

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

ANA BEATRIZ RODRIGUES

**ALGORITMOS GENÉTICOS E DE EVOLUÇÃO DIFERENCIAL APLICADOS À
OTIMIZAÇÃO DE CONTROLADORES UTILIZANDO A NORMA H_2 COMO
CRITÉRIO DE DESEMPENHO**

DISSERTAÇÃO

CORNÉLIO PROCÓPIO
2019

ANA BEATRIZ RODRIGUES

**ALGORITMOS GENÉTICOS E DE EVOLUÇÃO DIFERENCIAL APLICADOS À
OTIMIZAÇÃO DE CONTROLADORES UTILIZANDO A NORMA H_2 COMO
CRITÉRIO DE DESEMPENHO**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica (PPGEE) da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para a obtenção do título de Mestre em Engenharia Elétrica.

Orientador: Prof. Dr. Cristiano Marcos Agulhari

CORNÉLIO PROCÓPIO
2019

Dados Internacionais de Catalogação na Publicação

R696 Rodrigues, Ana Beatriz

Algoritmos genéticos e de evolução diferencial aplicados à otimização de controladores utilizando a norma H_2 como critério de desempenho / Ana Beatriz Rodrigues. – 2019.

81 p. : il. color. ; 31 cm.

Orientador: Cristiano Marcos Agulhari.

Dissertação (Mestrado) – Universidade Tecnológica Federal do Paraná. Programa de Pós-Graduação em Engenharia Elétrica. Cornélio Procópio, 2019.

Bibliografia: p. 78-81.

1. Algoritmos genéticos. 2. Otimização matemática. 3. MATLAB (Programa de computador). 4. Engenharia Elétrica – Dissertações. I. Agulhari, Cristiano Marcos, orient. II. Universidade Tecnológica Federal do Paraná. Programa de Pós-Graduação em Engenharia Elétrica. III. Título.

CDD (22. ed.) 621.3

Biblioteca da UTFPR - Câmpus Cornélio Procópio

Bibliotecário/Documentalista responsável:
Romeu Righetti de Araujo – CRB-9/1676

TERMO DE APROVAÇÃO

Algoritmos genéticos e de evolução diferencial aplicados à otimização de controladores utilizando a norma H_2 como critério de desempenho

por

Ana Beatriz Rodrigues

Orientador: **Cristiano Marcos Agulhari**

Esta dissertação foi apresentada como requisito parcial à obtenção do grau de MESTRE EM ENGENHARIA ELÉTRICA – Área de Concentração: Sistemas Eletrônicos Industriais, pelo Programa de Pós-Graduação em Engenharia Elétrica – PPGEE – da Universidade Tecnológica Federal do Paraná – UTFPR – Câmpus Cornélio Procópio, às 09:00 horas do dia 03 de maio de 2019. O trabalho foi aprovado pela Banca Examinadora, composta pelos professores:

Prof. Dr. Cristiano Marcos Agulhari
Presidente

Prof. Dr. Marcelo Favoretto Castoldi
UTFPR – Cornélio Procópio

Prof. Dr. Rafael Fontes Souto
IFB – Brasília

Visto da coordenação:

Prof. Dr. Alessandro do Nascimento Vargas
Coordenador do Programa de Pós-Graduação em Engenharia Elétrica
UTFPR – Cornélio Procópio

“A Folha de Aprovação assinada encontra-se na Coordenação do Curso”

Dedico este trabalho aos meus pais, Aderson Negrão e Suely Zanoni.

AGRADECIMENTOS

Agradeço primeiramente à Deus por sempre guiar meus passos e me abençoar com a escolha dos melhores caminhos.

Aos meus pais Aderson e Suely que fizeram mais por mim do que por eles próprios durante estes anos de vida acadêmica, meus exemplos de força e honestidade. Por sempre me apoiarem nas minhas decisões e me orientarem.

Agradeço aos meus irmãos Afranio e Adeilton que foram base da minha formação e graças a eles sou quem sou.

Ao meu namorado Victor, por estar sempre ao meu lado nos momentos de estresse e tristeza me incentivando a ser mais forte e paciente. Também por cada risada e descontração nos momentos de alegrias e projetos para o futuro.

Agradeço as minhas tias Eni e Jacira que sempre agem como minhas mães, com todo amor e carinho.

A todos os familiares que de alguma forma me ajudaram na minha formação acadêmica e pessoal.

Agradeço imensamente ao meu orientador Prof. Dr. Cristiano Marcos Agulhari, primeiramente, por ter aceito me orientar e, principalmente, por toda dedicação e carinho com que me orientou em todos estes meses apesar das dificuldades.

Agradeço aos meus amigos, aos que passaram e aos que permaneceram, sejam eles de infância, de festa, de faculdade ou de todas as horas, cada um de alguma forma contribuiu para a realização deste sonho e serei eternamente grata. Agradeço também a todos os colegas de pesquisa do LACOS e também do LSI por sempre darem apoio e ajudarem onde fosse necessário.

A todos os servidores e professores da UTFPR Cornélio Procópio em especial aos servidores do CIPECA por toda dedicação e contribuição para meus estudos, mesmo que de forma indireta.

Enfim, a todos os que por algum motivo contribuíram para a realização deste sonho.

Agradeço especialmente à instituição UTFPR por ter me dado a oportunidade de realizar minha graduação e pós-graduação.

Não divido o mundo entre os fracos e os fortes, ou entre sucessos e fracassos [...] divido o mundo entre os que aprendem e os que não aprendem.

Benjamin Barber

A persistência é o caminho do êxito.

Charles Chaplin

RESUMO

RODRIGUES, Ana Beatriz. **Algoritmos genéticos e de evolução diferencial aplicados à otimização de controladores utilizando a norma H_2 como critério de desempenho**. 2019. 81 f. Dissertação de Mestrado – Programa de Pós-Graduação em Engenharia Elétrica. Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2019.

A síntese de controladores utilizando a norma H_2 como critério de desempenho é comumente solucionada a partir de condições baseadas em desigualdades matriciais lineares ou por meio da resolução de equações de Riccati. No entanto, tais abordagens podem trazer resultados conservadores caso outras restrições ou critérios de projeto não-convexos sejam incorporadas ao projeto. Neste sentido, este trabalho propõe analisar a eficiência de heurísticas pela aplicação de algoritmos evolutivos na síntese de controles utilizando a norma H_2 como critério de desempenho, a fim de simplificar suas soluções. Os controladores obtidos pelos algoritmos genéticos e de evolução diferencial são simulados computacionalmente pelo *software* MATLAB/*Simulink*®, e também testados experimentalmente. Necessita-se também avaliar as implementações físicas em malha fechada deste sistema e analisar se há a necessidade de acrescentar diferentes restrições para garantir a implementabilidade do controlador obtido. Para tanto, os sistemas de controle são aplicados a um sistema torcional disponível no laboratório LACOS da UTFPR Cornélio Procópio. É importante ressaltar que os algoritmos implementados podem ser facilmente adaptados para qualquer sistema, basta se conhecer o modelo matemático deste. Os algoritmos mostram-se eficientes ferramentas para a síntese de controladores otimizados.

Palavras-chave: Algoritmo genético. Algoritmo de evolução diferencial. Norma H_2 . MATLAB/*Simulink*. Sistema torcional.

ABSTRACT

RODRIGUES, Ana Beatriz. **Genetics and differential evolution algorithms applied to the optimization of controllers using the H_2 norm as a performance criterion.** 2019. 81 f. Master's Degree – Postgraduate Program in Electrical Engineering. Federal Technological University of Paraná. Cornélio Procópio, 2019.

The synthesis of controllers using the H_2 norm as performance criterion is usually solved from conditions based on linear matrix inequalities or by solving Riccati equations. However, such approaches or non-convex design criteria are incorporated into the project. In this sense, this work proposes to analyze the efficiency of heuristics by the application of evolutionary algorithms in the synthesis of controls using H_2 norm as a performance criterion, in order to simplify their solutions. The controllers obtained by genetic and differential evolution algorithms are simulated computationally by MATLAB / Simulink® software, and also tested experimentally. It is also necessary to evaluate the closed-loop physical implementations of this system and to analyze if there is a need to add different constraints to ensure the implementation of the obtained controller. To do so, the control systems are applied to a torsional system available in the LACOS laboratory of the UTFPR Cornélio Procópio. It is important to emphasize that the implemented algorithms can be easily adapted to any system, if it is known the mathematical model of this one. The algorithms are efficient tools for the synthesis of optimized controllers.

Keywords: Genetic Algorithm. Differential Evolution Algorithm. H_2 Norm. MATLAB/Simulink. Torsional system.

LISTA DE FIGURAS

Figura 1 – Planta generalizada	21
Figura 2 – Relação das entradas e saídas	21
Figura 3 – Cruzamento AG	26
Figura 4 – Mutação AG	27
Figura 5 – Representação dos processos AG	27
Figura 6 – Mutação ED	29
Figura 7 – Cruzamento ED	30
Figura 8 – Representação dos processos ED	30
Figura 9 – Planta torcional ECP®	32
Figura 10 – Diagrama de blocos do <i>Simulink</i>	33
Figura 11 – Fluxograma dos métodos aplicados	36
Figura 12 – Controle em malha fechada	37
Figura 13 – População dos algoritmos	40
Figura 14 – Fluxograma do AG	43
Figura 15 – Fluxograma do ED	45
Figura 16 – Representação do sistema em malha fechada	46
Figura 17 – Sistema torcional aplicado	47
Figura 18 – Zeros e polos operação 1 – AG	56
Figura 19 – Zeros e polos operação 2 – AG	56
Figura 20 – Zeros e polos operação 3 – AG	57
Figura 21 – Zeros e polos operação 1 – AG com restrição	59
Figura 22 – Zeros e polos operação 2 – AG com restrição	59
Figura 23 – Zeros e polos operação 3 – AG com restrição	60
Figura 24 – Resposta ao degrau e energia de controle – operação 1 AG	61
Figura 25 – Resposta senoidal e energia de controle – operação 1 AG	61
Figura 26 – Resposta ao degrau e energia de controle – operação 2 AG	62
Figura 27 – Resposta senoidal e energia de controle – operação 2 AG	62
Figura 28 – Resposta ao degrau e energia de controle – operação 3 AG	63
Figura 29 – Resposta senoidal e energia de controle – operação 3 AG	63
Figura 30 – Zeros e polos operação 1 – ED	65
Figura 31 – Zeros e polos operação 2 – ED	66
Figura 32 – Zeros e polos operação 3 – ED	66

Figura 33 – Zeros e polos operação 1 – ED com restrição	68
Figura 34 – Zeros e polos operação 2 – ED com restrição	68
Figura 35 – Zeros e polos operação 3 – ED com restrição	69
Figura 36 – Resposta ao degrau e energia de controle – operação 1 ED	70
Figura 37 – Resposta senoidal e energia de controle – operação 1 ED	70
Figura 38 – Resposta ao degrau e energia de controle – operação 2 ED	71
Figura 39 – Resposta senoidal e energia de controle – operação 2 ED	71
Figura 40 – Resposta ao degrau e energia de controle – operação 3 ED	72
Figura 41 – Resposta senoidal e energia de controle – operação 3 ED	72
Figura 42 – Decaimento do valor FO a cada iteração do AG – operação 2	74
Figura 43 – Decaimento do valor FO a cada iteração do ED – operação 2	74

LISTA DE TABELAS

Tabela 1 – Ponderações relacionadas à saída artificial	39
Tabela 2 – Variação dos parâmetros AG	42
Tabela 3 – Variação dos parâmetros ED	44
Tabela 4 – Número de indivíduos AG	48
Tabela 5 – Porcentagem de cruzamento AG	49
Tabela 6 – Taxa de cruzamento AG	49
Tabela 7 – Porcentagem de mutação AG	50
Tabela 8 – Taxa de mutação AG	50
Tabela 9 – Parâmetros AG	51
Tabela 10 – Número de indivíduos ED	51
Tabela 11 – Taxa de mutação ED	52
Tabela 12 – Taxa de cruzamento ED	52
Tabela 13 – Parâmetros ED	53
Tabela 14 – Função objetivo AG	54
Tabela 15 – Função objetivo AG com restrição	58
Tabela 16 – <i>Fitness</i> ED	64
Tabela 17 – <i>Fitness</i> ED com restrição	67
Tabela 18 – Comparações entre AG e ED	73

LISTA DE SIGLAS E ACRÔNIMOS

A/D	Analógico/Digital
AG	Algoritmo Genético
CIPECA	Centro Integrado de Pesquisa em Controle e Automação
D/A	Digital/Analógico
DSP	<i>Digital Signal Processor</i>
FO	Função Objetivo
ED	Evolução Diferencial
LACOS	Laboratório de Controle e Otimização de Sistemas
LIT	Linear e Invariante no Tempo
LMI	<i>Linear Matrix Inequalities</i>
PSO	<i>Particle Swarm Optimization</i>
SISO	<i>Single Input Single Output</i>
UTFPR	Universidade Tecnológica Federal do Paraná

SUMÁRIO

1	INTRODUÇÃO	14
1.1	MOTIVAÇÃO	16
1.2	OBJETIVOS	16
1.2.1	Objetivo geral	16
1.2.2	Objetivos específicos	17
1.3	JUSTIFICATIVA	17
1.4	ORGANIZAÇÃO DO TRABALHO	17
2	MODELAGEM MATEMÁTICA EM ESPAÇO DE ESTADOS	19
2.1	NORMA H_2	20
2.2	CONCLUSÃO DO CAPÍTULO	23
3	ALGORITMOS EVOLUTIVOS	24
3.1	ALGORITMO GENÉTICO	24
3.2	ALGORITMO DE EVOLUÇÃO DIFERENCIAL	28
3.3	CONCLUSÃO DO CAPÍTULO	31
4	IDENTIFICAÇÃO DO SISTEMA	32
5	MÉTODOS	36
5.1	SISTEMA EM MALHA FECHADA	37
5.2	MINIMIZAÇÃO DA NORMA H_2	38
5.3	PARÂMETROS DOS ALGORITMOS	40
5.5	PROCEDIMENTOS SIMULADOS	45
5.6	PROCEDIMENTOS EXPERIMENTAIS	46
5.6	CONCLUSÃO DO CAPÍTULO	47
6	PARÂMETROS	48
6.1	PARÂMETROS DO ALGORITMO GENÉTICO	48
6.2	PARÂMETROS DA EVOLUÇÃO DIFERENCIAL	51
6.3	CONCLUSÃO DO CAPÍTULO	53
7	SINTESE DE CONTROLADORES	54
7.1	CONTROLADORES OBTIDOS PELO ALGORITMO GENÉTICO	54
7.2	CONTROLADORES OBTIDOS PELO ALGORITMO DE EVOLUÇÃO DIFERENCIAL	64
7.3	COMPARAÇÕES ENTRE AG E ED	72

7.3	CONCLUSÃO DO CAPÍTULO	75
8	CONSIDERAÇÕES FINAIS	76
8.1	TRABALHOS FUTUROS	77
8.2	PRODUÇÃO CIENTÍFICA.....	77
	REFERÊNCIAS.....	78

1 INTRODUÇÃO

Em processos industriais é comum o uso de plantas e sistemas de controles projetados a fim de atingir mais de uma especificação de trabalho, como por exemplo, um sistema que necessite de robustez a ruídos otimizando certos critérios de desempenho. Uma das formas de atingir tais objetivos é modelar o desempenho pela minimização de sua norma H_2 , enquanto que a robustez necessária pode ser garantida por meio de sua norma H_∞ (CHUN; CHOI; KIM, 2017).

A abordagem mais utilizada para a síntese de controladores mistos H_2 / H_∞ é baseada no conceito de estabilidade quadrática, ou também chamada de otimização por desigualdades matriciais lineares (*Linear Matrix Inequalities (LMI)*) (BOYD et al., 1994). O maior problema desta abordagem é que as condições menos conservadoras dependem de vários parâmetros que devem ser pré-determinados, o que prejudica a síntese do controlador ou mesmo pode apresentar uma difícil factibilidade em sua resolução. Outro problema apresentado a este tipo de resolução é que certas restrições de projeto podem ser não-convexas, o que aumenta veemente a complexidade desta abordagem (WU et al., 2013).

O algoritmo apresentado por Gonçalves, Palhares e Takahashi (2004) possibilita uma eficiente síntese do controle misto robusto a partir de realimentação de estados, por meio da resolução de um conjunto de LMIs. Outros trabalhos desenvolveram pesquisas sobre a resolução de LMIs para a obtenção da síntese de controladores mistos como apresentado em Boyd et al. (1994), Scherer (1996), Ghaoui e Niculescu (2000), Leibfritz (2001), Yu (2004), Sato (2010), Emam e Fraharian (2016) e também em Andrade e Agulhari (2016). Nota-se que há diversas pesquisas a respeito, mas a grande maioria é centrada na resolução de LMIs, o que torna interessante pesquisar e analisar outros tipos de abordagens.

