3(a) Second Order Population Growth system

Author: Mohammed Ijaz PS Number: 99003728 Date: 11th April 2021. Version: MATLAB 2020b.

Plant Description

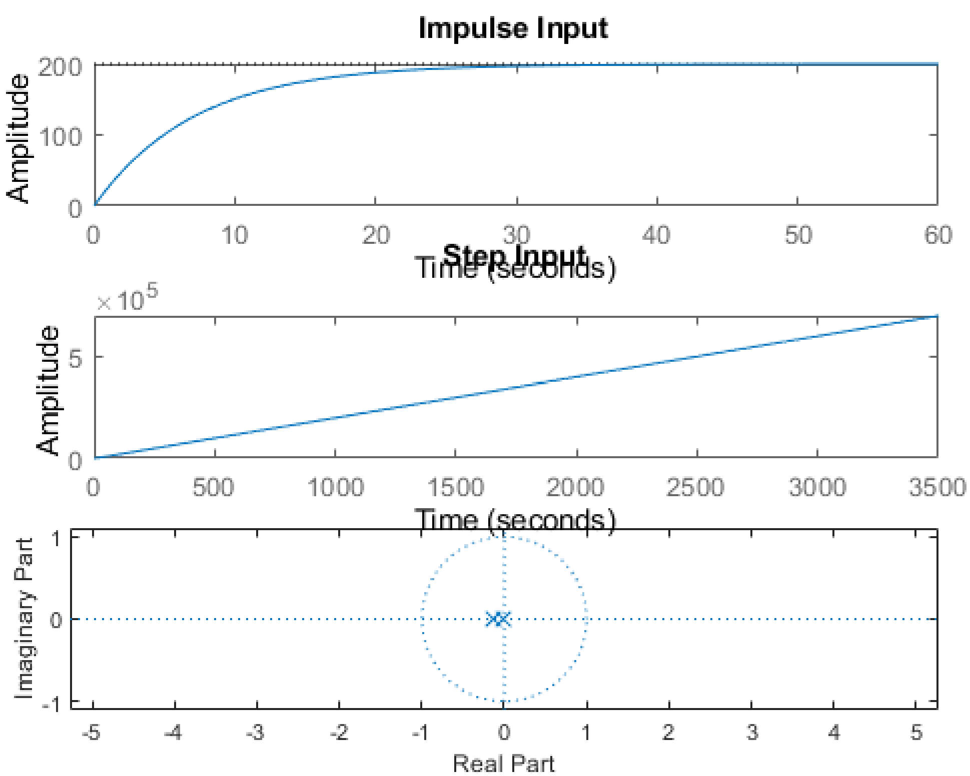
It is a exponential increasing system of a Population Growth Equation- dN(t)/dt=kN(t) N=Population a time t, k=constant, t=time Values- k=0.14, A=200

Without Controller

```
clc;
k= 0.14;
A= 200;
sys = tf([k*A],[1,k,0])
figure(1);
subplot(3,1,1);
impulse(sys);
title('Impulse Input');
subplot(3,1,2);
step(sys);
title('Step Input');
[z,p,k] = tf2zp([k*A],[1,k,0])
subplot(3,1,3);
zplane(z,p);
S = stepinfo(sys)
```

```
sys =

       28
  ------------
  s^2 + 0.14 s

Continuous-time transfer function.


z =

  0×1 empty double column vector
```

```
p =

         0
   -0.1400


k =

   28.0000


S =

  struct with fields:

        RiseTime: NaN
    SettlingTime: NaN
     SettlingMin: NaN
     SettlingMax: NaN
       Overshoot: NaN
      Undershoot: NaN
            Peak: Inf
        PeakTime: Inf
```

