```
% Tank Level Estimator
% Jonas Wagner - 2020-10-29
clear
close all

% System Parameters ------------------------------------------
% Constants - Defined symbolic
syms H A_s A_r K_vv Gamma

% Constants - Estimates
h = 5;
a_s = pi * (0.25)^2;
a_r = pi * (0.75)^2;
k_vv = 0.25;
gamma = 0.1;

% State Varibles
syms x_s x_r
syms X_s_e X_r_e

X_s = x_s - X_s_e;
X_r = x_r - X_r_e;

x_s_e = 10;
x_r_e = 0.25;
x_e = [x_s_e, x_r_e];

x = [x_s; x_r];
X = [X_s; X_r];
X_e = [X_s_e; X_r_e];

% System Inputs
syms q_p u_v
syms Q_p_e U_v_e

Q_p = q_p - Q_p_e;
U_v = u_v - U_v_e;

q_p_e = 2;
u_v_e = 0.25;
u_e = [q_p_e, u_v_e];

u = [q_p; u_v];
U = [Q_p; U_v];
U_e = [Q_p_e; U_v_e];

% Symplifed Non-linear System Dynamics -----------------------------
% Simplified State Equations
f_1 = (1/A_s) * (Q_p - U_v * K_vv * sqrt(Gamma * (X_s + H - X_r)));
f_2 = (1/A_r) * (-Q_p + U_v * K_vv * sqrt(Gamma * (X_s + H - X_r)));

f = [f_1; f_2]
```

```
% Output Equations
g_1 = X_s;
g_2 = X_r;
% g_3 = Q_p;
% g_4 = U_v;

g = [g_1; g_2]%; g_3; g_4]

% Linearization ---------------------------------------
% Equalibrium Points
f_e = subs(f, [X_s, X_r, Q_p, U_v], [X_s_e, X_r_e, Q_p_e, U_v_e]);
g_e = subs(g, [X_s, X_r, Q_p, U_v], [X_s_e, X_r_e, Q_p_e, U_v_e]);

% System Matrices
A = subs(jacobian(f,x),[X,U],[X_e,U_e])
B = subs(jacobian(f,u),[X,U],[X_e,U_e])
C = subs(jacobian(g,x),[X,U],[X_e,U_e])
D = subs(jacobian(g,u),[X,U],[X_e,U_e])

% Linearized Equations
x_dot = A * x + B * u
y = C * x + D * u

X_dot = A * x + B * u + f_e;
Y = C * x + D * u;


% Discretization -------------------------------------------------
syms T
t_step = 1;
% System Matrices
F = exp(A * T)
% G = F*(eye(2) - exp(-A*T)) * inv(A) * B
% error... A is apparently singular... inversion doesnt work...
det(A);

% This method works though... idk
syms tau
G = F * int(exp(-A*tau), tau, 0, T) * B

% Linear DT System -----------------------------------------
F_num = double(subs(F,[H A_s A_r K_vv Gamma T X_s_e X_r_e Q_p_e
 U_v_e],...
                    [h a_s a_r k_vv gamma t_step x_s_e x_r_e q_p_e
 u_v_e]));
G_num = double(subs(G,[H A_s A_r K_vv Gamma T X_s_e X_r_e Q_p_e
 U_v_e],...
                    [h a_s a_r k_vv gamma t_step x_s_e x_r_e q_p_e
 u_v_e]));
C_num = double(C);
D_num = double(D);

sys = ss(F_num, G_num, C_num, D_num, t_step)
```

```
% Full-Order Observer Design ---------------------------------
p = [0.5 0.5];

L = place(F_num',C_num',p).'

