

# Utilização de Algoritmo Genético para Estimação e Otimização de Parâmetros de um Controlador Nebuloso Adaptativo Usado no Controle da Atitude de um Drone Quadrotor

1<sup>st</sup> Lucas Jorge Caldeira Carvalho  
Programa de Pós-Graduação em Engenharia Elétrica  
Universidade Federal de Minas Gerais  
Av. Antônio Carlos 6627, 31270-901, Belo Horizonte, MG, Brasil  
lucasjcc@ufmg.br

2<sup>nd</sup> Mauro de Oliveira Prates  
Departamento de Engenharia Elétrica  
Universidade Federal de Viçosa  
Viçosa, Brasil  
mauroprates@ufv.br

3<sup>rd</sup> Víctor Costa da Silva Campos  
Departamento de Engenharia Eletrônica  
Universidade Federal de Minas Gerais  
Belo Horizonte, Brasil  
victor@cpdee.ufmg.br

**Resumo**—O quadrotor é um tipo de VANT (Veículos Aéreos Não Tripulados) que possui diversas vantagens e aplicações. O acoplamento dinâmico e o comportamento altamente não-linear impõem um carácter desafiador ao controle destas aeronaves. Neste trabalho um controlador nebuloso adaptativo capaz de resolver um problema de rastreamento de trajetória é desenvolvido. Os parâmetros do controlador são estimados e otimizados pelo uso de um algoritmo genético, já que a utilização de controladores nebulosos implica no ajuste de vários parâmetros e, à medida que a complexidade do processo aumenta, torna-se difícil estabelecer a configuração ideal destes componentes de um sistema nebuloso. Através de resultados obtidos por simulação computacional, mostrou-se que o controlador projetado é eficiente para resolver o problema proposto e que o algoritmo genético conseguiu otimizar os seus parâmetros.

**Palavras-chaves**—algoritmo genético, controle adaptativo, quadrotor, otimização, sistema nebuloso

## I. INTRODUÇÃO

O interesse em relação às pesquisas sobre Veículos Aéreos Não Tripulados (VANTs) vem aumentando de forma significativa nos últimos anos, principalmente devido à vasta gama de aplicações que vão desde o uso para o entretenimento até a utilização no campo civil e militar [1]. Essa classe de aeronave representa um avanço tecnológico por se tratar de sistemas completamente autônomos, não requerendo intervenção humana para executar uma determinada tarefa ou reduzindo esta intervenção por meio do manuseio de um controle conectado a um rádio [2].

O quadrotor, ou quadricóptero, é um tipo de VANT que é elevado e impulsionado por quatro rotores [3]. Ele possui

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

a vantagem de locomoção em locais com pouco espaço, capacidade de realizar manobras precisas em baixa e alta velocidade, pode levantar voo vertical, aterrissar em locais de difícil acesso, além de possuir a capacidade de realizar voos pairados [4].

A menor complexidade mecânica destas aeronaves em relação a outros modelos resulta em facilidades de construção e manutenção. Por outro lado, o acoplamento dinâmico e o comportamento altamente não-linear impõem um carácter desafiador ao controle dos quadrotoros [5].

A dificuldade de encontrar algoritmos de controle analítico para plantas complexas contribuiu para a expansão do uso de algoritmos genéticos (AG) e da programação genética na engenharia de controle [6], e seu uso juntamente com a lógica nebulosa vem crescendo rapidamente no mundo industrial para resolver problemas do mundo real com eficácia [7].

A utilização de controladores nebulosos implica no ajuste de vários parâmetros e, à medida que a complexidade do processo aumenta, torna-se difícil estabelecer a configuração ideal destes componentes de um sistema nebuloso [8].

O objetivo deste trabalho foi projetar um controlador nebuloso adaptativo capaz de, dada uma entrada de referência  $y_d(t)$ , fazer com que a saída do sistema  $y(t)$  seja tal que o erro  $e(t) = y_d(t) - y(t)$  seja o menor possível e que esse objetivo seja atingido com o menor esforço possível do controlador. Esse controlador usa o método de inferência Takagi-Sugeno de ordem zero para estimar o sinal de controle  $u(t)$ , e os parâmetros do controlador são estimados e otimizados de maneira *off-line* com o uso de AG, conforme mostrado na Figura 1. Um trabalho intitulado *Controle Nebuloso Adaptativo Direto de Drone Quadrotor* contendo o controle de mais malhas do quadrotor, mas que não utiliza AG, foi aceito no

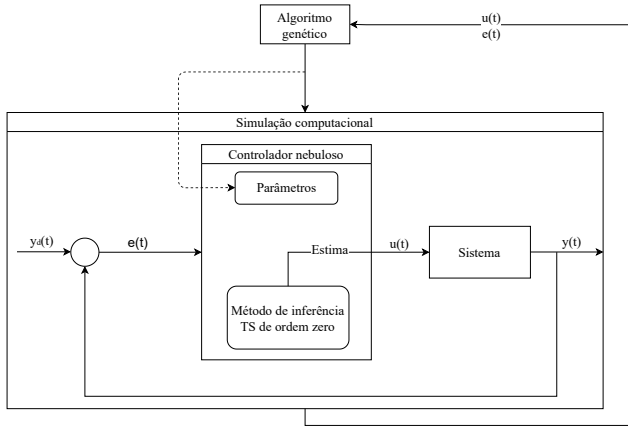


Figura 1. Diagrama do controlador projetado

Este artigo está assim dividido: a seção II apresenta a modelagem matemática de um drone quadrotor. Uma breve descrição do problema de controle tratado neste trabalho, assim como a definição de sistemas MIMO não-lineares, é mostrada na seção III. Uma introdução à sistemas nebulosos é feita na seção IV e, posteriormente, na seção V, é detalhado o controlador projetado, com suas leis de controle e adaptação. A seção VI traz uma breve teoria sobre diferenciadores exatos e robustos. Definições e conceitos, assim como a descrição do AG usado neste trabalho, são apresentados na seção VII. Por fim, os resultados de uma simulação computacional são apresentados na seção VIII.

