

**UNIVERSIDADE DE SÃO PAULO  
ESCOLA DE ENGENHARIA DE SÃO CARLOS  
DEPARTAMENTO DE ENGENHARIA AERONÁUTICA**

**SAA0168 - Sistemas de Controle de Aeronaves I**

## **Projeto Final**

Giulia Auzier Lima Vogel, 13673447

Julia Félix Salles, 12544138

Nathan Boteon Caneias, 11846877

**São Carlos,  
Julho, 2024**

# 1 Introdução

---

O projeto consiste em 2 etapas, nas quais sendo elas o motor DC, sendo uma etapa de identificação e modelagem, com a finalidade de gerar a partir de dados fornecidos, uma planta controlável, e o Drone 2D, um sistema controlador com o objetivo de implementar uma estratégia de controle para sua trajetória.

O motor DC é um circuito simples capaz de transformar energia mecânica em elétrica utilizando o princípio da indução, sendo acionado por uma fonte de alimentação de corrente contínua. O torque do motor é produzido pelo rotor, sendo gerado a partir do campo magnético produzido pela corrente de entrada.

Nesse estudo, foi analisada a dinâmica de um motor DC e seu funcionamento em um modelo real de motor. Para tal, foram desenvolvidas simulações computacionais que descrevem um motor DC, as implementado em um Motor ECX 32 flat UAV.

A implementação de um modelo de drone 2D no Simulink, com o objetivo de desenvolver uma estratégia de controle para sua trajetória. Através do uso de técnicas de controle avançadas, buscamos otimizar a estabilidade e precisão do drone, garantindo uma navegação eficiente em um ambiente simulado. A implementação visa demonstrar a eficácia do controle de trajetória em sistemas dinâmicos e não lineares, servindo como uma base para futuras pesquisas e aplicações.

## 2 Motor DC

### 2.1 Teoria

No desenho esquemático apresentado a seguir, podemos observar o funcionamento básico do motor DC, em que são representados o indutor, a resistência e o rotor.

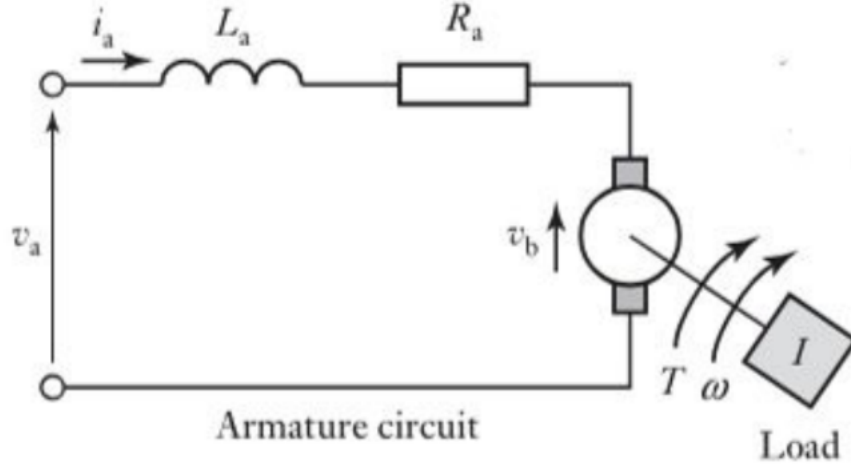


Figura 1: Esquema de circuito do motor DC

Nesse esquema, a parte elétrica é descrita pela tensão na armadura  $v_a$ , pela tensão da força contra eletromotriz  $v_b$ , pela resistência  $R_a$  e pela indutância entre os terminais da armadura do motor  $L_a$ . Todos esses elementos são percorridos pela corrente  $i_a$ . Esses elementos podem ser relacionados pela Lei de Kirchhoff para malhas fechadas:

$$v_a(t) - v_b(t) + L \frac{di(t)}{dt} + Ri(t) = 0 \quad (1)$$

Analisando a parte mecânica do motor, obtemos uma relação de torques no circuito:

$$T_l(t) = J \frac{d\omega(t)}{dt} + B\omega(t) + \tau(t) \quad (2)$$

sendo  $T_l$  o torque elétrico líquido do motor,  $c\omega$  o componente de torque referente ao atrito,  $\tau$  o torque externo aplicado na ponta do eixo,  $J$  a inércia do rotor e  $\omega$  sua velocidade angular.