Diante do sucesso e eficiência das meta-heurísticas em pesquisas científicas, estas também passaram a ser utilizadas para a otimização de controles mistos (CHUN; CHOI; KIM, 2017) (PEREIRA; ARAUJO, 2004) (SALIM; BETTAYEB, 2009) (ALI; AL-THUWAINY, 2010). As meta-heurísticas mais utilizadas nas últimas décadas são baseadas no comportamento e na evolução das espécies, denominadas algoritmos evolutivos. Estes algoritmos vêm sendo utilizados em diversos campos de

pesquisa e passaram inclusive a serem utilizados em otimização de sistemas complexos de engenharia (YANG et al., 2013).

Neste contexto, Chun, Choi e Kim (2017) propõem a otimização de uma estrutura controlada a partir da Otimização por Enxame de Partículas (*Particle Swarm Optimization (PSO)*), a qual apresenta uma função objetivo que mescla ambas as normas H_2 e H_∞ , diminuindo assim, o esforço computacional do algoritmo quando comparado às duas funções objetivos separadas. Há também algoritmos híbridos aplicados à síntese de controladores, como o desenvolvido por Pereira e Araújo (2004), o qual utiliza rotinas baseadas em LMI para a minimização do desempenho e um algoritmo genético para obter a população de controladores. Salim e Bettayeb (2009) e Ali e Al-Thuwainy (2010) também utilizam o algoritmo genético para obter a otimização de controles mistos. Outras propostas são apresentadas abordando algoritmos evolutivos para a otimização da síntese de controladores mistos, como o algoritmo inspirado no sistema imunológico de acordo com Guimarães et al. (2007) e também a partir do algoritmo de evolução diferencial proposto por Wu et al. (2013).

Por outro lado, há diversas aplicações industriais onde o sistema necessita de alto desempenho e sua robustez a perturbações já é tratada por outros agentes, assim, a síntese do controlador não precisa considerar este critério. Para estas aplicações, é possível analisar e otimizar apenas o critério de desempenho do sistema de controle associado a norma H_2 , tornando sua resolução mais simples e eficiente. A solução mais comum para a otimização da norma H_2 associado ao desempenho de sistemas de controle é a partir das equações algébricas de Riccati. Em Boers, Weiland e Damen (1997) é apresentado um estudo comparativo entre a solução de Riccati e uma distribuição probabilística proposta pelos autores.

Outros trabalhos também utilizam a solução de Riccati a fim de minimizar a norma H_2 de sistemas como a proposta apresentada por Dragan; Morozan e Stoica (2004) e em Kucera (2006), onde é comparada a solução de Riccati obtida pelo modelo em espaço de estados com a otimização de Wiener-Hopf obtida a partir da função de transferência do mesmo sistema.

As equações de Riccati podem também ser modeladas em termos de LMIs, que, por sua vez, é uma das abordagens mais utilizadas para resolver o problema de minimização da norma H_2 . Alguns trabalhos propuseram tal resolução para a

otimização de sistemas de controle como os apresentados em Farias et al. (2000), Xie e Chen (2010) e Gomez et al. (2015).

Conforme apresentado pela revisão bibliográfica, o uso de meta-heurísticas aplicadas para a otimização de desempenho associado à norma H_2 e para controles mistos H_2 e H_∞ ainda é pouco estudado, assim, há muito espaço para novos estudos e pesquisas utilizando esta abordagem. Além disso, tais trabalhos são raramente implementados fisicamente a fim de comprovar seus resultados.

1.1 MOTIVAÇÃO

Com a revisão literária apresentada em relação à otimização mista (robustez e desempenho) e à otimização de desempenho de sistemas de controle, notou-se que a grande maioria das técnicas de resolução apresentadas são algébricas e não há numerosas pesquisas utilizando técnicas meta-heurísticas, principalmente relacionadas à otimização apenas de desempenho.

Neste contexto, mostrou-se viável a análise apresentada neste trabalho, a qual propõe otimizar o desempenho H_2 de um sistema utilizando duas técnicas meta-heurísticas evolutivas que, de acordo com a pesquisa bibliográfica realizada, não foram ainda aplicadas para este fim. As meta-heurísticas a serem analisadas são o algoritmo genético (AG) e o de evolução diferencial (ED).

1.2 OBJETIVOS

Para a elaboração deste trabalho alguns objetivos são definidos, conforme segue.

1.2.1 Objetivo geral

Implementar uma técnica de síntese de controladores utilizando os algoritmos genéticos e de evolução diferencial, aplicados à otimização de controladores utilizando a norma H_2 como critério de desempenho e comparar tanto as soluções obtidas quanto a eficácia de ambos os algoritmos.

1.2.2 Objetivos específicos

Os objetivos específicos utilizados neste trabalho são:

- Conceituar de forma teórica o modelo em espaço de estados, a norma H_2 como critério de desempenho, o algoritmo genético e, por fim, o algoritmo de evolução diferencial;
- Apresentar a identificação do sistema de estudo;
- Implementar os algoritmos genéticos e de evolução diferencial;
- Realizar a síntese dos controladores a partir de ambos os algoritmos;
- Validar computacionalmente e experimentalmente os controladores obtidos pelo AG e pelo algoritmo ED;
- Analisar os resultados simulados e experimentais apresentados às técnicas de controle estudadas.

1.3 JUSTIFICATIVA

Algoritmos evolutivos vêm sendo utilizados em diversas pesquisas complexas de otimização e em diversas áreas, contudo não foi constatado em literatura sua aplicação em otimização da norma H_2 de sistemas de controle.

Assim, a maior contribuição desta pesquisa é a investigação da aplicação dos algoritmos genéticos e de evolução diferencial na otimização do desempenho de sistemas. Há contribuição também em relação à análise de confiabilidade dos algoritmos no que diz respeito à variância e estabilidade dos resultados apresentados, tornando possível avaliar qual dos algoritmos é o mais apropriado para a aplicação descrita.

1.4 ORGANIZAÇÃO DO TRABALHO

No Capítulo 2 é apresentada a modelagem de sistemas em espaço de estados e a norma H_2 de sistemas e controle. No Capítulo 3 são definidos os algoritmos evolutivos a serem estudados, sendo estes, o algoritmo genético e o de evolução diferencial. No capítulo 4 é determinado o modelo em espaço de estados do sistema torcional ECP, utilizado como objeto de estudo. No Capítulo 5 são

apresentadas as metodologias de desenvolvimento computacional e experimental para a obtenção de resultados. No Capítulo 6 são apresentados os testes para definir os parâmetros dos algoritmos genético e de evolução diferencial. No Capítulo 7 são apresentados os resultados simulados e experimentais obtidos pelos controladores em malha fechada. No Capítulo 8 são discutidas as considerações a respeito dos resultados obtidos relacionados ao algoritmo genético e de evolução diferencial.

2 MODELAGEM MATEMÁTICA EM ESPAÇO DE ESTADOS

O modelo matemático consiste em um conjunto de equações que são utilizadas para representar a dinâmica de um sistema. Dependendo das propriedades do sistema, um mesmo modelo pode ser descrito utilizando diferentes representações, que é determinada de acordo com suas características e aplicações. Por exemplo, a representação em variáveis de estado é comumente utilizada caso se deseje analisar ou otimizar o sistema em termos de sua energia ou de outros aspectos temporais, já a função de transferência é mais utilizada se certas características frequenciais são desejadas como filtragem, etc (OGATA; MAYA, LEONARDI, 2010).

Espaço de estados é denominado como o espaço n -dimensional cujas coordenadas são formadas pelas variáveis de estados, as quais representam o menor número de variáveis capaz de representar completamente o estado atual de um sistema. O número das variáveis de estado de um sistema não se altera, independentemente do modelo representado (OGATA; MAYA, LEONARDI, 2010). Um modelo característico utilizado na representação por espaço de estados é apresentado por (1) e (2).

$$\dot{x}(t) = f(x(t), u(t)), \quad (1)$$

$$y(t) = g(x(t), u(t)), \quad (2)$$

sendo:

$x(t)$ as variáveis de estado no tempo de dimensão n_x ;

$u(t)$ as variáveis de entrada no tempo de dimensão n_u ;

$y(t)$ as variáveis de saída no tempo de dimensão n_y ;

$f(x(t), u(t))$ e $g(x(t), u(t))$ as funções que representam, respectivamente, a dinâmica interna e a saída do modelo.

As Equações (1) e (2) são representações genéricas de sistemas dinâmicos, porém, como neste trabalho serão estudados sistemas lineares e invariantes no tempo (LIT), os sistemas serão portanto representados como descrito nas Equações (3) e (4).

$$\dot{x}(t) = Ax(t) + Bu(t), \quad (3)$$

$$y(t) = Cx(t) + Du(t), \quad (4)$$

sendo:

A a matriz de estados;

B a matriz de entrada;

C a matriz de saída;

D a matriz de transmissão direta.

2.1 NORMA H_2

Projetos de sistemas de controle são realizados a fim de que o sistema estudado apresente determinadas especificações de funcionamento e desempenho. Assim, uma das maneiras mais utilizadas para classificar o desempenho de um sistema é uma medida de energia para determinados sinais de interesse (TROFINO; COUTINHO; BARBOSA, 2005).

A norma H_2 , definida formalmente na sequência, é uma medida da energia média do sistema. Para que a norma H_2 seja utilizada como um índice de desempenho ou custo funcional do sistema em malha fechada, é introduzida uma saída artificial a ser controlada neste sistema. A resolução deste sistema em malha fechada é equivalente à substituição de suas condições iniciais por uma perturbação persistente não-correlacionada entrando em todas as direções do espaço de estados (TROFINO; COUTINHO; BARBOSA, 2005). O sistema descrito por (5), também denominado de planta generalizada, é ilustrado pela Figura 1.

$$G(t) = \left\{ \begin{array}{l} \dot{x}(t) = Ax(t) + Bu(t) + B_w w(t) \\ y(t) = Cx(t) \\ z(t) = C_z x(t) + D_z u(t) + D_{z_w} w(t) \end{array} \right\}, \quad (5)$$

as variáveis na Equação (5) são:

$z(t)$ a saída artificial (controlada) do sistema;

$w(t)$ a entrada exógena;

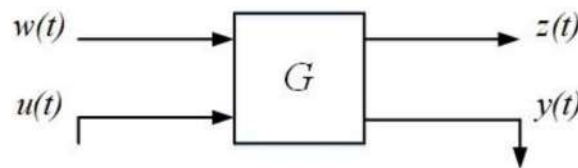
B_w a matriz relacionada ao valor resultante da norma H_2 ;

C_z a matriz relacionada com a velocidade de resposta dos estados do sistema;

D_z a matriz relacionada com a energia de controle aplicada à entrada do sistema;

D_{zw} a matriz de transmissão direta entre a entrada exógena e a saída controlada.

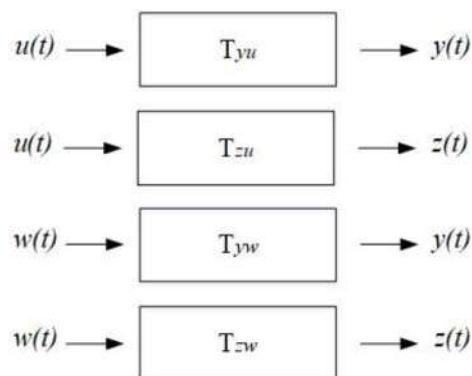
Figura 1 – Planta generalizada



Fonte: Autoria própria

Assim, o desempenho do sistema pode ser descrito em termos da relação direta entre a entrada exógena e a saída artificial controlada, a qual pode estar relacionada à robustez do sistema quando há ruídos ou, conforme abordado neste trabalho, relacionada à medida de energia. Em termos de função de transferência, este sistema pode ser caracterizado por quatro funções independentes, conforme Figura 2, sendo:

Figura 2 – Relação das entradas e saídas



Fonte: Autoria própria

T_{yu} a função de transferência que relaciona a entrada $u(t)$ à saída $y(t)$;

T_{zu} a função de transferência que relaciona a entrada $u(t)$ à saída $z(t)$;

T_{yw} a função de transferência que relaciona a entrada $w(t)$ à saída $y(t)$;

T_{zw} a função de transferência que relaciona a entrada $w(t)$ à saída $z(t)$.

Como se deseja minimizar a energia entre a entrada exógena $w(t)$ e a saída controlada $z(t)$, definidas pelo usuário a fim de otimizar os aspectos desejados do sistema, a norma H_2 do sistema é analisada pela função de transferência T_{zw} , conforme Definição 1.

Definição 1 (*Norma H_2*) A norma H_2 de um sistema $G(s)$ é definida por (6).

$$\|G\|_2^2 = \sqrt{\frac{1}{2\pi} \int_{-\infty}^{\infty} \text{traço}(G(j\omega)^T G(j\omega)) d\omega}, \quad (6)$$

sendo $G(j\omega)$ a resposta em frequência da função de transferência $G(s)$.

Propriedade (*Norma H_2 finita*) A norma H_2 de um sistema é finita se, e somente se, o sistema G for estável e estritamente próprio, ou seja, se a matriz de estados A for Hurwitz e se $D_{zw} = 0$.

A norma H_2 do sistema (5) também pode ser definida conforme (7) (TROFINO; COUTINHO; BARBOSA, 2005).

$$\|G\|_2^2 = \int_{-\infty}^{\infty} z(t)^T z(t) dt \quad (7)$$

Substituindo $z(t)$ em (7) e considerando que o sistema seja Hurwitz, a Equação apresentada em (8) é similar à integral do desempenho obtida pelo Regulador Quadrático Linear (*Linear Quadratic Regulator* (LQR)).

$$\|G\|_2^2 = \int_{-\infty}^{\infty} \{x^T(t) C_z^T C_z x(t) + u^T(t) D_z^T D_z u(t)\} dt \quad (8)$$

Assim, minimizar a norma H_2 é equivalente a minimizar uma ponderação entre a energia dos estados e a energia do sinal de controle, sendo tais ponderações relacionadas, respectivamente, às matrizes C_z e D_z (ZHOU; DOYLE; GLOVER, 1996). As matrizes C_z e D_z , e conseqüentemente a variável artificial $z(t)$, são definidas pelo usuário para determinar a ponderação de minimização dos aspectos mencionados.

Além das matrizes do sistema aumentado C_z e D_z , necessita-se determinar uma lei de controle $u(t)$, cuja estrutura utilizada neste trabalho é detalhada no Capítulo 5, para que a norma H_2 do sistema controlado seja minimizada. Para atingir este objetivo, os parâmetros do controlador que compõem $u(t)$ são determinados pelos algoritmos heurísticos apresentados no Capítulo 3.

2.2 CONCLUSÃO DO CAPÍTULO

Neste capítulo foram apresentados os conceitos teóricos sobre o modelo de sistemas em espaços de estados, necessário para a determinação do modelo da planta torcional e sobre a norma H_2 , utilizada como critério de desempenho do sistema para a implementação do algoritmo genético e de evolução diferencial. O Capítulo 3 apresenta o funcionamento de ambos os algoritmos estudados.

3 ALGORITMOS EVOLUTIVOS

Os algoritmos evolutivos são inspirados na teoria de evolução natural de Darwin, que afirma que, com o passar dos milhares de anos, espécies precisavam se adaptar para se enquadrarem ao meio ambiente que habitavam. A comparação entre os processos de otimização e a evolução biológica possibilitou o desenvolvimento de uma importante área da inteligência computacional: as técnicas computacionais evolutivas (DAS; SUGANTHAN, 2011).

Algoritmos evolutivos utilizam progressão iterativa, ou seja, os indivíduos da população são desenvolvidos para atingirem uma adequação desejada, onde tal população é selecionada aleatoriamente dentro de um campo de busca (DAS; SUGANTHAN, 2011). O principal inconveniente deste algoritmo é a necessidade de correta adequação de seus parâmetros, que, caso não estejam corretamente definidos, podem demandar um alto custo computacional (QIN; HUANG; SUGANTHAN, 2009).

O interesse por algoritmos de computação evolucionária vem aumentando nas últimas décadas para resolução de problemas de otimização, pois fornecem uma abordagem mais robusta e eficiente para resolução de problemas reais complexos (KAO; ZAHARA, 2007).

Apesar do aumento de interesse e aplicação destes algoritmos, os algoritmos evolucionários são estocásticos. Assim, devido à sua dependência randômica, não garantem um resultado ótimo. Contudo, tais algoritmos tendem a encontrar a solução ótima ou ficar bem próximos destas (LINDEN, 2008).

Neste capítulo, serão apresentados dois algoritmos baseados na computação evolutiva, os quais posteriormente terão seus resultados comparados para avaliar qual algoritmo apresenta melhor desempenho, sendo eles: o algoritmo genético e o algoritmo de evolução diferencial.

3.1 ALGORITMO GENÉTICO

O algoritmo genético foi desenvolvido na década de 1960 por John Holland, porém teve sua primeira publicação apenas 1975. Baseado nos fundamentos dos algoritmos evolutivos, o AG simula a evolução da população de soluções, a fim de solucionar problemas de otimização (ROBERGE; TARBOUCHI; LABONTÉ, 2013).

O algoritmo é inicializado com uma população de soluções candidatas obtidas aleatoriamente a partir de um espaço de busca. Tal população é representada por uma matriz $R^{\{NP \times D\}}$, ou seja, de dimensões iguais a NP cromossomos/indivíduos, que são as linhas desta matriz, por D genes, suas colunas. Após sua inicialização, operadores genéticos são aplicados, a fim de reproduzir seus descendentes com melhor grau de adequação em relação à função objetivo (PEZZELLA; MORGANTI; CIASCHETTI, 2008).

