

## **Parte II**

### **O vírus CODIG – Espaço de Estados**

Diogo Alexandre Brandão da Silva  
FEUP/DEEC Porto, Portugal  
up201809213@fe.up.pt

Grupo 07

Manuel João Narciso Ribeiro  
FEUP/DEEC Porto, Portugal  
up201603213@fe.up.pt

José Maria Queirós Rodrigues Sarmento  
FEUP/DEEC Porto, Portugal  
up201909931@fe.up.pt

Paulo Alexandre Sousa Silva  
FEUP/DEEC Porto, Portugal  
up201909556@fe.up.pt

## **Introdução**

O conceito de modelo epidemiológico, é termo recorrente nos dias de hoje, devido à situação que nos encontramos.

O presente documento pretende demonstrar o trabalho desenvolvido para a segunda parte de uma aplicação de um destes modelos numa situação em concreto (vírus CODIG) utilizando para isto a análise com ferramentas de espaço estados após linearização do problema.

### **1. Análise do sistema por espaço de estados em tempo contínuo.**

#### **A. Verificação do ponto de funcionamento ( $S^*$ , $I_D^*$ , $I_H^*$ , $R^*$ ) como ponto de equilíbrio.**

Para as demonstrações que se seguem, como se está a trabalhar em volta de um ponto escrever por exemplo  $S(t)$  ou  $S^*$  significa a mesma coisa.

Do enunciado sabe-se que:

$$\dot{S}(t) = -(\alpha_1 + \alpha_2)S(t) - vS(t) + \rho R(t) \quad (1)$$

$$\dot{I}_D(t) = \alpha_1 S(t) - \gamma_1 I_D(t) \quad (2)$$

$$\dot{I}_H(t) = \alpha_2 S(t) - \gamma_2 I_H(t) \quad (3)$$

$$\dot{R}(t) = vS(t) + \gamma_1 I_D(t) + \gamma_2 I_H(t) - \rho R(t) \quad (4)$$

Para o ponto de equilíbrio sabe-se que:

$$U^* = R_0 - \frac{N}{0.85 S^*} \quad (5)$$

$$\frac{I_H^*}{I_D^*} = \frac{0.15}{0.85} \quad (6)$$

$$S^* = \frac{\gamma}{\alpha^*} (I_H^* + I_D^*) \quad (7)$$

$$R^* = \frac{(\alpha^* + v^*) S^*}{\rho} \quad (8)$$

Onde:

$$\alpha = \alpha_1 + \alpha_2 \quad (9)$$

$$\alpha_1 = 0.85\alpha \quad (10)$$

$$\alpha_2 = 0.15\alpha \quad (11)$$

$$\gamma = \gamma_1 = \gamma_2 = \frac{1}{T_{inf}} \quad (12)$$

Pretende-se obter:

$$\dot{S}(t) = 0$$

$$\dot{I}_D(t) = 0$$

$$\dot{I}_H(t) = 0$$

$$\dot{R}(t) = 0$$

Com (1), (7), (8) e (9) tem-se:

$$\begin{aligned} \dot{S}(t) &= -\alpha \frac{\gamma}{\alpha^*} (I_H^* + I_D^*) - v \frac{\gamma}{\alpha^*} (I_H^* + I_D^*) + \rho \frac{(\alpha^* + v^*)}{\rho} \frac{\gamma}{\alpha^*} (I_H^* + I_D^*) \\ &= -\gamma (I_H^* + I_D^*) - v \frac{\gamma}{\alpha^*} (I_H^* + I_D^*) + \gamma (I_H^* + I_D^*) + v^* \frac{\gamma}{\alpha^*} (I_H^* + I_D^*) = 0 \end{aligned}$$

Com (2), (6), (7), (10) e (12) tem-se:

$$\dot{I}_D(t) = 0.85\alpha \frac{\gamma}{\alpha^*} \left( \frac{0.15}{0.85} I_D^* + I_D^* \right) - \gamma I_D(t) = \gamma I_D^* - \gamma I_D(t) = 0$$

Com (3), (6), (7), (11) e (12) tem-se:

$$\dot{I}_H(t) = 0.15\alpha \frac{\gamma}{\alpha^*} \left( \frac{0.85}{0.15} I_H^* + I_H^* \right) - \gamma I_H(t) = \gamma I_H^* - \gamma I_H(t) = 0$$

Com (4), (7), (8) e (12) tem-se:

$$\begin{aligned} \dot{R}(t) &= v \frac{\gamma}{\alpha^*} (I_H^* + I_D^*) + \gamma I_D(t) + \gamma I_H(t) - \rho \frac{(\alpha^* + v^*)}{\rho} \frac{\gamma}{\alpha^*} (I_H^* + I_D^*) \\ &= v \frac{\gamma}{\alpha^*} (I_H^* + I_D^*) + \gamma (I_D(t) + I_H(t)) - (I_H^* + I_D^*) - v^* \frac{\gamma}{\alpha^*} (I_H^* + I_D^*) = 0 \end{aligned}$$

**B. Verificação que valores ( $S^*$ ,  $I_D^*$ ,  $I_H^*$ ,  $R^*$ ) formam ponto de equilíbrio com  $v^* = 0$ . Cálculo de  $U^*$  nessas condições.**

Do enunciado temos que:

$$N = 10 \times 10^6$$

$$I^* = 2 \times 10^5 = I_H^* + I_D^*$$

$$R^* = 10I^* = 20 \times 10^5$$

$$S^* = N - I^* - R^* = 78 \times 10^5$$

$$\rho = 1/60$$

$$\gamma = 1/6$$

$$R_0 = 2.4 \text{ (enunciado parte I)}$$

Para (7) temos:

$$S^* = \frac{\gamma}{\alpha^*} (I_H^* + I_D^*) = \frac{I^*}{6\alpha^*} \leftrightarrow I^* = 6\alpha^* S^* \quad (13)$$

Para (8) temos:

$$R^* = \frac{(\alpha^* + v^*)S^*}{\rho} = \frac{\alpha^* S^*}{\frac{1}{60}} \leftrightarrow 10I^* = \frac{\alpha^* S^*}{\frac{1}{60}} \leftrightarrow I^* = 6\alpha^* S^* \quad (14)$$

Como através de (13) e (14) se obtém o mesmo resultado, é possível comprovar que para  $v^* = 0$  os valores do ponto definido formam um ponto de equilíbrio.

Para o cálculo de  $U^*$  utiliza-se (5):

$$U^* = R_0 - \frac{N}{0.85 S^*} = 2.4 - \frac{10 \times 10^6}{0.85 \times 78 \times 10^5} = 0.8917$$

**C. Linearização do sistema dinâmico em torno do ponto de equilíbrio ( $S^*$ ,  $I_D^*$ ,  $I_H^*$ ,  $R^*$ ).**

Do enunciado temos que:

$$\alpha = (R_0 - U)\gamma \frac{I_D(t)}{N} \quad (15)$$

Queremos linearizar (1), (2), (3) e (4).

Utilizando então (9), (10), (11), (12) e (15) em (1), (2), (3) e (4) tem-se:

$$\dot{S}(t) = -(R_0 - U)\gamma \frac{I_D(t)}{N} S(t) - vS(t) + \rho R(t) = f_1(S, I_D, I_H, R, U, v) \quad (16)$$

$$\dot{I}_D(t) = 0.85(R_0 - U)\gamma \frac{I_D(t)}{N} S(t) - \gamma I_D(t) = f_2(S, I_D, I_H, R, U, v) \quad (17)$$

$$\dot{I}_H(t) = 0.15(R_0 - U)\gamma \frac{I_D(t)}{N} S(t) - \gamma I_H(t) = f_3(S, I_D, I_H, R, U, v) \quad (18)$$

$$\dot{R}(t) = vS(t) + \gamma I_D(t) + \gamma I_H(t) - \rho R(t) = f_4(S, I_D, I_H, R, U, v) \quad (19)$$

Como queremos linearizar (16), (17), (18) e (19) em torno de um ponto temos:

$$\begin{cases} \dot{S}(t) = \frac{\partial f_1}{\partial S} S + \frac{\partial f_1}{\partial I_D} I_D + \frac{\partial f_1}{\partial I_H} I_H + \frac{\partial f_1}{\partial R} R + \frac{\partial f_1}{\partial U} U + \frac{\partial f_1}{\partial v} V \\ \dot{I}_D(t) = \frac{\partial f_2}{\partial S} S + \frac{\partial f_2}{\partial I_D} I_D + \frac{\partial f_2}{\partial I_H} I_H + \frac{\partial f_2}{\partial R} R + \frac{\partial f_2}{\partial U} U + \frac{\partial f_2}{\partial v} V \\ \dot{I}_H(t) = \frac{\partial f_3}{\partial S} S + \frac{\partial f_3}{\partial I_D} I_D + \frac{\partial f_3}{\partial I_H} I_H + \frac{\partial f_3}{\partial R} R + \frac{\partial f_3}{\partial U} U + \frac{\partial f_3}{\partial v} V \\ \dot{R}(t) = \frac{\partial f_4}{\partial S} S + \frac{\partial f_4}{\partial I_D} I_D + \frac{\partial f_4}{\partial I_H} I_H + \frac{\partial f_4}{\partial R} R + \frac{\partial f_4}{\partial U} U + \frac{\partial f_4}{\partial v} V \end{cases} \quad (20)$$

Sendo:

$$\begin{cases} \frac{\partial f_1}{\partial S} = -(R_0 - U^*)\gamma \frac{I_D^*}{N} - v \\ \frac{\partial f_1}{\partial I_D} = -(R_0 - U^*)\gamma \frac{S^*}{N} \\ \frac{\partial f_1}{\partial I_H} = 0 \\ \frac{\partial f_1}{\partial R} = \rho \\ \frac{\partial f_1}{\partial U} = \gamma \frac{I_D^*}{N} S^* \\ \frac{\partial f_1}{\partial v} = -S^* \end{cases} \quad (21)$$

$$\begin{cases} \frac{\partial f_2}{\partial S} = 0.85(R_0 - U^*)\gamma \frac{I_D^*}{N} \\ \frac{\partial f_2}{\partial I_D} = 0.85(R_0 - U^*)\gamma \frac{S^*}{N} - \gamma \\ \frac{\partial f_2}{\partial I_H} = 0 \\ \frac{\partial f_2}{\partial R} = 0 \\ \frac{\partial f_2}{\partial U} = -0.85\gamma \frac{I_D^*}{N} S^* \\ \frac{\partial f_2}{\partial v} = 0 \end{cases} \quad (22)$$

$$\left\{ \begin{array}{l} \frac{\partial f_3}{\partial S} = 0.15(R_0 - U^*)\gamma \frac{I_D^*}{N} \\ \frac{\partial f_3}{\partial I_D} = 0.15(R_0 - U^*)\gamma \frac{S^*}{N} \\ \frac{\partial f_3}{\partial I_H} = -\gamma \\ \frac{\partial f_3}{\partial R} = 0 \\ \frac{\partial f_3}{\partial U} = -0.15\gamma \frac{I_D^*}{N} S^* \\ \frac{\partial f_3}{\partial v} = 0 \end{array} \right. \quad (23)$$

$$\left\{ \begin{array}{l} \frac{\partial f_1}{\partial S} = v \\ \frac{\partial f_1}{\partial I_D} = \gamma \\ \frac{\partial f_1}{\partial I_H} = \gamma \\ \frac{\partial f_1}{\partial R} = -\rho \\ \frac{\partial f_1}{\partial U} = 0 \\ \frac{\partial f_1}{\partial v} = S^* \end{array} \right. \quad (24)$$

Por fim através de (20), (21), (22), (23) e (24) obtém-se o sistema linearizado:

$$\left\{ \begin{array}{l} \dot{S}(t) = \left( -(R_0 - U^*)\gamma \frac{I_D^*}{N} - v \right) S(t) - (R_0 - U^*)\gamma \frac{S^*}{N} I_D(t) + \rho R(t) + \gamma \frac{I_D^*}{N} S^* U(t) - S^* V \\ \dot{I}_D(t) = 0.85(R_0 - U^*)\gamma \frac{I_D^*}{N} S(t) + \left( 0.85(R_0 - U^*)\gamma \frac{S^*}{N} - \gamma \right) I_D(t) - 0.85\gamma \frac{I_D^*}{N} S^* U(t) \\ \dot{I}_H(t) = 0.15(R_0 - U^*)\gamma \frac{I_D^*}{N} S(t) + 0.15(R_0 - U^*)\gamma \frac{S^*}{N} I_D(t) - \gamma I_H(t) - 0.15\gamma \frac{I_D^*}{N} S^* U(t) \\ \dot{R}(t) = vS(t) + \gamma I_D(t) + \gamma I_H(t) - \rho R(t) + S^* V \end{array} \right. \quad (25)$$

