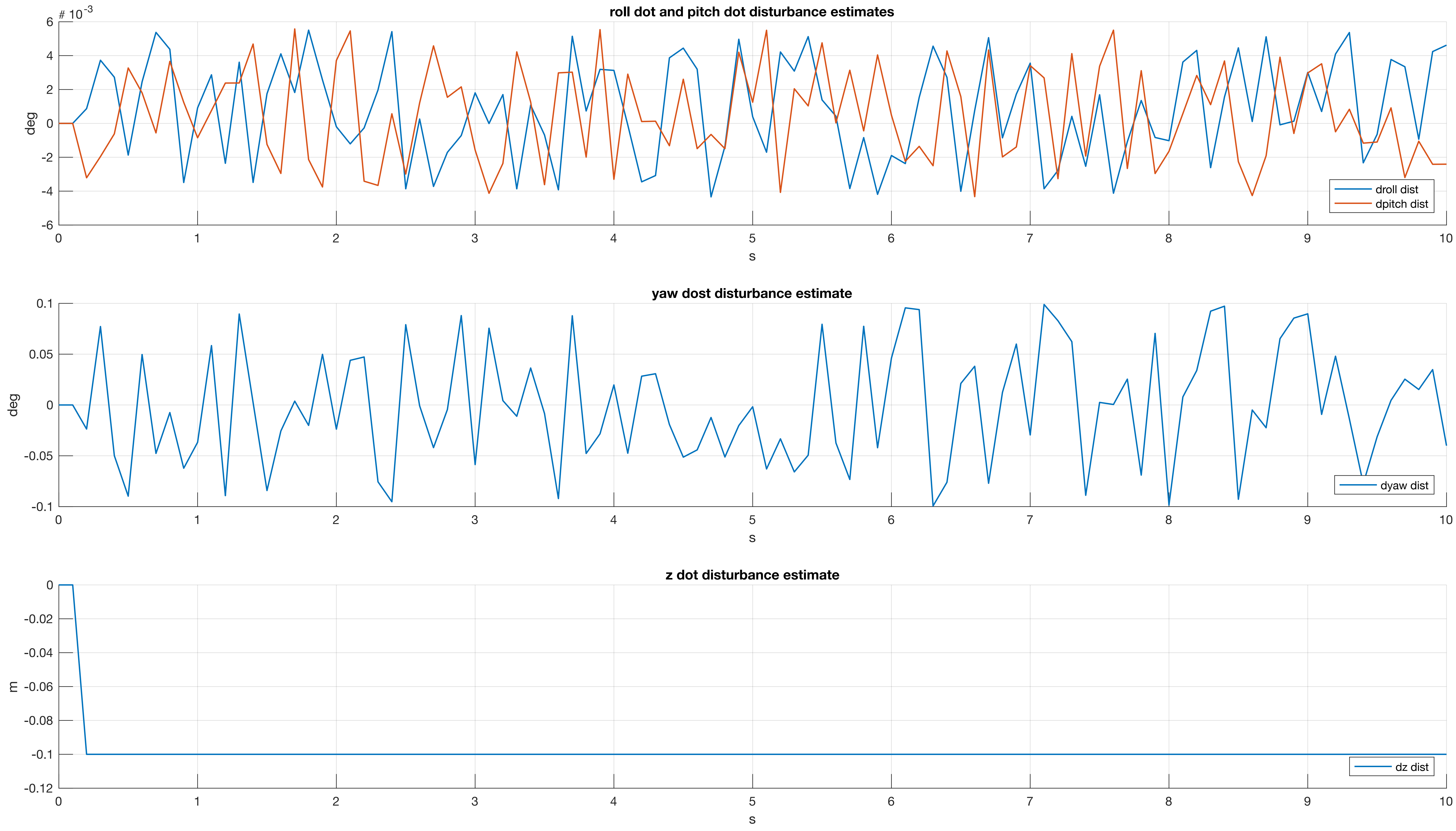
17. Comparison of running times:

MPC reference tracking (task 5):
Forces: 0.0081s
Quadprog: 0.0256s

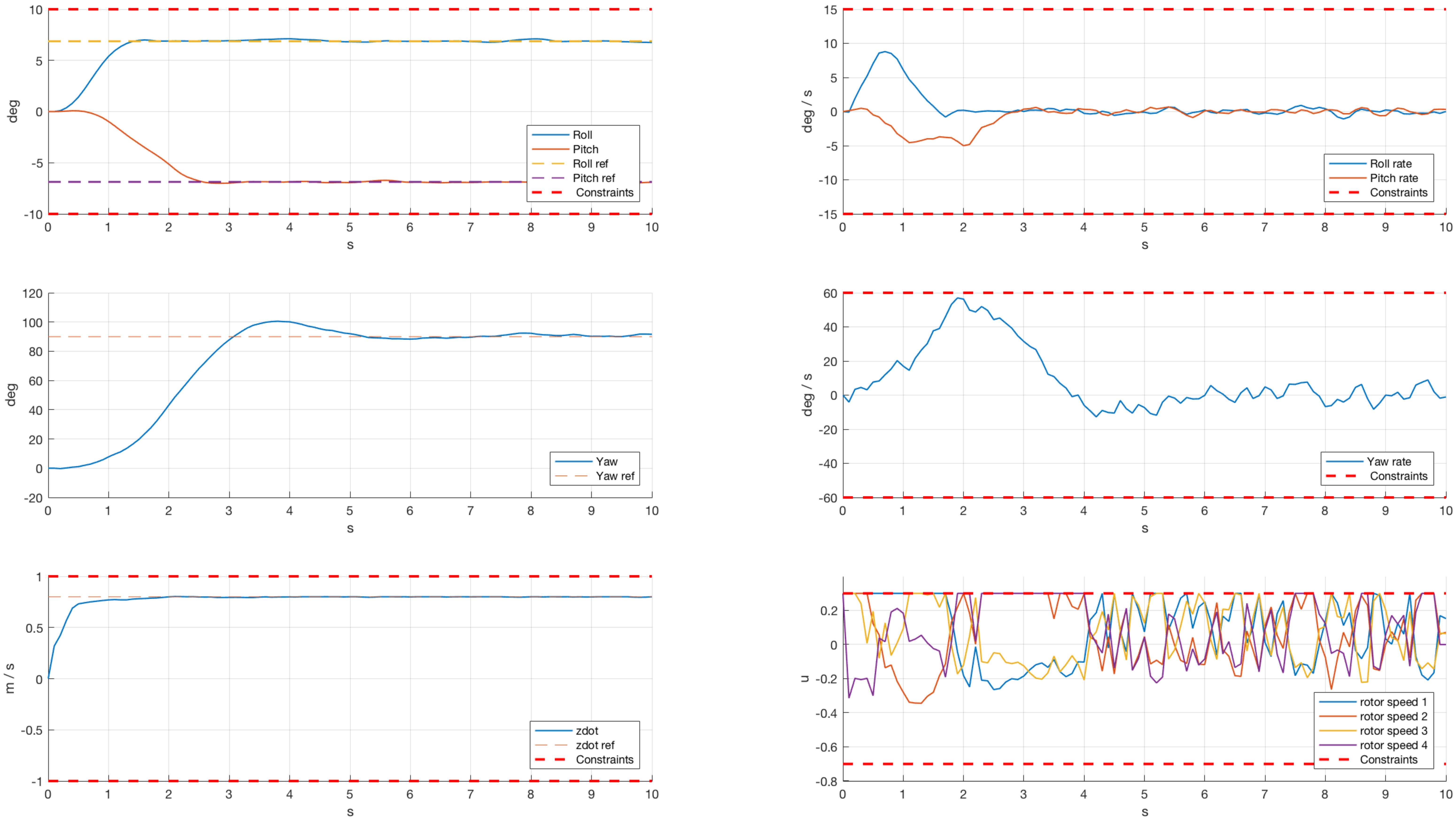
Offset free MPC (task 9):
Forces: 0.0082s
Quadprog: 0.0198s