Após a criação da população inicial de soluções, todos os indivíduos são avaliados em relação ao seu grau de aptidão de acordo com a função objetivo e, em sequência, os operadores genéticos são iniciados. O primeiro processo realizado é o da seleção, onde duas técnicas são utilizadas, a seleção por elitismo e a seleção por torneio (OUYANG; PANO, 2015).

Ao fim de uma geração, há a possibilidade de se perder os indivíduos mais aptos da população inicial, assim, para evitar que isso ocorra, o melhor indivíduo da população inicial (mais apto) é diretamente selecionado para a nova população de soluções. Este processo é denominado seleção por elitismo e pode melhorar consideravelmente o desempenho do algoritmo (LACERDA; CARVALHO, 1999).

O processo de seleção é caracterizado pela escolha dos indivíduos mais aptos a fim de que “sobrevivam” para próximas gerações. Assim, no segundo método de seleção utilizado, o torneio realiza a escolha aleatória de grupos de n indivíduos, onde o indivíduo mais apto de cada grupo é selecionado e enviado para a nova população. Este torneio é realizado até que a nova população possua dimensão igual à população inicial. Geralmente são gerados grupos de três indivíduos e esta técnica de seleção é conhecida como torneio de três (LACERDA; CARVALHO, 1999).

Outras técnicas de seleção também podem ser utilizadas, como a seleção por roleta, onde os indivíduos são ordenados de acordo com a função objetivo e são atribuídas probabilidades decrescentes de serem escolhidos, a seleção por classificação, onde a seleção é relacionada a posição na ordenação dos indivíduos ou a seleção por truncamento, onde os N melhores indivíduos são selecionados e os demais são descartados (LACERDA; CARVALHO, 1999).

O próximo processo a ser realizado é o cruzamento. Para que parte das soluções sejam preservadas, define-se uma porcentagem de indivíduos da população inicial que serão cruzados, onde dois destes indivíduos se recombina de forma a

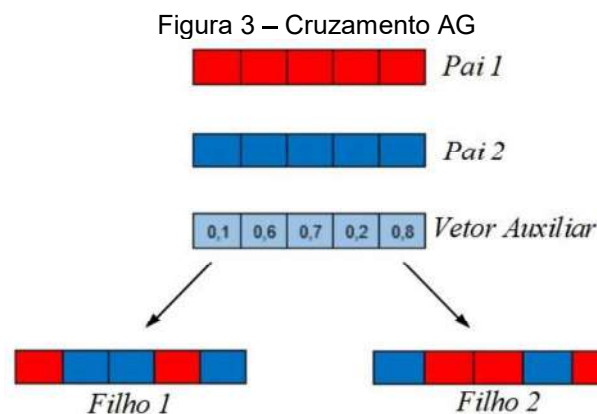
obter dois filhos que apresentem material genético (genes) de ambos pais (LINDEN, 2008).

A técnica de cruzamento pode ser realizada de diversas formas, como por exemplo, a técnica de corte, onde dois cromossomos pais transmitem seus genes para dois cromossomos filhos. Esta técnica pode ser considerada “fraca”, pois comumente apresenta apenas um ou dois pontos de cortes para que haja o cruzamento dos indivíduos pais.

Por isso, torna-se vantajoso aplicar um cruzamento uniforme, pois, desta forma, o número de genes cruzados pode ser maior do que a técnica apresentada anteriormente, e este fato pode melhorar a busca do algoritmo pelo *fitness* ótimo.

Para realizar o cruzamento uniforme, um vetor auxiliar de dimensão igual ao número de genes da população inicial é gerado, onde cada gene deste vetor auxiliar é comparado com a taxa de cruzamento definida pelo usuário. Quando o valor do gene do vetor auxiliar for menor do que a taxa de cruzamento, seleciona-se o gene do *Pai 1* para o *Filho 1* e o gene correspondente do *Pai 2* para o *Filho 2*, caso seja maior, o processo inverso é realizado. Desta forma, os filhos resultantes do processo de cruzamento são sempre um o complemento do outro (PRICE; STORN; LAMPINEN, 2006).

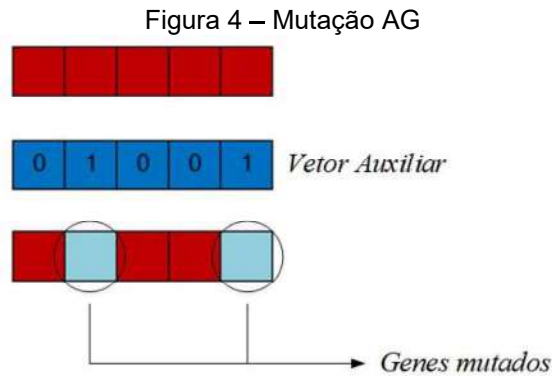
A Figura 3 apresenta um exemplo de cruzamento uniforme supondo uma taxa de cruzamento igual a 0,5.



Fonte: Autoria própria

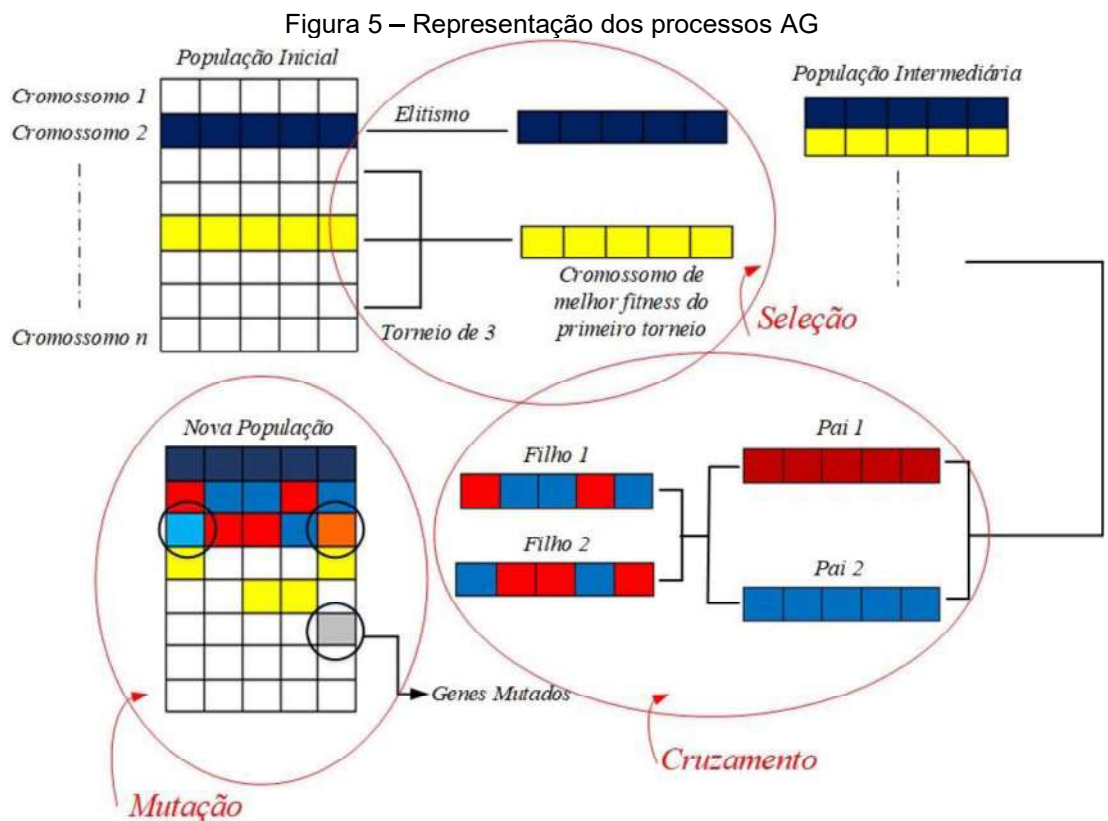
Por fim, na mutação há uma taxa que determina o número de indivíduos aleatórios que passarão por este processo. Assim, é gerado um vetor auxiliar de dimensão D genes com valores iguais a 0 ou 1, este cromossomo determinará quais genes dos indivíduos sofrerão a mutação. Com isso, quando o gene do vetor auxiliar

for igual a 1, há variação do gene correspondente do indivíduo a ser mutado, caso o gene do vetor auxiliar seja 0, não há mutação no gene correspondente. A Figura 4 ilustra este processo (LINDEN, 2008).



Fonte: Adaptado de Linden (2008)

Após a realização dos operadores, avalia-se o grau de aptidão de todos os indivíduos e determina-se o cromossomo mais apto de acordo com a função objetivo. Caso a condição de parada do algoritmo seja satisfeita, o algoritmo é finalizado, caso não seja, uma nova geração é iniciada e a população inicial é substituída pela que percorreu os processos anteriormente. A Figura 5 ilustra todos os processos do algoritmo genético descrito.



Fonte: Autoria própria

3.2 ALGORITMO DE EVOLUÇÃO DIFERENCIAL

O algoritmo de evolução diferencial, desenvolvido em 1995 por Rainer Storn e Kenneth Price, vem sendo utilizado em diversos campos de pesquisas científicas e de engenharia para a resolução de problemas avançados de otimização em espaços contínuos. O aumento do número e da variedade de aplicações deste algoritmo é consequência de sua eficiente busca populacional estocástica (QIN; HUANG; SUGANTHAN, 2009).

A população de possíveis soluções inicializada aleatoriamente sofre manipulações a cada iteração, também chamada de geração. Com o desenvolvimento das gerações, o grau de aptidão de seus indivíduos em relação à função objetivo é melhorado (maximizado ou minimizado), e este grau de adequação é denominado o *fitness* do algoritmo (BOSCO et al., 2017).

As manipulações que a população sofre em cada geração são: mutação, cruzamento e seleção, a serem explicados na sequência. Os processos ocorrem por um total de G gerações até que o critério de parada do algoritmo seja satisfeito, o qual pode ser a limitação do número de gerações ou o erro estabelecido entre a diferença do *fitness* de $G - 1$ e G .

Uma vez inicializado, o algoritmo de evolução diferencial passa pelo processo de mutação, o qual seleciona aleatoriamente três indivíduos da população inicial, e então determina a diferença ponderada entre dois indivíduos e soma o resultado ao terceiro, conforme Equação (9) (PRICE; STORN; LAMPINEN, 2006).

$$v_{i,G+1} = x_{r3,G} + F(x_{r2,G} - x_{r1,G}), \quad (9)$$

sendo:

$v_{i,G+1}$ o indivíduo i da população mutada;

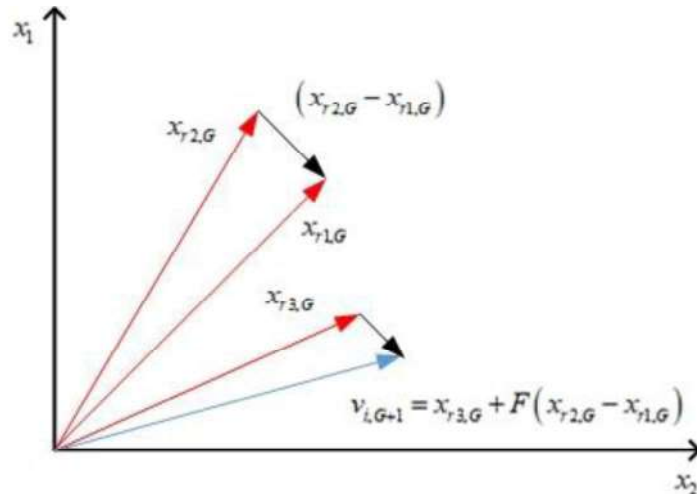
F o fator de ponderação ou taxa de mutação;

$x_{r1,G}$, $x_{r2,G}$ e $x_{r3,G}$ os cromossomos da população antes da mutação, obtidos aleatoriamente.

O fator de ponderação F é especificado como um valor entre 0 e 1, o qual determina o tamanho do passo a ser dado na direção denotada pelo vetor diferença. Assim, quanto maior for este valor, maior será a perturbação apresentada. Este

processo evita que as soluções fiquem presas em regiões locais. A Figura 6 ilustra o processo de mutação.

Figura 6 – Mutação ED



Fonte: Adaptado de Price, Stone e Lampinen (2006)

Após todos os indivíduos sofrerem mutação, inicia-se o processo de cruzamento, onde cada indivíduo será alvo e servirá de comparação, denominado de *target vector*. Neste processo, cada *target vector* é cruzado com seu respectivo cromossomo mutado, a fim de possibilitar maior aptidão dos novos indivíduos. Assim, cria-se um vetor auxiliar com valores aleatórios entre 0 e 1 e dimensão igual ao número de genes do indivíduo.

O cromossomo cruzado depende diretamente do valor de cada gene deste vetor auxiliar e da taxa de cruzamento definida. Quando o valor do gene do vetor auxiliar for menor do que a taxa de cruzamento, seleciona-se o gene do cromossomo mutado, caso seja maior, seleciona-se o gene do *target vector*, conforme Equação (10).

$$u_{i,G+1} = \begin{cases} v_{i,G+1} & \rightarrow \text{Se } r_n \leq CR \\ x_{i,G} & \rightarrow \text{Se } r_n > CR \end{cases}, \quad (10)$$

sendo:

$u_{i,G+1}$ o indivíduo i da população cruzada;

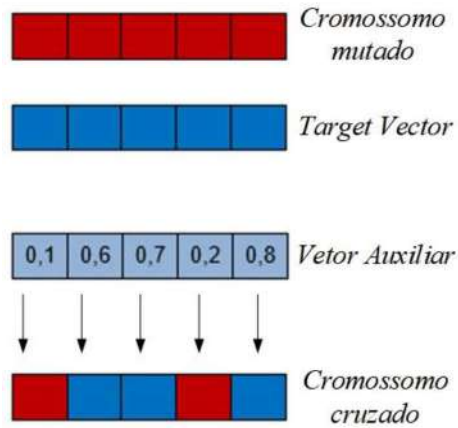
$x_{i,G}$ o *target vector*;

r_n o n -ésimo gene do cromossomo auxiliar;

CR a taxa de cruzamento.

A Figura 7 apresenta o método do processo de cruzamento supondo taxa de cruzamento igual a 0,5.

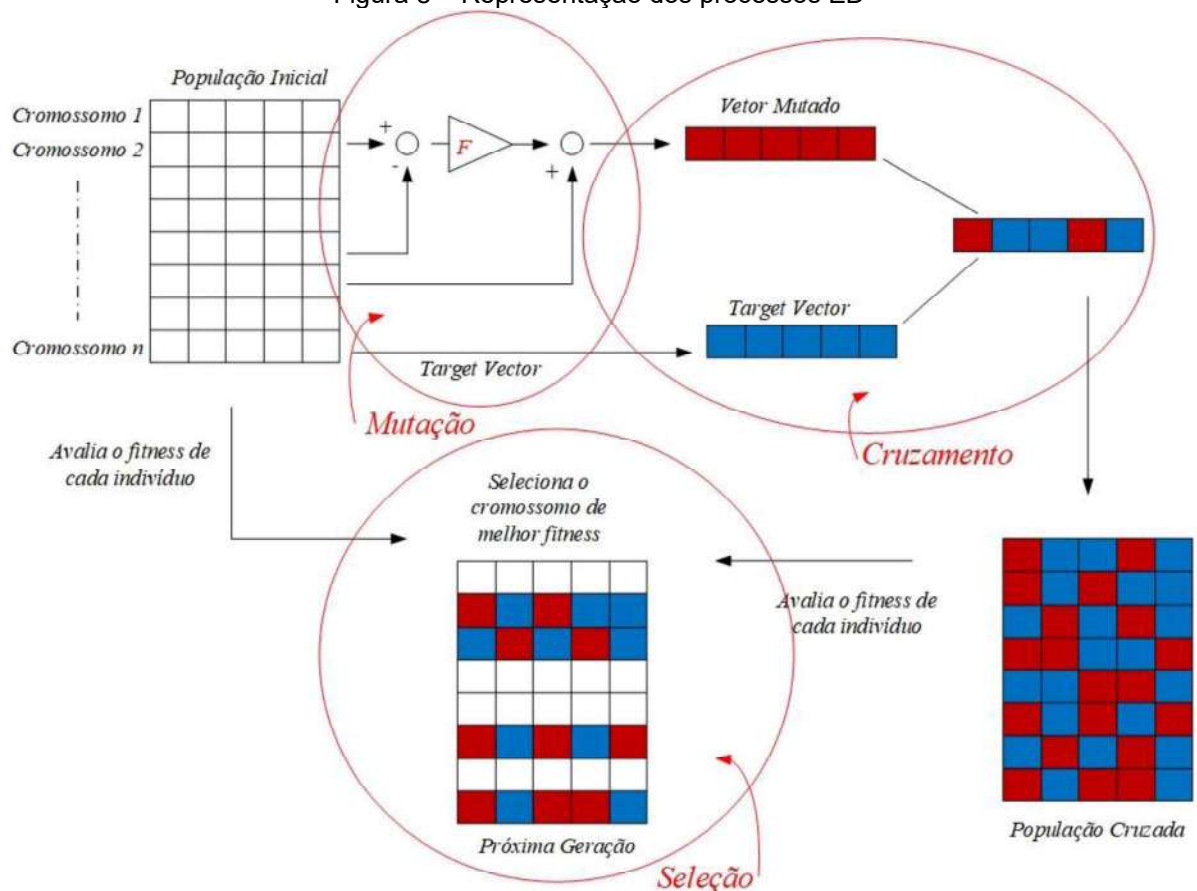
Figura 7 – Cruzamento ED



Fonte: Adaptado de Price, Stone e Lampinen (2006)

Por fim, o último processo do algoritmo é a seleção, a qual avalia os indivíduos de melhor aptidão, ou seja, de melhor *fitness*. A Figura 8 exemplifica todos os processos do algoritmo de evolução diferencial.