## II. MODELAGEM DO QUADROTOR

Na literatura existem duas abordagens clássicas para a modelagem matemática de veículos aéreos: uma baseada no formalismo de Newton-Euler e outra no formalismo de Euler-Lagrange [4]. Essa última abordagem será a usada neste trabalho.

Além disso, a modelagem foi dividida em dois subsistemas de acordo com o tipo de movimentos analisado: rotação e translação.

### A. Princípio de funcionamento de um quadrotor

Embora o quadricóptero possua seis graus de liberdade (GDL), ele é equipado apenas com quatro hélices, não sendo, portanto, possível controlar independentemente todos eles [10].

Devido ao seu formato físico, os quatro movimentos controláveis são os de atitude (rolagem  $[\phi]$ , arfagem  $[\theta]$  e guinada  $[\psi]$ ) e altitude. Os movimentos translacionais acabam sendo dependentes desses outros [11].

### B. Cinemática

Cinemática é o ramo da Física que se dedica ao estudo do movimento sem se preocupar com o que o causou.

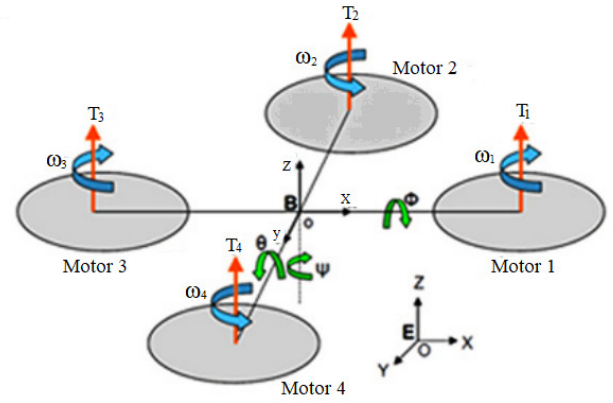


Figura 2. Diagrama de corpo livre de um quadricóptero [12].

1) *Posição e orientação*: A posição linear absoluta do quadricóptero é definida, em relação ao referencial fixo E (Figura 2), pelo vetor  $\epsilon$  (1). A atitude, ou seja, a posição angular, é definida na estrutura inercial com os três ângulos de Euler  $\eta_e$  (2) [13].

$$\epsilon = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (1)$$

$$\eta_e = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \quad (2)$$

2) *Rotação*: Embora o sistema de coordenadas do corpo (B) seja útil para expressarmos as velocidades de translação e rotação da aeronave, ele não serve para indicar a posição e orientação da mesma [14]. Para isso, utilizaremos o sistema de coordenadas fixo na Terra (E).

Uma matriz de rotação transforma o conjunto de coordenadas que representa um objeto, em um conjunto cartesiano ortogonal, sem alterar sua forma ou tamanho [15]. Essas matrizes são amplamente utilizadas para descrever a orientação de qualquer corpo móvel, devido à facilidade da utilização da álgebra matricial [5].

A matriz de rotação do sistema de coordenadas do corpo para o sistema de coordenadas fixo é dada por [13]:

$$R = \begin{bmatrix} c_\psi c_\theta & c_\psi s_\theta s_\phi + s_\psi c_\phi & c_\psi s_\theta c_\phi - s_\psi s_\phi \\ -s_\psi c_\theta & -s_\psi s_\theta s_\phi + c_\psi c_\phi & -s_\psi s_\theta c_\phi - c_\psi s_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix} \quad (3)$$

Em que  $c_\theta = \cos\theta$ ,  $s_\psi = \sin\psi$ , e assim por diante.

### C. Dinâmica

Dinâmica é o ramo da Física que estuda a relação entre o movimento e as forças que o gerou.

As equações foram desenvolvidas utilizando as considerações feitas por [16]:

- A estrutura é de um corpo rígido.
- A estrutura é simétrica.

- O centro de gravidade coincide com o centro do referencial inercial.
- As hélices são rígidas.

1) *Dinâmica translacional*: A equação de Euler-Lagrange é baseada na energia cinética e na energia potencial [16].

$$T_i = \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} \quad (4)$$

Onde

- L: função Lagrangiana (T-V).
- T: energia cinética total.
- V: energia potencial total.
- $q_i$ : coordenada generalizada.
- $\dot{q}_i$ : primeira derivada da coordenada generalizada em relação ao tempo.
- $T_i$ : conjunto de forças generalizadas que regem o sistema.

A Equação (4) e as considerações feitas em relação ao sistema nos permitem chegar no seguinte modelo dinâmico para o movimento de translação do quadrotor [5]:

$$\begin{aligned} \ddot{x} &= (c\psi s\theta c\phi + s\psi s\theta) \frac{U_1}{m} \\ \ddot{y} &= (s\psi s\theta c\phi - c\psi s\theta) \frac{U_1}{m} \\ \ddot{z} &= (-g + c\theta c\phi) \frac{U_1}{m} \end{aligned} \quad (5)$$

Onde  $g$  é a gravidade,  $m$  é a massa do quadricóptero e  $U_1 = \Omega_1 + \Omega_2 + \Omega_3 + \Omega_4$ , com  $\Omega_i$ ,  $i = 1, \dots, 4$  sendo as forças produzidas em cada um dos quatro motores.

2) *Dinâmica rotacional*: Segundo [5], a energia rotacional total do sistema é

$$L_{rot} = \frac{1}{2} \dot{\eta}_e^T J(\eta_e) \dot{\eta}_e \quad (6)$$

$J(\eta_e)$  é o Jacobiano das coordenadas gerais de E.

Como a estrutura do quadrotor considerada é simétrica, temos a seguinte matriz inercial

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \quad (7)$$

$I_{xx}$ ,  $I_{yy}$  e  $I_{zz}$  são os vetor de inércia do corpo B.