A_obsv = F_num - L * C_num;
B_obsv = [G_num - L * D_num, L];
C_obsv = C_num;
D_obsv = [D_num, zeros(2)];

sys_obsv = ss(A_obsv,B_obsv,C_obsv,D_obsv,t_step)


% ode45 simulation method -----------------------------------
T1 = 0;
T2 = 500;
N = (T2-T1+1)/t_step;

t = linspace(T1,T2,N);
gt = [t;t];
g = [u_e(1) * sin(2*gt(1,:)/N);
    u_e(2) * ones(1,N)];

% tspan = [T1 T2];

% Simplified Nonlinear Modeling
[t,x] = ode45(@(t,x)
 TankLevelDynamics_SimplifiedNonlinear(t,x,gt,g,h,a_s,a_r,k_vv,gamma),
 t, x_e);
y = awgn(x,10,'measured');


% Estimator Testing
x_est = lsim(sys_obsv, [y, g'], t, zeros(1,2));

x_error = x(:,1:2) - x_est(:,1:2);

figure()
hold on
plot(t,x(:,1))
plot(t,y(:,1))
plot(t,x_est(:,1))
title('X_s Level Estimate')
hold off

figure()
hold on
plot(t,x(:,2))
plot(t,y(:,2))
plot(t,x_est(:,2))
title('X_r Level Estimate')
hold off
```

```
figure()
plot(t,x_error)
title('Full Order Observer Error')
```

```
f =

 (q_p - Q_p_e + K_vv*(U_v_e - u_v)*(Gamma*(H + X_r_e - X_s_e - x_r +
 x_s))^(1/2))/A_s
-(q_p - Q_p_e + K_vv*(U_v_e - u_v)*(Gamma*(H + X_r_e - X_s_e - x_r +
 x_s))^(1/2))/A_r
```

```
g =

x_s - X_s_e
x_r - X_r_e
```

```
A =

[-(Gamma*K_vv*U_v_e)/(2*A_s*(Gamma*(H - X_r_e + X_s_e))^(1/2)),
 (Gamma*K_vv*U_v_e)/(2*A_s*(Gamma*(H - X_r_e + X_s_e))^(1/2))]
[ (Gamma*K_vv*U_v_e)/(2*A_r*(Gamma*(H - X_r_e + X_s_e))^(1/2)), -
(Gamma*K_vv*U_v_e)/(2*A_r*(Gamma*(H - X_r_e + X_s_e))^(1/2))]
```

```
B =

[ 1/A_s, -(K_vv*(Gamma*(H - X_r_e + X_s_e))^(1/2))/A_s]
[-1/A_r,  (K_vv*(Gamma*(H - X_r_e + X_s_e))^(1/2))/A_r]