Figura 8 – Representação dos processos ED



Fonte: Autoria própria

Na seleção é realizada uma comparação entre o indivíduo após passar pelos processos de mutação e cruzamento com seu respectivo *target vector* da população inicial, assim, o indivíduo melhor adaptado é encaminhado para nova população.

Desta forma, conclui-se a primeira geração do algoritmo. Caso o critério de parada seja satisfeito, o algoritmo é finalizado e apresenta o cromossomo de melhor *fitness* entre todos os indivíduos da nova população. Caso não seja satisfeito, esta nova população passa a ser a população inicial e inicia-se uma nova geração.

3.3 CONCLUSÃO DO CAPÍTULO

Neste capítulo foi apresentado o princípio de funcionamento dos dois algoritmos evolutivos que serão comparados neste trabalho, o algoritmo genético e o de evolução diferencial. Ambos os algoritmos são amplamente utilizados em processos de otimização e fundamentais para o desenvolvimento deste trabalho, os quais serão avaliadas quanto a sua aplicação em otimização de controladores, os quais terão seus métodos detalhados no Capítulo 5.

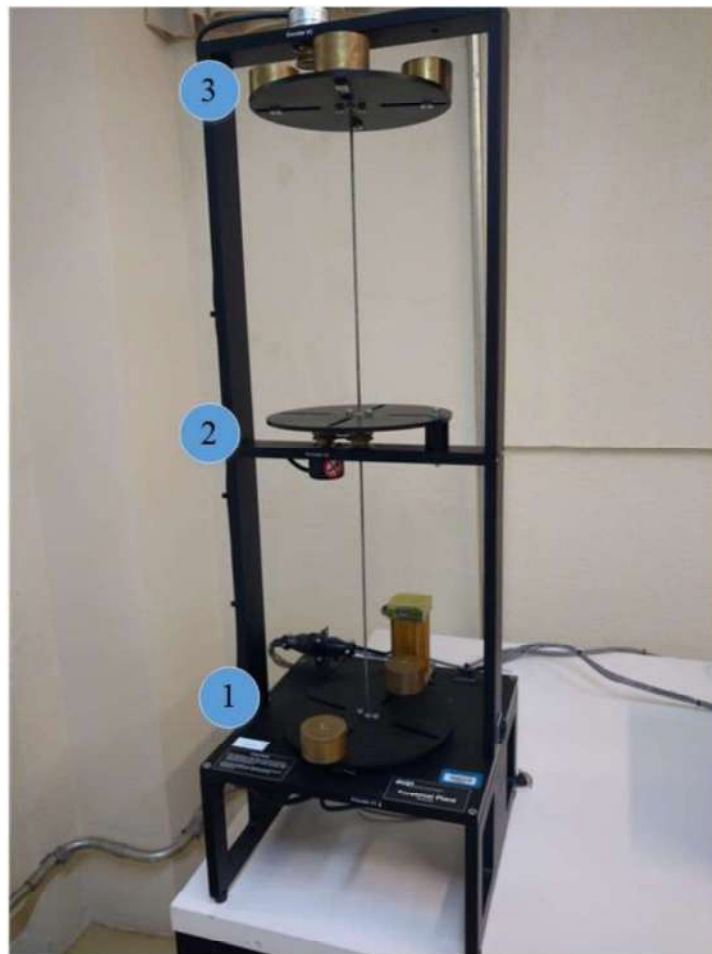
O Capítulo 4 apresenta a identificação do modelo do sistema da planta torcional didática, objeto de controle neste estudo.

4 IDENTIFICAÇÃO DO SISTEMA

Neste trabalho, os algoritmos genético e de evolução diferencial são aplicados a fim de otimizar a norma H_2 de um sistema torcional didático disponível no Laboratório de Controle e Otimização de Sistemas (LACOS) do Centro Integrado de Pesquisa em Controle e Automação (CIPECA) da Universidade Tecnológica Federal do Paraná (UTFPR).

A Planta Torcional ECP® - Modelo 205 é uma plataforma didática versátil para estudos de sistemas de controle. Esta planta é composta por três discos, os quais apresentam respectivos *encoders*, responsáveis por sua medida de posição. O acionamento dos discos é realizado por um atuador, o qual fica disposto no Disco 1. A Figura 9 ilustra a planta descrita.

Figura 9 – Planta torcional ECP®

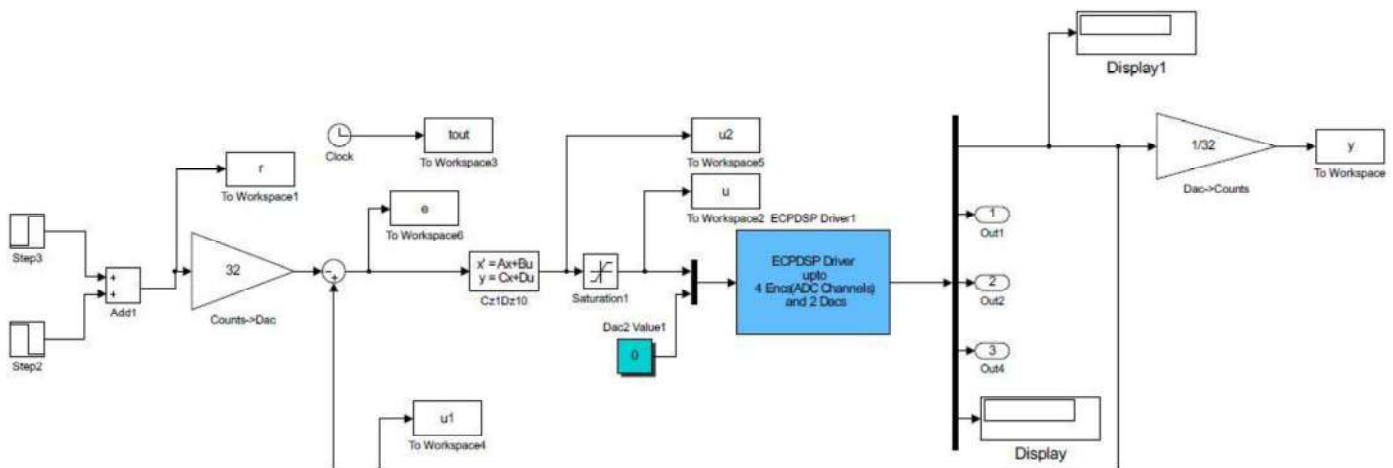


Fonte: Autoria própria.

O controle deste sistema é implementado utilizando o *software* MATLAB/Simulink® como interface de controle, permitindo assim, total versatilidade

para gerar e alterar o controlador do sistema, inclusive de forma *online*. É possível também alterar a configuração do sistema, seja o controlador apresentado em malha direta ou em malha de realimentação. A Figura 10 apresenta o diagrama de blocos do *Simulink* utilizado para os procedimentos experimentais deste trabalho.

Figura 10 – Diagrama de blocos do *Simulink*



Fonte: Autoria própria.

O bloco em azul na Figura 10 é a representação do sistema torcional no simulink. Este bloco aceita duas entradas, que são a tensão aplicada ao motor e a tensão aplicada em um dispositivo responsável pela inserção de perturbação, sendo que apenas a primeira entrada é utilizada neste trabalho. As saídas do bloco são as posições angulares de cada um dos discos e também um bit que informa se o sistema excedeu as regiões de operação, a fim de evitar danos físicos ao sistema. Os dados experimentais são coletados e enviados por meio do Processador Digital de Sinais (*Digital Signal Processor (DSP)*) acoplado ao computador por uma entrada do tipo PCI.

Diferentes configurações podem ser aplicadas à planta, ao implementá-la com um, dois ou três discos, resultando em sistemas de ordem dois, quatro ou seis, respectivamente. Há também a possibilidade de se acoplar massas aos discos, a fim de alterar sua configuração de inércia. Neste trabalho, a planta será implementada apenas com o Disco 1 (inferior) com duas massas de cobre localizadas a 9 centímetros do centro do disco, onde cada massa possui 0,5 kg e 5 cm de diâmetro (PARKS, 1999).

A modelagem do sistema realizada por Parks (1999) considera sua entrada e saída em unidades de *counts*, que é a unidade utilizada nos conversores

Digital/Analógico (D/A) e Analógico/Digital (A/D). Assim, os fatores de conversão são obtidos por (11) e (12).

$$T(t) = 7,866500885084628 \cdot 10^{-5} T_c(t) \quad (11)$$

$$\theta(t) = \frac{\pi}{64000} \theta_c(t) \quad (12)$$

sendo:

$T(t)$ o torque (entrada) do sistema em N.m;

$T_c(t)$ o torque (entrada) do sistema em *counts*;

$\theta(t)$ a posição (saída) do disco em rad;

$\theta_c(t)$ a posição (saída) do disco em *counts*.

A identificação do sistema é realizada a partir de uma série de procedimentos em malha aberta, para a identificação do coeficiente de amortecimento, da frequência natural e da frequência amortecida do sistema. A partir destes parâmetros, se calcula o momento de inércia e o coeficiente de atrito viscoso a serem utilizados na Equação da dinâmica do sistema, obtida pela modelagem física de acordo com o Parks (1999), é expressa por (13).

$$J_{d1} \ddot{\theta}(t) + c_{d1} \dot{\theta}(t) = T(t) \quad (13)$$

sendo:

J_{d1} o momento de inércia associado ao Disco 1;

$\ddot{\theta}$ a aceleração do Disco 1;

c_{d1} o coeficiente de atrito viscoso associado ao Disco 1;

$\dot{\theta}$ a velocidade do Disco 1.

Assim, a função de transferência e o modelo em espaços de estados identificados do sistema torcional são apresentados respectivamente por (14) e (15) (PARKS, 1999).

$$\frac{\theta_c(s)}{T_c(s)} = G(s) = \frac{1597}{s^2 + 0,948s} \quad (14)$$

$$\begin{aligned} \begin{bmatrix} \dot{\theta}_c \\ \ddot{\theta}_c \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ 0 & -0,948 \end{bmatrix} \begin{bmatrix} \theta_c \\ \dot{\theta}_c \end{bmatrix} + \begin{bmatrix} 0 \\ 32 \end{bmatrix} u(t) \\ y(t) &= \begin{bmatrix} 49,91 & 0 \end{bmatrix} \begin{bmatrix} \theta_c \\ \dot{\theta}_c \end{bmatrix} \end{aligned} \quad (15)$$

Com o sistema devidamente identificado, é possível realizar seu controle em malha fechada para fins de análises de estabilidade e desempenho, o qual será tratado a seguir.

4.1 CONCLUSÃO DO CAPÍTULO

Neste capítulo foi apresentada a identificação do modelo em função de transferência e em espaço de estados do sistema torcional definida por Parks (1999), o qual será utilizado a fim de otimizar seu controlador. Vale ressaltar que os algoritmos implementados podem ser aplicados a qualquer sistema, basta conhecer seu modelo matemático, sendo que o sistema torcional será utilizado a fim de possibilitar a validação dos dados de forma experimental.

O Capítulo 5 apresenta os métodos e procedimentos utilizados na elaboração dos algoritmos evolutivos, obtenção dos controladores ótimos e análises em malha fechada do sistema de estudo.

5 MÉTODOS

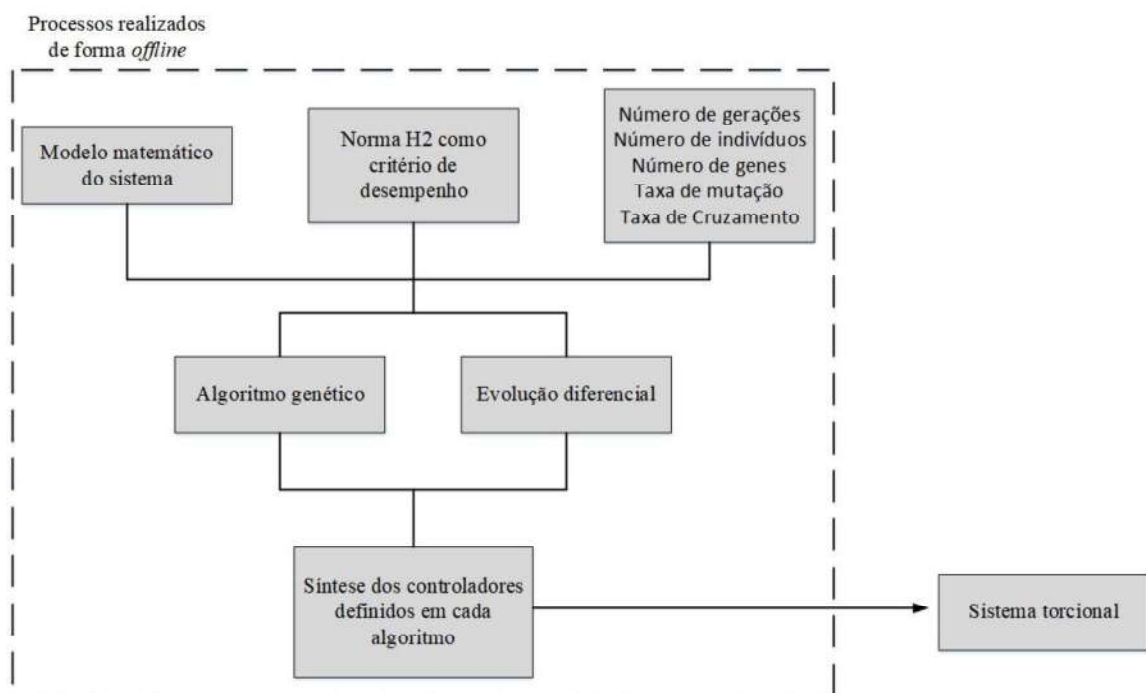
Os algoritmos genético e de evolução diferencial são implementados pelo *software* MATLAB®, com um notebook Intel Core i5 7th Gen, 8GB de memória, HD de 1 TB e placa de vídeo externa NVIDIA® Geforce de 2GB.

Para que os algoritmos possam ser implementados, três etapas são fundamentais para seu desenvolvimento e elaboração. Primeiramente, faz-se necessário conhecer o modelo matemático da planta do sistema que será controlado, em segundo lugar, é necessário determinar a função objetivo do sistema, ou seja, qual critério de desempenho será utilizado e, por fim, deve-se definir os parâmetros de cada algoritmo.

A versátil variação destes parâmetros torna o algoritmo flexível, o que possibilita sua aplicação em diferentes sistemas e permite identificar o controlador ótimo com base em diversos e possíveis critérios de desempenho, como a norma H_2 , norma H_∞ , ITAE, conforme apresentado em Bosco et al. (2017), ISE e diversos outros.

A Figura 11 ilustra a ordem das metodologias necessárias para elaboração do trabalho proposto, sendo que todos os métodos realizados até se obter a síntese dos controladores são realizados de forma *off-line* e, apenas após se conhecer o controlador otimizado, este é aplicado ao sistema torcional.

Figura 11 – Fluxograma dos métodos aplicados

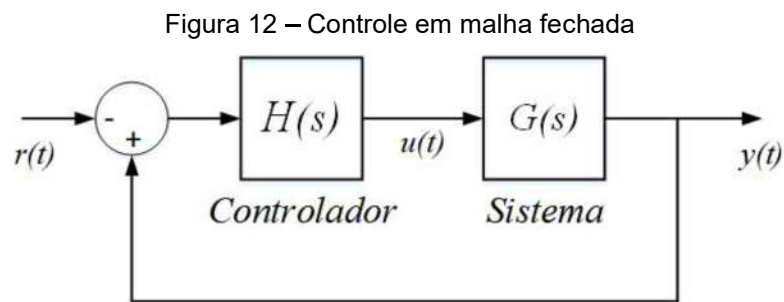


Fonte: Autoria própria

Assim, neste capítulo serão apresentados os métodos utilizados para que seja possível aplicar os algoritmos evolutivos a fim de determinar o controlador otimizado do sistema descrito.

5.1 SISTEMA EM MALHA FECHADA

Neste trabalho foi considerado um sistema com controle aplicado em malha direta e realimentação unitária, conforme diagrama de blocos apresentado na Figura 12. Este tipo de controle, em malha fechada, tem como principal objetivo garantir a estabilidade do sistema e permite ao projetista especificar características de funcionamento, como máximo sobressinal, tempo de assentamento, redução do *overshoot*, etc.