Supondo uma corrente de campo constante, podemos também supor um fluxo de campo constante, que fornece aproximações próximas aos valores reais. Dessa forma, chegamos em expressões para  $v_b$  e  $T_l$  em função, respectivamente da velocidade angular e corrente do circuito, como representado a seguir:

$$\begin{aligned} v_b(t) &= K_e \cdot \omega(t) \\ T_l(t) &= K_t \cdot i(t) \end{aligned} \quad (3)$$

em que  $K_t$  é a constante de torque do motor e  $K_e = 1/K_v$ , tal que  $K_v$  é a constante de velocidade.

Reorganizando as Equações (1) e (2) e fazendo as substituições necessárias, obtemos:

$$\begin{aligned} L \frac{di(t)}{dt} &= -K_e \cdot \omega(t) - R \cdot i(t) + v_a(t) \\ J \frac{d\omega(t)}{dt} &= -B \cdot \omega(t) + K_t \cdot i(t) - \tau(t) \end{aligned} \quad (4)$$

Reescrevendo as equações na forma de um sistema em espaço de estados, sendo as entradas a voltagem sobre a armadura  $v_a$  e o torque externo  $\tau$ , obtemos:

$$\frac{d}{dt} \begin{bmatrix} i \\ \omega \end{bmatrix} = \begin{pmatrix} \frac{-R}{L} & \frac{-K_e}{J} \\ \frac{K_t}{J} & \frac{-B}{J} \end{pmatrix} \cdot \begin{bmatrix} i \\ \omega \end{bmatrix} + \begin{pmatrix} \frac{1}{L} & 0 \\ 0 & \frac{1}{J} \end{pmatrix} \cdot \begin{bmatrix} v_a \\ \tau \end{bmatrix}$$

Além da forma matricial, é possível visualizar o sistema obtido por meio de um esquema de blocos, apresentado na Figura 2:

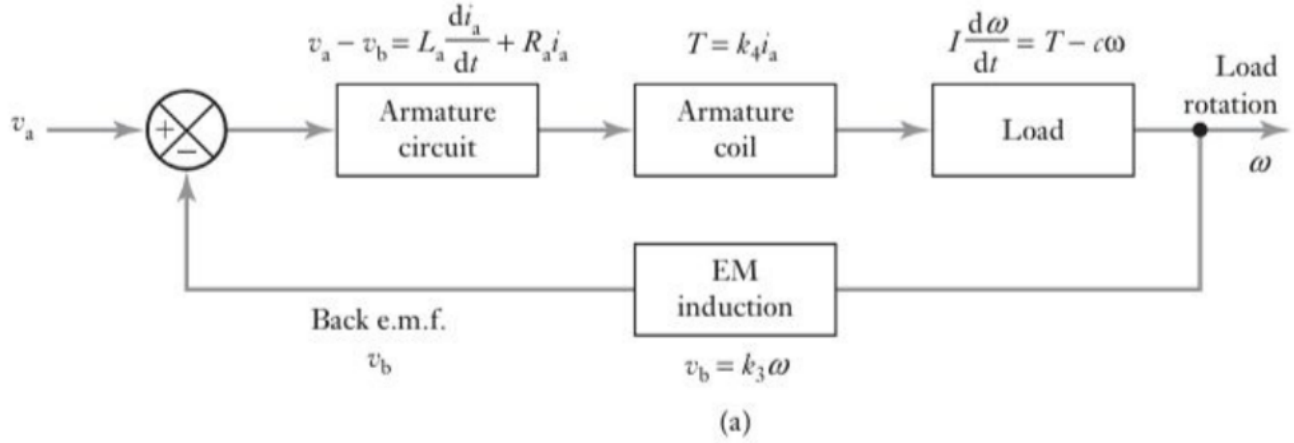


Figura 2: Esquema de blocos do circuito do motor DC

Até o momento, foi apresentado o modelo em função do tempo e aplicando a Transformada de Laplace nas equações apresentadas, podemos obtê-las no domínio da frequência, como abaixo:

$$V_a(s) - K_e \Omega(s) + (sL + R)I(s) = 0 \quad (5)$$

$$K_t I(s) = (sJ + B)\Omega(s) + \tau(s) \quad (6)$$

A partir das equações no domínio da frequência, podemos desenvolver a função transferência tensão-velocidade  $G_1(s)$  para  $\tau(s) = 0$  e a função transferência para o torque da carga  $G_2(s)$  para  $v_a(s) = 0$ :

$$G_1(s) = \frac{\Omega(s)}{V_i(s)} \Big|_{\tau=0} = \frac{K_t}{-(L_s + R)(J_s + B) + K_t \cdot K_e} \quad (7)$$

$$G_2(s) = \frac{\Omega(s)}{\tau(s)} \Big|_{v_a=0} = \frac{-\frac{1}{J_s+B}}{1 + \frac{K_t \cdot K_e}{(L_s+R)(J_s+B)}} \quad (8)$$

## 2.2 Eficiências

A simulação apresentada também permite a avaliação das eficiências do motor, que são calculadas pela razão das potências do circuito.