Para rotação, a equação Euler-Lagrange fica

$$\frac{d}{dt} \left( \frac{\partial L_{rot}}{\partial \dot{\eta}_e} \right) - \frac{\partial L_{rot}}{\partial \eta_e} = \tau_{\eta_e} \quad (8)$$

Sendo  $l$  a distância do centro do quadrotor à cada hélice,  $\tau_{\eta_e}$  é o vetor de torques, definido como

$$\tau_{\eta} = \begin{bmatrix} lU_2 \\ lU_3 \\ U_4 \end{bmatrix} \quad (9)$$

$U_2 = \Omega_3 - \Omega_1$ ,  $U_3 = \Omega_2 - \Omega_4$  e  $U_4 = \Omega_x + \Omega_z - \Omega_w - \Omega_y$ . A força  $\Omega_e$ , com  $e = (x, z, w, y)$ , é a força que aparece quando as hélices giram. Essa força é ortogonal a  $\Omega_i$ , com  $i = (1, 2, 3, 4)$ .

A equação dinâmica que rege o quadrotor é dada por [5]

$$J(\eta_e) \ddot{\eta} + C(\eta, \dot{\eta}) \dot{\eta} = \tau_{\eta} \quad (10)$$

$C(\eta_e, \dot{\eta}_e) \dot{\eta}_e$  é a matriz de coriolis, que define os efeitos giroscópicos e o sistema centrífugo [13].

A partir de (6) a (10) obtém-se as equações da dinâmica rotacional

$$\begin{aligned} \ddot{\phi} &= -\dot{\psi} \dot{\theta} c\phi + \frac{lc\psi}{I_{yy}} U_2 - \frac{ls\psi}{I_{yy}} U_3 + \frac{I_{yy} - I_{zz}}{I_{yy}} (\dot{\psi} - \dot{\theta} s\phi) \dot{\theta} c\phi \\ \ddot{\theta} &= \frac{\dot{\psi}}{c\phi} + \dot{\phi} \dot{t}\phi + \frac{ls\psi}{I_{yy} c\phi} U_2 + \frac{lc\psi}{I_{yy} c\phi} U_3 - \frac{I_{yy} - I_{zz}}{I_{yy}} (\dot{\psi} - \dot{\theta} s\phi) \frac{\dot{\phi}}{c\phi} \\ \ddot{\psi} &= \dot{\psi} \dot{t}\phi + \frac{\dot{\phi} \dot{\theta}}{c\phi} + \frac{ls\dot{\psi} \dot{t}\phi}{I_{yy}} U_2 + \frac{lc\dot{\psi} \dot{t}\phi}{I_{yy}} U_3 + \frac{1}{I_{zz}} U_4 - \frac{I_{yy} - I_{zz}}{I_{yy}} (\dot{\psi} - \dot{\theta} s\phi) \dot{\phi} \dot{t}\phi \end{aligned} \quad (11)$$

### III. DESCRIÇÃO DO PROBLEMA

Neste trabalho, um sistema multivariável (MIMO) não-linear com  $n$  estados é definido como

$$\mathbf{y}^{(r)} = \mathbf{f}(\mathbf{x}) + \mathbf{G}(\mathbf{x}) \mathbf{u} \quad (12)$$

$\mathbf{x} = [x_1, \dots, x_n]^T \in \mathbb{R}^n$  é o vetor de estado,  $\mathbf{u} = [u_1, \dots, u_p]^T \in \mathbb{R}^p$  o vetor de entrada de controle,  $\mathbf{y} = [y_1, \dots, y_p]^T \in \mathbb{R}^p$  o vetor de saída, o vetor  $\mathbf{f}(\mathbf{x})$  e a matriz  $\mathbf{G}(\mathbf{x})$  são formados por funções não-lineares desconhecidas, e  $r$  representa o grau da equação diferencial.

O controlador projetado neste trabalho foi desenvolvido para um problema de rastreamento de trajetória, em que a saída  $\mathbf{y}(t)$  do sistema controlado em malha fechada deve seguir uma trajetória desejada  $\mathbf{y}_d(t) = [y_{d1}(t), \dots, y_{dp}(t)]^T$ . Para isso, assume-se que cada componente do vetor de entrada de controle ideal  $\mathbf{u}^* = [u_1^*, \dots, u_p^*]$  possa ser aproximado por meio de sistema nebuloso.

O foco do controlador é o controle dos ângulos de rolagem e arfagem. O ângulo de guinada ( $\psi$ ) é considerado uma entrada não controlada e, por simplificação, o sinal de controle é o vetor das forças que interferem nos ângulos  $\phi$  e  $\theta$ , com  $\mathbf{u} = [U_2, U_3]^T$ .

### IV. SISTEMAS NEBULOSOS

#### A. Conjuntos Nebulosos

O advento da lógica nebulosa, ou lógica fuzzy, foi causado pela necessidade de um método capaz de expressar de uma maneira sistemática quantidades imprecisas, vagas, mal definidas [17].

Claramente, a "classe de todos os reais muito maiores que 1", ou "a classe de mulheres bonitas" ou "a classe de homens altos" não constituem classes ou conjuntos no senso matemático comum [18].

Um conceito fundamental na teoria dos conjuntos nebulosos é o grau de pertinência. O grau de pertinência de um determinado objeto a um conjunto nebuloso é um número real definido no intervalo  $[0, 1]$  que representa quão verdadeiro é

a afirmação de que um certo elemento pertence a um dado conjunto. Um conjunto nebuloso "A" pode ser caracterizado por sua função característica, também denominada função de pertinência e denotada por  $\mu_A(x)$ , assim definida:

$$\mu_A(x) = U \rightarrow [0,1] \quad (13)$$

U é denominado "universo do discurso", definido como o espaço ou domínio das variáveis deste conjunto, e  $x \in U$  [19].