Fonte: Autoria própria

Considerando que o sistema a ser controlado possui ordem dois, como apresentado em (15), para garantir sua estabilidade, não há necessidade do controlador projetado possuir ordem superior que a própria planta (OGATA; MAYA; LEONARDI, 2010). Assim, especificou-se que o controlador possui ordem igual à planta do sistema, ou seja, ordem dois. Faz-se necessário conhecer o modelo genérico característico do controlador, sendo que sua representação em espaço de estados pode ser ilustrada em (16) e desmembrada em (17).

$$\begin{aligned} \dot{x}_c(t) &= A_c x_c(t) + B_c y(t) \\ u(t) &= C_c x_c(t) + D_c y(t) \end{aligned} \quad (16)$$

$$\begin{aligned} \dot{x}_c(t) &= \begin{bmatrix} a & c \\ b & d \end{bmatrix} x_c(t) + \begin{bmatrix} e \\ f \end{bmatrix} y(t) \\ u(t) &= \begin{bmatrix} g & h \end{bmatrix} x_c(t) + \begin{bmatrix} i \end{bmatrix} y(t) \end{aligned} \quad (17)$$

em que:

A_c é a matriz de estados do controlador;

B_c é a matriz de entrada do controlador;

C_c é a matriz de saída do controlador;

D_c é a matriz de transmissão direta do controlador;

x_c é o vetor dos estados do controlador H de dimensão n_{x_c} ;

u_c é a energia interna de controle com dimensão n_{u_c} ;

$a, b, c, d, e, f, g, h, i$ são os coeficientes do controlador.

Conhecendo a representação em espaço de estados tanto da planta quanto do controlador, pode-se desenvolver o sistema em malha fechada apresentado na Figura 12, o qual possui uma modelagem matemática genérica conforme (18).

$$\begin{aligned} \begin{bmatrix} \dot{x}(t) \\ \dot{x}_c(t) \end{bmatrix} &= \begin{bmatrix} A+BD_cC & BC_c \\ B_cC & A_c \end{bmatrix} \begin{bmatrix} x(t) \\ x_c(t) \end{bmatrix} + \begin{bmatrix} BD_c \\ B_c \end{bmatrix} u(t) \\ y(t) &= [C+D_cC \quad C_c] \begin{bmatrix} x(t) \\ x_c(t) \end{bmatrix} + [D_c]u(t) \end{aligned} \quad (18)$$

Desta forma, torna-se possível determinar os coeficientes do controlador que torne o sistema apresentado estável e otimizado. Lembrando que os coeficientes e a ordem do controlador $H(s)$ podem ser modificados de acordo com a planta do sistema, sendo este um caso particularizado para $G(s)$.

5.2 MINIMIZAÇÃO DA NORMA H_2

Conforme mencionado, para analisar o desempenho pela norma H_2 , uma saída controlada é inserida ao sistema, conforme apresentado na Equação (5). Esta saída artificial é determinada afim de especificar como deve ser a ponderação desejada para otimização tanto da energia interna quanto do tempo de resposta dos estados do sistema. Neste sentido, a matriz D_z possui relação inversa com a energia de controle, ou seja, quanto maior o valor da matriz, menor será a energia dispensada pelo controlador, resultando em um controle mais econômico. Por outro lado, a matriz C_z possui relação direta com a energia dos estados e, conseqüentemente, com o

tempo de resposta. Quanto maior o valor da matriz C_z mais rápida será a resposta deste sistema. Os dois objetivos apresentados são conflitantes, uma vez que para realizar busca rápida dos estados, demanda-se uma alta energia de controle.

Assim, três testes são abordados neste trabalho, onde o primeiro especifica um tempo rápido de resposta dos estados e com uma alta energia de controle. Já o segundo teste mantém tanto o tempo de resposta, quanto a energia interna em equilíbrio e, por fim, o terceiro teste especifica uma resposta lenta com baixa energia de controle.

As três situações são passíveis de implementação computacional, porém em circunstâncias físicas, há a necessidade de se fazer um mapeamento para verificar se o sistema será implementável. As ponderações relacionadas à saída artificial correspondente aos três pontos de operação estão representadas na Tabela 1.

Tabela 1 – Ponderações relacionadas à saída artificial

	Operação 1	Operação 2	Operação 3
C_z	[10 10]	[1 1]	[1 1]
D_z	[1]	[1]	[10]

Fonte: Autoria própria.

Portanto, conforme mencionado, os critérios de desempenho são analisados pela relação entre a entrada exógena e à saída controlada do sistema. O sistema resultante, também denominado de sistema aumentado, é uma relação entre o modelo do sistema $G(s)$, do controlador $H(s)$ e das matrizes relacionadas à saída artificial. Assim, o sistema aumentado pode ser definido por (19).

$$\begin{aligned}
 \begin{bmatrix} \dot{x}(t) \\ \dot{x}_c(t) \end{bmatrix} &= \begin{bmatrix} A + (BD_cC) & BC_c \\ B_cC & A_c \end{bmatrix} \begin{bmatrix} x(t) \\ x_c(t) \end{bmatrix} + \begin{bmatrix} B_w \\ 0 \end{bmatrix} w(t) \\
 z(t) &= [C_z + (D_zD_cC) \quad D_zC_c] \begin{bmatrix} x(t) \\ x_c(t) \end{bmatrix}
 \end{aligned} \tag{19}$$

Com o modelo proposto, é possível implementar tanto o algoritmo genético quanto o de evolução diferencial a fim de determinar os coeficientes do controlador que minimizem a norma H_2 e, conseqüentemente, minimizem a energia interna e tempo de resposta do sistema.

5.3 PARÂMETROS DOS ALGORITMOS

Para a implementação dos algoritmos, alguns parâmetros devem ser determinados de forma comum para o AG e para o ED, como o número de genes da população inicial, número de gerações e a função objetivo.

Para a representação em espaço de estados, conforme Equação generalizada apresentada em (17), nove genes são necessários para representar os coeficientes do controlador. Vale lembrar que este valor pode ser alterado de acordo com o grau do controlador. A Equação (20) exemplifica o cálculo necessário para se obter o número de genes do algoritmo.

$$genes = n_{x_c} n_{x_c} + n_{x_c} n_y + n_{x_c} n_x + n_y n_x \quad (20)$$

Para fins de ilustração, a Figura 13 mostra como deve ser formada a população dos cromossomos do algoritmo, onde cada cromossomo representa um controlador diferente.

Figura 13 – População dos algoritmos

<i>Controlador 1</i>	<i>a1</i>	<i>b1</i>	<i>c1</i>	<i>d1</i>	<i>e1</i>	<i>f1</i>	<i>g1</i>	<i>h1</i>	<i>i1</i>
<i>Controlador 2</i>	<i>a2</i>	<i>b2</i>	<i>c2</i>	<i>d2</i>	<i>e2</i>	<i>f2</i>	<i>g2</i>	<i>h2</i>	<i>i2</i>
<i>Controlador 3</i>	<i>a3</i>	<i>b3</i>	<i>c3</i>	<i>d3</i>	<i>e3</i>	<i>f3</i>	<i>g3</i>	<i>h3</i>	<i>i3</i>
<i>Controlador 4</i>	<i>a4</i>	<i>b4</i>	<i>c4</i>	<i>d4</i>	<i>e4</i>	<i>f4</i>	<i>g4</i>	<i>h4</i>	<i>i4</i>
<i>Controlador NP</i>	<i>a_n</i>	<i>b_n</i>	<i>c_n</i>	<i>d_n</i>	<i>e_n</i>	<i>f_n</i>	<i>g_n</i>	<i>h_n</i>	<i>i_n</i>

Fonte: Autoria própria

sendo:

a_n, b_n, c_n, d_n, e_n, f_n, g_n, h_n e *i_n* os coeficientes dos *NP* controladores.

Mil gerações são definidas como limite máximo para os algoritmos, valor este que é suficientemente grande para garantir a menor norma H_2 considerando os parâmetros apresentados. Um elevado número de gerações pode aumentar o tempo de convergência do algoritmo, contudo, por se tratar de um sistema *offline*, o tempo de convergência não prejudica a qualidade e eficiência do algoritmo de otimização.

Assim, o próximo parâmetro a ser determinado é a função objetivo, também denominada de *fitness*, a qual é caracterizada pela minimização da norma H_2 do sistema apresentado, conforme (21).

$$FO = \text{Min}_{H(s)} \{ \text{norma } H_2(G(s)) \} \quad (21)$$

Os parâmetros definidos até o momento são comuns entre os algoritmos genético e de evolução diferencial. A seguir serão definidos os métodos utilizados para a obtenção dos parâmetros individuais de cada algoritmo. A técnica utilizada é empírica, porém os valores iniciais são embasados em literatura, lembrando que os parâmetros podem ser alterados de acordo com a aplicação dos algoritmos de otimização, sendo necessária uma pré-análise para avaliar quais conjuntos de parâmetros são os interessantes.

O critério utilizado para avaliar qual o melhor valor para dado parâmetro é a média e a variância do valor da norma H_2 obtida a partir de dez ciclos de trabalho do algoritmo, conforme (22) e (23), respectivamente. Mais detalhes a respeito dos parâmetros considerados para teste são apresentados no Capítulo 6. Desta forma, é possível avaliar quais parâmetros do algoritmo resultam em uma maior consistência de resultados, sendo que o desempenho do algoritmo é diretamente dependente de seus parâmetros.

$$\text{média} = \frac{\sum_1^{10} \text{norma}}{10} \quad (22)$$

$$\text{variancia} = \frac{\sum (\text{norma}_i - \text{média})^2}{10-1} \quad (23)$$

sendo:

norma_i o valor da norma H_2 das amostras $i = 1, 2, \dots, 10$.

5.3.1 Algoritmo genético

Os parâmetros restantes que devem ser conhecidos do algoritmo genético são a taxa de cruzamento, o limiar de cruzamento, a taxa de mutação e o limiar de

mutação. Não há em literatura valores pré-definidos para tais parâmetros, pois estes podem variar de acordo com a aplicação do algoritmo.

Contudo, Lacerda e Carvalho (1999) sugerem que a taxa de cruzamento deve estar entre 60% e 95% dos indivíduos, sendo que quando não ocorre cruzamento os indivíduos são iguais aos pais e não há variação genética. Os autores também sugerem que a taxa de mutação deva estar entre 1% e 5%. Já o número de indivíduos possui uma relação direta com o número de genes dos cromossomos, sendo que, quanto maior o número de genes, maior deverá ser o número de indivíduos desta população.

Já as especificações para os limiares de mutação e de cruzamento não foram encontradas em literatura, pois os métodos utilizados foram adaptados neste trabalho, a fim de melhorar o rendimento do algoritmo genético.

Observou-se com os testes realizados com a taxa de mutação sugerida por Lacerda e Carvalho (1999), deixava o tempo de convergência do algoritmo muito alto. Assim, de forma a otimizá-la, a taxa de mutação analisada foi entre 5% a 15%, sendo que tais valores foram determinados de forma empírica e apresentaram melhores respostas.

Conforme mencionado e conhecendo apenas possíveis estimativas dos parâmetros do algoritmo, sucessivos testes empíricos foram realizados a fim de determinar faixas limites de possíveis parâmetros possibilitando assim aumentar a eficiência do AG. O campo de busca dos parâmetros definidos para a análise do algoritmo é apresentado na Tabela 2 e serão analisados no Capítulo 6.

Tabela 2 – Variação dos parâmetros AG

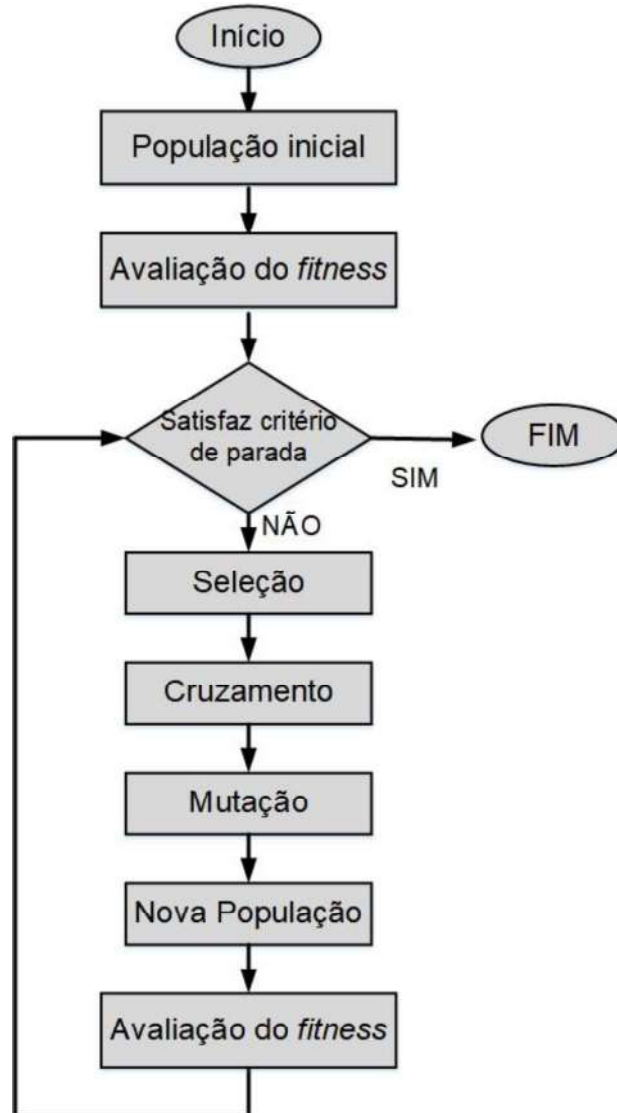
Parâmetros	Testes
Número de indivíduos	[45 67 90]
Taxa de cruzamento	[0,85 0,90 0,95]
Limiar de cruzamento	[0,40 0,50 0,60]
Taxa de mutação	[0,20 0,25 0,30]
Limiar de Mutação	[0,05 0,10 0,15]

Fonte: Autoria própria.

Após definir a população inicial aleatória, todos os indivíduos são avaliados de acordo com a função objetivo, que neste caso está associada à norma H_2 . Assim, é realizado o primeiro processo do algoritmo, que é a seleção por elitismo, onde o cromossomo de melhor *fitness* é diretamente encaminhado para a próxima população.

A Figura 14 ilustra a ordem dos processos realizados pelo algoritmo genético.

Figura 14 – Fluxograma do AG



Fonte: Autoria própria

Em sequência os demais processos são realizados, a seleção por torneio, o cruzamento e por fim a mutação. Após todos os processos mencionados há uma nova avaliação da população que já passou por todas as etapas e, novamente o cromossomo de melhor *fitness* é determinado. Contudo, se o critério de parada do algoritmo não for satisfeito, esta população já alterada assume o lugar da população inicial e todas as etapas são novamente realizadas. Cada ciclo de processos do algoritmo é determinado como uma geração.

5.3.2 Algoritmo de evolução diferencial

Há alguns trabalhos que abordam a interação dos parâmetros do ED à sua eficiência e robustez. Storn e Price (1997) especificam que o número de cromossomos deve estar entre $5D$ e $10D$, onde D é o número de genes de um indivíduo e que este valor deve ser no mínimo 4, para garantir que o algoritmo terá vetores suficientes para realizar o processo de mutação. A taxa de mutação e de cruzamento são valores entre $[0, 1]$, assim, Storn e Price (1997) sugerem que $F = 0,5$ e $CR = 0,9$ são boas escolhas para valores iniciais.

Já para Gämperle, Müller e Koumoutsakos (2002) o número de indivíduos deve estar entre $3D$ e $6D$, enquanto as escolhas iniciais de taxa de mutação e cruzamento são $F = 0,6$ e $CR = [0,3 \ 0,9]$. Já para Rankkonen, Kukkonen e Price (2005) o número de indivíduos deve estar entre $2D$ e $4D$, $F = [0,4 \ 0,95]$ e $CR = [0,8 \ 1,0]$.

Conforme pode ser observado, não há em literatura parâmetros específicos para o algoritmo. Portanto, neste trabalho alguns testes empíricos foram realizados, tendo como campo de busca os parâmetros apresentados na Tabela 3.

Tabela 3 – Variação dos parâmetros ED

Parâmetros	Testes
Número de indivíduos	[18 27 36]
Taxa de mutação	[0,45 0,60 0,75]
Taxa de cruzamento	[0,4 0,5 0,6]

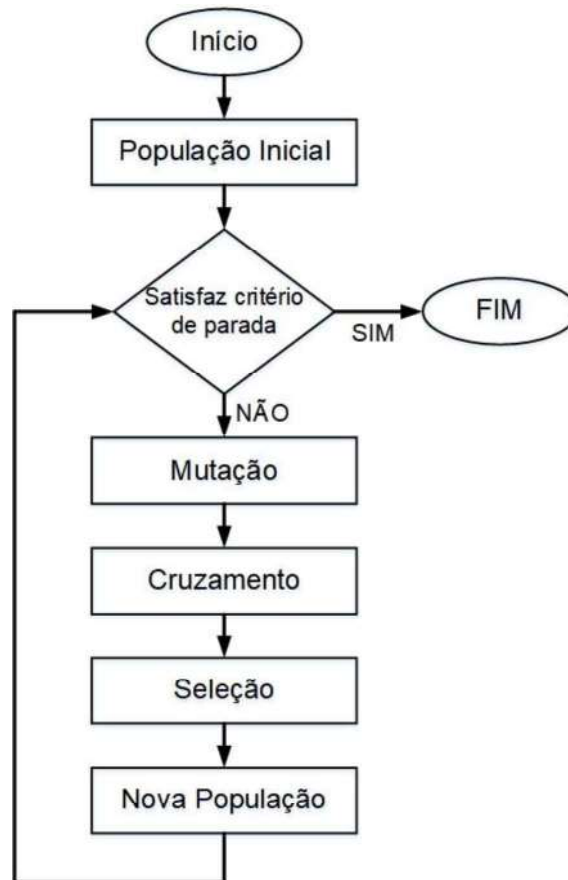
Fonte: Autoria própria.