A eficiência elétrica do motor é dada pela relação entre a potência mecânica de saída e a potência total média de entrada, ou seja,

$$\eta_p = \frac{T_L}{\frac{R_a i_a^2 K_t}{K_e \omega} + B\omega + T_f + T_L} \times 100 \quad (9)$$

Já a eficiência mecânica é a relação entre a potência mecânica de saída e a potência mecânica desenvolvida:

$$\eta_{cm} = \frac{T_L}{B\omega + T_f + T_L} \times 100 \quad (10)$$

As perdas de energia em motores DC desse porte são elevadas, da ordem de 50%. A energia perdida é transformada em calor, podendo causar aquecimento excessivo na armadura.

## 2.3 MATLAB

A partir do modelo experimental apresentado anteriormente, podemos realizar uma simulação do motor DC no MATLAB, já utilizando os dados do Motor ECX 32 flat UAV a ser analisado, que estão descritos abaixo:

$$\begin{aligned} R &= 0.0179[\Omega] \\ L &= 0.0456 [\text{mH}] \\ J &= 35.1 [\text{gcm}^2] \\ B &= 11.9 [\text{rpm/mNm}] \\ K_t &= 12 [\text{mNm/A}] \\ K_v &= 797 [\text{rpm/V}] \\ K_e &= \frac{1}{K_v} \end{aligned}$$