#### D. Representação em espaço de estados.

Pretende-se representar (25) na forma:

$$\dot{x} = Ax + Bu$$

$$y = Cx + Du$$

Com:

Estado:

$$x = (S, I_D, I_H, R)$$

Saída:

$$y = (I_H, I_D + I_H)$$

Controlo:

$$u = (U, V)$$

Posto isto obtém-se:

$$\dot{x} = \begin{bmatrix} -(R_0 - U)\gamma \frac{I_D^*}{N} - v & -(R_0 - U)\gamma \frac{S^*}{N} & 0 & \rho \\ 0.85(R_0 - U)\gamma \frac{I_D^*}{N} & 0.85(R_0 - U)\gamma \frac{S^*}{N} - \gamma & 0 & 0 \\ 0.15(R_0 - U)\gamma \frac{I_D^*}{N} & 0.15(R_0 - U)\gamma \frac{S^*}{N} & -\gamma & 0 \\ v & \gamma & \gamma & -\rho \end{bmatrix} \begin{bmatrix} S \\ I_D \\ I_H \\ R \end{bmatrix} + \begin{bmatrix} \gamma \frac{I_D^*}{N} S^* & -S^* \\ -0.85\gamma \frac{I_D^*}{N} S^* & 0 \\ -0.15\gamma \frac{I_D^*}{N} S^* & 0 \\ 0 & S^* \end{bmatrix} \begin{bmatrix} U \\ V \end{bmatrix} \quad (26)$$

$$y = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} S \\ I_D \\ I_H \\ R \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U \\ V \end{bmatrix} \quad (27)$$

**E. Verificação se o sistema é estável e estabilizável. Verificação se o sistema é controlável para  $u = U$  e para  $u = (U, V)$ . Verificação se o sistema é observável para  $y = I_H$  e para  $y = (I_H, I_D + I_H)$ .**

Para responder às questões que se seguem são utilizados os valores definidos em **B.)** para as matrizes apresentadas.

### → Controlável

Sendo a matriz de controlabilidade dada por:  $M = [B \ AB \ A^2B \ \dots \ A^{n-1}B]$

▪ Entrada ( $u = U$ )

$$A = \begin{bmatrix} -(R_0 - U)\gamma \frac{I_D^*}{N} - v & -(R_0 - U)\gamma \frac{S^*}{N} & 0 & \rho \\ 0.85(R_0 - U)\gamma \frac{I_D^*}{N} & 0.85(R_0 - U)\gamma \frac{S^*}{N} - \gamma & 0 & 0 \\ 0.15(R_0 - U)\gamma \frac{I_D^*}{N} & 0.15(R_0 - U)\gamma \frac{S^*}{N} & -\gamma & 0 \\ v & \gamma & \gamma & -\rho \end{bmatrix}; B = \begin{bmatrix} \gamma \frac{I_D^*}{N} S^* \\ -0.85\gamma \frac{I_D^*}{N} S^* \\ -0.15\gamma \frac{I_D^*}{N} S^* \\ 0 \end{bmatrix}$$

$$M = [B \ AB \ A^2B \ A^3B]$$

M =			
22100	32300/9	-65374/707	-3053/3487
-18785	1445/18	7131/547	-439/1307
-3315	85/6	1615/702	-798/13463
0	-11050/3	4165/54	2103/1655

**Figura 1** – Matriz de controlabilidade M para uma entrada  $u = U$

Verificando se é controlável tem-se:

$$\det(M) = 0$$

$$\text{rank}(M) = 2 \neq n = 4$$

Através dos resultados obtidos, verifica-se que a característica da matriz M (Figura 1) é igual a 2, isto é, não tem característica total ( $\text{rank}(M) = n$ ), e o seu determinante é igual zero. Assim sendo, conclui-se que o sistema não é controlável.

▪ Entrada ( $u = (U, V)$ )

$$A = \begin{bmatrix} -(R_0 - U)\gamma \frac{I_D^*}{N} - v & -(R_0 - U)\gamma \frac{S^*}{N} & 0 & \rho \\ 0.85(R_0 - U)\gamma \frac{I_D^*}{N} & 0.85(R_0 - U)\gamma \frac{S^*}{N} - \gamma & 0 & 0 \\ 0.15(R_0 - U)\gamma \frac{I_D^*}{N} & 0.15(R_0 - U)\gamma \frac{S^*}{N} & -\gamma & 0 \\ v & \gamma & \gamma & -\rho \end{bmatrix}; B = \begin{bmatrix} \gamma \frac{I_D^*}{N} S^* & -S^* \\ -0.85\gamma \frac{I_D^*}{N} S^* & 0 \\ -0.15\gamma \frac{I_D^*}{N} S^* & 0 \\ 0 & S^* \end{bmatrix}$$

$$M = [B \ AB \ A^2B \ A^3B]$$

M =							
22100	*	32300/9	490000/3	-65374/707	45745/17	-3053/3487	-16957/92
-18785	0	1445/18	-85000/3	7131/547	13646/23	-439/1307	1691/173
-3315	0	85/6	-5000	1615/702	12250/117	-798/13463	1737/1007
0	*	-11050/3	-130000	4165/54	-30500/9	2103/1655	6567/38

**Figura 2** - Matriz de controlabilidade M para uma entrada  $u = (U, V)$

**Nota:** \* significa que é um número muito próximo de 0.

Verificando se é controlável tem-se:

Não é possível determinar o determinante da matriz pois não é quadrada.

$$\text{rank}(M) = 2$$

Através dos resultados obtidos, observa-se que a matriz M (Figura 2) não tem característica total e assim, conclui-se que o sistema não é controlável.

➔ **Observável**

Sendo a matriz de observabilidade dada por:  $N = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}$

▪  $y = I_H$

$$A = \begin{bmatrix} -(R_0 - U)\gamma \frac{I_D^*}{N} - v & -(R_0 - U)\gamma \frac{S^*}{N} & 0 & \rho \\ 0.85(R_0 - U)\gamma \frac{I_D^*}{N} & 0.85(R_0 - U)\gamma \frac{S^*}{N} - \gamma & 0 & 0 \\ 0.15(R_0 - U)\gamma \frac{I_D^*}{N} & 0.15(R_0 - U)\gamma \frac{S^*}{N} & -\gamma & 0 \\ v & \gamma & \gamma & -\rho \end{bmatrix}; C = [0 \ 0 \ 1 \ 0]$$

$$N = \begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{bmatrix}$$

N =			
0	0	1	0
1/1560	1/34	-1/6	0
-1/365040	-10/1989	1/36	1/93600
*	13/15867	-373/80599	*

**Figura 3** - Matriz de observabilidade N para  $y = I_H$

**Nota:** \* significa que é um número muito próximo de 0.

Verificando se é observável tem-se:

$$\det(N) = -5.0479 * 10^{-12} \neq 0$$

$$\text{rank}(N) = 4 = n$$

Através dos resultados obtidos, verifica-se que a característica da matriz N (Figura 3) é igual a 4, isto é, tem característica total ( $\text{rank}(N) = n$ ), e o seu determinante é diferente de zero. Assim sendo, conclui-se que o sistema é observável.



$$y = (I_H, I_D + I_H)$$

$$A = \begin{bmatrix} -(R_0 - U)\gamma \frac{I_D^*}{N} - v & -(R_0 - U)\gamma \frac{S^*}{N} & 0 & \rho \\ 0.85(R_0 - U)\gamma \frac{I_D^*}{N} & 0.85(R_0 - U)\gamma \frac{S^*}{N} - \gamma & 0 & 0 \\ 0.15(R_0 - U)\gamma \frac{I_D^*}{N} & 0.15(R_0 - U)\gamma \frac{S^*}{N} & -\gamma & 0 \\ v & \gamma & \gamma & -\rho \end{bmatrix}; C = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

$$N = \begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{bmatrix}$$

N =

0	0	1	0
0	1	1	0
1/1560	1/34	-1/6	0
1/234	1/34	-1/6	0
-1/365040	-10/1989	1/36	1/93600
-1/54756	-137/23868	1/36	1/14040
*	13/15867	-373/80599	*
-1/337182	25/30032	-389/84240	-1/670482

**Figura 4** - Matriz de observabilidade N para uma entrada  $y = (I_H, I_D + I_H)$

**Nota:** \* significa que é um número muito próximo de 0.

Verificando se é observável tem-se:

Não é possível determinar o determinante da matriz pois não é quadrada.

$$\text{rank}(N) = 4$$

Através dos resultados obtidos, observa-se que a matriz N (Figura 4) tem característica total e assim, conclui-se que o sistema é observável.

➔ **Estável**

De modo a verificar a estabilidade do sistema foram calculados os polos da função do sistema, a partir da matriz A já apresentada anteriormente.

```
>> eig(A)

ans =

    -49/4680 + 44/16951i
    -49/4680 - 44/16951i
    -1/248775553394772320 + 0i
    -1/6 + 0i
```

Figura 5 - Eigenvalues da matriz A

Como é possível observar na Figura 5, todos os polos calculados encontram-se no semiplano esquerdo, pelo que podemos concluir que o sistema é estável.

### ➔ Estabilizável

Sendo o sistema não controlável, pode-se assumir que é não possível anular a existência de polos no semiplano direito, apesar de neste caso isso não se verificar. Logo, conclui-se que não é estabilizável.

### F. Representações nas formas canónica controlável, observável e diagonal para $u = U$ e $y = I_H$ .

Para a representação nas formas canónicas temos como partida o sistema dinâmico de controlo linear e invariante no tempo:

$$y^{(n)} + a_1 y^{(n-1)} + \dots + a_{n-1} y' + a_n y = b_0 u^{(n)} + b_1 u^{(n-1)} + \dots + b_{n-1} u' + b_n u \quad (28)$$

### ➔ Controlável

Uma vez que o sistema não é controlável, não é possível escrever a representação canónica controlável do mesmo.

### ➔ Observável

De acordo com (28) temos para a forma canónica observável que:

$$A = \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} b_n - a_n b_0 \\ b_{n-1} - a_{n-1} b_0 \\ \vdots \\ b_2 - a_2 b_0 \\ b_1 - a_1 b_0 \end{bmatrix}, \quad C^T = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}, \quad D = [b_0]$$

O procedimento para se obter esta forma consiste em realizar uma mudança de base  $x = U\tilde{x}$ , onde  $U = (WN)^{-1}$ , sendo  $N$  a matriz de observabilidade (já calculada na Figura 3).