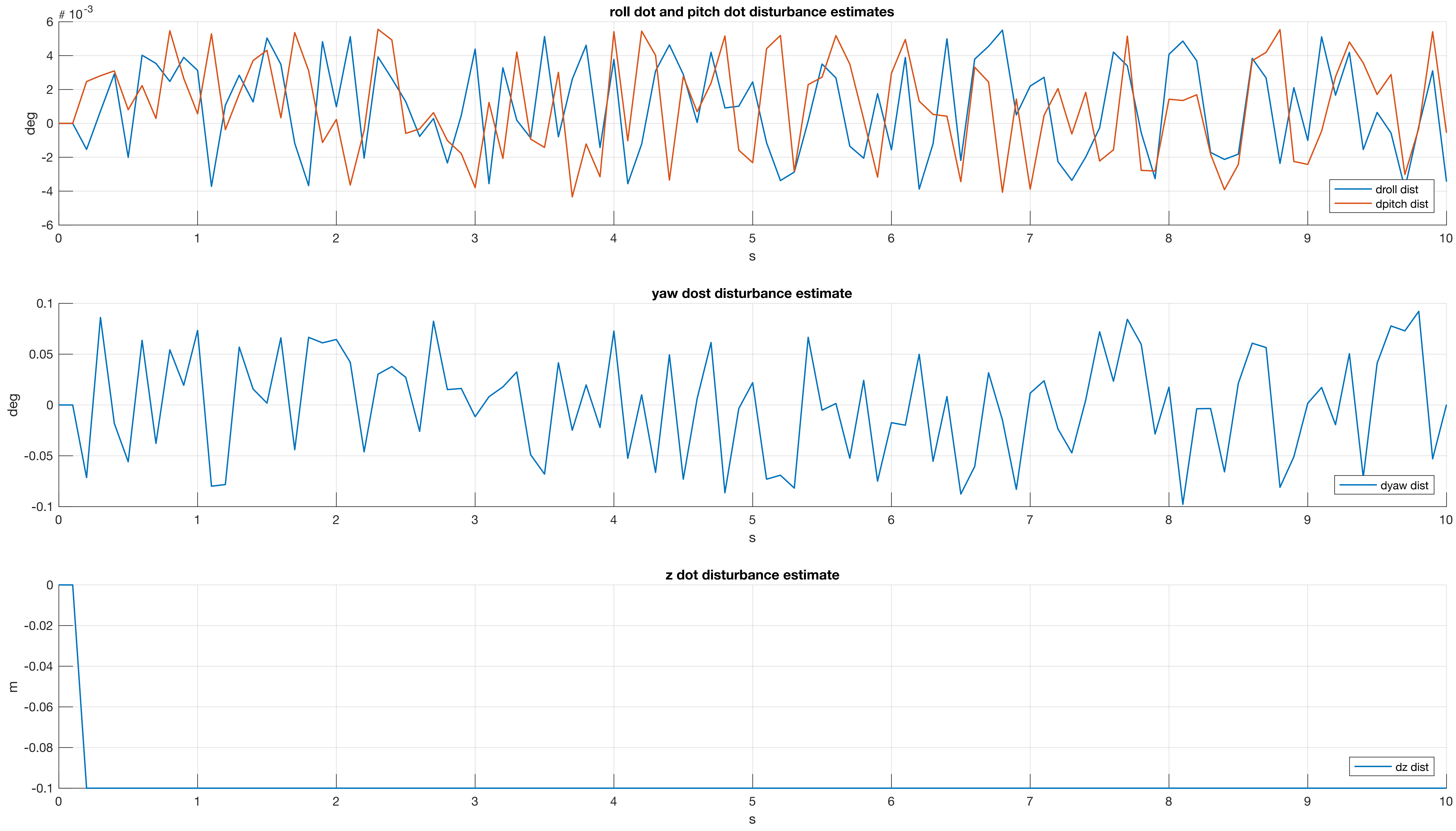


deg
s
Roll
Pitch
Roll ref
Pitch ref
Constraints
deg / s
Roll rate
Pitch rate
Constraints
deg
Yaw
Yaw ref
deg / s
Yaw rate
Constraints
m / s
zdot
zdot ref
Constraints
u
rotor speed 1
rotor speed 2
rotor speed 3
rotor speed 4
Constraints



roll dot and pitch dot disturbance estimates
10^-3
6
4
2
0
-2
-4
-6
deg
0
1
2
3
4
5
6
7
8
9
10
s
droll dist
dpitch dist
yaw dost disturbance estimate
0.1
0.05
0
-0.05
-0.1
deg
dyaw dist
z dot disturbance estimate
0
-0.02
-0.04
-0.06
-0.08
-0.1
-0.12
m
dz dist


deg
s
Roll
Pitch
Roll ref
Pitch ref
Constraints
deg / s
s
Roll rate
Pitch rate
Constraints
deg
s
Yaw
Yaw ref
deg / s
s
Yaw rate
Constraints
m / s
s
zdot
zdot ref
Constraints
u
s
rotor speed 1
rotor speed 2
rotor speed 3
rotor speed 4
Constraints


roll dot and pitch dot disturbance estimates
10-3
deg
s
droll dist
dpitch dist
yaw dost disturbance estimate
deg
s
dyaw dist
z dot disturbance estimate
m
s
dz dist