Assim como nos demais conjuntos, podemos fazer operações com conjuntos nebulosos:

- Interseção (e):  $\mu_{A \cup B} = \text{prod}(\mu_A, \mu_B)$
- União (ou):  $\mu_{A \cap B} = \text{max}(\mu_A, \mu_B)$
- Complemento (não):  $\mu_{\bar{A}} = 1 - \mu_A$

#### B. Método de Inferência Takagi-Sugeno (TS) de Ordem Zero

O método de inferência TS é usado para determinar  $\mathbf{u}^*$ . Ele mapeia cada uma das m entradas do vetor  $\mathbf{z} = [z_1, \dots, z_m]^T$ , entrada do sistema nebuloso, em um escalar de saída  $y_f$ . Sendo  $A_i^k$ ,  $i = 1, \dots, m$ , conjuntos nebulosos, temos, para N regras ( $R^k$ )

$$R^k : \text{Se } z_1 \text{ é } A_1^k \text{ e } \dots \text{ e } z_m \text{ é } A_m^k \text{ ENTÃO} \\ u_f \text{ é } u_f^k \quad (k = 1, \dots, N) \quad (14)$$

A saída final do sistema, usando a fuzzificação singleton e o operador produto como operador de interseção, é dada por [20]

$$u_f(\mathbf{z}) = \frac{\sum_{k=1}^N \mu_k(\mathbf{z}) u_f^k}{\sum_{k=1}^N \mu_k(\mathbf{z})} \quad (15)$$

com

$$\mu_k(\mathbf{z}) = \prod_{i=1}^m \mu_{A_i^k}(z_i) \quad (16)$$

Sendo

$$w_k(\mathbf{z}) = \frac{\mu_k(\mathbf{z})}{\sum_{j=1}^N \mu_j(\mathbf{z})}, \quad k = 1, \dots, N \quad (17)$$

a saída pode ser reescrita de forma compacta

$$u_f(\mathbf{z}) = \mathbf{w}^T(\mathbf{z})\Theta \quad (18)$$

Em que  $\Theta = [u_f^1, \dots, u_f^N]^T$  e  $\mathbf{w}(\mathbf{z}) = [w_1(\mathbf{z}), \dots, w_N(\mathbf{z})]^T$ .

#### V. CONTROLADOR NEBULOSO ADAPTATIVO

[20] mostrou que para a lei de controle

$$\mathbf{u}(\mathbf{z}) = \mathbf{w}^T(\mathbf{z})\Theta \quad (19)$$

a seguinte lei de adaptação pode ser usada

$$\dot{\Theta} = \eta \mathbf{w}(\mathbf{z}) \left[ \dot{\mathbf{s}} + \mathbf{Ks} + \mathbf{K}_0 \tanh\left(\frac{\mathbf{s}}{\epsilon_0}\right) \right] - \eta \sigma \Theta \quad (20)$$

Em que  $\mathbf{K} = \text{diag}[k_1, \dots, k_n]$ ,  $\mathbf{K}_0 = \text{diag}[k_{0_1}, \dots, k_{0_n}]$ , com  $K_i > 0$  e  $K_{0_i} > 0$ , para  $i = 1, \dots, n$ ,  $\epsilon_0$  e  $\sigma$  são pequenas constantes positivas e  $\eta$  é uma constante positiva.

O termo depois da subtração é chamado de modificação- $\sigma$ . Ele deixa a lei de adaptação robusta aos desvios pelos erros de aproximação.

O vetor  $\mathbf{s}$  é chamado por [20] de erro de rastreamento filtrado.

$$s_i(t) = \left( \frac{d}{dt} + \lambda_i \right)^{r_i-1} e_1(t), \quad \lambda_i > 0, \quad i = 1, \dots, n \quad (21)$$

$\lambda_i$  são constantes positivas e  $e_i(t)$  são os erros de rastreamento de trajetória.

$$e_i(t) = y_{d_i}(t) - y_i(t), \quad i = 1, \dots, n \quad (22)$$

O erro de rastreamento filtrado representa uma equação diferencial cuja solução implica que o erro de trajetória converge para zero com uma constante de tempo igual a  $(r_i - 1)/\lambda_i$ . Usando o erro filtrado, o objetivo de controle torna-se manter o escalar  $s_i(t)$  igual a zero, ao invés do problema original de estabilização do vetor  $[e_1, \dots, e_n^{r_i-1}]$  de  $r_i - \text{dimensões}$  [20].

A derivada temporal do erro rastreamento de trajetória é dada por

$$\dot{\mathbf{s}} = \mathbf{Ks} - \mathbf{K}_0 \tanh\left(\frac{\mathbf{s}}{\epsilon_0}\right) + \mathbf{G}(\mathbf{x})(\mathbf{u}^* - \mathbf{u}) \quad (23)$$

Como assume-se  $\mathbf{u}^*$ ,  $\mathbf{G}(\mathbf{x})$  e  $\mathbf{f}(\mathbf{x})$  desconhecidos, não é possível o cálculo do valor de  $\dot{\mathbf{s}}$ . [20] propõem uma implementação digital de (23). Como maioria das especificações, projetos e análise de comportamento dos processos, sistemas e plantas são feitas em tempo contínuo [21], e uma das estratégias para o desenvolvimento de controladores digitais é a aproximação discreta de controladores contínuos [22], neste trabalho, adotou-se o método sugerido por [23], que estima o valor de  $\dot{\mathbf{s}}$  com o uso de um diferenciador robusto e exato.

#### VI. DIFERENCIADOR ROBUSTO E EXATO (RED)

A diferenciação de sinais em tempo real é uma questão que enfrenta problemas de ordem prática. A tarefa de gerar uma estimativa da derivada temporal de um sinal base é dificultada quando o sinal de entrada do diferenciador é corrompido por ruído de alta frequência, uma vez que esta classe de ruídos possui derivada com amplitudes elevadas [24].

A maioria dos diferenciadores conhecidos é capaz de gerar estimativas próximas da derivada do sinal base, além de rejeitar razoavelmente ruídos de alta frequência. Para isso, eles utilizam uma largura de banda finita, de modo a filtrar as componentes de alta frequência do sinal de entrada. Deste modo, eles não conseguem fornecer uma resposta exata na ausência de ruído, sendo, portanto, diferenciadores robustos, mas não exatos [24].