A matriz  $W$  neste caso é dada por:

$$W = \begin{bmatrix} a_3 & a_2 & a_1 & 1 \\ a_2 & a_1 & 1 & 0 \\ a_1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Sendo a função de transferência do sistema com  $u = U$  e  $y = I_H$  a da Figura 6:

```
tf =

      -3315 s^3 - 607.7 s^2 - 9.208 s - 1.131e-17
-----
      s^4 + 0.1876 s^3 + 0.004274 s^2 + 0.0001306 s + 5.249e-22

Continuous-time transfer function.
```

**Figura 6** - Função de transferência do sistema para  $u = U$  e  $y = I_H$

Conclui-se então que:

$$\begin{cases} a_4 = 5.249 \cdot 10^{-22} \\ a_3 = 0.0001306 = \frac{11}{84240} \\ a_2 = 0.004274 = \frac{1}{234} \\ a_1 = 0.1876 = \frac{439}{2340} \\ a_0 = 1 \end{cases}$$

E que:

$$W = \begin{bmatrix} 0.0001306 & 0.004274 & 0.1876 & 1 \\ 0.004274 & 0.1876 & 1 & 0 \\ 0.1876 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Assim tem-se que a matriz  $U$  e a sua inversa são dadas pelas Figuras 7 e 8 respetivamente:

```
U =

    -248997/4      82999/8     -8119/48     -828/185
      9497/7     -9497/42      9497/252    -1909/3107
           0           0           0           1
      622493    -40595/4      4864/37      7309/1787
```

**Figura 7** - Matriz  $U$  para forma canónica observável com  $u = U$  e  $y = I_H$

```
invU =
```

1/561600	1/561600	1/561600	1/561600
11/93600	1/2040	11/14040	1/93600
1/1560	1/34	49/2340	*
0	0	1	0

**Figura 8** - Matriz U inversa para forma canónica observável com  $u = U$  e  $y = I_H$

Posto isto a nova representação é:  $\tilde{x}_{k+1} = U^{-1}AU\tilde{x}_k + U^{-1}Bu_k$   
 $y_k = CU\tilde{x}_k + Du_k$

```
A =
```

*	*	*	*
1	*	*	-11/84240
0	1	*	-1/234
*	*	1	-439/2340

**Figura 9** - Matriz A para representação na forma canónica observável

```
B =
```

-1/917931134241120
-221/24
-2431/4
-3315

**Figura 10** - Matriz B para representação na forma canónica observável

$$\tilde{x}_{k+1} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -\frac{11}{84240} \\ 0 & 1 & 0 & -\frac{1}{234} \\ 0 & 0 & 1 & -\frac{439}{2340} \end{bmatrix} [\tilde{x}_k] + \begin{bmatrix} -1.1 \cdot 10^{-15} \\ \frac{221}{24} \\ \frac{2431}{4} \\ -3315 \end{bmatrix} [u_k]$$

```
C =
```

0	0	0	1
---	---	---	---

**Figura 11** - Matriz C para representação na forma canónica observável

```
D =
```

0
---

**Figura 12** - Matriz D para representação na forma canónica observável

$$y_k = [0 \ 0 \ 0 \ 1][\tilde{x}_k] + [0][u_k]$$

## ➔ Diagonal

De acordo com (28) temos para a forma canónica diagonal que:

$$A = \begin{bmatrix} -p_1 & 0 & \dots & 0 & 0 \\ 0 & -p_2 & \dots & 0 & 0 \\ \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & \dots & -p_{n-1} & 0 \\ 0 & 0 & \dots & 0 & -p_n \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \\ 1 \end{bmatrix}, \quad C^T = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_{n-1} \\ c_n \end{bmatrix}, \quad D = [b_0]$$

A forma canónica diagonal corresponde a uma representação numa base de vetores próprios da matriz  $A$  associados respetivamente aos seus valores próprios. Considerando a mudança de base  $x = Vz$  onde  $V = [v1 \ v2 \ v3 \ v4]$ , estes vetores próprios foram calculados através da *Jordan Canonical Form* do MATLAB em que é devolvida a matriz de mudança de base,  $V$  (Figura 13), e uma matriz diagonal,  $E$  (Figura 14), com os valores próprios da matriz  $A$ .

```
[V,E]=eig(A);
```

```
V =
-665/923 + 0i -665/923 + 0i * + 0i -283/3822 + 0i
346/9893 + 355/4094i 346/9893 - 355/4094i -355/4192 + 0i 29/17970 + 0i
83/13448 + 197/12874i 83/13448 - 197/12874i -117/7829 + 0i -1153/1726 + 0i
447/658 - 552/5411i 447/658 + 552/5411i -269/270 + 0i 659/890 + 0i
```

**Figura 13** - Matriz mudança de base  $V$  para a forma canónica diagonal com  $u = U e y = I_H$

```
E =
-49/4680 + 44/1695i 0 + 0i 0 + 0i 0 + 0i
0 + 0i -49/4680 - 44/1695i 0 + 0i 0 + 0i
0 + 0i 0 + 0i * + 0i 0 + 0i
0 + 0i 0 + 0i 0 + 0i -1/6 + 0i
```

**Figura 14** - Matriz diagonal  $E$  com valores próprios da matriz  $A$

```
invV =
-796/1147 + 247/1491i -53/3906 - 2747/516i 361/4695 + 985/2129i * + 176/395i
-796/1147 - 247/1491i -53/3906 + 2747/516i 361/4695 - 985/2129i * - 176/395i
-1178/1291 - * -1178/1291 + * -1178/1291 + * -1178/1291 - *
* - * 587/2223 - * -1833/1225 - * * - *
```

**Figura 15** - Matriz mudança de base  $V$  inversa

A nova representação de estado é:  $z_{k+1} = V^{-1}AVz_k + V^{-1}Bu_k$   
 $y_k = CV + Du_k$

```
A =
-49/4680 + 44/1695i * - * * + * * - *
* - * -49/4680 - 44/1695i * - * * + *
* - * * + * * - * * + *
* - * * + * * - * -1/6 - *
```

**Figura 16** - Matriz  $A$  para representação na forma canónica diagonal

```
B =
-291404/19 +102132i
-291404/19 -102132i
-1/23902426691 - 1/201312184098i
1/1099511627776 - 1/21167619609667i
```

**Figura 17** - Matriz  $B$  para representação na forma canónica diagonal

$$z_{k+1} = \begin{bmatrix} -\frac{49}{4680} + \frac{44}{1695}i & 0 & 0 & 0 \\ 0 & -\frac{49}{4680} - \frac{44}{1695}i & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{1}{6} \end{bmatrix} [z_k] + \begin{bmatrix} -\frac{291404}{19} + 102132i \\ -\frac{291404}{19} - 102132i \\ 0 \\ 0 \end{bmatrix} [u_k]$$

$$C = \begin{bmatrix} 83/13448 & + & 197/12874i & & 83/13448 & - & 197/12874i & & -117/7829 & + & 0i & & -1153/1726 & + & 0i \end{bmatrix}$$

**Figura 18** - Matriz C para representação na forma canónica diagonal

$$D = \begin{bmatrix} 0 \end{bmatrix}$$

**Figura 19** - Matriz D para representação na forma canónica diagonal

$$y_k = \left[ \frac{83}{13448} + \frac{197}{12874}i, \quad \frac{83}{13448} - \frac{197}{12874}i, \quad -\frac{117}{7829}, \quad -\frac{1153}{1726} \right] [z_k] + [0][u_k]$$

### G. Função de transferência a partir do espaço de estados com $u = U$ e $y = I_H$ .

Tendo a representação do sistema em espaço de estados a sua função de transferência é dada por:

$$G(s) = \frac{Y(s)}{U(s)} = C(sI - A)^{-1}B + D \quad (29)$$

Para este caso temos:

$$A = \begin{bmatrix} -(R_0 - U)\gamma \frac{I_D^*}{N} - v & -(R_0 - U)\gamma \frac{S^*}{N} & 0 & \rho \\ 0.85(R_0 - U)\gamma \frac{I_D^*}{N} & 0.85(R_0 - U)\gamma \frac{S^*}{N} - \gamma & 0 & 0 \\ 0.15(R_0 - U)\gamma \frac{I_D^*}{N} & 0.15(R_0 - U)\gamma \frac{S^*}{N} & -\gamma & 0 \\ v & \gamma & \gamma & -\rho \end{bmatrix}, \quad B = \begin{bmatrix} \gamma \frac{I_D^*}{N} S^* \\ -0.85\gamma \frac{I_D^*}{N} S^* \\ -0.15\gamma \frac{I_D^*}{N} S^* \\ 0 \end{bmatrix}$$

$$C = [0 \ 0 \ 1 \ 0], \quad D = [0]$$

A função de transferência a partir da representação no espaço dos estados para o caso de uma entrada e uma saída ( $u = U, y = I_H$ ), foi encontrada tanto pela expressão (29) como pela função do MATLAB que devolve uma função de transferência de uma representação no espaço de estados.

➔ **Pela expressão (29)**

$$TF = \frac{-31028400 * s * (6485183463413514240 * s^2 + 1188950301625811124 * s + 18014398509481987)}{60701317217550493286400 * s^4 + 11387982161754132690240 * s^3 + 259407338536540885680 * s^2 + 7926335344172078960 * s + 17}$$

O que simplificando fica:

$$TF = \frac{-3315s^3 - 607.75s^2 - 9.208s}{s^4 + 0.1876s^3 + 0.0042735s^2 + 1.306 * 10^{-4} * s + 2.8 * 10^{-22}}$$

➔ **MATLAB**

```
SS =

A =

      S      Id      Ih      R
S   -0.004274   -0.1961      0   0.01667
Id    0.003632  -2.776e-17      0      0
Ih    0.000641   0.02941  -0.1667      0
R           0    0.1667   0.1667  -0.01667

B =

      u
S   2.21e+04
Id  -1.879e+04
Ih   -3315
R      0

C =

      S      Id      Ih      R
y    0      0      1      0

D =

      u
y    0

Continuous-time state-space model.
```

**Figura 20** - Espaço de estados do sistema

```
TF =

From input "u" to output "y":
      -3315 s^3 - 607.7 s^2 - 9.208 s - 1.131e-17
-----
      s^4 + 0.1876 s^3 + 0.004274 s^2 + 0.0001306 s + 5.249e-22

Continuous-time transfer function.
```

**Figura 21** - Função de transferência do sistema

Posto isto pode-se verificar que as funções de transferência são praticamente iguais, apenas diferem em valores de reduzida escala comparando com os outros.

H. Gráfico para resposta temporal transitória: (i) condições iniciais nulas e controlo em degrau de  $U^*$  para  $0.9U^*$ ; (ii) controlo nulo e condições iniciais não nulas com  $I_D$  igual a  $1.2I_D^*$ .

→ Condições iniciais nulas e controlo em degrau

U passa de  $U^*$  para  $0.9U^*$

```
SS =

A =
      S      Id      Ih      R
S  -0.004526  -0.2077      0  0.01667
Id  0.003847  0.009853      0      0
Ih  0.0006789  0.03115 -0.1667      0
R      0      0.1667  0.1667 -0.01667

B =
      u
S  2.21e+04
Id -1.879e+04
Ih   -3315
R      0

C =
      S  Id  Ih  R
y  0    0   1   0

D =
      u
y  0
```

Continuous-time state-space model.

Figura 22 - Espaço de estados para os critérios definidos

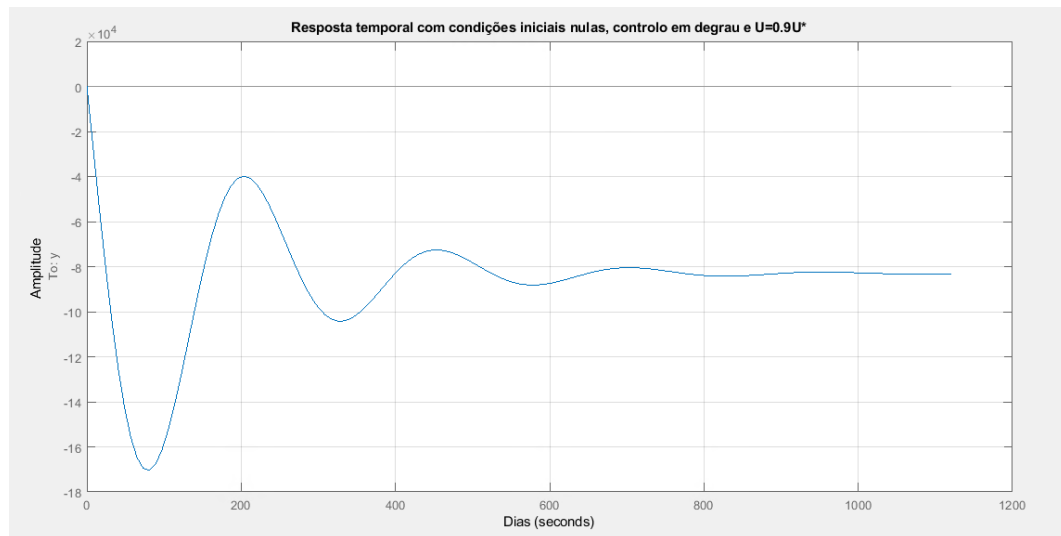
```
TFI =

From input "u" to output "y":
      -3315 s^3 - 607.7 s^2 - 9.208 s + 5.925e-17
-----
s^4 + 0.178 s^3 + 0.002555 s^2 + 0.0001109 s - 9.008e-22

Continuous-time transfer function.
```

Figura 23 - Função de transferência para os critérios definidos





**Figura 24** - Resposta temporal transitória para os critérios definidos

➔ **Condições iniciais não nulas e controlo nulo**

$I_D$  é  $1.2 I_D^*$

$U$  é 0

SS =

A =

	S	Id	Ih	R
S	-0.00816	-0.312	0	0.01667
Id	0.006936	0.09853	0	0
Ih	0.001224	0.0468	-0.1667	0
R	0	0.1667	0.1667	-0.01667

B =

	u
S	2.652e+04
Id	-2.254e+04
Ih	-3978
R	0

C =

	S	Id	Ih	R
y	0	0	1	0

D =

	u
y	0

Continuous-time state-space model.