Após definir os parâmetros que apresentam menor variância nos resultados, juntamente com seu modelo em espaço de estados e os demais parâmetros necessários, o algoritmo é inicializado.

Com a população inicial definida, torna-se possível implementar os demais processos do algoritmo, sendo a mutação, o cruzamento e a seleção. O *fitness* do algoritmo de evolução diferencial é a norma H_2 do sistema em malha fechada, sendo o controlador determinado pelos genes dos indivíduos da população do ED.

Todos os processos descritos são realizados repetidamente até que o critério de parada seja satisfeito, em que neste caso é determinado pelo número máximo de gerações. A Figura 15 ilustra o fluxograma dos processos do algoritmo apresentado.

Figura 15 – Fluxograma do ED



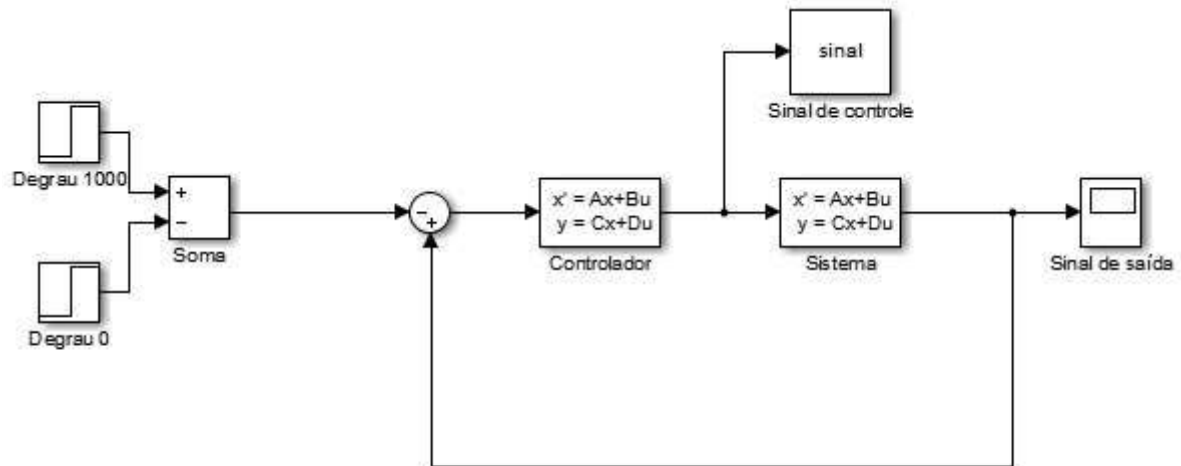
Fonte: Autoria própria

5.5 PROCEDIMENTOS SIMULADOS

Ao se conhecer o controlador ótimo a ser aplicado na planta do sistema torcional de estudo, primeiramente, faz-se seus testes computacionais para validação. Para isso, é utilizado o *software* MATLAB/*Simulink*, o qual permite simular as condições reais do sistema em malha fechada.

A Figura 16 ilustra o sistema em malha fechada com realimentação unitária simulado pelo *software* mencionado, considerando uma entrada degrau como referência. A simulação também retorna o sinal de controle deste sistema e plota o gráfico da sua saída, a fim de realizar comparações com os sinais de entrada e calcular seu erro.

Figura 16 – Representação do sistema em malha fechada



Fonte: Autoria própria

Desta forma, com os controladores selecionados e devidamente simulados computacionalmente, realiza-se os testes experimentais do sistema torcional de estudo, mostrado a seguir. Estes testes são necessários para que os controladores sejam validados e para que assim, possa se confirmar a efetividade dos algoritmos de otimização implementados.

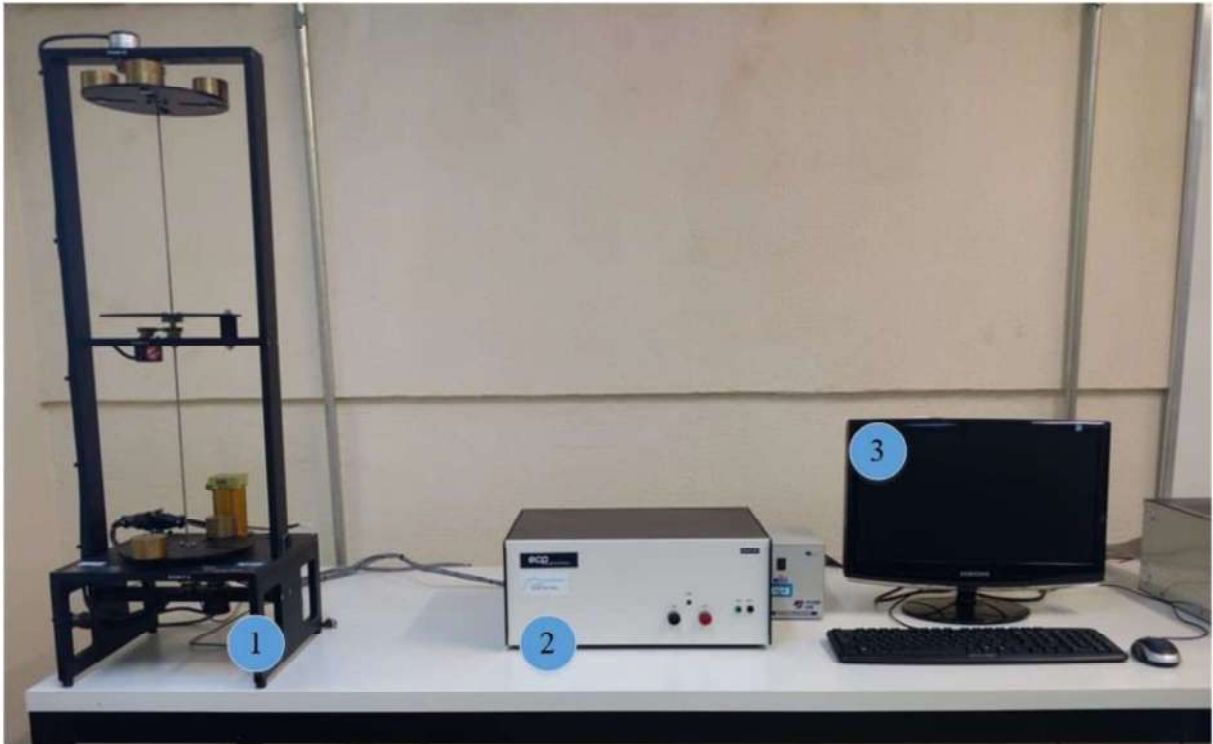
5.6 PROCEDIMENTOS EXPERIMENTAIS

A planta torcional é operada em tempo real utilizando o *software* MATLAB/Simulink® como interface de controle. Isso permite que as ferramentas de desenvolvimento e análise disponibilizadas pelo *software* possam ser utilizadas enquanto o controle é implementado, possibilitando inclusive a alteração online de parâmetros do sistema.

A placa de processamento de dados, denominada como Processador Digital de Sinais (DSP - *Digital Signal Processor*), está localizado no computador ligado à planta e é utilizada no sistema a fim de fornecer alta performance e precisão para a execução de controle da planta torcional em tempo real. Outro funcionamento indispensável do DSP é o monitoramento de limites máximos de segurança, como velocidade, corrente e tensão.

Assim, a Figura 17 apresenta a estrutura do sistema disponível no LACOS, a qual possibilita a implementação física do controle da planta torcional ECP.

Figura 17 – Sistema torcional aplicado



Fonte: Autoria própria

Os elementos necessários para os testes são: (1) a planta torcional; (2) fonte de alimentação; (3) computador para a utilização do MATLAB como interface e utilização do DSP acoplado à sua CPU.

Como neste trabalho é utilizado apenas o Disco 1 acoplado com dois pesos, a configuração dos demais discos não interfere no desempenho do controlador.

5.6 CONCLUSÃO DO CAPÍTULO

Neste capítulo foi apresentada a metodologia para o desenvolvimento desse trabalho, descrevendo os procedimentos para a determinação dos parâmetros e implementação dos algoritmos genéticos e de evolução diferencial, bem como os procedimentos experimentais realizados.

No Capítulo 6 são apresentados os principais resultados obtidos relacionados a determinação dos parâmetros dos algoritmos e estudo.

6 PARÂMETROS

Neste capítulo são apresentados os resultados obtidos para a determinação dos parâmetros do algoritmo genético, bem como do algoritmo de evolução diferencial. Tais parâmetros serão posteriormente utilizados a fim de realizar as sínteses dos controladores otimizados do sistema de estudo.

6.1 PARÂMETROS DO ALGORITMO GENÉTICO

Considerando as variações de parâmetros apresentados na Tabela 2, foram realizados diversos testes com o algoritmo genético onde, para cada teste, foram realizados dez ciclos de trabalho do algoritmo, ou seja, o algoritmo foi executado dez vezes para cada configuração de parâmetros.

O primeiro teste foi realizado variando o número de indivíduos, onde os demais parâmetros foram mantidos constantes em seu valor mínimo atribuído. A variância e a média são relacionadas ao valor resultante da função objetivo do AG, ou seja, é relacionado à norma H_2 do sistema. A Tabela 4 apresenta os parâmetros atribuídos ao algoritmo, bem como resultados estatísticos obtidos.

Tabela 4 – Número de indivíduos AG

Parâmetros	Teste 1	Teste 2	Teste 3
Número de indivíduos	45	67	90
Taxa de cruzamento	0,85	0,85	0,85
Limiar de cruzamento	0,40	0,40	0,40
Taxa de mutação	0,20	0,20	0,20
Limiar de Mutação	0,05	0,05	0,05
Média	7,6699	3,8968	3,0491
Variância	68,2257	5,0173	3,9238

Fonte: Autoria própria.

Nota-se que quanto maior o número de indivíduos, menor o valor da norma e da variância apresentados. Determinou-se o limite máximo de indivíduos igual a 90 (10 vezes o número de genes), pois populações extensas aumentam consideravelmente o tempo de convergência do algoritmo e apresentam pouca melhoria, fato mostrado pelas médias entre os testes com 67 e 90 indivíduos.

Assim, determina-se o número de indivíduos igual a 90 e para os próximos testes, a taxa de cruzamento é variada a fim de análise conforme Tabela 5.

Tabela 5 – Porcentagem de cruzamento AG

Parâmetros	Teste 1	Teste 2	Teste 3
Número de indivíduos	90	90	90
Taxa de cruzamento	0,85	0,90	0,95
Limiar de cruzamento	0,40	0,40	0,40
Taxa de mutação	0,20	0,20	0,20
Limiar de Mutação	0,05	0,05	0,05
Média	3,0491	2,3344	2,1965
Variância	3,9238	1,4904	0,4025

Fonte: Autoria própria.

De acordo com o apresentado, a melhor taxa de cruzamento é de 95% dos indivíduos, isso quer dizer que apenas 5% dos indivíduos filhos serão mantidos iguais aos pais. Assim, mostra-se que aumentar a diversidade destes indivíduos melhora a eficiência do algoritmo.

A Tabela 6 apresenta os testes ao se variar o limiar de cruzamento.

Tabela 6 – Taxa de cruzamento AG

Parâmetros	Teste 1	Teste 2	Teste 3
Número de indivíduos	90	90	90
Taxa de cruzamento	0,95	0,95	0,95
Limiar de cruzamento	0,40	0,50	0,60
Taxa de mutação	0,20	0,20	0,20
Limiar de Mutação	0,05	0,05	0,05
Média	2,1965	3,0362	1,9316
Variância	0,4025	2,2192	0,4321

Fonte: Autoria própria.

Conforme mencionado, a forma de se realizar o cruzamento já apresenta melhorias em relação a forma mais convencional, que a técnica de corte, portanto, notou-se que a taxa de 60% proporciona menor média dos valores de norma. É interessante notar que com uma taxa de 50% apresentou-se uma média maior do que com 40% ou 60%, o que mostra que é mais vantajoso apresentar a probabilidade de gerar filhos com mais genes de um dos pais do que uma probabilidade igual entre ambos.

A taxa de indivíduos mutados é menor do que a porcentagem de indivíduos cruzados, pois há a necessidade de se manter o campo de busca inicial do algoritmo de forma majoritária e permitir que apenas alguns indivíduos busquem campos de convergência diferentes. Este processo é necessário para que se evitem mínimos

locais. A Tabela 7 apresenta os testes realizados com a variação da taxa de indivíduos mutados.

Tabela 7 – Porcentagem de mutação AG

Parâmetros	Teste 1	Teste 2	Teste 3
Número de indivíduos	90	90	90
Taxa de cruzamento	0,95	0,95	0,95
Limiar de cruzamento	0,60	0,60	0,60
Taxa de mutação	0,20	0,25	0,30
Limiar de Mutação	0,05	0,05	0,05
Média	1,9316	1,9288	1,5753
Variância	0,4321	0,4556	0,2404

Fonte: Autoria própria.

Ao proporcionar que 30% dos indivíduos sofram mutação apresentou-se o menor valor de média e variância da norma do sistema estudado. Isso mostra que o campo de busca inicial não está totalmente adequado para esta aplicação, contudo, o próprio processo do algoritmo possibilita corrigir esta busca.

Por fim, o último parâmetro analisado é o limiar de mutação, apresentado na Tabela 8.

Tabela 8 – Taxa de mutação AG

Parâmetros	Teste 1	Teste 2	Teste 3
Número de indivíduos	90	90	90
Taxa de cruzamento	0,95	0,95	0,95
Limiar de cruzamento	0,60	0,60	0,60
Taxa de mutação	0,30	0,30	0,30
Limiar de Mutação	0,05	0,10	0,15
Média	1,5753	0,7359	1,0114
Variância	0,2404	0,1208	0,5594

Fonte: Autoria própria.

Limiares de mutação muito altos podem direcionar a população para fora do campo ótimo de busca, da mesma forma que um limiar muito baixo não consegue realizar a busca necessária. Desta forma, o limiar de 0,10 foi o que apresentou melhores resultados nos valores de norma.

A Tabela 9 apresenta os parâmetros do algoritmo genético determinados de forma empírica e estatística, os quais apresentam indivíduos de melhor *fitness* e de alta confiabilidade do algoritmo, representado pela baixa variância de seus valores de norma.

Tabela 9 – Parâmetros AG

Parâmetros	
Número de indivíduos	90
Taxa de cruzamento	0,95
Limiar de cruzamento	0,60
Taxa de mutação	0,30
Limiar de Mutação	0,10

Fonte: Autoria própria.

6.2 PARÂMETROS DA EVOLUÇÃO DIFERENCIAL

Já o algoritmo de evolução diferencial apresenta apenas três parâmetros a serem determinados, os quais estão apresentados na Tabela 3. O método empírico e estatístico utilizado para a determinação de seus parâmetros é o mesmo que foi apresentado no algoritmo genético.

A Tabela 10 ilustra os testes realizados considerando a variação dos indivíduos da população deste algoritmo.

Tabela 10 – Número de indivíduos ED

Parâmetros	Teste 1	Teste 2	Teste 3
Número de indivíduos	18	27	36
Taxa de mutação	0,45	0,45	0,45
Taxa de cruzamento	0,4	0,4	0,4
Média	2,3491	1,2490	1,6708
Variância	8,2898	0,5628	1,0452

Fonte: Autoria própria.

Como o principal objetivo é minimizar a norma H_2 , a escolha mais interessante de parâmetros é a que apresenta a menor média e variância de sua função objetivo. Desta forma, os parâmetros realizados no Teste 2 são os mais aptos a serem utilizados, assim, o número de indivíduos do algoritmo é definido como 27.

Tal resultado é esperado, pois aumentar excessivamente o número de indivíduos da população pode dificultar que o algoritmo encontre o campo de busca ótimo, caso os operadores genéticos não estejam devidamente ajustados. Isso ocorre devido à característica diferencial de seus vetores, assim, populações grandes aumentam a diversidade de suas soluções.

Os demais parâmetros são testados substituindo-se o número de indivíduos já determinado anteriormente. Logo, a Tabela 11 apresenta os parâmetros de teste para a variação do valor da taxa de mutação atribuídas ao algoritmo.

Tabela 11 – Taxa de mutação ED

Parâmetros	Teste 1	Teste 2	Teste 3
Número de indivíduos	27	27	27
Taxa de mutação	0,45	0,6	0,75
Taxa de cruzamento	0,4	0,4	0,4
Média	1,2490	4,0149	49,8013
Variância	0,5628	19,1843	2095,5190

Fonte: Autoria própria.

Fica evidente que ao se aumentar a taxa de mutação, maiores serão a média e a variância dos resultados, portanto a taxa determinada para implementação é de 0,45. A taxa de mutação é, por definição, o tamanho do passo dado em direção ao vetor diferença criado por dois indivíduos, assim, é possível inferir que passos muito grandes levam os indivíduos para campos de busca fora do campo ótimo.