Este trabalho utiliza o diferenciador baseado em modos deslizantes proposto por [25]. Seu objetivo é que dado um sinal de entrada mensurável e localmente limitado  $f(t)$  a saída  $u_{RED}$  do RED deve ser tal que  $u_{RED} - \dot{f}(t) = 0$ , ou seja, que sua saída convirja para a derivada temporal do sinal de entrada.

$$\dot{x}_{RED} = u_{RED} \quad (24)$$

$$\begin{aligned} u_{RED} &= u_{RED_1} - \kappa |x_{RED} - f(t)|^{\frac{1}{2}} \text{sign}(x_{RED} - f(t)) \\ \dot{u}_{RED_1} &= -\alpha \text{sign}(x_{RED} - f(t)), \quad \kappa, \alpha > 0. \end{aligned} \quad (25)$$

Seja  $f(t)$  um sinal cuja derivada possua constante de Lipschitz  $C > 0$ , a observância da seguinte expressão é condição suficiente para a convergência de  $u_{RED}(t)$  para  $f(t)$  [25]:

$$\alpha > C, \quad \kappa^2 \geq 4C \frac{\alpha+C}{\alpha-C} \quad (26)$$

Este diferenciador pode fornecer, na ausência de ruído, a derivada exata. Na presença de ruído o RED possui acurácia proporcional a raiz quadrada da magnitude do ruído. Deve-se destacar, ainda, que o estado do RED não pode escapar em tempo finito, dado que o sinal de entrada possui derivada segunda limitada [26].

## VII. ALGORITMO GENÉTICO (AG)

A computação evolutiva é constituída pelos métodos computacionais inspirados na teoria da evolução natural das espécies. Tais métodos são baseados em uma população de indivíduos sujeitos a avaliação, modificações e seleção. As instâncias dos algoritmos fundamentados em princípios evolutivos são chamadas de algoritmos evolutivos [27].

AG é uma classe particular de algoritmos evolutivos que usam técnicas inspiradas pela biologia evolutiva como hereditariedade, mutação, seleção natural e recombinação para encontrar soluções aproximadas em problemas de otimização e busca. Esses algoritmos são implementados como uma simulação de computador, em que uma população de representações abstratas de solução é selecionada em busca de soluções melhores. A evolução geralmente se inicia a partir de um conjunto de soluções criado aleatoriamente e é realizada por meio de gerações. A cada geração, a adaptação de cada solução na população é avaliada, alguns indivíduos são selecionados para a próxima geração, são recombinados ou sofrem mutação para formar uma nova população. A nova população então é utilizada como entrada para a próxima iteração do algoritmo [9].

A função de custo ( $J$ ) a ser otimizada pelo AG é a raiz do erro quadrático médio (RMSE, em inglês) somada à raiz do sinal de controle quadrático médio.

$$J = \left[ \frac{1}{N} (\mathbf{e}\mathbf{e}^T + \mathbf{u}\mathbf{u}^T) \right]^{1/2} \quad (27)$$

Sendo  $N$  o tamanho dos vetores  $\mathbf{e}$  e  $\mathbf{u}$ .

O algoritmo deve encontrar a configuração dos parâmetros  $\mathbf{K}$ ,  $\mathbf{K}_0$ ,  $\lambda$ ,  $\epsilon_0$ ,  $\sigma$  e  $\eta$  do controlador que minimiza (27).

O pseudo-código do AG desenvolvido neste trabalho se encontra abaixo.

**Entrada:** Quantidade de gerações, probabilidade de

1► mutação, tamanho da população ( $N$ ), tamanho do torneio

**Saída:** Melhor indivíduo encontrado

2► Gerar população inicial aleatoriamente

3► Avaliar indivíduos

4► **enquanto** máximo de gerações não for atingido **faça**

5► Selecionar pais por torneio

6► Realizar cruzamento

7► Gerar número aleatório entre 0 e 1

8► **se** número aleatório  $\leq$  probabilidade de mutação **então**

9► Realizar mutação

10► **fim se**

11► Avaliar filhos

12► Inserir filhos na população

13► Ordenar por aptidão

14► Manter os  $N$  melhores indivíduos

15► **fim enquanto**

16► Retornar o melhor indivíduo encontrado

### A. Representação dos indivíduos

Um indivíduo é uma possível solução para o problema a ser otimizado. No caso deste trabalho, sua representação computacional é dada por um vetor de números reais, chamado de cromossomo, em que cada posição deste vetor, chamada de gene, representa um parâmetro do controlador a ser otimizado.

Dois exemplos de cromossomos:

$$cromossomo_1 = [K_1^1 \ K_2^1 \ K_{0_1}^1 \ K_{0_2}^1 \ \lambda_1^1 \ \lambda_2^1 \ \epsilon_0^1 \ \sigma^1 \ \eta^1]$$

$$cromossomo_2 = [K_1^2 \ K_2^2 \ K_{0_1}^2 \ K_{0_2}^2 \ \lambda_1^2 \ \lambda_2^2 \ \epsilon_0^2 \ \sigma^2 \ \eta^2]$$

⋮

$$cromossomo_N = [K_1^N \ K_2^N \ K_{0_1}^N \ K_{0_2}^N \ \lambda_1^N \ \lambda_2^N \ \epsilon_0^N \ \sigma^N \ \eta^N]$$

Os parâmetros vetoriais ( $\mathbf{K}$ ,  $\mathbf{K}_0$  e  $\lambda$ ) foram decompostos de forma que seus componentes são representados em genes diferentes.

### B. Inicialização da população

A população é um conjunto de indivíduos. Sua inicialização foi feita de forma aleatória.

Tomando como base os valores dos parâmetros usados por [23], realizou-se simulações computacionais para definir os extremos dos intervalos em que os valores dos parâmetros não levam o sistema à instabilidade. Posteriormente, valores aleatórios nesses intervalos foram gerados para compor os cromossomos dos indivíduos da população inicial.

### C. Avaliação da população

Para cada indivíduos da população é executada uma simulação do sistema usando os valores dos parâmetros do controlador contidos em seu cromossomo. A avaliação usa o erro e o sinal de controle obtido na simulação em (27).

#### D. Seleção dos pais

A seleção dos pais que realizarão o cruzamento foi feita por torneio. Nele um número estipulado de indivíduos são sorteados aleatoriamente e os dois melhores indivíduos (menor valor de erro quadrático) dentre eles são escolhidos para serem os indivíduos pais.

Quanto maior o número de indivíduos selecionados para participarem do torneio, mais elitista torna-se o algoritmo, já que a probabilidade dos melhores indivíduos serem selecionados é maior. Caso essa quantidade seja igual ao tamanho da população, os dois melhores indivíduos sempre serão selecionados como pais, o que pode provocar uma convergência precoce do algoritmo.