O sistema na forma matricial foi representado como um modelo de espaço de estados como apresentado abaixo:

```
model =

A =
      x1      x2
x1  -3.925  -0.02752
x2   0.3419  -0.339

B =
      u1      u2
x1   21.93      0
x2      0  -0.02849

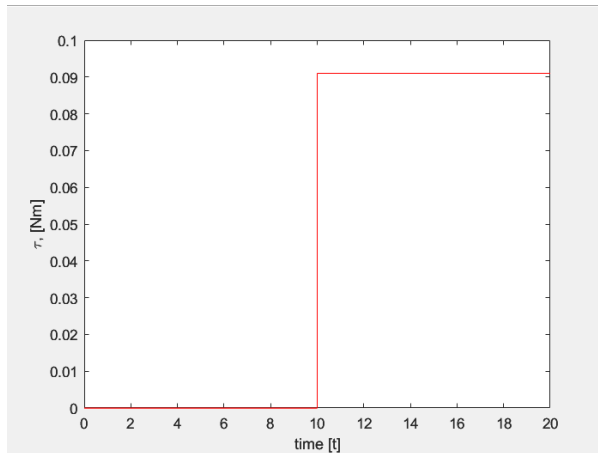
C =
      x1  x2
y1      1   0
y2      0   1

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

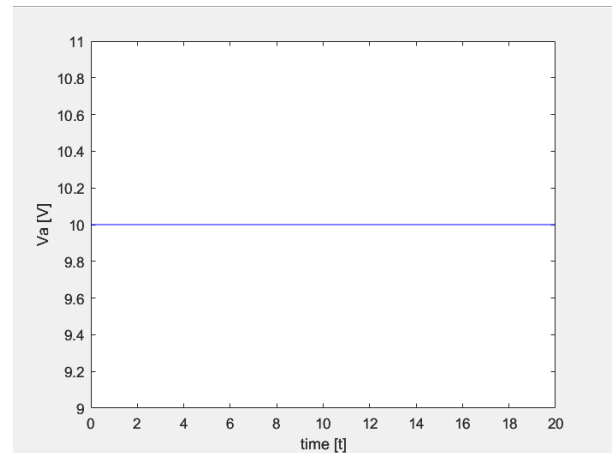
Figura 3: Espaço de estados

A partir dessa representação foram traçadas duas entradas para o sistema. A primeira entrada, "Entrada  $v_a$ ", é a condição "no load", caracterizada pela ausência de um torque externo. É uma entrada

degrau multiplicada por 10(V) no tempo  $t = 0$ . Já a segunda entrada, "Entrada torque", é a condição nominal, determinada pelo fabricante e de maior potência. Esta também é uma entrada degrau, mas multiplicada por 0.091 (Nm) no tempo  $t = 10s$



((a)) Entrada degrau em  $t=10s$



((b)) Entrada degrau em  $t=0$

Figura 4: Entradas do sistema

Após a definição do modelo, podemos, então encontrar a resposta do sistema, representada na imagem abaixo.

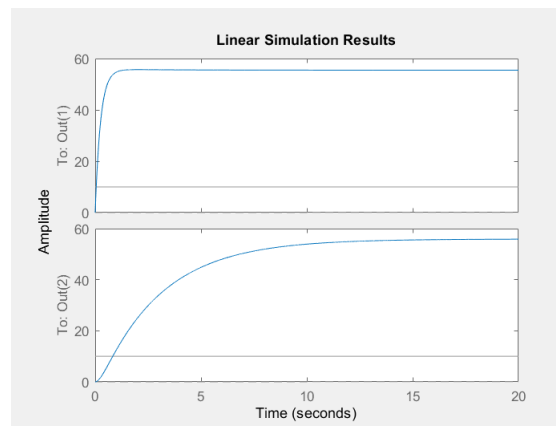


Figura 5: Resposta do sistema

## 2.4 Simulink

Em complemento ao Matlab, podemos realizar uma análise visualmente melhor do sistema, por meio do Simulink. A simulação é realizada utilizando duas entradas Step e um bloco de Espaço de Estados, com as mesmas variáveis e condições utilizadas no Matlab. A resposta é dada em função da corrente e da velocidade angular. O esquema de blocos tal como as condições impostas e gráficos obtidos estão nas Figuras abaixo:

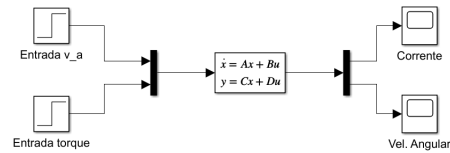


Figura 6: Esquema de blocos no Simulink

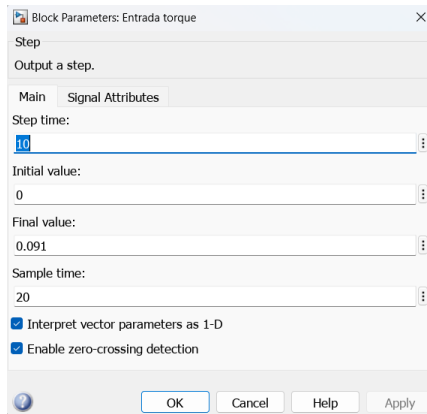


Figura 7: Entrada torque

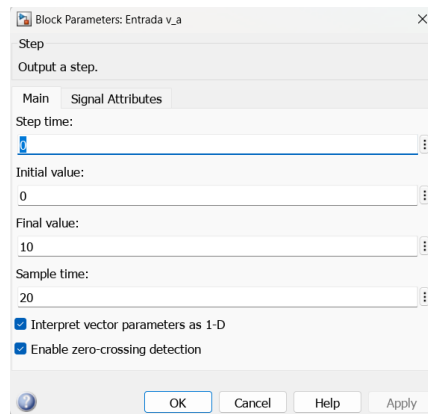


Figura 8: Entrada v.a.

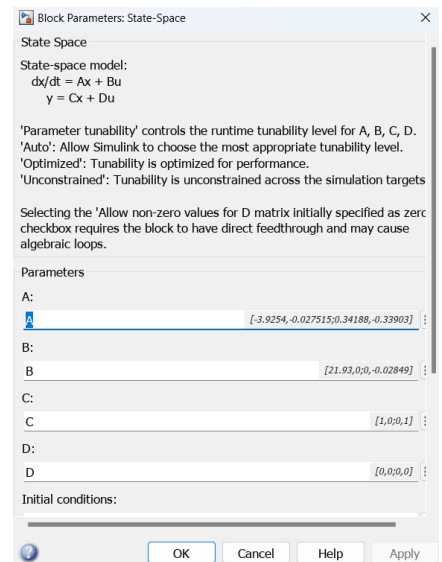


Figura 9: Entrada espaço de estados

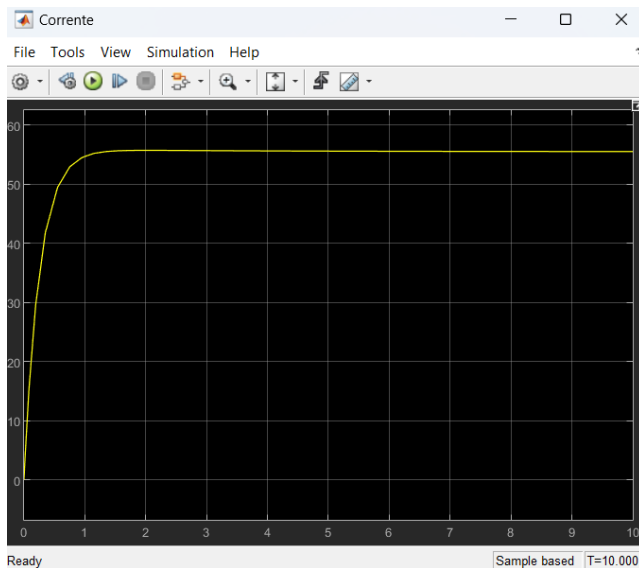


Figura 10: Saída corrente

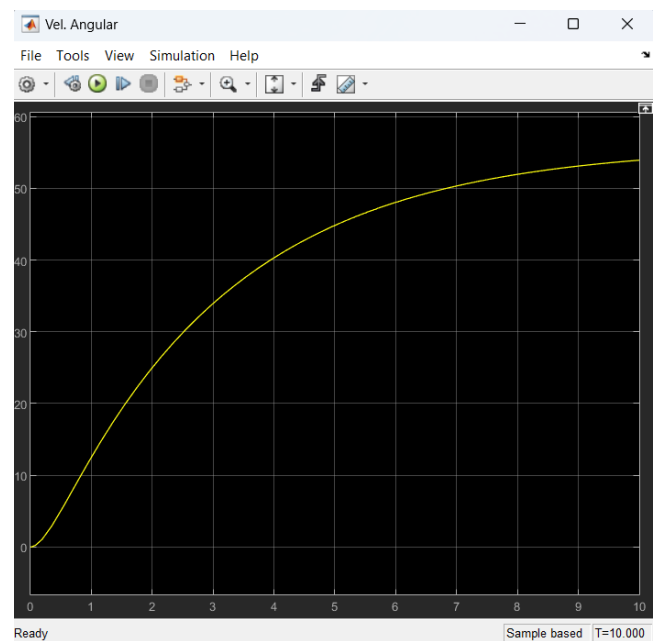


Figura 11: Saída velocidade angular

## 3 Drone 2D

### 3.1 Modelo simples

Um modelo fornecido pelo docente foi transferido para o Simulink e a nele construímos entradas e saídas que possibilitassem a análise do funcionamento. A imagem abaixo ilustra o modelo simplificado o qual ilustra o movimento de um quadricóptero em duas dimensões.

Observamos que foram colocadas duas entradas degraus, as quais foram ajustadas da seguinte forma:



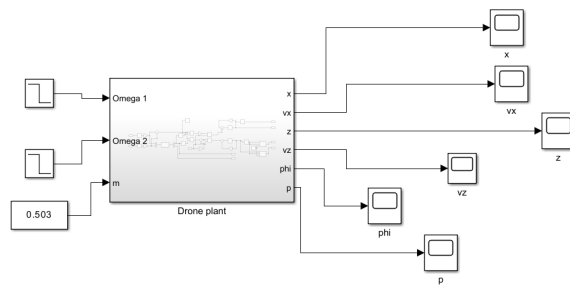


Figura 12: Esquema de blocos no Simulink modelo simples

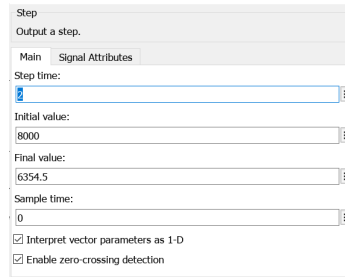


Figura 13: Configurações entrada step

Além disso, para as saídas, substituímos por blocos "Scope" os quais fazem o plot automático da variável de entrada em função do tempo. O resultado das variáveis está nas imagens abaixo.

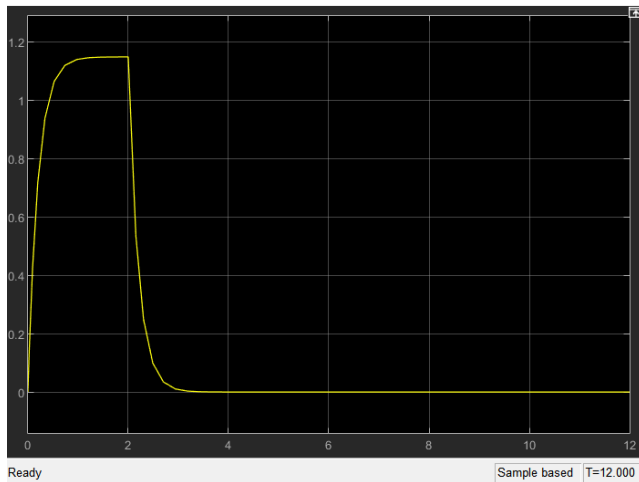


Figura 14: Saída da velocidade em z para modelo simples

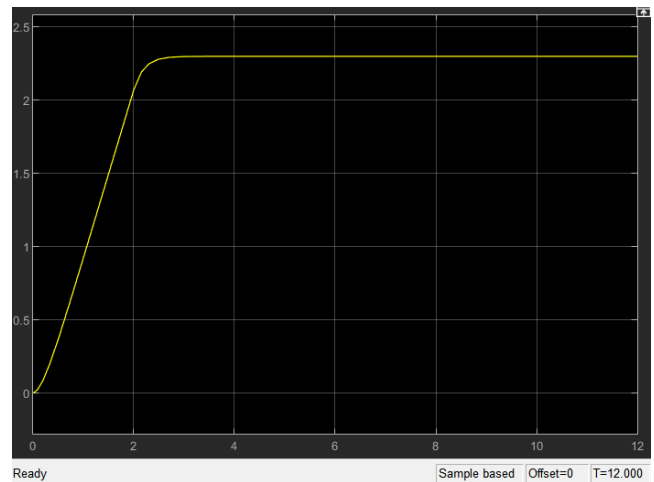