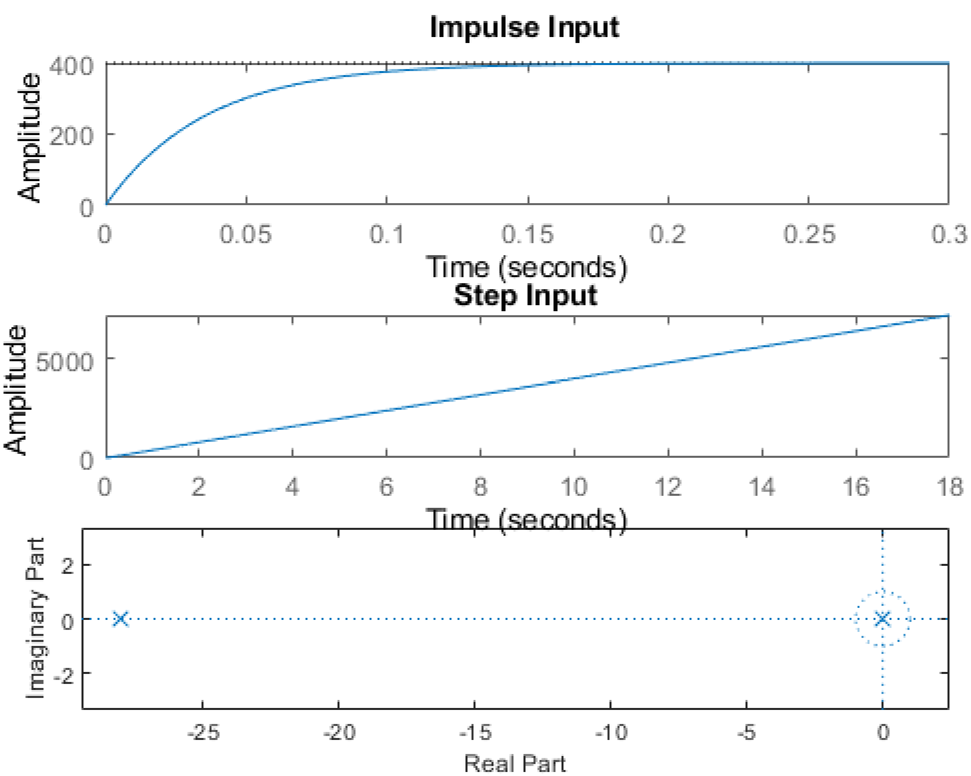


Open Loop with Controller (P)

```
P= 2;
sys = tf([P*(k)*A],[1,k,0])
figure(2);
subplot(3,1,1);
impulse(sys);
title('Impulse Input');
subplot(3,1,2);
step(sys);
title('Step Input');
[z,p,k] = tf2zp([P*k*A],[1,k,0])
subplot(3,1,3);
zplane(z,p);
S = stepinfo(sys)
```

```
sys =

   1.12e04
  ----------
  s^2 + 28 s

Continuous-time transfer function.


z =

  0×1 empty double column vector


p =

         0
  -28.0000


k =

   1.1200e+04


S =

  struct with fields:

        RiseTime: NaN
    SettlingTime: NaN
     SettlingMin: NaN
     SettlingMax: NaN
       Overshoot: NaN
      Undershoot: NaN
            Peak: Inf
```

```
    PeakTime: Inf
```