**Figura 25** - Espaço de estados para os critérios definidos

TFII =

From input "u" to output "y":

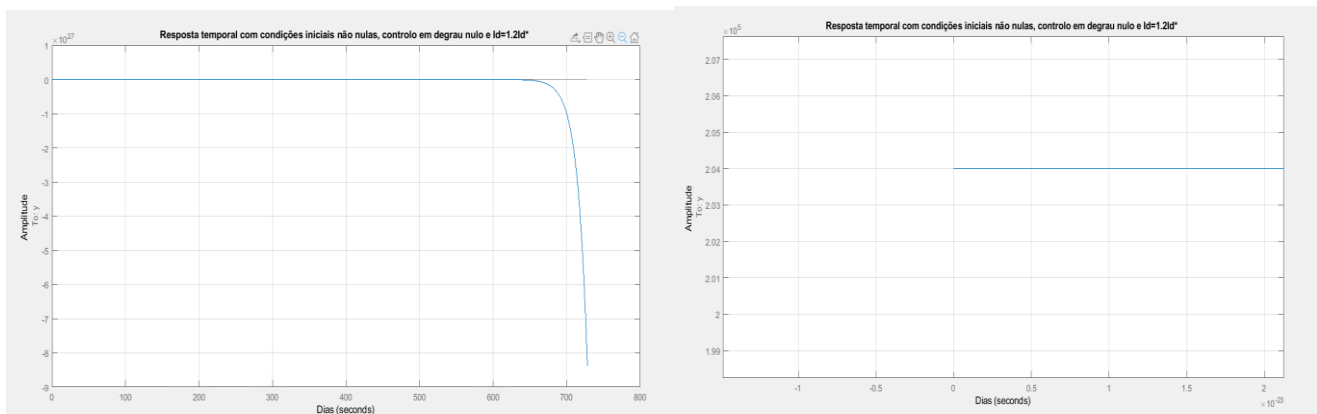
$$-3978 s^3 - 729.3 s^2 - 11.05 s + 3.696e-18$$

---


$$s^4 + 0.09296 s^3 - 0.01243 s^2 - 2.437e-05 s + 2.056e-21$$

Continuous-time transfer function.

**Figura 26** – Função de transferência para os critérios definidos



**Figura 27** - Resposta temporal transitória para os critérios definidos

Analisando os gráficos é fácil de perceber as diferenças existentes. Na primeira ocasião em que o controlo existe, visualiza-se que a curva dos afetados acaba por atenuar o que resulta numa melhor contenção do vírus em si. Apesar de ter um grande pico inicial, os picos sucessivos vão ser cada vez mais pequenos até por fim, ficar a rondar o 0 do eixo vertical.

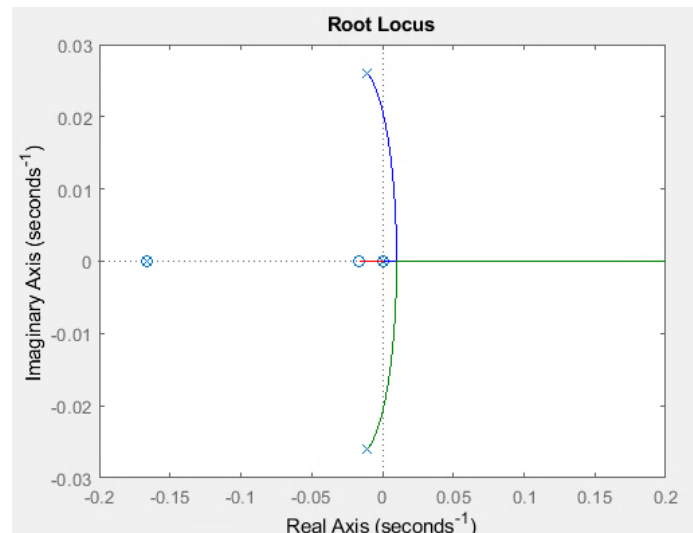
Na segunda ocasião em que não existe controlo e o número de infetados é multiplicado por 1.2, podemos constatar que, tanto a condição inicial de 204000 infetados a recuperar no domicílio é cumprida, assim como este caso é pior em termos de contenção e dizimação do vírus. Apesar de no primeiro caso, demorar bastante tempo até se verificar uma quase anulação do número de infetados, no segundo nunca chegamos a visualizar a atenuação do número de infetados e há um ponto esse número cresce a uma velocidade muito grande até atingir um valor que supera a própria população total N.

## 2. Projeto de controlador em tempo contínuo por espaço de estados.

### A. Projeção de um controlador através da técnica de colocação de polos.

Para esta parte do trabalho começou-se por analisar o sistema em malha aberta vendo a posição dos seus polos no lugar de raízes.

Utilizando a função **ss()** para criou-se um espaço de estados em tempo contínuo. Após isso utilizou-se a função **rlocus()** para se poder ver o lugar de raízes do sistema. Obtendo assim o que está representado na Figura 28.



**Figura 28** - Lugar de raízes do sistema

Pela análise decidiu-se que uma boa abordagem seria colocar os polos no eixo real para menos oscilações e tentar polos mais próximos dos zero para que a estabilização seja mais lenta e tenha por isso menor *overshoot*.

Para fazer isso utilizou-se o método direto usando a fórmula presente no formulário, sendo ela a seguinte:

$$\det([\lambda I - (A - K_e C)])$$

Para colocar os polos em  $[-0.4 \ -0.04 \ 0 \ -1/6]$  (de realçar que se manteve os dois polos que já estavam no eixo real, sendo eles 0 e  $-1/6$ ) resolveu-se o sistema em ordem a  $K_e$  por forma a conseguir calcular a nova matriz A, sendo ela dada pela seguinte formula:

$$A_{cl} = A - BK_e$$

Para resolver isto no MATLAB utilizou-se o código da Figura 29.

```

%% Colocação de polos pelo metodo direto
syms lamb k1 k2 k3 k4;
polos = [-0.4 -0.04 0 -1/6]; %Polos desejados para o nosso sistema, mantivemos 2 polos o -1/6 e o zero,
                                %visto que so conseguimos controlar dois polos
                                %se tentassemos controlar mais polos o sistema
                                %e perder a estabilidade

Kcontrolo = [k1 k2 k3 k4];

metDireto = det(lamb*eye(4)-(A-B*Kcontrolo));% aplicação do metodo segundo o formulario

fator = sym2poly((lamb-polos(1))*(lamb-polos(2))*(lamb-polos(3))*(lamb-polos(4))); % obter o valor que multiplica por Lamb
                                                %isto é exemplo: a1*X + a2*X retorna uma matriz
                                                %com [a1 a2]

coeficientes = coeffs(metDireto,lamb);% retorna o coeficientes do lambda

equacao1 = coeficientes(5) == fator(1);%cria as equações
equacao2 = coeficientes(4) == fator(2);
equacao3 = coeficientes(3) == fator(3);
equacao4 = coeficientes(2) == fator(4);
equacao5 = coeficientes(1) == fator(5);

resultado = solve([equacao1 equacao2 equacao3 equacao4 equacao5], Kcontrolo);% resolve as equações em ordem a K

Kcontrolo=[double(resultado.k1),double(resultado.k2),double(resultado.k3),double(resultado.k4)];
  
```

Figura 29 - Código para exercício 2.A

Assumindo como condições iniciais o vetor  $x_0 = [10e6, 100, 1, 1]$  pode-se tentar agora comparar as respostas em malha aberta e em malha fechada.

Começando por mostrar a evolução em malha aberta (Figura 30):

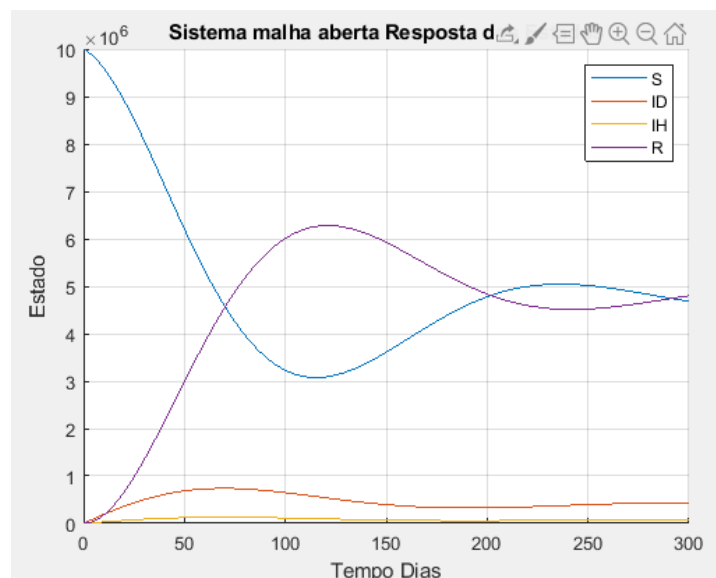
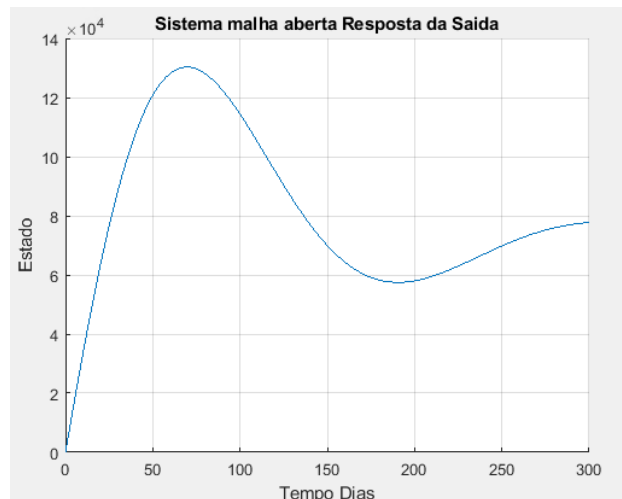


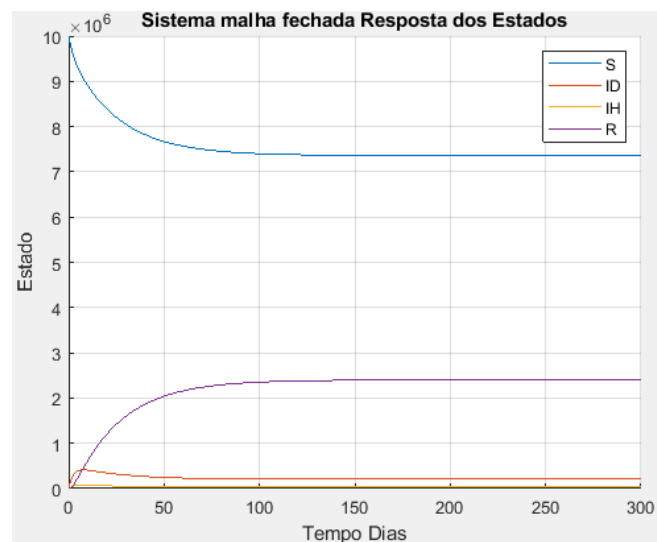
Figura 30 - Resposta em malha aberta dos estados

O comportamento da saída em malha aberta, que se considera como o número de infetados no Hospital (de notar que esta saída já podia ser vista nos estados acima, mas assim torna-se mais fácil de comparar com malha fechada) pode ser visto na Figura 31:



**Figura 31** - Saída do sistema em malha aberta

Em malha fechada obteve-se os resultados das Figura 32 e 33:



**Figura 32** - Resposta em malha fechada dos estados



**Figura 33** - Saída do sistema em malha fechada

Como é visível pela análise das Figuras 30, 31, 32 e 33 o sistema em malha fechada tende a estabilizar não oscilando como em malha aberta. É possível ver também a dinâmica do sistema, por exemplo em malha fechada vemos que o número de infetados no hospital tende a estabilizarem mais ou menos em  $3.7 \times 10^4$  abaixo da capacidade máxima do serviço nacional de saúde estabelecida na parte 1 do trabalho.