Por fim, o último parâmetro a ser determinado é a taxa de cruzamento, onde seus valores de testes podem ser verificados na Tabela 12.

Tabela 12 – Taxa de cruzamento ED

Parâmetros	Teste 1	Teste 2	Teste 3
Número de indivíduos	27	27	27
Taxa de mutação	0,45	0,45	0,45
Taxa de cruzamento	0,4	0,5	0,6
Média	1,2490	0,6406	0,4691
Variância	0,5628	0,1205	0,0044

Fonte: Autoria própria.

A taxa de cruzamento é a probabilidade de escolha entre os genes do vetor doador e os genes do *target vector*. Com os resultados obtidos mostra-se que ao apresentar uma probabilidade de 60%, é possível obter um algoritmo mais eficiente com os menores valores de média e variância da norma H_2 . Este resultado é esperado, pois aumentar ou diminuir a taxa de cruzamento possibilita que haja excesso ou escassez de diversidade do vetor doador.

Portanto, os parâmetros do algoritmo de evolução diferencial utilizados para determinar o controlador ótimo do sistema torcional descrito são apresentados na Tabela 13.

Tabela 13 – Parâmetros ED

Parâmetros	
Número de indivíduos	27
Taxa de mutação	0,45
Taxa de cruzamento	0,6

Fonte: Autoria própria.

Com os parâmetros definidos para ambos os algoritmos, pode-se realizar a síntese dos controladores otimizados, o valor do critério de desempenho e as saídas do sistema sujeitas à diversas entradas. Neste trabalho foram realizados testes para entrada degrau, a qual é analisada a seguir.

6.3 CONCLUSÃO DO CAPÍTULO

A eficiência dos algoritmos evolutivos está diretamente relacionada com seus parâmetros. Desta forma, neste capítulo foram apresentadas avaliações estatísticas a fim de selecionar os parâmetros que configuram maior confiabilidade dos algoritmos.

O capítulo 7 apresenta a síntese dos controladores ótimos obtidos por ambos os algoritmos de estudo, assim como a análise da aplicação de possíveis restrições aos algoritmos de otimização.

7 SINTESE DE CONTROLADORES

Com o modelo matemático do sistema, a função objetivo e os parâmetros definidos, os algoritmos propostos podem ser aplicados à otimização de controladores. Neste capítulo serão apresentados os principais controladores obtidos e aplicados ao sistema torcional de estudo.

7.1 CONTROLADORES OBTIDOS PELO ALGORITMO GENÉTICO

Por se tratar de uma meta-heurística, o algoritmo genético não garante o mínimo global, porém apresenta um resultado sub-ótimo dentro dos parâmetros apresentados. Assim, para obter o controlador otimizado, foram realizados dez ciclos de trabalho do algoritmo. A Tabela 14 apresenta o valor da norma H_2 obtida em cada ciclo para as três operações apresentadas na Tabela 1.

Tabela 14 – Função objetivo AG

Ciclo	Operação 1	Operação2	Operação 3
1	26,2822	1,3495	0,5919
2	17,3183	0,7965	1,0791
3	9,4304	0,7402	0,6878
4	24,0170	0,7372	0,7309
5	10,0321	0,7760	0,5428
6	7,3922	0,9453	0,4557
7	14,7910	2,5313	1,5640
8	9,9159	1,3527	0,5386
9	15,2475	2,0123	0,7571
10	4,2627	2,0931	0,4114
Média	13,8689	1,3334	0,7359
Variância	50,6663	0,4367	0,1208

Fonte: Autoria própria.

Nota-se que o valor das normas e suas médias variam conforme a operação estabelecida. Para fim de análise, não é viável realizar comparações entre os pontos de operação, pois estes estão submetidos a diferentes condições de funcionamento já mencionados anteriormente.

Desta forma, será considerado o melhor controlador para cada ponto de operação dentre os dez ciclos realizados, ou seja, o controlador que resulta na menor

norma do sistema. Para o primeiro ponto de operação, em que é especificada uma prioridade na velocidade de convergência dos estados em detrimento da energia de controle, o melhor controlador obtido apresenta norma igual a 4,2627 e é apresentado em (24).

$$\begin{aligned}\dot{x}_c &= \begin{bmatrix} -37,8096 & -16,4733 \\ 5,3085 & -20,5704 \end{bmatrix} x_c + \begin{bmatrix} -15,7565 \\ 0,4539 \end{bmatrix} y \\ u &= [-0,3997 \quad 0,3909] x_c + [-0,1238] y\end{aligned}\tag{24}$$

O segundo ponto de operação analisado possui o tempo de resposta da saída do sistema e a energia interna de controle com mesma ponderação, assim, da mesma forma que apresentado anteriormente, entre os dez testes realizados, a menor norma obtida foi 0,7372. O controlador ótimo para estas condições é expresso por (25).

$$\begin{aligned}\dot{x}_c &= \begin{bmatrix} -12,4916 & -20,2044 \\ 60,8989 & -45,4672 \end{bmatrix} x_c + \begin{bmatrix} -5,3122 \\ -8,4667 \end{bmatrix} y \\ u &= [-0,5428 \quad -0,5022] x_c + [-0,1417] y\end{aligned}\tag{25}$$

Por fim, o terceiro e último ponto de operação analisado, apresenta um controlador ótimo, dentro dos parâmetros apresentados, conforme (26). A norma apresentada pelo sistema nestas condições é 0,4114.

$$\begin{aligned}\dot{x}_c &= \begin{bmatrix} -235,4180 & -2,3589 \\ 23,4289 & -17,3173 \end{bmatrix} x_c + \begin{bmatrix} -0,5634 \\ -0,0675 \end{bmatrix} y \\ u &= [-4,6019 \quad 5,5760] x_c + [-0,0152] y\end{aligned}\tag{26}$$

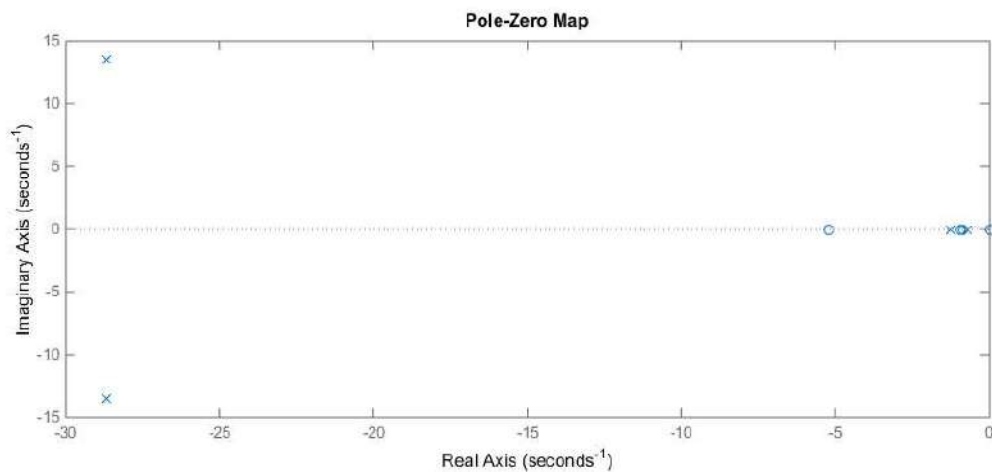
Ao realizar os testes experimentais relacionados a cada um dos controladores apresentados até o momento, em nenhum caso o sistema teve energia suficiente para acionar o atuador e mover o disco. Ou seja, os três controladores possuem norma H_2 tão baixa, que a energia de controle do sistema não é suficiente para tirar o disco da inércia.

Nota-se também que os controladores apresentados resultam em sistemas em que seus polos possuem parte real muito negativas, o que teoricamente caracteriza um sistema de busca rápida dos estados. Porém, como a energia de

controle é mínima, não é possível que tais controladores sejam fisicamente implantados.

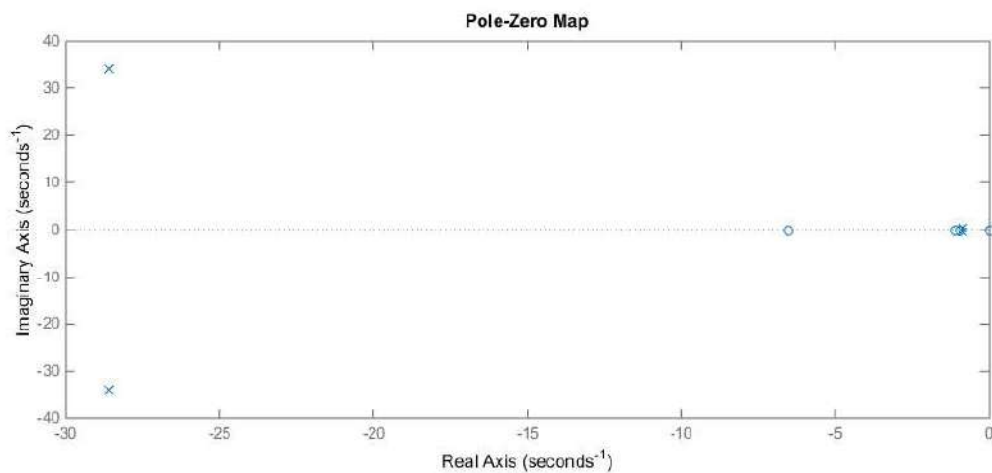
As Figuras 18, 19 e 20 mostram os zeros e polos do sistema em malha fechada ao aplicar os controladores (24), (25) e (26), respectivamente.

Figura 18 – Zeros e polos operação 1 - AG



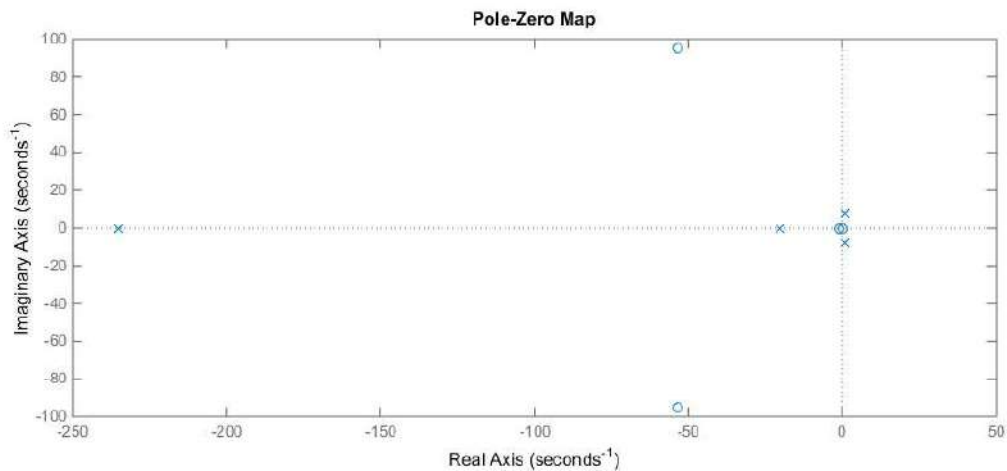
Fonte: Autoria própria.

Figura 19 – Zeros e polos operação 2 - AG



Fonte: Autoria própria.

Figura 20 – Zeros e polos operação 3 - AG



Fonte: Autoria própria.

Em termos de otimização, o algoritmo se mostrou eficiente, pois conseguiu de fato otimizar a norma do sistema de estudo. Contudo, há a necessidade de inserir restrições ao problema de otimização a fim de que o controlador obtido possa ser fisicamente implementável.

Após diversas análises, foi proposta uma alteração na função objetivo do algoritmo, a fim de ponderar seus polos e, com isso, permitir que os polos do sistema em malha fechada estejam próximos. A Equação (27) ilustra a nova função objetivo aplicada ao algoritmo.

$$FO = \text{Min}_{H(s)} \{ \text{norma } H_2(G(s)) - \text{Min}|\lambda_i| + \text{Max}|\lambda_i| \} \quad (27)$$

sendo:

$\text{Min}|\lambda_i|$ o polo de menor módulo do sistema em malha fechada, ou seja, é o polo menos negativo;

$\text{Max}|\lambda_i|$ o polo de maior módulo do sistema, ou seja, é o polo mais negativo.

Com a nova função objetivo devidamente apresentada ao algoritmo genético, torna-se possível implementá-lo a fim de obter controladores otimizados e que sejam passíveis de realizar o controle em malha fechada do sistema físico estudado.

Novamente, por se tratar de um algoritmo meta-heurístico, dez ciclos de trabalho são realizados para cada operação apresentados anteriormente. A Tabela 15

mostra os valores das novas funções objetivos, bem como a média e variância obtida em cada operação.

Tabela 15 – Função objetivo AG com restrição

Ciclo	Operação 1	Operação2	Operação 3
1	42,9246	16,4438	33,9808
2	62,7239	23,5019	45,6440
3	43,1701	16,5223	29,3941
4	42,9833	12,8292	28,9180
5	47,3002	18,6536	37,7846
6	62,5719	22,2855	27,9722
7	52,1413	8,7275	49,2077
8	44,1125	22,1059	46,4566
9	46,0252	8,6465	30,8605
10	60,2368	12,7053	36,7043
Média	50,4189	16,2422	36,6923
Variância	70,1406	29,7628	62,7121

Fonte: Autoria própria.

Com a função objetivo apresentada por (27) e seus respectivos valores mostrados na Tabela 15, o controlador minimizado na operação 1 apresenta uma função objetivo igual a 42,9246 e seus coeficientes são apresentados em (28).

$$\begin{aligned} \dot{x}_c &= \begin{bmatrix} -59,4138 & 4,0125 \\ -12,7890 & -38,2653 \end{bmatrix} x_c + \begin{bmatrix} 5,2727 \\ 3,0492 \end{bmatrix} y \\ u &= [4,4320 \quad 7,8228] x_c + [-0,9399] y \end{aligned} \quad (28)$$

Já para o ponto de operação 2, a menor função objetivo é igual e 8,6465, sendo o controlador correspondente expresso por (29).

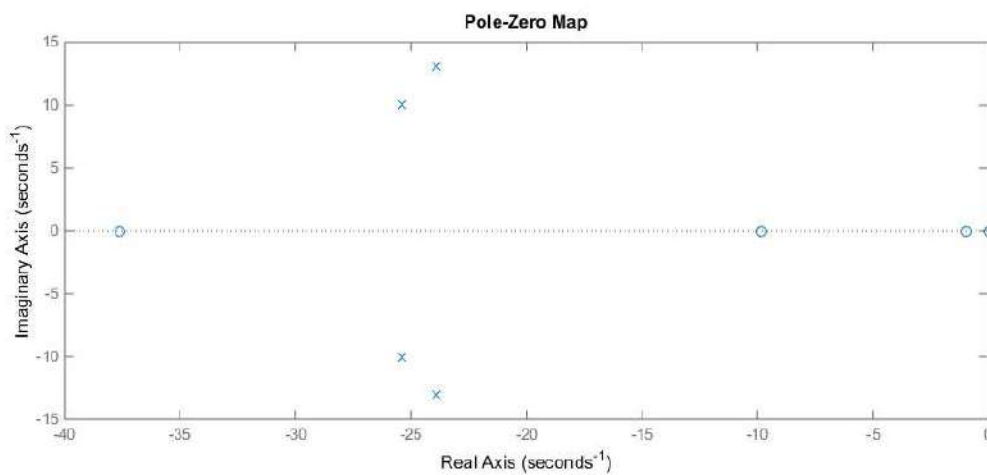
$$\begin{aligned} \dot{x}_c &= \begin{bmatrix} -19,0485 & 2,3525 \\ -13,2716 & -51,4745 \end{bmatrix} x_c + \begin{bmatrix} 6,9437 \\ 0,9005 \end{bmatrix} y \\ u &= [4,6738 \quad 12,6241] x_c + [-1,1194] y \end{aligned} \quad (29)$$

Por fim, o terceiro e último ponto de operação, apresenta um controlador otimizado conforme (30), cuja função objetivo é igual 27,9722.

$$\begin{aligned} \dot{x}_c &= \begin{bmatrix} -28,3142 & -4,5028 \\ -5,8870 & -38,3134 \end{bmatrix} x_c + \begin{bmatrix} -2,2705 \\ -6,6171 \end{bmatrix} y \\ u &= \begin{bmatrix} -6,8776 & -0,0790 \end{bmatrix} x_c + \begin{bmatrix} -0,4613 \end{bmatrix} y \end{aligned} \quad (30)$$

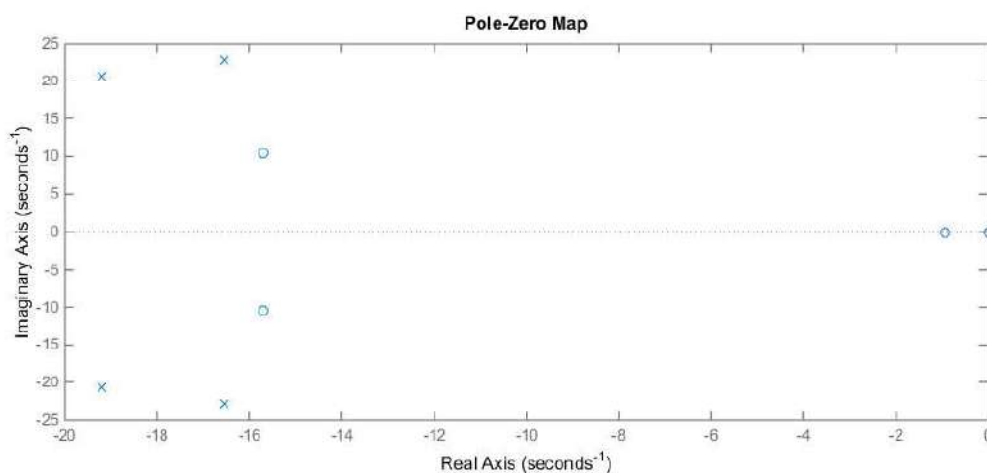
Com a nova função objetivo apresentada, é possível inferir que os polos do sistema possuem uma diferença menor entre si e, com isso, a busca pelos estados torna-se um pouco mais lenta e, assim, o sistema pode ser fisicamente implementado. As Figuras 21, 22 e 23 mostram os polos e zeros do sistema torcional em malha fechada quando são aplicados os controladores (28), (29) e (30), respectivamente.