#### E. Cruzamento

O cruzamento recombina-se as características das soluções selecionadas (pais) para gerar outros indivíduos (filhos).

O operador de cruzamento usado tem os seguintes passos: um número aleatório ( $\beta$ ) entre 0 e 1 é gerado. Os cromossomos dos filhos são gerados pelas equações:

$$\begin{aligned} cromossomo_{filho_1} &= \beta \cdot cromossomo_{pai_1} + \\ & (1 - \beta) \cdot cromossomo_{pai_2} \\ cromossomo_{filho_2} &= (1 - \beta) \cdot cromossomo_{pai_1} + \\ & \beta \cdot cromossomo_{pai_2} \end{aligned} \quad (28)$$

#### F. Mutação

A mutação permite que algumas características dos indivíduos resultantes sejam alteradas para gerar mais variedade genética. Com certa porcentagem de acontecimento, cada filho pode sofrer uma alteração em seus genes.

No operador escolhido, um gene aleatório do indivíduo é selecionado e seu valor é substituído por outro dentro do intervalo válido definido para o parâmetro do controlador que aquele gene representa.

#### G. Exclusão dos piores indivíduos

Os indivíduos filhos gerados são inseridos na população e os  $N$  piores indivíduos desta população aumentada são excluídos, de forma que a tamanho da população permaneça sempre constante.

### VIII. RESULTADOS

Conforme dito anteriormente, o ângulo de guinada foi tratado como uma entrada não controlada com valor definido pela função  $\psi = \text{sen}(0.1t)$ .

Os valores das constantes do quadrotor foram retirados do trabalho de [23], com a gravidade  $g = 9.78 \frac{m}{s^2}$  e os parâmetros do quadrotor iguais a  $m = 1.2 \text{ kg}$ ,  $l = 0.31 \text{ m}$ ,  $I_{yy} = 9.8 \cdot 10^{-3} \text{ Nms}^2$  e  $I_{zz} = 16.3 \cdot 10^{-3} \text{ Nms}^2$ .

Os valores dos parâmetros para o derivador robusto e exato, usado para estimar o valor de  $\dot{s}$ , são  $\kappa = \sqrt{10}$  e  $\alpha = 11$ .

Para cada variável de entrada  $z_i$ ,  $i = 1, \dots, 4$ , três funções de pertinência Gaussianas são usadas:

$$\begin{aligned} \mu_{A_1^1}(z_i) &= \exp\left(-\frac{1}{2} \left(\frac{z_i + 1.25}{0.6}\right)^2\right) \\ \mu_{A_1^2}(z_i) &= \exp\left(-\frac{1}{2} \left(\frac{z_i}{0.6}\right)^2\right) \\ \mu_{A_1^3}(z_i) &= \exp\left(-\frac{1}{2} \left(\frac{z_i - 1.25}{0.6}\right)^2\right) \end{aligned}$$

Dois sistemas nebulosos na forma (19) são usados para gerar os sinais de controle  $U_2$  e  $U_3$ , e cada sistema possui como entrada o vetor  $\mathbf{z} = [e_\phi(t), \dot{e}_\phi(t), e_\theta(t), \dot{e}_\theta(t)]^T$ .

O universo de discurso  $U$  das variáveis de entrada do vetor  $\mathbf{z}$  não foi limitado a nenhum valor, de modo que os erros e as variações dos erros pudessem variar no conjunto  $\mathbb{R}$ .

As equações diferenciais que regem a saída  $\mathbf{y}$  são de segunda ordem. Portanto, os elementos do vetor  $\mathbf{r}$  são iguais 2.

O objetivo de controle é fazer com que as saídas  $\phi$  e  $\theta$  sigam as trajetórias descritas pelos sinais  $y_{d_\phi}(t)$  e  $y_{d_\theta}(t)$ , respectivamente. Neste trabalho, esses sinais possuem parte senoidal e parte rampas com inclinações elevadas.

Nos intervalos de tempo  $8 \text{ s} \leq t \leq 12 \text{ s}$  e  $18 \text{ s} \leq t \leq 22 \text{ s}$  inseriu-se ruídos de processo e medição, respectivamente, ambos brancos e com amplitude de aproximadamente  $0,16 \text{ rad}$ .

Os valores iniciais dos parâmetros estimados  $\Theta_1(0)$  e  $\Theta_2(0)$  são iguais a zero e as condições iniciais são  $\mathbf{x} = [0 \ 0 \ 0 \ 0]^T$ .

Os intervalos encontrados usados pelo AG para gerar os valores aleatórios são:  $K_i \in [8,15]$ ,  $K_{0_i} \in [20,60]$ ,  $\lambda \in [8,16]$ ,  $\epsilon_0 \in [0,1,0,9]$ ,  $\sigma \in [0,1,0,9]$  e  $\eta \in [10,40]$ , com  $i = 1,2$ .

Os parâmetros do AG são: tamanho da população igual a 40, quantidade máxima de gerações igual a 50 e a quantidade de indivíduos participantes do torneio igual a 6. A probabilidade de mutação inicia-se igual a 0,1 e cresce 0,05 a cada 10 iterações do algoritmo. O único critério de parada usado foi atingir o número máximo de gerações estipulado.

As Figuras 3 e 4 mostram, respectivamente, as respostas para os ângulos de rolagem e arfagem encontradas manualmente e as respostas encontradas para o controlador com os parâmetros otimizados pelo AG.

Assim como qualquer controlador adaptativo, existe um primeiro momento de aprendizado do controlador no qual o erro é maior. Posteriormente, notamos que ambos os controladores, com parâmetros sintonizados manualmente e via AG, conseguem fazer com que os sinais de saída da planta convirjam para os sinais de referência.

Outra observação importante diz respeito ao comportamento do controlador nebuloso adaptativo proposto. Ele apresentou maior robustez ao ruído de processo se comparado ao ruído de medição.