Figura 15: Saída da posição em z para modelo simples

De 0 a 2 segundos, o drone precisa operar com uma rotação maior do que a necessária para o hover (sustentação básica), a fim de ganhar altura, já de 2 a 12 segundos, o drone deve ajustar sua rotação para apenas o suficiente para manter-se na mesma altura, sem subir ou descer.

É importante notar que as entradas que utilizamos para controlar o drone não são práticas. Elas exigiram cálculos prévios e, mesmo assim, não foram capazes de atingir a altura desejada de 5 metros. Além disso, essas entradas não proporcionam uma interface realística entre o drone e o operador, tornando o controle manual complicado e impreciso. Por essa razão, há uma necessidade clara de

implementar um sistema de controle automático para garantir que o drone possa alcançar e manter a altura desejada de maneira eficiente e confiável.

### 3.1.1 Motores

A primeira interferência no modelo que faremos será na entrada, substituindo pelo modelo descrito na imagem abaixo:

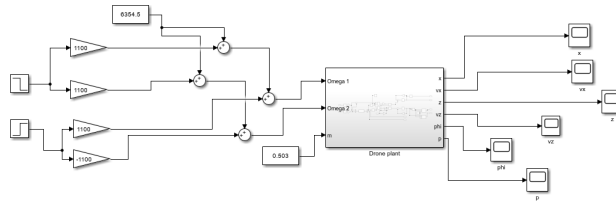


Figura 16: Esquema de blocos no Simulink modelo com lógica de motores

Assim, se estabelece uma interface mais lógica entre o comando de entrada e a resposta de saída. Variando o controle step de 0 a 1, ajusta-se a rotação dos motores de forma proporcional, resultando na elevação do drone. Quando o comando é "solto" e retorna a zero, o valor de referência mantém o drone em sua altura atual, evitando que ele caia.