```

```
C =

[1, 0]
[0, 1]
```

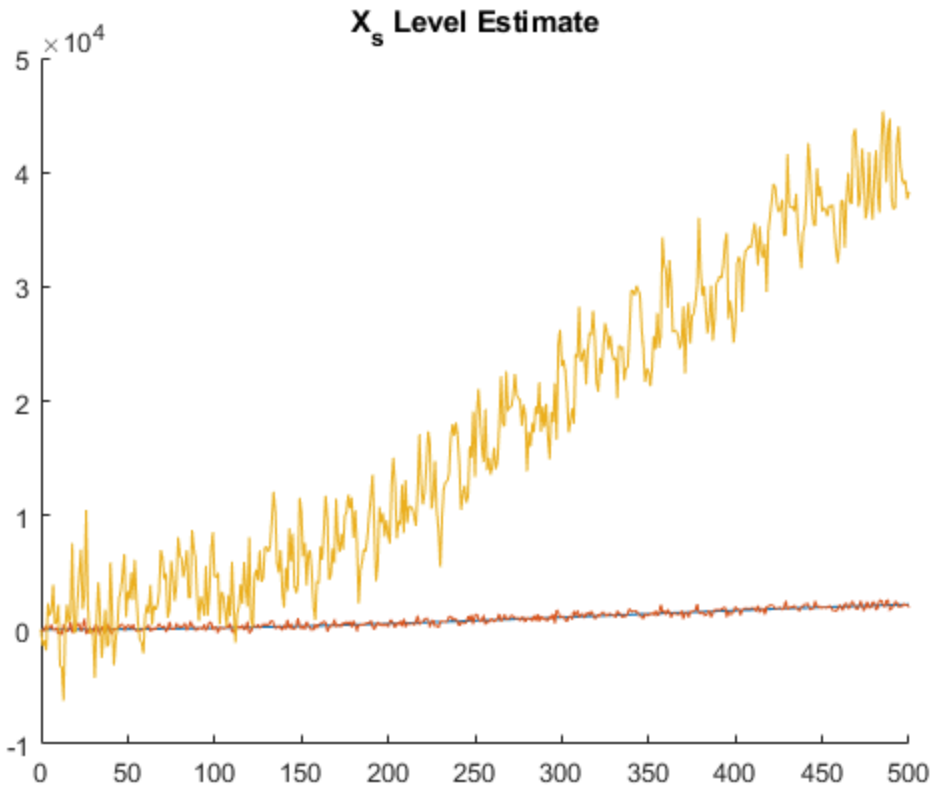
```
D =

[0, 0]
[0, 0]
```

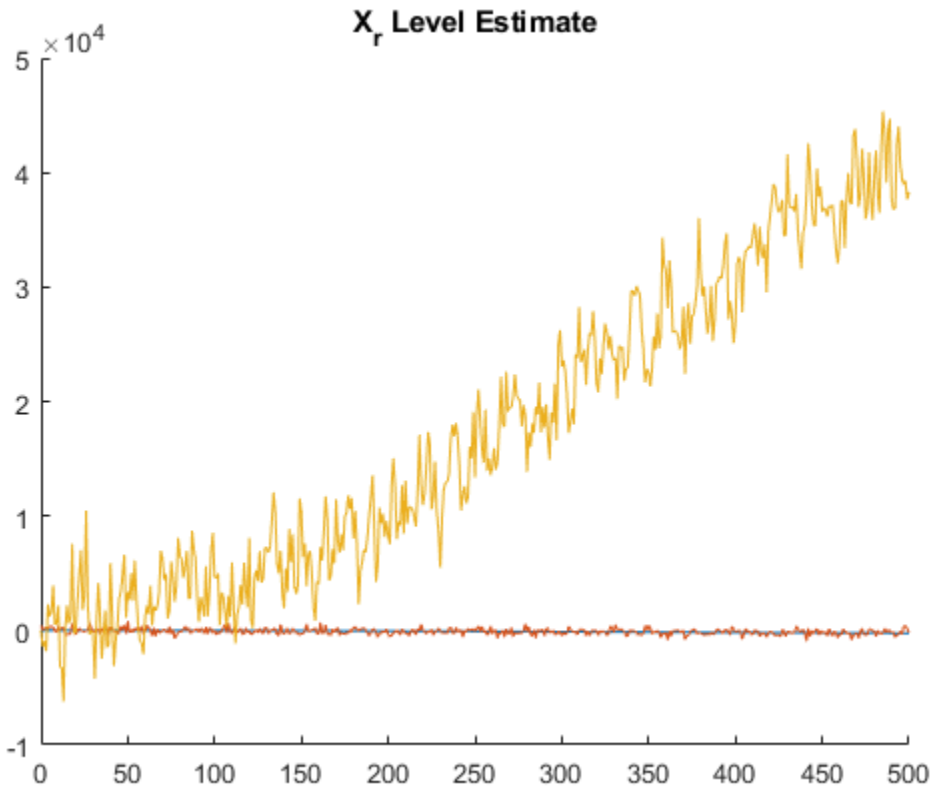
```
x_dot =

q_p/A_s - (K_vv*u_v*(Gamma*(H - X_r_e + X_s_e))^(1/2))/A_s +
 (Gamma*K_vv*U_v_e*x_r)/(2*A_s*(Gamma*(H - X_r_e + X_s_e))^(1/2)) -
 (Gamma*K_vv*U_v_e*x_s)/(2*A_s*(Gamma*(H - X_r_e + X_s_e))^(1/2))
(K_vv*u_v*(Gamma*(H - X_r_e + X_s_e))^(1/2))/A_r - q_p/A_r -
 (Gamma*K_vv*U_v_e*x_r)/(2*A_r*(Gamma*(H - X_r_e + X_s_e))^(1/2)) +
 (Gamma*K_vv*U_v_e*x_s)/(2*A_r*(Gamma*(H - X_r_e + X_s_e))^(1/2))
```

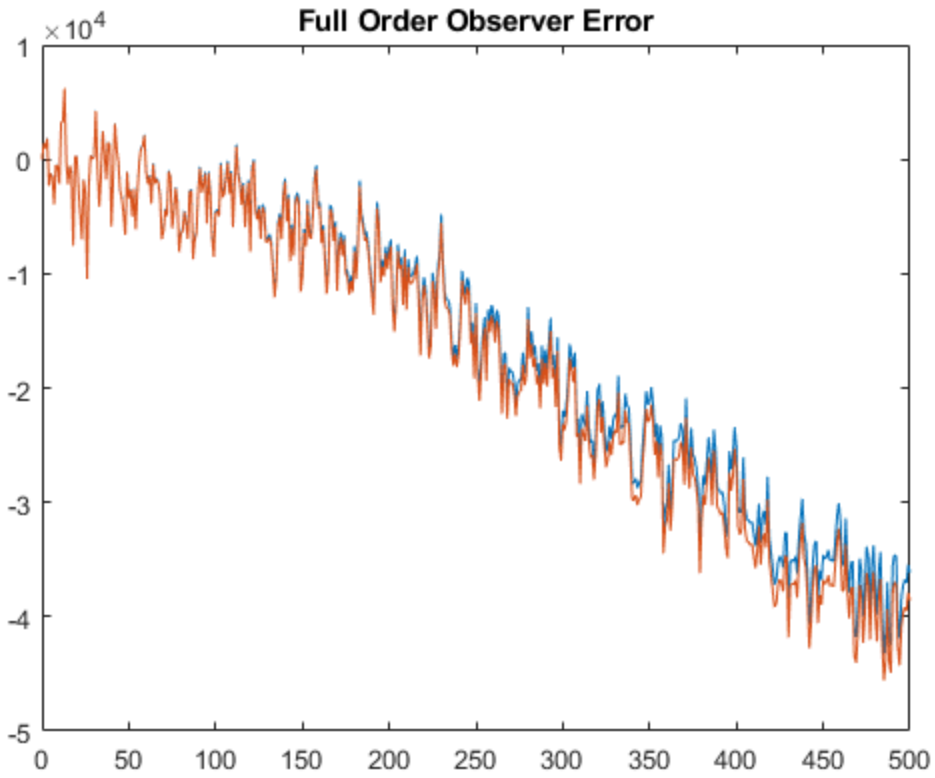
```
y =

x_s
x_r


F =

[exp(-(Gamma*K_vv*T*U_v_e)/(2*A_s*(Gamma*(H - X_r_e + X_s_e))^(1/2))),
  exp((Gamma*K_vv*T*U_v_e)/(2*A_s*(Gamma*(H - X_r_e + X_s_e))^(1/2)))]
[ exp((Gamma*K_vv*T*U_v_e)/(2*A_r*(Gamma*(H - X_r_e + X_s_e))^(1/2))),
 exp(-(Gamma*K_vv*T*U_v_e)/(2*A_r*(Gamma*(H - X_r_e + X_s_e))^(1/2)))]


G =

[(exp((Gamma*K_vv*T*U_v_e)/(2*A_s*(Gamma*(H - X_r_e
 + X_s_e))^(1/2)))*((2*A_r*(Gamma*(H - X_r_e +
 X_s_e))^(1/2))/(Gamma*K_vv*U_v_e) - (2*A_r*exp(-
(Gamma*K_vv*T*U_v_e)/(2*A_r*(Gamma*H - Gamma*X_r_e +
 Gamma*X_s_e)^(1/2)))*(Gamma*(H - X_r_e + X_s_e))^(1/2))/
(Gamma*K_vv*U_v_e)) + (2*A_s*exp(-(Gamma*K_vv*T*U_v_e)/
(2*A_s*(Gamma*(H - X_r_e + X_s_e))^(1/2)))*(Gamma*(H - X_r_e
 + X_s_e))^(1/2)*(exp((Gamma*K_vv*T*U_v_e)/(2*A_s*(Gamma*H -
 Gamma*X_r_e + Gamma*X_s_e)^(1/2))) - 1))/(Gamma*K_vv*U_v_e))/
A_s - (exp(-(Gamma*K_vv*T*U_v_e)/(2*A_s*(Gamma*(H - X_r_e +
 X_s_e))^(1/2)))*((2*A_s*(Gamma*(H - X_r_e + X_s_e))^(1/2))/