O código utilizado para gerar os gráficos em malha aberta e malha fechada são idênticos por isso na Figura 34 a título de exemplo o usado na malha aberta.

```

%% Representacao grafica do sistema
Utf = 0.8917; %
u=[];
for i = 0:0.1:300
    u = [u Utf];
end
T_final = 300;
t=0:0.1:T_final;
x0=[10e6,100,1,1]; %% Valores iniciais, parecidos com os usados no trabalho 1
[y1, t1, x1]=lsim(sys,u,t,x0);
figure(2)
hold on
title('Sistema malha aberta Resposta dos Estados')
xlabel('Tempo Dias')
ylabel('Estado')
plot(t,x1)
legend('S','ID','IH','R');
grid
figure(3)
hold on
title('Sistema malha aberta Resposta da Saida')
xlabel('Tempo Dias')
ylabel('Estado')
plot(t,y1)
grid

```

**Figura 34** - Código usado para gerar a resposta do sistema

**B. Projeção de um estimador de estado para uma norma de erro ao fim de dois dias de um décimo do erro inicial.**

Para este exercício começou-se por atribuir um valor para o erro do estimador de  $[-0.03 \ -0.06 \ 0 \ -0.5]$ , este valor é dez vezes menor que o erro ao fim de 2 dias. Em seguida para calcular os valores de  $K_e$ , tendo em conta os valores de erro de estimação, utilizou-se a fórmula de *Ackermann* sendo esta a seguinte:

$$K_e = p_e(A)N^{-1} \begin{bmatrix} 0 \\ 0 \\ : \\ 1 \end{bmatrix}$$

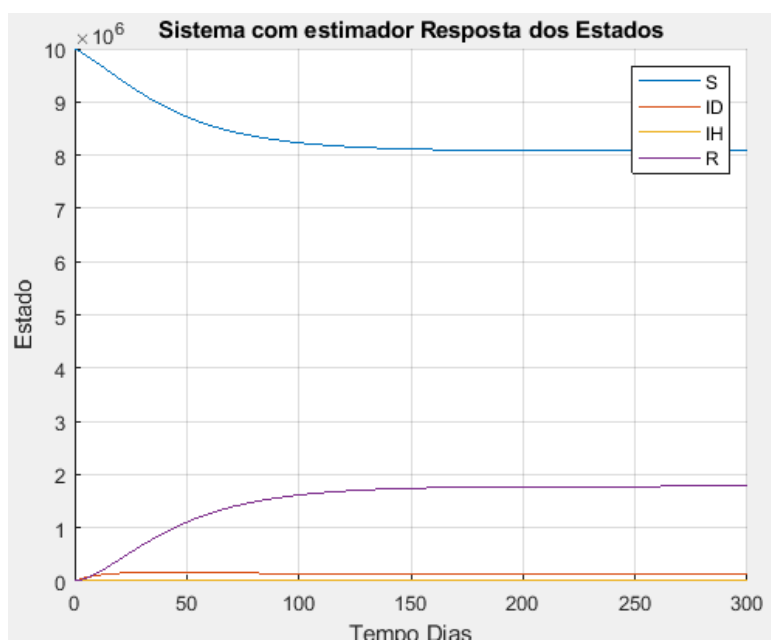
De ressaltar que  $p_e$  é o polinómio característico que neste caso é igual ao erro do estimador.

Para resolver isto no MATLAB utilizou-se a função **acker()** como se pode ver na Figura 35:

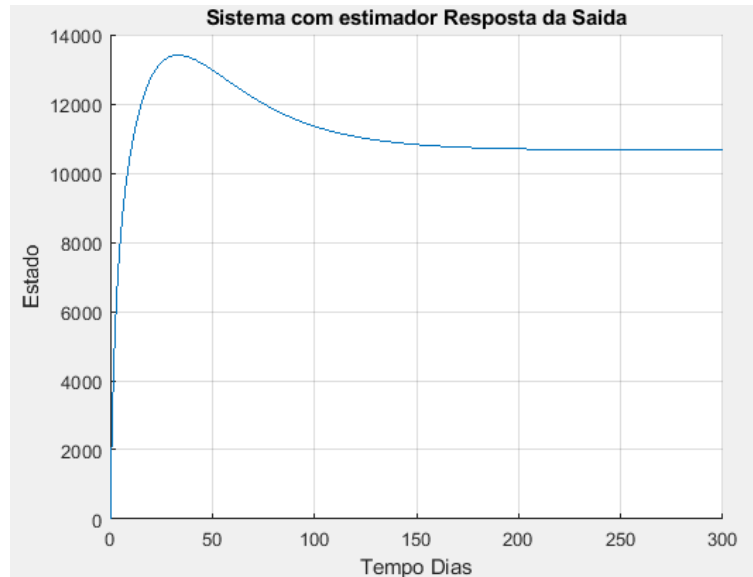
```
%% Estimador
Valor_erro_estimado = [-0.03 -0.06 0 -0.5];
Kestimado=acker(A',C',Valor_erro_estimado)
disp('Novo valor proprio de A');
disp(eig(A-Kestimado*C));
```

**Figura 35** - Cálculo do estimador no MATLAB

A evolução do sistema neste caso foi a apresentada das Figuras 36 e 37:



**Figura 36** - Resposta dos estados em malha fechada com estimador



**Figura 37** - Resposta da saída em malha fechada com estimador

Como é visível na Figura 37 o número de infetados no hospital estabiliza num número abaixo do valor que obtivemos no sistema com controlador. Porém isto apenas está relacionado com a escolha dos polos. É de esperar que neste caso, visto não termos acesso a alguns estados, a resposta do sistema seja pior que no anterior.

### C. Equações que definem o sistema com controlador e estimador.

Pode-se definir o sistema com controlador e estimador da seguinte forma

$$\begin{cases} \dot{\hat{x}} = A\hat{x} + Bu + K_e(y - C\hat{x}) \\ u = -K\hat{x} \end{cases}$$

$$\begin{cases} \dot{\hat{x}} = A\hat{x} - BK\hat{x} + K_e(Cx - C\hat{x}) \\ Bu = -BK\hat{x} \end{cases}$$

$$\begin{cases} \dot{x} - \dot{\tilde{x}} = A(x - \tilde{x}) - BK(x - \tilde{x}) + K_e(Cx - C(x - \tilde{x})) \\ \dot{x} - Ax = -BK(x - \tilde{x}) \end{cases}$$

$$\begin{cases} \dot{x} = \dot{\tilde{x}} + (A - BK)x - (A - BK)\tilde{x} + K_e C \tilde{x} \\ \dot{x} = (A - BK)x + BK\tilde{x} \end{cases}$$

$$\begin{cases} BK\tilde{x} = \dot{\tilde{x}} - A\tilde{x} + BK\tilde{x} + K_e C \tilde{x} \\ \dot{x} = (A - BK)x + BK\tilde{x} \end{cases}$$

$$\begin{cases} \dot{\tilde{x}} = (A - K_e C)\tilde{x} \\ \dot{x} = (A - BK)x + BK\tilde{x} \end{cases}$$



Passando agora para a forma matricial temos o seguinte:

$$\begin{bmatrix} \dot{x} \\ \dot{\tilde{x}} \end{bmatrix} = \begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix} \begin{bmatrix} x \\ \tilde{x} \end{bmatrix}$$

$$y = [C \quad 0] \begin{bmatrix} x \\ \tilde{x} \end{bmatrix} \quad (30)$$

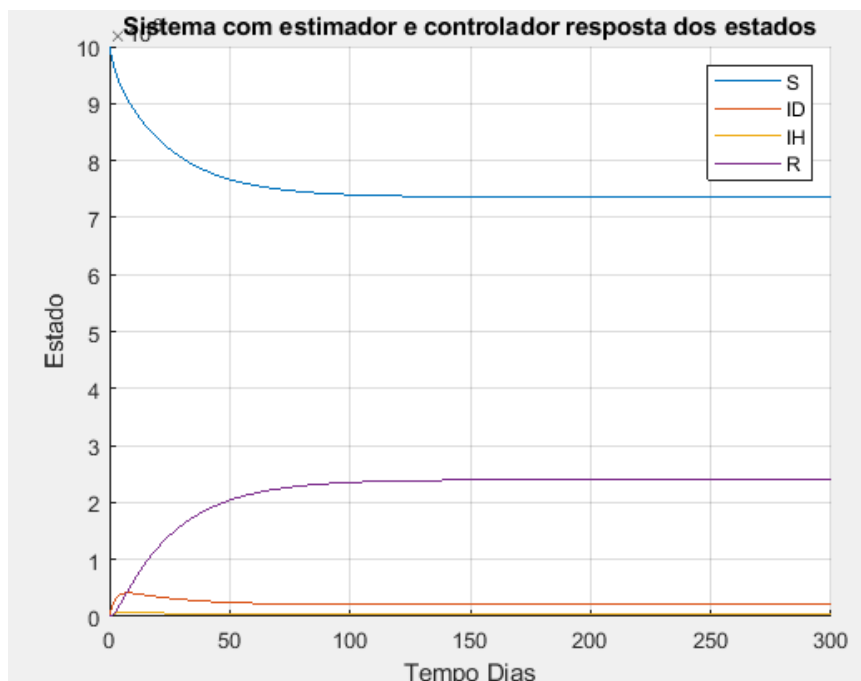
#### D. Análise da resposta do sistema controlado.

Para a última parte implementou-se a fórmula (30) no MATLAB da como se pode ver na Figura 38:

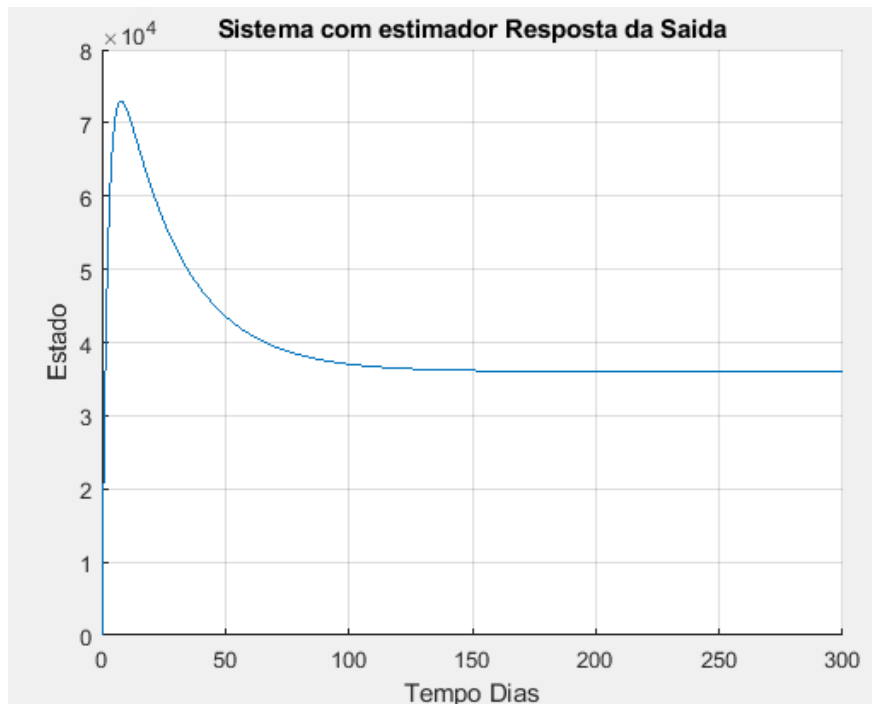
```
% Sistema com controlador e estimador
sys_controlo_estimador= ss([A-B*Kcontrolo,B*Kcontrolo;zeros(4),A-Kestimado*C],[B;zeros(4,1)],[C,zeros(1,4)],D);
```

**Figura 38** - Implementação do estimador e controlador

Obtendo os resultados das Figuras 39 e 40:



**Figura 39** - Resposta dos estados



**Figura 40-** Resposta da saída (infetados no hospital)

Os resultados obtidos são idênticos aos obtidos pelo método de colocação de polos. Parte-se do suposto que isto deve-se ao facto de uma escolha não correta dos valores para o estimador. Deveriam ser vistas algumas diferenças pois neste caso alguns dos estados não são acessíveis.