Nota-se que os erros angulares de ambos os controladores ficaram muito próximos (Figura 5). Isso pode ser explicado pelo esforço significativo dedicado à sintonização manual do controlador, obtendo-se, assim, parâmetros relativamente próximos aos encontrados pela otimização via AG.

O mesmo não é observado pelo esforço de controle (Figura 6). Claramente, o controlador otimizado apresentou um esforço de controle menor.

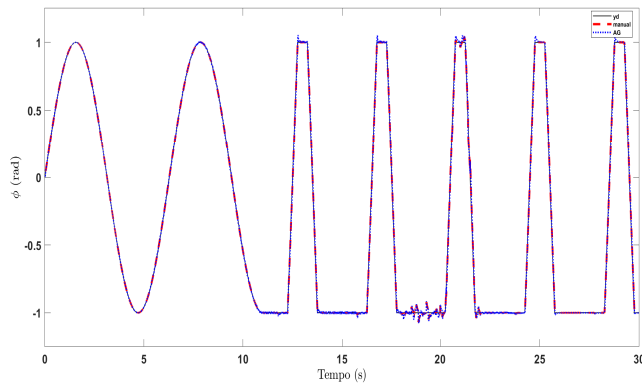


Figura 3. Rastreamento de trajetória do ângulo  $\phi$  ao longo do tempo;  $y_d(t)$  (preto),  $y(t)$  com parâmetros manuais (vermelho) e  $y(t)$  com parâmetros estimados pelo AG (azul).

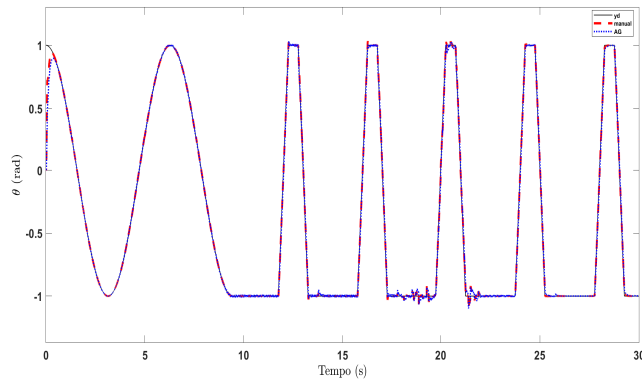


Figura 4. Rastreamento de trajetória do ângulo  $\theta$  ao longo do tempo;  $y_d(t)$  (preto),  $y(t)$  com parâmetros manuais (vermelho) e  $y(t)$  com parâmetros estimados pelo AG (azul).

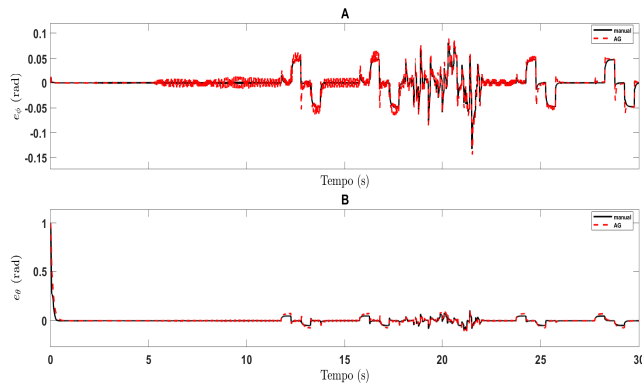


Figura 5. Erro de rastreamento de trajetória do ângulo  $\phi$  ao longo do tempo (A) e erro de rastreamento de trajetória do ângulo  $\theta$  ao longo do tempo (B); erro com parâmetros manuais (preto) e erro com parâmetros otimizados pelo AG (vermelho).

A Figura 7 mostra a evolução da aptidão do melhor indivíduo de cada geração e a da aptidão média da população ao longo das gerações. A partir, aproximadamente, da 35ª geração todos os indivíduos da população convergiram para o valor do melhor indivíduo encontrado. Próximo à 40ª geração, o algoritmo encontrou valores mais otimizados, demonstrando

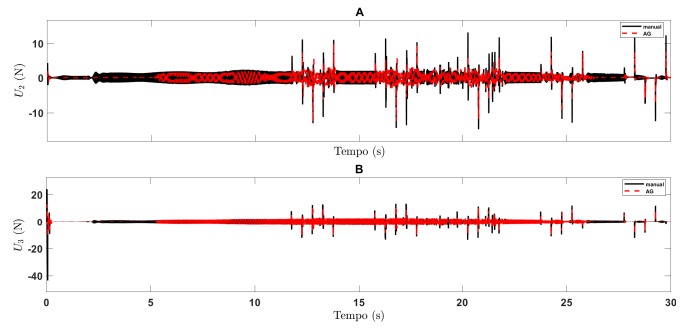


Figura 6. Sinais de controle  $U_2$  (A) e  $U_3$  (B); controlador com parâmetros manuais (preto) e com parâmetros otimizados pelo AG (vermelho).

que a convergência anterior era devido a um mínimo local. A probabilidade de mutação crescente, adotada nesse trabalho, visa evitar esses mínimos locais, já que, com o passar das gerações, o algoritmo entra em regiões mais otimizadas, cuja tendência é de diminuição da diversidade genética da população.

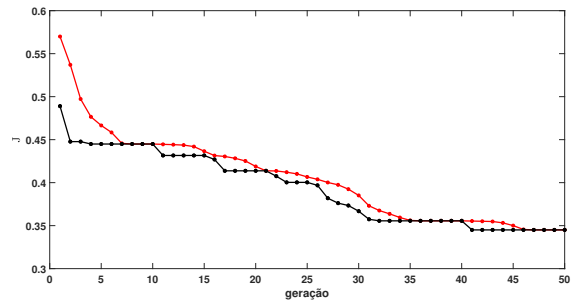


Figura 7. Evolução da aptidão do melhor indivíduo de cada geração (preto) e aptidão média da população (vermelha) ao longo das gerações.

## IX. CONCLUSÃO

O controlador projetado foi capaz de lidar com o problema de rastreamento de trajetória proposto, mesmo apresentando sensibilidade ao ruídos de medição adotado. Além disso, o controlador demonstrou robustez ao ruídos de processo.