(Gamma*K_vv*U_v_e) - (2*A_s*exp(-(Gamma*K_vv*T*U_v_e)/(2*A_s*(Gamma*H
 - Gamma*X_r_e + Gamma*X_s_e)^(1/2)))*(Gamma*(H - X_r_e +
 X_s_e))^(1/2))/(Gamma*K_vv*U_v_e)) + (2*A_r*exp((Gamma*K_vv*T*U_v_e)/
(2*A_s*(Gamma*(H - X_r_e + X_s_e))^(1/2)))*(Gamma*(H - X_r_e
 + X_s_e))^(1/2)*(exp((Gamma*K_vv*T*U_v_e)/(2*A_r*(Gamma*H -
 Gamma*X_r_e + Gamma*X_s_e)^(1/2))) - 1))/(Gamma*K_vv*U_v_e))/
A_r, (K_vv*(exp(-(Gamma*K_vv*T*U_v_e)/(2*A_s*(Gamma*(H - X_r_e
 + X_s_e))^(1/2)))*((2*A_s*(Gamma*(H - X_r_e + X_s_e))^(1/2))/
(Gamma*K_vv*U_v_e) - (2*A_s*exp(-(Gamma*K_vv*T*U_v_e)/(2*A_s*(Gamma*H
 - Gamma*X_r_e + Gamma*X_s_e)^(1/2)))*(Gamma*(H - X_r_e +
 X_s_e))^(1/2))/(Gamma*K_vv*U_v_e)) + (2*A_r*exp((Gamma*K_vv*T*U_v_e)/
(2*A_s*(Gamma*(H - X_r_e + X_s_e))^(1/2)))*(Gamma*(H - X_r_e +
 X_s_e))^(1/2)*(exp((Gamma*K_vv*T*U_v_e)/(2*A_r*(Gamma*H - Gamma*X_r_e
 + Gamma*X_s_e)^(1/2))) - 1))/(Gamma*K_vv*U_v_e))*(Gamma*(H -
 X_r_e + X_s_e))^(1/2))/A_r - (K_vv*(exp((Gamma*K_vv*T*U_v_e)/
(2*A_s*(Gamma*(H - X_r_e + X_s_e))^(1/2)))*((2*A_r*(Gamma*(H
 - X_r_e + X_s_e))^(1/2))/(Gamma*K_vv*U_v_e) - (2*A_r*exp(-
(Gamma*K_vv*T*U_v_e)/(2*A_r*(Gamma*H - Gamma*X_r_e +
 Gamma*X_s_e)^(1/2)))*(Gamma*(H - X_r_e + X_s_e))^(1/2))/
(Gamma*K_vv*U_v_e)) + (2*A_s*exp(-(Gamma*K_vv*T*U_v_e)/
(2*A_s*(Gamma*(H - X_r_e + X_s_e))^(1/2)))*(Gamma*(H - X_r_e +
 X_s_e))^(1/2)*(exp((Gamma*K_vv*T*U_v_e)/(2*A_s*(Gamma*H - Gamma*X_r_e
 + Gamma*X_s_e)^(1/2))) - 1))/(Gamma*K_vv*U_v_e))*(Gamma*(H - X_r_e +
 X_s_e))^(1/2))/A_s]
```

```