Tirando este aspeto, os resultados obtidos tal como na alínea **3A** são satisfatórios, e consegue-se ver bem a dinâmica do sistema. Como se pode observar existe uma descida de suscetíveis que passam a infetados e depois a recuperados, estes valores tendem todos a estabilizar mantendo sempre os infetados nos hospitais abaixo da capacidade máxima.

### 3. Espaço de estados em tempo discreto.

#### A. Modelo de espaço de estado discreto com período de amostragem de um dia.

A partir da alínea C obteve-se a o sistema em espaço de estados na D, equação (26). De seguida, retirou-se as matrizes A, B, C e D, onde se substituiu os respetivos valores de **1B**, para obter as matrizes A, B, C e D:

$$A = \begin{bmatrix} -0.0043 & -0.1960 & 0 & 1/60 \\ 0.0036 & 0 & 0 & 0 \\ 0.0006 & 0.0294 & -1/6 & 0 \\ 0 & 1/6 & 1/6 & -1/6 \end{bmatrix} \quad B = \begin{bmatrix} 22100 & -7800000 \\ -18785 & 0 \\ -3315 & 0 \\ 0 & 7800000 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix} \quad D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

De modo a discretizar A e B, iremos utilizar as seguintes expressões:

$$\tilde{A} = e^{At} \quad (31)$$

$$\tilde{B} = \left( \int_0^T e^{At} dt \right) B \quad (32)$$

$$e^{At} = L^{-1}[(sI - A)^{-1}] \quad (33)$$

Começando então pela discretização de A, utilizando as expressões (31) e (33) obtém-se:

$$(sI - A)^{-1} = \begin{bmatrix} s + 0.0043 & 0.1960 & 0 & -\frac{1}{60} \\ -0.0036 & s & 0 & 0 \\ -0.0006 & -0.0294 & s + \frac{1}{6} & 0 \\ 0 & -\frac{1}{6} & -\frac{1}{6} & s + \frac{1}{6} \end{bmatrix}^{-1}$$

De seguida, através da matriz adjunta realizamos a inversa, resulta a seguinte matriz que iremos nomear de matriz J por razões de simplificação:

$$\frac{1}{\det} \begin{bmatrix} s^3 + 0.1833s^2 + 0.0028s & -0.1960s^2 - 0.03312s & 0.00278s & 0.01667s^2 + 0.0028s \\ 0.0036s^2 + 0.0007s & s^3 + 0.1876s^2 + 0.0036s & 0 & 0 \\ 0.0006s^2 + 0.0001s & 0.0294s^2 + 0.005s & s^3 + 0.0209s^2 + 0.0008s & 0 \\ 0.0007s + 0.0001 & 0.1667s^2 + 0.0334s + 1e^{-4} & 0.1667s^2 + 1e^{-4}s - 4s + 1e^{-4} & s^3 + 0.1709s^2 + 0.0014s + 1e^{-4} \end{bmatrix}$$

Onde,

$$\det = s^4 + 0.1876s^3 + 0.0042s^2 + 0.0001s$$

De seguida encontrou-se os zeros do polinómio do determinante de modo a fatorizá-los. Para facilitar a representação mais tarde nomeou-se os polos de p1, p3, p3 e p4.

$$p1 = -0.1667$$

$$p2 = 0$$

$$p3 = -0.0105 - 0.0260i$$

$$p4 = -0.0105 + 0.0260i$$

Através das frações parciais chegou-se às matrizes m1, m2, m3 e m4:

$$\frac{1}{(s-p2)(s-p3)(s-p4)} J|_{s=p1} = \begin{bmatrix} 0 & -0.0196 & 0.1108 & 0 \\ 0 & 4.2614e-04 & -0.0024 & 0 \\ 0 & -0.1764 & 0.9996 & 0 \\ 0 & 0.1955 & -1.1080 & 0 \end{bmatrix}$$

$$\frac{1}{(s-p1)(s-p3)(s-p4)} J|_{s=p2} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0.0773 & 0.0773 & 0.0773 & 0.0773 \\ 0.0136 & 0.0136 & 0.0136 & 0.0136 \\ 0.9091 & 0.9091 & 0.9091 & 0.9091 \end{bmatrix}$$

$$\frac{1}{(s-p1)(s-p2)(s-p4)} J|_{s=p3} = \begin{bmatrix} 0.5000 + 0.1194i & 0.0098 - 3.8356i & -0.0554 + 0.3333i & 0.3210i \\ -0.0386 + 0.0544i & 0.4612 + 0.1874i & -0.0374 - 0.0228i & -0.0386 - 0.0156i \\ -0.0068 + 0.0096i & 0.0814 + 0.0331i & -0.0066 - 0.0040i & -0.0068 - 0.0028i \\ -0.4545 - 0.1833i & 0.5523 + 3.6151i & 0.0994 - 0.3065i & 0.0455 - 0.3027i \end{bmatrix}$$

$$\frac{1}{(s-p1)(s-p2)(s-p3)} J|_{s=p4} = - \begin{bmatrix} 0.5000 + 0.1194i & 0.0098 - 3.8356i & -0.0554 + 0.3333i & 0.3210i \\ -0.0386 + 0.0544i & 0.4612 + 0.1874i & -0.0374 - 0.0228i & -0.0386 - 0.0156i \\ -0.0068 + 0.0096i & 0.0814 + 0.0331i & -0.0066 - 0.0040i & -0.0068 - 0.0028i \\ -0.4545 - 0.1833i & 0.5523 + 3.6151i & 0.0994 - 0.3065i & 0.0455 - 0.3027i \end{bmatrix}$$

Resultando então em:

$$\frac{1}{(s-p1)} m1 + \frac{1}{(s-p2)} m2 + \frac{1}{(s-p3)} m3 + \frac{1}{(s-p4)} m4$$

Aplicando a transformação inversa de Z, resultando em:

$$e^{At} = e^{p1.t} m1 + e^{p2.t} m2 + e^{p3.t} m3 + e^{p4.t} m4$$

Sendo t=1, obtém-se então  $\tilde{A}$ :

$$e^{At} = \tilde{A} = \begin{bmatrix} 0.9954 & -0.1942 & 0.0013 & 0.0165 \\ 0.0036 & 0.9996 & 0 & 0 \\ 0.0006 & 0.0270 & 0.8465 & 0 \\ 0.0004 & 0.1676 & 0.1522 & 0.9835 \end{bmatrix}$$

De seguida de modo a calcular  $\tilde{B}$ , visto que A não é invertível utilizou-se a seguinte expressão da definição (32), resultando em:

$$\tilde{B} = \begin{bmatrix} 2.3879 * 10^4 & -7.7179 * 10^6 \\ -1.8743 * 10^4 & -1.4067 * 10^4 \\ -3.3075 * 10^3 & -2.4825 * 10^3 \\ -1.8288 * 10^3 & 7.7344 * 10^6 \end{bmatrix}$$

Obtendo-se por fim o seguinte espaço de estados discreto:

$$\dot{x} = \begin{bmatrix} 0.9954 & -0.1942 & 0.0013 & 0.0165 \\ 0.0036 & 0.9996 & 0 & 0 \\ 0.0006 & 0.0270 & 0.8465 & 0 \\ 0.0004 & 0.1676 & 0.1522 & 0.9835 \end{bmatrix} \begin{bmatrix} S \\ I_D \\ I_H \\ R \end{bmatrix} + \begin{bmatrix} 2.3879 * 10^4 & -7.7179 * 10^6 \\ -1.8743 * 10^4 & -1.4067 * 10^4 \\ -3.3075 * 10^3 & -2.4825 * 10^3 \\ -1.8288 * 10^3 & 7.7344 * 10^6 \end{bmatrix} \begin{bmatrix} U \\ V \end{bmatrix}$$

$$y = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} S \\ I_D \\ I_H \\ R \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U \\ V \end{bmatrix}$$

Para verificar os valores calculados utilizou-se a função **c2d()** do MATLAB, convertendo o sistema em espaço de estados contínuo obtido em 1D para discreto utilizando a taxa de amostragem igual a 1 e obteve-se:

$$\dot{x} = \begin{bmatrix} 0.9954 & -0.1942 & 0.0013 & 0.0165 \\ 0.0035 & 0.9996 & 0 & 0 \\ 0.0006 & 0.0270 & 0.8465 & 0 \\ 0.0004 & 0.1686 & 0.1522 & 0.9835 \end{bmatrix} \begin{bmatrix} S \\ I_D \\ I_H \\ R \end{bmatrix} + \begin{bmatrix} 2.388 * 10^4 & -7.718 * 10^6 \\ -1.875 * 10^4 & -1.407 * 10^4 \\ -3308 & -2482 \\ -1829 & 7.734 * 10^6 \end{bmatrix} \begin{bmatrix} U \\ V \end{bmatrix}$$

$$y = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} S \\ I_D \\ I_H \\ R \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U \\ V \end{bmatrix}$$

### B. Função de transferência em Z a partir do espaço de estados. Comparação com a discretização de 1G.

De forma a comparar com o com a função de transferência obtida em 1G, utilizou-se de igual forma apenas uma entrada e uma saída ( $u = U$  e  $y = I_H$ ).

Utilizando a seguinte equação:

$$\frac{Y(z)}{U(z)} = \tilde{C}(zI - \tilde{A})^{-1} \tilde{B} + \tilde{D}$$

Obteve-se então a seguinte função de transferência:

$$FT = \frac{-3308z^3 + 9360z^2 - 8806z + 2753}{z^4 - 3.825z^3 - 5.479z^2 - 3.483z + 0.8289}$$

De seguida, obtendo a discretização da expressão em 1G recorrendo **c2d()**, com uma taxa de amostragem igual a 1, obteve-se exatamente a mesma função de transferência que se obteve utilizando o espaço de estados discreto.

**C. Projeção de um controlador, sem saturação da entrada, que leva o sistema ao ponto de equilíbrio em quatro instantes de tempo.**

Por motivos de tempo não foi possível chegar a nenhum resultado apresentável nesta alínea.

**Conclusão**

Dado por terminado o trabalho, de uma forma geral pode-se dizer que apesar de não, se ter implementado o controlador discreto corretamente em MATLAB, seria possível controlar este sistema, por forma a manter o caso de infetados abaixo do limite máximo dos Serviço de Saúde á semelhança do que se observou para tempo contínuo.

Este trabalho permitiu também ficar com uma melhor ideia de como abordar sistemas através da utilização de espaço de estados e das vantagens que este apresenta no que toca à sua versatilidade na análise de sistemas contínuos e discretos.