Conforme mencionado, a utilização de controladores nebulosos implica no ajuste de vários parâmetros e, à medida que a complexidade do processo aumenta, torna-se difícil estabelecer a configuração ideal. Foi necessário um esforço considerável e massante para realizar a sintonia manual. O algoritmo genético desenvolvido mostrou-se eficaz para estimar e otimizar os parâmetros do controlador nebuloso, sendo uma alternativa viável para solucionar o desafio de ajuste dos parâmetros.

## REFERÊNCIAS

- [1] D. Q. M. Resende et al., *Desenvolvimento de um controlador híbrido fuzzy-PID e supervisor para estabilização de drone quadricóptero*, II Congresso Internacional de Gestão e Tecnologias, 2018.
- [2] L. P. Lisboa, *Controlador não-linear para Veículo Aéreo Não Tripulado*, Dissertação (Mestrado em Engenharia Elétrica), Pontifícia Universidade do Rio Grande do Sul, Porto Alegre, 2014.

- [3] M. Rabah et al., *Design of fuzzy-PID controller for quadcopter trajectory-tracking*, International Journal of Fuzzy Logic and Intelligent Systems, vol. 18, no. 3, pp. 204–213, 2018.
- [4] E. R. D. Morais, *Desenvolvimento de um controlador híbrido fuzzy-PID para estabilização de um VANT do tipo quadricóptero na realização de um voo autônomo vertical*, Dissertação (Mestrado em Sistemas de Comunicação e Automação), Universidade Federal Rural do Semi-Árido, Mossoró, 2017.
- [5] T. C. P. Benigno, *Modelagem matemática e controle de atitude e posição do quadricóptero*, Dissertação (Mestrado em Sistemas de Comunicação e Automação), Universidade Federal Rural do Semi-Árido, Mossoró, 2015.
- [6] F. Herrera et al., *Tuning fuzzy logic controllers by genetic algorithms*, International Journal of Approximate Reasoning, pp. 299–315, 1995.
- [7] A. Plerou, *Fuzzy genetic algorithms: fuzzy logic controllers and genetics algorithms*, Global Journal for research analysis, vol. 5, pp. 497-500, 2016.
- [8] F. M. U. Araujo et al., *Algoritmo genético para otimização de um controlador nebuloso para suspensão de vibrações*, VI Simpósio Brasileiro de Automação Inteligente, 2003.
- [9] J. S. de Morais, *Algoritmo diferencial evolutivo aplicado à identificação de sistemas dinâmicos não-lineares*, Tese (Doutorado em Engenharia Elétrica), Universidade Federal de Uberlândia, Uberlândia, 2013.
- [10] T. Bresciani, *Modelling, identification and control of a quadrotor helicopter*, Department of Automatic Control, Lund University, Lund, 2008.
- [11] J. C. D. Paula, *Desenvolvimento de um VANT do tipo quadricóptero para obtenção de imagens aéreas em alta definição*, Dissertação (Mestrado em Engenharia Elétrica), Universidade Federal do Paraná, Curitiba, 2012.
- [12] Y. Naidoo et al., *Quad-Rotor unmanned aerial vehicle helicopter modelling & control*, Int J Adv Robotic Sy, vol. 8, no. 4, pp. 139–149, 2011.
- [13] T. Luukkonen, *Modelling and control of quadcopter*, School of Science, 2011.
- [14] C. P. Duarte, *Modelagem e controle de um míssil de cruzeiro com propulsão foguete*, Trabalho de Conclusão de Curso (Engenharia Eletrônica), Instituto Tecnológico da Aeronáutica, São José dos Campos, 2003.
- [15] P. R. Evans, *Rotations and rotation matrices*, Acta Crystallographica, pp. 1355-1359, 2001.
- [16] S. Bouabdallah, *Design and control of quadrotors with application to autonomous flying*, École Polytechnique Fédérale de Lausanne, Lausanne, 2007.
- [17] M. G. Simões, I. S. Shaw, *Controle e modelagem fuzzy*, Blucher, São Paulo, 2007.
- [18] L. A. Zadeh, *Fuzzy sets*, Information and Control, pp. 338-353, 1965.
- [19] I. J. C. Nunez, *O uso de controladores fuzzy adaptativos, implementados em microcontroladores, no controle de vibrações de sistemas mecânicos*, Dissertação (Mestrado em Engenharia Mecânica), Universidade Federal de Uberlândia, Uberlândia, 2001.
- [20] S. Labiod, T. M. Guerra, *Direct adaptive fuzzy control for a class of MIMO nonlinear systems*, International Journal of Systems Science, vol. 38, no. 8, pp. 665–675, 2007.
- [21] C. N. Moreira, F. S. da Silva, *Automação da discretização de controladores PID e filtros digitais*, Revista Ciência e Tecnologia, vol. 17, n. 31, pp. 12-22, 2014.
- [22] F. C. Conceição, W. Beccaro, J. F. Justo, *Ensinando Métodos de Discretização para Projeto de Sistemas de Controle: Comparação de Controladores PID em Sistemas Embarcados*, Journal of Applied Instrumentation and Control, pp. 8-15, 2020.
- [23] L. J. C. Carvalho, V. C. S. Campos, *Controle nebuloso adaptativo direto de drone quadricóptero*, Simpósio Brasileiro de Automação Inteligente, 2021.
- [24] P. V. N. M. Vidal, *Rastreamento global e exato de sistemas incertos usando diferenciadores exatos não-lineares*, Projeto de graduação (Curso de Engenharia Eletrônica), Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2014.
- [25] A. Levant, *Robust exact differentiation via sliding Mode technique*, Automatica, vol. 34, no. 3, pp. 379–384, 1998.
- [26] E. V. L. Nunes et al., *Controle por realimentação de saídas para rastreamento exato de sistemas incertos*, Brazilian Conference on Automatic Control, 2004.
- [27] A. V. de Medeiros, *Modelagem de sistemas dinâmicos não lineares utilizando sistemas fuzzy, algoritmos genéticos e funções de base ortogonal*, Dissertação (Mestrado em Engenharia Elétrica), Universidade Estadual de Campinas, Campinas, 2006.