[(exp(-(Gamma*K_vv*T*U_v_e)/(2*A_r*(Gamma*(H - X_r_e +
 X_s_e))^(1/2)))*((2*A_r*(Gamma*(H - X_r_e + X_s_e))^(1/2))/
(Gamma*K_vv*U_v_e) - (2*A_r*exp(-(Gamma*K_vv*T*U_v_e)/(2*A_r*(Gamma*H
 - Gamma*X_r_e + Gamma*X_s_e)^(1/2)))*(Gamma*(H - X_r_e +
 X_s_e))^(1/2))/(Gamma*K_vv*U_v_e)) + (2*A_s*exp((Gamma*K_vv*T*U_v_e)/
(2*A_r*(Gamma*(H - X_r_e + X_s_e))^(1/2)))*(Gamma*(H
 - X_r_e + X_s_e))^(1/2)*(exp((Gamma*K_vv*T*U_v_e)/
(2*A_s*(Gamma*H - Gamma*X_r_e + Gamma*X_s_e)^(1/2))) -
 1))/(Gamma*K_vv*U_v_e))/A_s - (exp((Gamma*K_vv*T*U_v_e)/
(2*A_r*(Gamma*(H - X_r_e + X_s_e))^(1/2)))*((2*A_s*(Gamma*(H
 - X_r_e + X_s_e))^(1/2))/(Gamma*K_vv*U_v_e) - (2*A_s*exp(-
(Gamma*K_vv*T*U_v_e)/(2*A_s*(Gamma*H - Gamma*X_r_e +
 Gamma*X_s_e)^(1/2)))*(Gamma*(H - X_r_e + X_s_e))^(1/2))/
(Gamma*K_vv*U_v_e)) + (2*A_r*exp(-(Gamma*K_vv*T*U_v_e)/
(2*A_r*(Gamma*(H - X_r_e + X_s_e))^(1/2)))*(Gamma*(H
 - X_r_e + X_s_e))^(1/2)*(exp((Gamma*K_vv*T*U_v_e)/
(2*A_r*(Gamma*H - Gamma*X_r_e + Gamma*X_s_e)^(1/2))) - 1))/
(Gamma*K_vv*U_v_e))/A_r, (K_vv*(exp((Gamma*K_vv*T*U_v_e)/
(2*A_r*(Gamma*(H - X_r_e + X_s_e))^(1/2)))*((2*A_s*(Gamma*(H
 - X_r_e + X_s_e))^(1/2))/(Gamma*K_vv*U_v_e) - (2*A_s*exp(-
(Gamma*K_vv*T*U_v_e)/(2*A_s*(Gamma*H - Gamma*X_r_e +
 Gamma*X_s_e)^(1/2)))*(Gamma*(H - X_r_e + X_s_e))^(1/2))/
(Gamma*K_vv*U_v_e)) + (2*A_r*exp(-(Gamma*K_vv*T*U_v_e)/
(2*A_r*(Gamma*(H - X_r_e + X_s_e))^(1/2)))*(Gamma*(H - X_r_e +
 X_s_e))^(1/2)*(exp((Gamma*K_vv*T*U_v_e)/(2*A_r*(Gamma*H - Gamma*X_r_e
 + Gamma*X_s_e)^(1/2))) - 1))/(Gamma*K_vv*U_v_e))*(Gamma*(H -