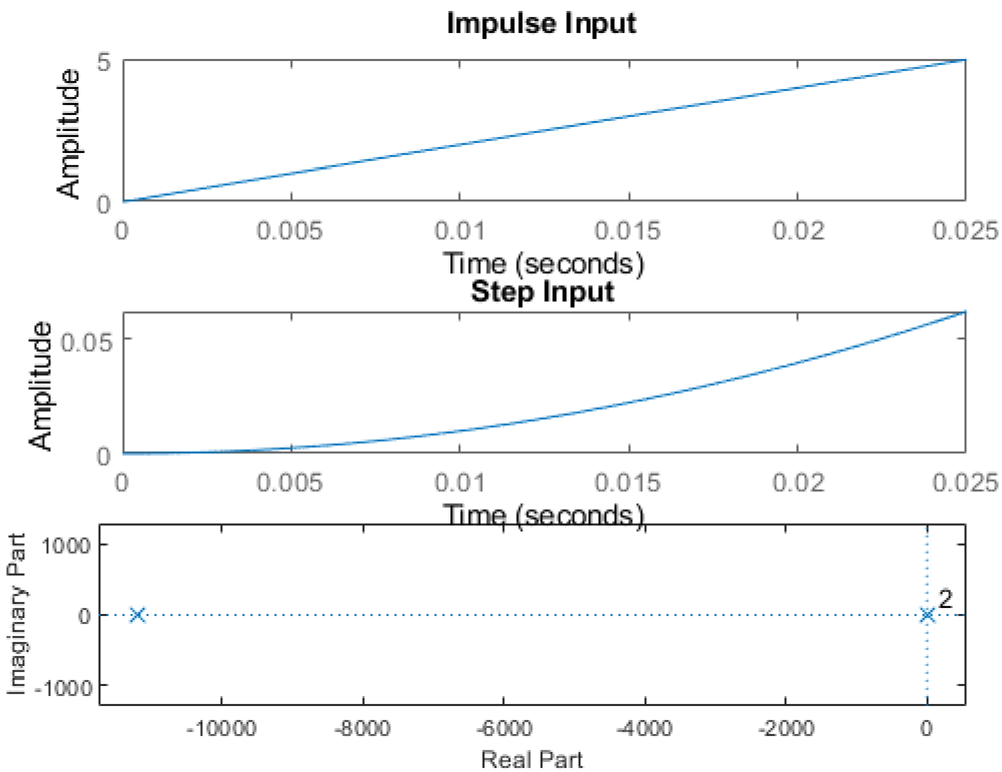


Open Loop with Controller (I)

```
sys = tf([k*A],[1,k,0,0])
figure(3);
subplot(3,1,1);
impulse(sys);
title('Impulse Input');
subplot(3,1,2);
step(sys);
title('Step Input');
[z,p,k] = tf2zp([k*A],[1,k,0,0])
subplot(3,1,3);
zplane(z,p);
S = stepinfo(sys)
```

```
sys =

       2.24e06
  -----------------
  s^3 + 1.12e04 s^2

Continuous-time transfer function.
```

```
z =

  0×1 empty double column vector


p =

   1.0e+04 *

         0
         0
   -1.1200


k =

   2.2400e+06


S =

  struct with fields:

        RiseTime: NaN
    SettlingTime: NaN
     SettlingMin: NaN
     SettlingMax: NaN
       Overshoot: NaN
      Undershoot: NaN
            Peak: Inf
        PeakTime: Inf
```

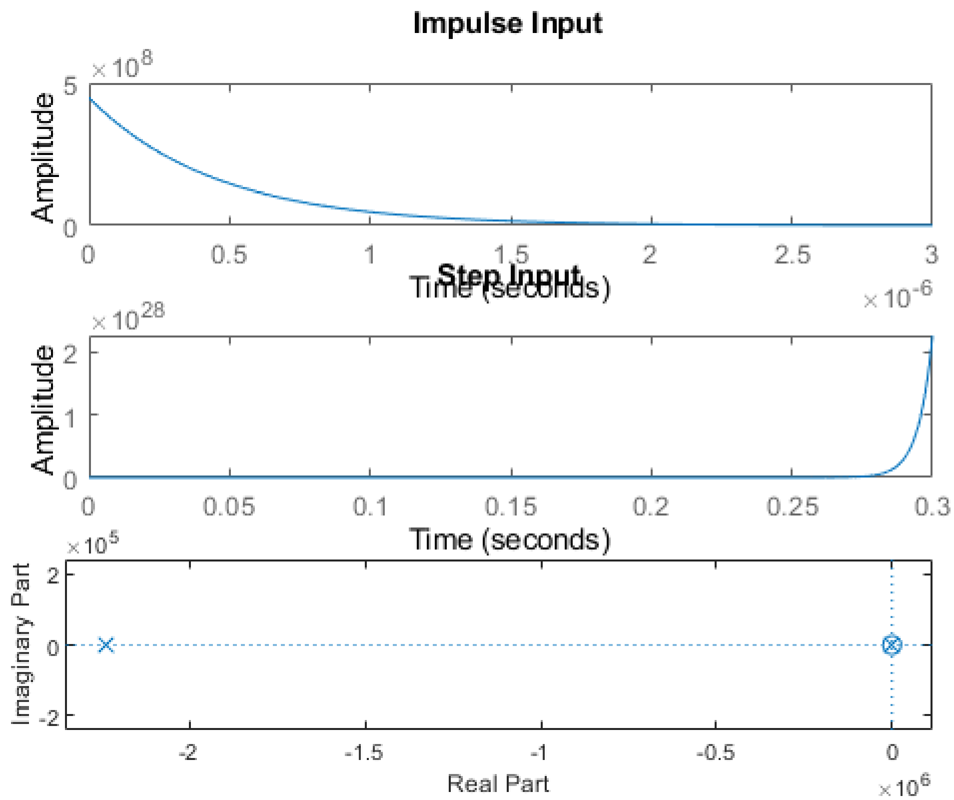


Closed Loop- Negative feedback with Controller (D)

```
sys = tf([k*A,0],[1,k,(-k)*A])
figure(4);
subplot(3,1,1);
impulse(sys);
title('Impulse Input');
subplot(3,1,2);
step(sys);
title('Step Input');
[z,p,k] = tf2zp([k*A,0],[1,k,(-k)*A])
subplot(3,1,3);
zplane(z,p);
S = stepinfo(sys)
```

```
sys =

          4.48e08 s
  -------------------------
  s^2 + 2.24e06 s - 4.48e08

Continuous-time transfer function.


z =
```

```
     0


p =

   1.0e+06 *

   -2.2402
    0.0002


k =

   4.4800e+08


S =

  struct with fields:

        RiseTime: NaN
    SettlingTime: NaN
     SettlingMin: NaN
     SettlingMax: NaN
       Overshoot: NaN
      Undershoot: NaN
            Peak: Inf
        PeakTime: Inf
```