Contudo, nesse modelo ainda não resolvemos a questão do feedback da posição em z, portando não se trata ainda de uma malha fechada. Ainda assim, abaixo temos as respostas de velocidade e posição em z para a configuração reproduzida:

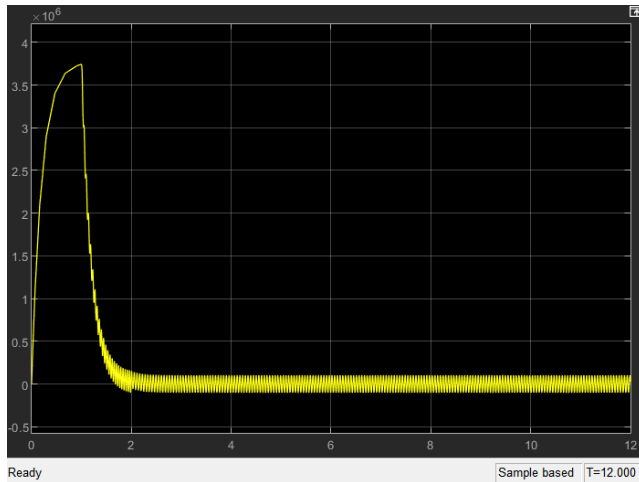


Figura 17: Saída da velocidade em z para modelo com lógica de motores

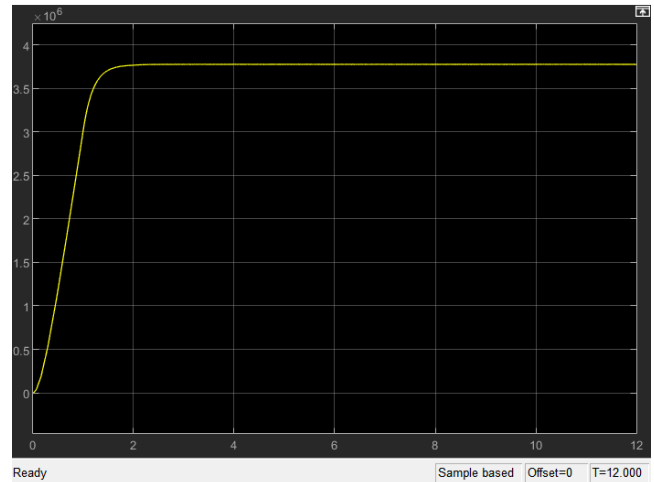


Figura 18: Saída da posição em z para modelo com lógica de motores

## 3.2 Controlador de altitude

Na próxima configuração, vamos introduzir o feedback do sistema adicionando um controlador de altitude, o que vai nos ajudar a mandar a altitude de referência de 5 metros do chão.

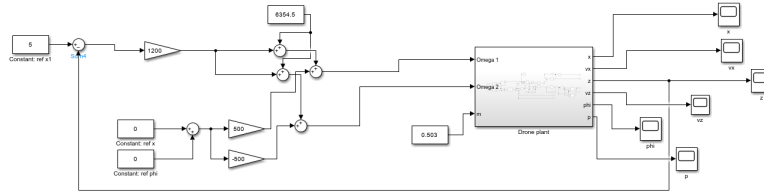


Figura 19: Esquema de blocos no Simulink modelo com controlador de altitude

Assim, o resultado para a saída se dá por:

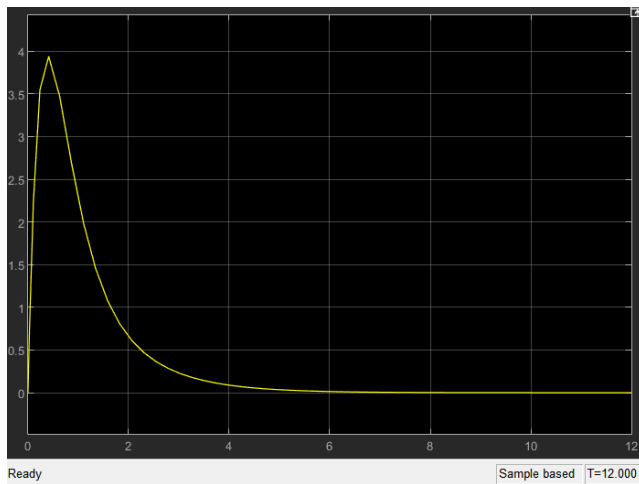


Figura 20: Saída da velocidade em z para modelo com controlador de altitude

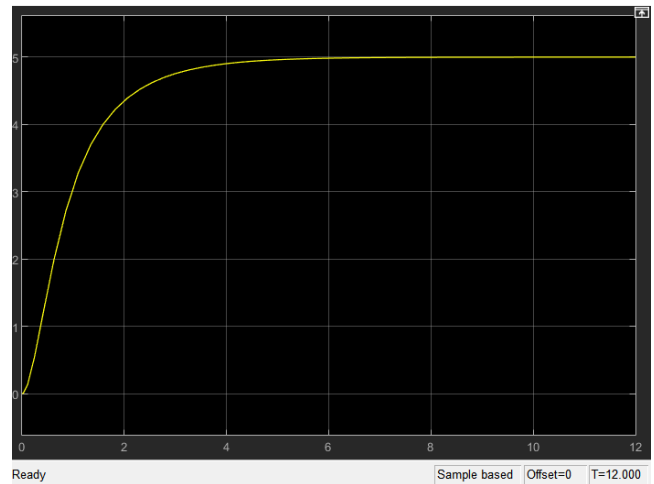


Figura 21: Saída da posição em z para modelo com controlador de altitude

Observamos que, apesar de melhorar o desempenho do drone, a resposta está lenta, demorando mais para estabilizar.

## 3.3 Controlador de altitude PID

Como forma de melhorar o desempenho e rapidez do nosso modelo, podemos implementar um controlador de altitude PID, como mostra a figura abaixo.

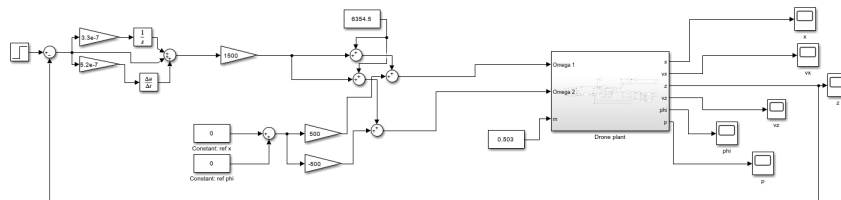


Figura 22: Modelo com controlador de altitude PID