 X_r_e + X_s_e))^(1/2))/A_r - (K_vv*(exp(-(Gamma*K_vv*T*U_v_e)/
(2*A_r*(Gamma*(H - X_r_e + X_s_e))^(1/2)))*((2*A_r*(Gamma*(H
 - X_r_e + X_s_e))^(1/2))/(Gamma*K_vv*U_v_e) - (2*A_r*exp(-
(Gamma*K_vv*T*U_v_e)/(2*A_r*(Gamma*H - Gamma*X_r_e +
 Gamma*X_s_e)^(1/2)))*(Gamma*(H - X_r_e + X_s_e))^(1/2))/
(Gamma*K_vv*U_v_e)) + (2*A_s*exp((Gamma*K_vv*T*U_v_e)/
(2*A_r*(Gamma*(H - X_r_e + X_s_e))^(1/2)))*(Gamma*(H - X_r_e +
 X_s_e))^(1/2)*(exp((Gamma*K_vv*T*U_v_e)/(2*A_s*(Gamma*H - Gamma*X_r_e
 + Gamma*X_s_e)^(1/2))) - 1))/(Gamma*K_vv*U_v_e))*(Gamma*(H - X_r_e +
 X_s_e))^(1/2))/A_s]


sys =

  A =
           x1      x2
   x1   0.987   1.013
   x2   1.001  0.9985

  B =
           u1      u2
   x1   9.087  -2.759
   x2   9.087  -2.759

  C =
       x1  x2
   y1   1   0
   y2   0   1
```

```
  D =
       u1  u2
   y1   0   0
   y2   0   0

Sample time: 1 seconds
Discrete-time state-space model.


L =

    0.4870    1.0132
    1.0015    0.4985


sys_obsv =

  A =
        x1   x2
   x1  0.5    0
   x2    0  0.5

  B =
           u1      u2      u3      u4
   x1   9.087  -2.759   0.487   1.013
   x2   9.087  -2.759   1.001  0.4985

  C =
       x1  x2
   y1   1   0
   y2   0   1

  D =
       u1  u2  u3  u4
   y1   0   0   0   0
   y2   0   0   0   0

Sample time: 1 seconds
Discrete-time state-space model.
```


X_s Level Estimate
$\times 10^4$
5
4
3
2
1
0
-1
0
50
100
150
200
250
300
350
400
450
500


X_r Level Estimate
$\times 10^4$
5
4
3
2
1
0
-1
0
50
100
150
200
250
300
350
400
450
500



Published with MATLAB® R2020b