# 1 Anexos

## 1.a Exercício 1

```
1 #####
2 %                               Codig          #
3 %Desenvolvido por :                #
4 %Diogo Silva up201809213            #
5 %Paulo Silva up201909556            #
6 %Emanuel Ribeiro up201603213        #
7 %Jose Sarmiento up201909931         #
8 #####
9
10 %% Exercicio 1 parte 1
11
12 clear
13 format rat;
14 syms Ro U Id Ih y N v S p s R I;
15
16 %VALORES


---


17
18 N=10*10^6;
19 p=1/60;
20 Ro=2.4;
21 S=(78*10^5);
22 U=Ro-N/(0.85*S);
23 v=0;
24 y=(1/6);
25 Id=((2*10^5)/(1+(0.15/0.85)));
26 Ih=((2*10^5)/(1+(0.85/0.15)));
27 I=Id+Ih;
28 R=10*I;
29
30 %MATRIZ A


---


31
32 A=[(-(Ro-U)*y*(Id/N)-v) (-(Ro-U)*y*(S/N)) 0 p;
33 (0.85*(Ro-U)*y*(Id/N)) 0.85*(Ro-U)*y*(S/N)-y 0 0;
34 (0.15*(Ro-U)*y*(Id/N)) 0.15*(Ro-U)*y*(S/N) -y 0;
35 v y y -p];
36
37 %MATRIZ B1 - 1 INPUT


---


38
39 B1=[y*(Id/N)*S;
```

```

40 ( -0.85*y*(Id/N)*S);
41 ( -0.15*y*(Id/N)*S);
42 0];
43
44 %MATRIZ B2 - 2 INPUT

```

---

```

45
46 B2=[y*(Id/N)*S -S;
47 ( -0.85*y*(Id/N)*S) 0;
48 ( -0.15*y*(Id/N)*S) 0;
49 0 S];
50
51 %MATRIZ C1 - 1 OUTPUT

```

---

```

52
53 C1=[0 0 1 0];
54
55 %MATRIZ C2 - 2 OUTPUT

```

---

```

56
57 C2=[0 0 1 0;
58      0 1 1 0];
59
60 %MATRIZ D1 - 1 INPUT / 1 OUTPUT

```

---

```

61
62 D11=[0];
63
64 %MATRIZ D1 - 2 INPUT / 1 OUTPUT

```

---

```

65
66 D21=[0 0];
67
68 %MATRIZ D1 - 1 INPUT / 2 OUTPUT

```

---

```

69
70 D12=[0;
71      0];
72
73 %MATRIZ D2 - 2 INPUT / 2 OUTPUT

```

---

```

74
75 D22=[0 0;
76      0 0];
77
78 %MATRIZ IDENTIDADE Ie

```

---

```

79

```



```

80 Ie=eye(4);
81
82 %
      #####

83
84 % EXERCICIO 1.E – CONTROLABILIDADE, OBSERVABILIDADE, ESTAVEL,
      ESTABILIZAVEL
85
86 disp('EXERCICIO 1.E');
87 fprintf('\n');
88
89 %%CONTROLABILIDADE – 1 INPUT
90 disp('Matriz M – 1 input:');
91 M1=ctrb(A,B1)
92 disp('Determinante M1');
93 detM1=det(M1)
94 disp('Caracteristica M1');
95 rankM1=rank(M1)
96
97 %%CONTROLABILIDADE – 2 INPUT
98 disp('Matriz M – 2 input:');
99 M2=ctrb(A,B2)
100 disp('Matriz n o quadrada');
101 fprintf('\n');
102 disp('Caracteristica M2');
103 rankM2=rank(M2)
104
105 %%OBSERVABILIDADE – 1 OUTPUT
106 disp('Matriz N – 1 output:');
107 N1=obsv(A,C1)
108 disp('Determinante N1');
109 detN1=det(N1)
110 disp('Caracteristica N1');
111 rankN1=rank(N1)
112
113 %%OBSERVABILIDADE – 2 OUTPUT
114 disp('Matriz N – 2 output:');
115 N2=obsv(A,C2)
116 disp('Matriz n o quadrada');
117 fprintf('\n');
118 disp('Caracteristica N2');
119 rankN2=rank(N2)
120
121 %%ESTABILIDADE
122
123 disp('Valores pr prios de A');
124 eigen=eig(A)

```

```

125
126 %%ESTABILIZ VEL
127
128 disp('N o      estabiliz vel pois n o      control vel');
129 fprintf('\n');
130
131 %
132 #####
133
134 %
135 #####
136
137
138
139
140
141 %%FORMA CAN NICA CONTROLVEL
142 disp('FORMA CAN NICA CONTROLVEL:')
143 disp('Sistema n o      control vel , n o      poss vel calcular'
144 );
145 fprintf('\n');
146
147 %%FORMA CAN NICA OBSERV VEL
148 disp('FORMA CAN NICA OBSERV VEL:')
149 polos=poly(A);
150 W=[polos(4) polos(3) polos(2) 1;
151     polos(3) polos(2) 1 0;
152     polos(2) 1 0 0;
153     1 0 0 0];
154 Uo=inv(W*N1);
155 invU=inv(Uo);
156 disp('Matriz A observ vel');
157 Ao=invU*A*Uo
158 disp('Matriz B observ vel');
159 Bo=invU*B1
160 disp('Matriz C observ vel');
161 Co=C1*Uo
162 disp('Matriz D observ vel');
163 Do=D11
164
165 %%FORMA CAN NICA DIAGONAL
166 disp('FORMA CAN NICA DIAGONAL:')
167 fprintf('\n');
168 [V,E]=eig(A);

```

```

168 invV=inv(V);
169 disp('Matriz A diagonal');
170 Ad=invV*A*V
171 disp('Matriz B diagonal');
172 Bd=invV*B1
173 disp('Matriz C diagonal');
174 Cd=C1*V
175 disp('Matriz D diagonal');
176 Dd=D11
177
178 %
179 #####
180
181 %
182 #####
183
184
185 states={'S' 'Id' 'Ih' 'R'};
186 inputs={'u'};
187 outputs={'y'};
188 SS=ss(A,B1,C1,D11,'state name',states,'input name',inputs,'
189 output name',outputs)
190 TF=tf(SS)
191
192 TF1=C1*inv(s*Ie-A)*B1+D11;
193 TF1=simplify(TF1)
194 %
195 #####
196
197 %
198 #####
199
200
201 %I
202
203 disp('CONDICÖES INICIAIS NULAS, CONTROLO EM DEGRAU E U=0.9U*')
204 U=U*0.9;
205 disp('Nova matriz A');

```

```

206 A=[(-(Ro-U)*y*(Id/N)-v) (-(Ro-U)*y*(S/N)) 0 p;
207 (0.85*(Ro-U)*y*(Id/N)) 0.85*(Ro-U)*y*(S/N)-y 0 0;
208 (0.15*(Ro-U)*y*(Id/N)) 0.15*(Ro-U)*y*(S/N) -y 0;
209 v y y -p];
210 SS=ss(A,B1,C1,D11,'statename',states,'inputname',inputs,'
    outputname',outputs)
211 TFI=tf(SS)
212 figure(1);clf
213 [x,t]=step(SS);
214 u=ones(length(t),1);
215 lsim(SS,u,t) %condi es iniciais nulas, x0 ignorado
216 title('Resposta temporal com condi es iniciais nulas,
    controlo em degrau e U=0.9U*')
217 xlabel('Dias')
218 ylabel('Amplitude')
219 grid on
220
221
222 %
    #####

1 %#####
2 % Codig #
3 %Desenvolvido por : #
4 %Diogo Silva up201809213 #
5 %Paulo Silva up201909556 #
6 %Emanuel Ribeiro up201603213 #
7 %Jose Sarmento up201909931 #
8 %#####
9
10 %% Exercicio 1 parte 2
11
12 clear
13 format shortEng;
14 syms Ro U Id Ih y N v S p s R I;
15
16 %VALORES

17
18 N=10*10^6;
19 p=1/60;
20 Ro=2.4;
21 S=7800000;
22 U=0; %controlo NULO
23 v=0;
24 y=(1/6);
25 Id=((2*10^5)/(1+(0.15/0.85)))*1.2;

```

```

26 Ih=((2*10^5)/(1+(0.85/0.15)));
27 I=Id+Ih;
28 R=10*I;
29
30 %MATRIZ A


---


31
32 A=[(-(Ro-U)*y*(Id/N)-v) (-(Ro-U)*y*(S/N)) 0 p;
33 (0.85*(Ro-U)*y*(Id/N)) 0.85*(Ro-U)*y*(S/N)-y 0 0;
34 (0.15*(Ro-U)*y*(Id/N)) 0.15*(Ro-U)*y*(S/N) -y 0;
35 v y y -p];
36
37 %MATRIZ B1 - 1 INPUT


---


38
39 B1=[y*(Id/N)*S;
40 (-0.85*y*(Id/N)*S);
41 (-0.15*y*(Id/N)*S);
42 0];
43
44 %MATRIZ B2 - 2 INPUT


---


45
46 B2=[y*(Id/N)*S -S;
47 (-0.85*y*(Id/N)*S) 0;
48 (-0.15*y*(Id/N)*S) 0;
49 0 S];
50
51 %MATRIZ C1 - 1 OUTPUT


---


52
53 C1=[0 0 1 0];
54
55 %MATRIZ C2 - 2 OUTPUT


---


56
57 C2=[0 0 1 0;
58 0 1 1 0];
59
60 %MATRIZ D1 - 1 INPUT / 1 OUTPUT


---


61
62 D11=[0];
63
64 %MATRIZ D1 - 2 INPUT / 1 OUTPUT


---


65

```

```

66 D21=[0 0];
67
68 %MATRIZ D1 – 1 INPUT / 2 OUTPUT


---


69
70 D12=[0;
71      0];
72
73 %MATRIZ D2 – 2 INPUT / 2 OUTPUT


---


74
75 D22=[0 0;
76      0 0];
77
78 disp('CONDIÇÕES INICIAIS NÃO NULAS, CONTROLO EM DEGRAU NULO
79      E Id=1.2Id*');
80 fprintf('\n');
81 states={'S' 'Id' 'Ih' 'R'};
82 inputs={'u'};
83 outputs={'y'};
84 SS=ss(A,B1,C1,D11,'statename',states,'inputname',inputs,'
85      outputname',outputs)
86 TFII=tf(SS)
87 figure(2);clf
88 [x,t]=step(SS);
89 u=ones(length(t),1);
90 x0=[Id Id Id Id];
91 lsim(SS,u,t,x0) %condições iniciais Id=1.2Id
92 title('Resposta temporal com condições iniciais não nulas,
93      controlo em degrau nulo e Id=1.2Id*')
94 xlabel('Dias')
95 ylabel('Amplitude')
96 grid on
97
98 %
99 #####

```

## 1.b Exercício 2

```

1 %#####
2 %                               Codig                               #
3 %Desenvolvido por :                               #
4 %Diogo Silva up201809213                               #
5 %Paulo Silva up201909556                               #
6 %Emanuel Ribeiro up201603213                               #
7 %Jose Sarmiento up201909931                               #
8 %#####
9

```

```

10 %%Exercicio 2
11 %% Limpar consola
12 close all;
13 clear all;
14 clc;
15
16 %% Elementos da matriz A com os valores calculados manualmente
17 a11 = -0.0042735166666667;
18 a12 = -0.196079;
19 a13 = 0;
20 a14 = 1/60;
21
22 a21 = 0.0036324891666667;
23 a22 = 4.833333333333333e-7;
24 a23 = 0;
25 a24 = 0;
26
27 a31 = 0.0006410275;
28 a32 = 0.02941185;
29 a33 = -1/6;
30 a34 = 0;
31
32 a41 = 0;
33 a42 = 1/6;
34 a43 = 1/6;
35 a44 = -1/60;
36
37 %% Matriz A
38 A = [a11 a12 a13 a14;
39      a21 a22 a23 a24;
40      a31 a32 a33 a34;
41      a41 a42 a43 a44];
42 disp('Matriz A');
43 display(A);
44 %% Matriz B
45 p=1/60;
46 R=2.4;
47 S=(78*10^5);
48 U=0.8917;
49 v=0;
50 y=(1/6);
51 N=10*10^6;
52 I=((2*10^5)/(1+(0.15/0.85)));
53 Ie=eye(4);
54
55 B=[y*(I/N)*S;
56    (-0.85*y*(I/N)*S);
57    (-0.15*y*(I/N)*S);

```

```

58 0];
59 disp('Matriz B');
60 display(B);
61 %% Matriz C
62 C=[0 0 1 0];
63 disp('Matriz C');
64 display(C);
65 %% Matriz D
66 D=[0];
67 disp('Matriz D');
68 display(D);
69 %% Verificar valor propio de A
70 valor_propioA=eig(A);
71 disp('Valor Propio de A');
72 display(valor_propioA);
73 %% Sistema em malha aberta
74 sys = ss(A,B,C,D);% Criar sistema em malha aberta
75 figure(1)
76 hold on
77 rlocus(sys);
78 %% Representacao grafica do sistema
79 Utf = 0.8917; %
80 u=[];
81 for i = 0:1:300
82     u = [u Utf];
83 end
84 T_final = 300;
85 t=0:1:T_final;
86 x0=[10e6,100,1,1];%% Valores iniciais , parecidos com os usados
    no trabalho 1
87 [y1, t1, x1]=lsim(sys,u,t,x0);
88 figure(2)
89 hold on
90 title('Sistema malha aberta Resposta dos Estados')
91 xlabel('Tempo Dias')
92 ylabel('Estado')
93 plot(t,x1)
94 legend('S','ID','IH','R');
95 grid
96 figure(3)
97 hold on
98 title('Sistema malha aberta Resposta da Saida')
99 xlabel('Tempo Dias')
100 ylabel('Estado')
101 plot(t,y1)
102 grid
103
104 %% Coloca ao de polos pelo metodo direto