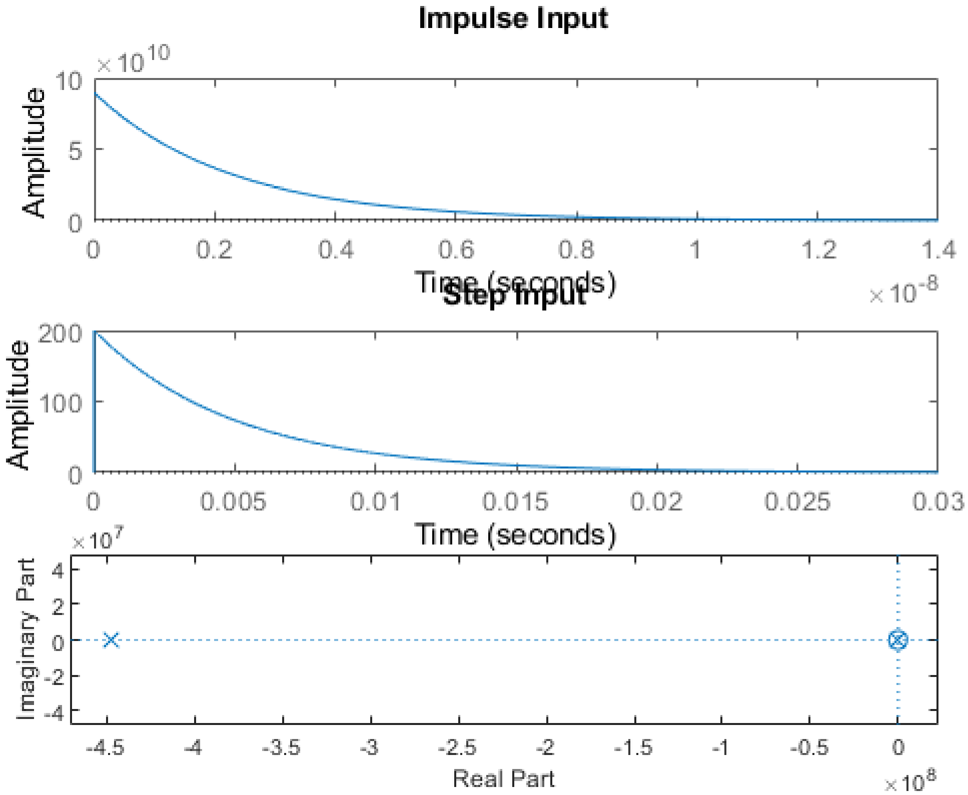


Closed Loop- Positive feedback with Controller (D)

```
sys = tf([k*A,0],[1,k,k*A])
figure(5);
subplot(3,1,1);
impulse(sys);
title('Impulse Input');
subplot(3,1,2);
step(sys);
title('Step Input');
[z,p,k] = tf2zp([k*A,0],[1,k,k*A])
subplot(3,1,3);
zplane(z,p);
S = stepinfo(sys)
```

```
sys =

         8.96e10 s
  -------------------------
  s^2 + 4.48e08 s + 8.96e10

Continuous-time transfer function.


z =
```

```
     0


p =

   1.0e+08 *

   -4.4800
   -0.0000


k =

   8.9600e+10


S = 

  struct with fields:

        RiseTime: 0
    SettlingTime: 0.0196
     SettlingMin: 0.2598
     SettlingMax: 199.9633
       Overshoot: Inf
      Undershoot: 0
            Peak: 199.9633
        PeakTime: 9.2103e-07
```



Math Analysis

Independent: Time(t) Dependent: Population(N) Constant: Constant(A)

Comparison Analysis

1) System without controller behaves exactly like an exponential Growth. with the system growing exponentially. 2) On adding a proportionality controller to system , the system becomes unstable. 3) On adding a Integrator controller to system, the response times have decreased hugely, making the system reach stablity faster than a P controller. 4) Integrator controller adds a pole to zero also. 5) On addition of a differentiator controller in negative feedback the system becomes unstable. 6) A zero gets added at origin due to the differentiator. 7) On addition of a differentiator controller in positive feedback the system becomes stable.