A partir dele, nossas saídas referentes a velocidade e posição em z são:

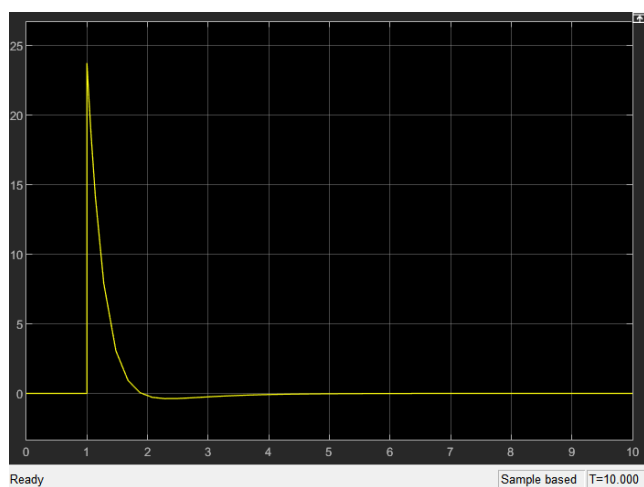


Figura 23: Saída da velocidade em z

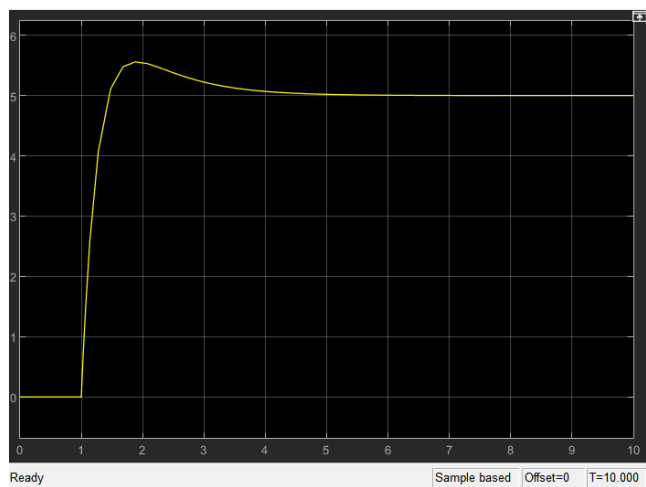


Figura 24: Saída da posição em z

Dessa forma, podemos partir para melhores arquiteturas de projeto, visando manter a estabilidade e refinar o desempenho do drone.

## 4 Conclusão

---

A implementação e simulação do drone 2D e do motor DC no Simulink foram concluídas com sucesso. Todas as estratégias de controle desenvolvidas para o drone demonstraram eficiência na manutenção da trajetória desejada, evidenciando a eficácia dos métodos utilizados. As simulações do motor DC também apresentaram resultados satisfatórios, comprovando a precisão dos modelos matemáticos e dos parâmetros empregados. Estes resultados reforçam a robustez do ambiente Simulink para a modelagem e controle de sistemas dinâmicos, fornecendo uma base sólida para futuras pesquisas e aplicações práticas em robótica e automação, tanto para o âmbito da graduação quanto para a engenharia de forma geral.

## 5 Referências

---

- DOUGLAS, Brian. Drone Simulation and Control, Part 1: Setting Up the Control Problem. 2018. Disponível em: [https : //youtu.be/hGcGPUqB67Q?si = GpuCxAPK3qWOZHm0](https://youtu.be/hGcGPUqB67Q?si=GpuCxAPK3qWOZHm0). Acesso em: 8 jul. 2024.
- COSTA, Cesar. Modelagem de um motor de corrente continua. Disponível em: [https : //professorcesarcosta.com.br](https://professorcesarcosta.com.br). Acesso em: 8 jul. 2024.