```



```

105 syms lamb k1 k2 k3 k4;
106 polos = [-0.4 -0.04 0 -1/6]; %Polos desejados para o nosso
    sistema, mantivemos 2 polos o -1/6 e o zero,
107                                     %visto que so conseguimos
                                     controlar dois polos
108                                     %se tentassemos controlar mais
                                     polos o sistema
109                                     %e perder a estabilidade
110
111 Kcontrole = [k1 k2 k3 k4];
112
113 metDireto = det(lamb*eye(4)-(A-B*Kcontrole));% aplica ao do
    metodo segundo o formulario
114
115 fator = sym2poly((lamb-polos(1))*(lamb-polos(2))*(lamb-polos(3)
    )*(lamb-polos(4))); % obter o valor que multiplica por Lamb
116

```

117

```

118
119 coeficientes = coeffs(metDireto,lamb);% retorna o coeficientes
    do lambda
120
121 equacao1 = coeficientes(5) == fator(1);%%cria as equa oes
122 equacao2 = coeficientes(4) == fator(2);
123 equacao3 = coeficientes(3) == fator(3);
124 equacao4 = coeficientes(2) == fator(4);
125 equacao5 = coeficientes(1) == fator(5);
126
127 resultado = solve([equacao1 equacao2 equacao3 equacao4 equacao5
    ], Kcontrole);% resolve as equa oes em ordem a K
128
129 Kcontrole=[double(resultado.k1),double(resultado.k2),double(
    resultado.k3),double(resultado.k4)];
130 %% Sistema em malha fechada
131 syscl = ss(A-B*Kcontrole,B,C,D);
132
133 %% resposta do sistema com controlador
134 [y1, t1, x1]=lsim(syscl, u, t, x0);
135 figure(4)
136 hold on
137 title('Sistema malha fechada Resposta dos Estados')
138 xlabel('Tempo Dias')
139 ylabel('Estado')
140 plot(t,x1)
141 legend('S','ID','IH','R');
142 grid
143 figure(5)
144 hold on
145 title('Sistema malha fechada Resposta da Saida')
146 xlabel('Tempo Dias')
147 ylabel('Estado')
148 plot(t,y1)
149 grid
150
151 %% Estimador
152 Valor_erro_estimado = [-0.03 -0.06 0 -0.5];
153 Kestimado=acker(A',C',Valor_erro_estimado)'
154 disp('Novo valor propio de A');
155 disp(eig(A-Kestimado*C));
156
157 %% Sistema com estimador
158
159 sys_estimado = ss(A-Kestimado*C,B,C,D);
160 x0=[10e6,100,1,1];
161 [y4, t4, x4]=lsim(sys_estimado, u, t, x0);

```

```

162 size_x = size(x4);
163 figure(6)
164 hold on
165 title('Sistema com estimador Resposta dos Estados')
166 xlabel('Tempo Dias')
167 ylabel('Estado')
168 plot(t,x4)
169 legend('S','ID','IH','R');
170 grid
171 figure(7)
172 hold on
173 title('Sistema com estimador Resposta da Saida')
174 xlabel('Tempo Dias')
175 ylabel('Estado')
176 plot(t,y4)
177 grid
178
179 %% Sistema com controlador e estimador
180 sys_controlo_estimador= ss([A-B*Kcontrolo ,B*Kcontrolo;zeros(4),
    A-Kestimado*C],[B;zeros(4,1)],[C,zeros(1,4)],D);
181
182 %% Resposta do sistema com com controlador e estimador
183 xt=[-0.3 -0.6 0 -5];
184 [y2, t1, x2]=lsim(sys_controlo_estimador, u, t,[ x0,xt]);
185 size_x = size(x2);
186 figure(8)
187 hold on
188 title('Sistema com estimador e controlador resposta dos estados
    ')
189 xlabel('Tempo Dias')
190 ylabel('Estado')
191 plot(t,x2(:,1:size_x(1,2)/2));
192 legend('S','ID','IH','R');
193 grid
194
195 figure(9)
196 hold on
197 title('Sistema com estimador Resposta da Saida')
198 xlabel('Tempo Dias')
199 ylabel('Estado')
200 plot(t,y2)
201 grid

```

### 1.c Exercício 3

```

1 %#####
2 %                               Codig                               #
3 %Desenvolvido por :                               #
4 %Diogo Silva up201809213                               #

```

```

5 %Paulo Silva up201909556 #
6 %Emanuel Ribeiro up201603213 #
7 %Jose Sarmiento up201909931 #
8 %#####
9
10 %% Exercicio 3 Parte 1
11
12 clear all;
13 clc;
14
15 syms R;
16 syms U I y N v S p s t;
17 format shortEng;
18 sympref('FloatingPointOutput',true)
19 N=10*10^6;
20 p=1/60;
21 R=2.4;
22 S=(78*10^5);
23 U=R-N/(0.85*S);
24 v=0;
25 y=(1/6);
26
27 I=((2*10^5)/(1+(0.15/0.85)));
28 Ie=eye(4);
29
30 %% duas entradas duas saidas
31 A1=[(-(R-U)*y*(I/N)-v) (-(R-U)*y*(S/N)) 0 p;
32 (0.85*(R-U)*y*(I/N)) (0.85*(R-U)*y*(S/N)-y) 0 0;
33 (0.15*(R-U)*y*(I/N)) (0.15*(R-U)*y*(S/N)) -y 0;
34 v y y -p]
35
36 B1=[y*(I/N)*S,-S;
37 (-0.85*y*(I/N)*S),0;
38 (-0.15*y*(I/N)*S),0;
39 0,S]
40
41 C1=[0 0 1 0;
42 0 1 1 0;]
43
44 D1=[0]
45
46 %% uma entradas uma saidas
47 % A1=[(-(R-U)*y*(I/N)-v) (-(R-U)*y*(S/N)) 0 p;
48 % (0.85*(R-U)*y*(I/N)) (0.85*(R-U)*y*(S/N)-y) 0 0;
49 % (0.15*(R-U)*y*(I/N)) (0.15*(R-U)*y*(S/N)) -y 0;
50 % v y y -p]
51 %
52 % B1=[y*(I/N)*S;

```

```

53 % (-0.85*y*(I/N)*S);
54 % (-0.15*y*(I/N)*S);
55 % 0]
56 %
57 % C1=[0 0 1 0]
58
59 % A1=[0 1 0;
60 %      0 0 1;
61 %      -8 -14 -7]
62 % B1=[0;0;1]
63 % C1=[2 1 0]
64
65 %D1=[0]
66
67
68
69 Ad_1=(s*Ie-A1) %e^(At)
70 Ad_det=det(Ad_1); %determinante
71 Ad_ajoint=adjoint(Ad_1) %adjun o
72 T_Ad_ajoint=transpose(Ad_ajoint);
73 zeros = solve(Ad_det, s, 'MaxDegree', 4); %encontra os zeros do
      polinomio do det
74 matriz_d=Ad_ajoint;
75
76
77 matriz1_aux=matriz_d*(1/((s-zeros(2))*(s-zeros(3))*(s-zeros(4))
      )); %fracoes parciais
78 matriz1=subs(matriz1_aux, s, zeros(1));
79 matriz2_aux=matriz_d*(1/((s-zeros(1))*(s-zeros(3))*(s-zeros(4))
      ));
80 matriz2=subs(matriz2_aux, s, zeros(2));
81 matriz3_aux=matriz_d*(1/((s-zeros(1))*(s-zeros(2))*(s-zeros(4))
      ));
82 matriz3=subs(matriz3_aux, s, zeros(3));
83 matriz4_aux=matriz_d*(1/((s-zeros(1))*(s-zeros(2))*(s-zeros(3))
      ));
84 matriz4=subs(matriz4_aux, s, zeros(4));
85
86 %inversa de Z
87 exp1=exp(t*zeros(1));
88 exp2=exp(t*zeros(2));
89 exp3=exp(t*zeros(3));
90 exp4=exp(t*zeros(4));
91 %t=1;
92 Ad_aux_var_1=vpa(exp1*matriz1);
93 Ad_aux_var_2=vpa(exp2*matriz2);
94 Ad_aux_var_3=vpa(exp3*matriz3);
95 Ad_aux_var_4=vpa(exp4*matriz4);

```

```

96 aux_var=Ad_aux_var_1+Ad_aux_var_2+Ad_aux_var_3+Ad_aux_var_4;
97 %substitui T
98 Ad_subs=subs(aux_var,t,1);
99 % A discreta
100 Ad_fin= vpa(round(Ad_subs*10000)/10000)
101
102 %B discreta pela integral(e^(At))*B
103 matrix_Ad=Ad_ajoint*(1/Ad_det);
104 int_exp1= int(Ad_aux_var_1,0,1);
105 int_exp2= int(Ad_aux_var_2,0,1);
106 int_exp3= int(Ad_aux_var_3,0,1);
107 int_exp4= int(Ad_aux_var_4,0,1);
108
109 sum_ints=int_exp1+int_exp2+int_exp3+int_exp4;
110 Bd_fin=real(sum_ints)*B1

1 #####
2 % Codig #
3 %Desenvolvido por : #
4 %Diogo Silva up201809213 #
5 %Paulo Silva up201909556 #
6 %Emanuel Ribeiro up201603213 #
7 %Jose Sarmiento up201909931 #
8 #####
9
10 %% Exercicio 3 Parte 2
11
12 clear all;
13
14 syms R;
15 syms U I y N v S p s z;
16 format shortEng;
17 N=10*10^6;
18 p=1/60;
19 R=2.4;
20 S=(78*10^5);
21 U=R-N/(0.85*S);
22 v=0;
23 y=(1/6);
24
25 I=((2*10^5)/(1+(0.15/0.85)));
26 Ie=eye(4);
27
28
29 % %% duas entradas duas saidas
30 % A=[(-(R-U)*y*(I/N)-v) (-(R-U)*y*(S/N)) 0 p;
31 % (0.85*(R-U)*y*(I/N) (0.85*(R-U)*y*(S/N)-y) 0 0;
32 % (0.15*(R-U)*y*(I/N) (0.15*(R-U)*y*(S/N)) -y 0;
33 % v y y -p]

```

```

34 %
35 % B=[y*(I/N)*S,-S;
36 % (-0.85*y*(I/N)*S),0;
37 % (-0.15*y*(I/N)*S),0;
38 % 0,S]
39 %
40 % C=[0 0 1 0;
41 %      0 1 1 0;]
42 %
43 % D=[0]
44
45 %% uma entrada uma saida codigo copiado do ex1
46 A=[(-(R-U)*y*(I/N)-v) (-(R-U)*y*(S/N)) 0 p;
47 (0.85*(R-U)*y*(I/N)) (0.85*(R-U)*y*(S/N)-y) 0 0;
48 (0.15*(R-U)*y*(I/N)) (0.15*(R-U)*y*(S/N)) -y 0;
49 v y y -p];
50
51 B=[y*(I/N)*S;
52 (-0.85*y*(I/N)*S);
53 (-0.15*y*(I/N)*S);
54 0];
55
56 C=[0 0 1 0];
57
58 D=[0];
59
60
61 states={'S' 'Id' 'Ih' 'R'};
62 inputs={'u'};
63 outputs={'Ih'};
64 SS=ss(A,B,C,D,'statename',states,'inputname',inputs,'outputname',
65      outputs);
66 display("espa o de estados continuo:")
67 G=ss(A,B,C,D)%espa o de estados continuo do ex 1
68 transf=tf(G);
69
70 %%ex 3
71 %a
72 display('espa o de estados discreto:')
73 G_d=c2d(G,1) %% discretiza o do espa o de estados G
74 %b
75 %%calcula da Ft
76 I_A=z*Ie-G_d.A;
77 I_A_adj=adjoint(I_A);
78 I_A_det=det(I_A);
79 Y_U=vpa(C*(I_A_adj)*G_d.B)*(1/vpa(I_A_det));
80

```

```

81 %show
82
83 TFstr = char(Y_U);
84 z = tf('z');
85 disp("funcao de transferencia discreta a partir de ss discreto
      :");
86 eval(['TF = ',TFstr])
87
88 %% discretizacao de tf de G
89 disp("funcao de transferencia discreta a partir de ss discreto
      :");
90 trasf_d=c2d(transf,1)
91
92
93
94 %rlocus(G_d);
95 %p=[1,2,2,3]
96 %K=place(G_d.A,G_d.B,p)

```