Figura 21 – Zeros e polos operação 1 – AG com restrição



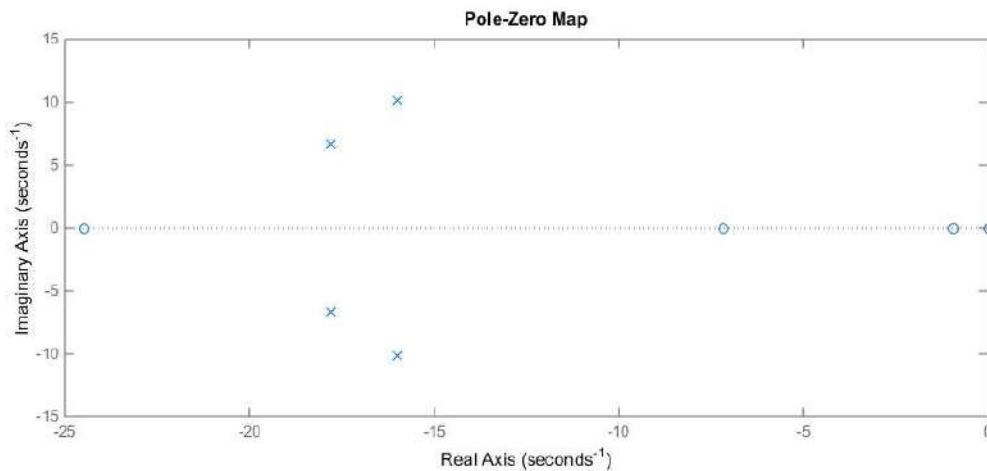
Fonte: Autoria própria.

Figura 22 – Zeros e polos operação 2 – AG com restrição



Fonte: Autoria própria.

Figura 23 – Zeros e polos operação 3 – AG com restrição



Fonte: Autoria própria.

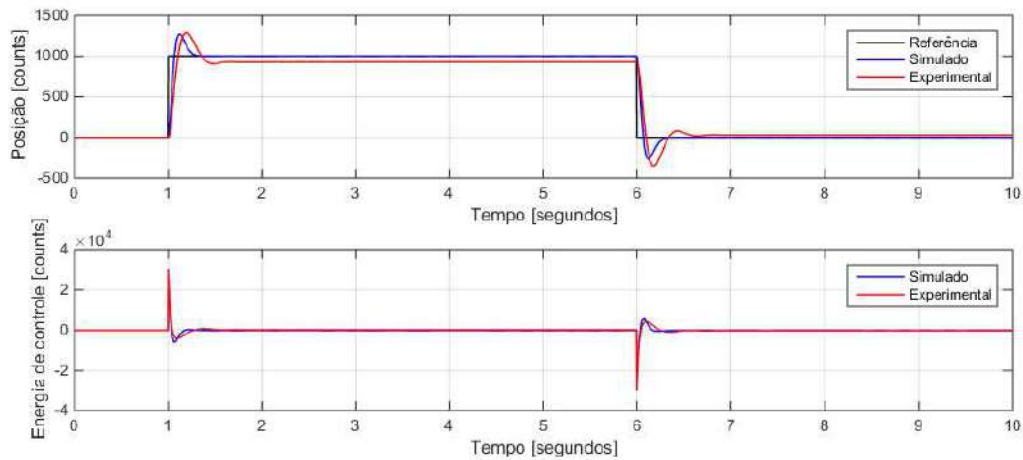
A partir dos gráficos de zeros e polos do sistema em malha fechada obtidos pela nova função objetivo em comparação com o sistema obtido pela função objetivo original, pode-se observar que todos os polos possuem parte real menos negativa. Outra característica interessante é que todos os polos dos sistemas possuem parte real e imaginária, o que caracteriza sistemas com repostas sub-amortecidas.

Com os controladores ótimos obtidos pelo algoritmo genético, iniciaram-se seus testes computacionais e experimentais, conforme apresentado no Capítulo 5, onde os controladores obtidos em cada um dos três pontos de operação foram analisados.

A fim de comprovar a validação destes controladores, aplicou-se ao sinal de entrada duas entradas distintas, sendo a degrau com amplitude de 1000 *counts* durante 5 segundos e senoidal com amplitude também de 1000 *counts* e frequência de 3 rad/s. Torna-se interessante também analisar a energia de controle do sistema em malha fechada, já que este é um dos parâmetros de otimização.

Assim, para o controlador apresentados em (28), o sinal de saída deste sistema à referência degrau, bem como, o sinal de sua energia de controle é apresentado pela Figura 24.

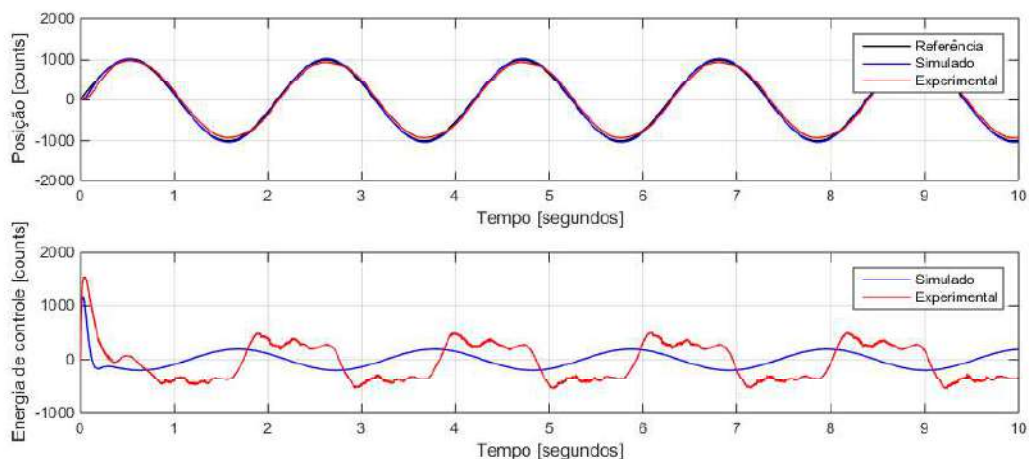
Figura 24 – Resposta ao degrau e energia de controle – operação 1 AG



Fonte: Autoria própria.

O controle conseguiu seguir a referência proposta tanto pelo sistema simulado no *Simulink*, quanto aplicado experimentalmente no sistema torcional. Nota-se um erro de 7,3% de seu valor em regime, porém, com resultados ainda satisfatórios. Para o sinal de entrada senoidal, a saída do sistema é ilustrada conforme Figura 25.

Figura 25 – Resposta senoidal e energia de controle – operação 1 AG

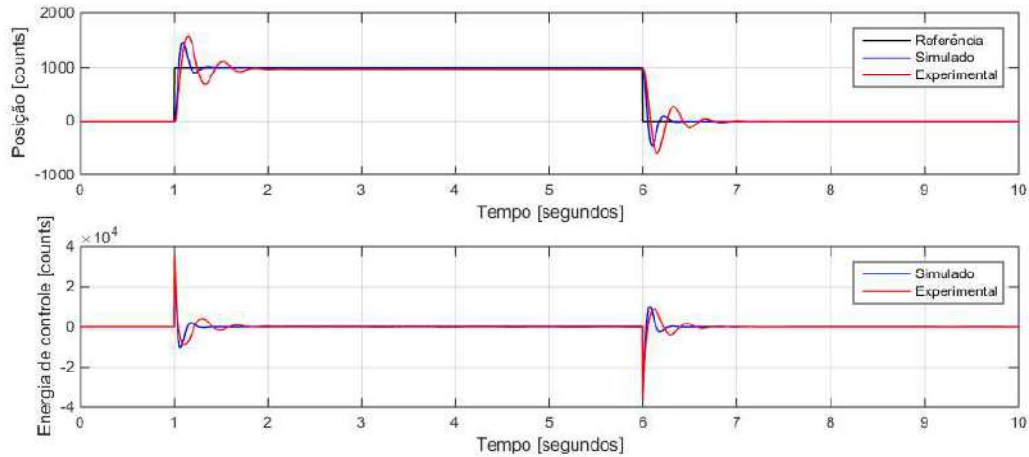


Fonte: Autoria própria.

O controle também conseguiu seguir o sinal de entrada senoidal de forma satisfatória. Percebe-se que a energia de controle experimental apresenta um atraso em relação à energia de controle simulada e também apresenta picos mais elevados, mostrando que o sistema físico necessita de maior energia para seu funcionamento.

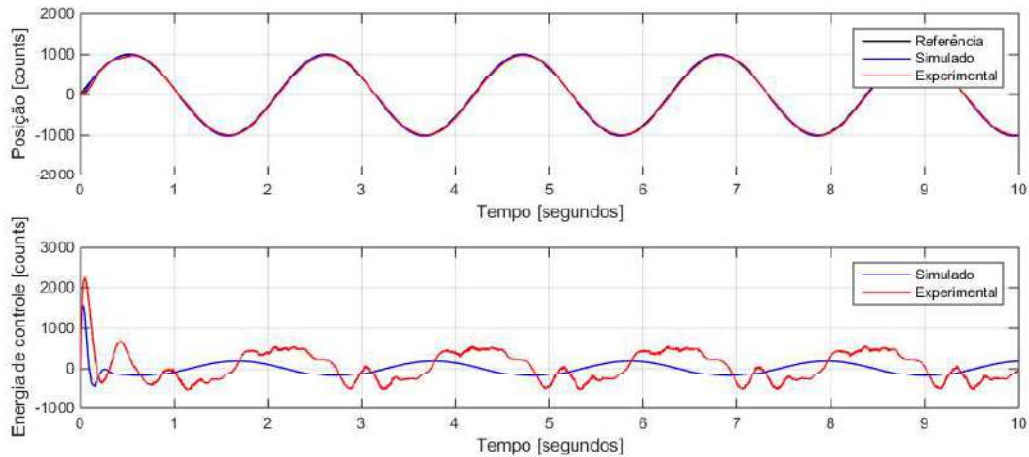
As Figura 26 e 27 mostram o sinal de saída do sistema aplicando-se o controlador (29) quando excitados por sinal de entrada degrau e senoidal, respectivamente.

Figura 26 – Resposta ao degrau e energia de controle – operação 2 AG



Fonte: Autoria própria.

Figura 27 – Resposta senoidal e energia de controle – operação 2 AG

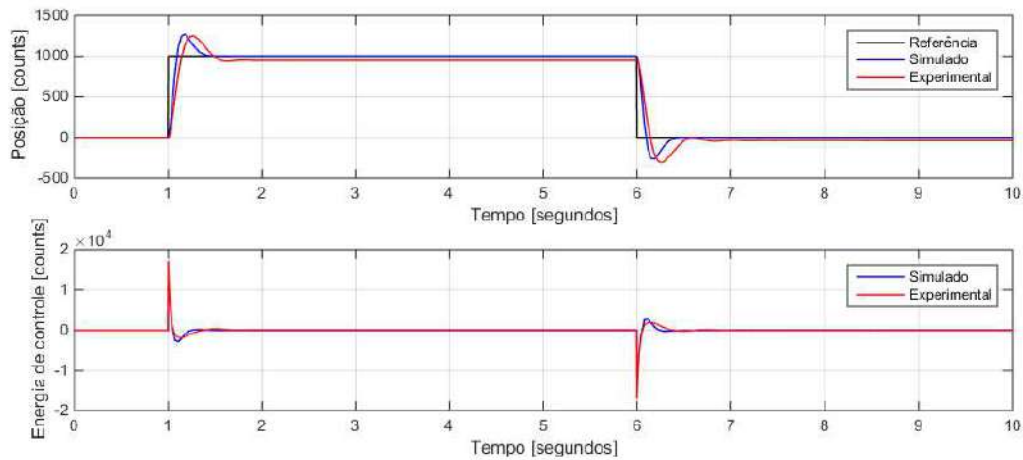


Fonte: Autoria própria.

Para ambos os sinais de entrada pode-se aferir que o controlador foi eficiente e apresentou erro em regime ainda menor do que o controlador apresentado no ponto de operação anterior, sendo em torno de 2,5% de seu valor em regime.

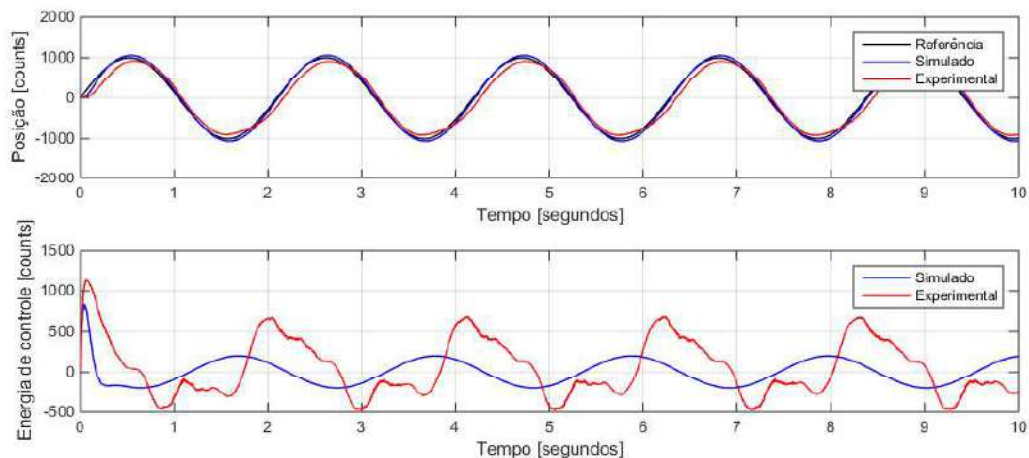
Por fim, os sinais de saída do terceiro ponto de operação obtido pelo controlador (30) quando aplicados à entrada degrau e senoidal são apresentados pelas Figura 28 e 29, respectivamente.

Figura 28 – Resposta ao degrau e energia de controle – operação 3 AG



Fonte: Autoria própria.

Figura 29 – Resposta ao senoidal e energia de controle – operação 3 AG



Fonte: Autoria própria.

Conforme as Figuras 28 e 29, tal controlador apresenta erro em regime maior do que os controladores apresentados nos pontos de operação anteriores, sendo aproximadamente 5,4% de seu valor. Contudo, o controlador se aproxima da referência considerando ambos os sinais de entrada.

Assim, os resultados obtidos pelos controladores correspondentes aos três pontos de operação estudados, mostram que o algoritmo genético, após realizar ajustes em sua função objetivo, é uma ferramenta eficiente na otimização de controladores utilizando a norma H_2 como critério de desempenho.

A seguir serão realizados e analisados os controladores obtidos pelo algoritmo de evolução diferencial, para que assim, seja possível realizar comparações entre os algoritmos.

7.2 CONTROLADORES OBTIDOS PELO ALGORITMO DE EVOLUÇÃO DIFERENCIAL

Da mesma forma que apresentado anteriormente ao algoritmo genético, o algoritmo de evolução diferencial também é uma meta-heurística e, assim, não resulta em um mínimo global, mas apresenta o melhor valor dentre os parâmetros dispostos. Portanto, também foram realizados dez ciclos de trabalho do algoritmo, ou seja, o algoritmo foi executado dez vezes, a fim de analisar a eficiência de suas respostas.

A Tabela 16 apresenta os valores da norma H_2 obtida, bem como a média e variância de seus valores para cada operação estipulados na Tabela 1.

Tabela 16 – *Fitness* ED

Ciclo	Operação 1	Operação 2	Operação 3
1	3,1714	0,7168	0,4961
2	3,3236	0,2317	0,5750
3	2,7575	0,3646	0,4601
4	5,7322	0,1669	0,3276
5	3,6969	0,5291	0,4891
6	4,8901	0,1731	0,4847
7	3,7803	0,1823	0,5218
8	2,7903	0,3477	0,4100
9	3,8617	0,2115	0,4488
10	5,7049	0,2506	0,4776
Média	3,9701	0,3174	0,4691
Variância	1,2225	0,0325	0,0044

Fonte: Autoria própria.

Novamente, as análises são realizadas de acordo com os pontos de operação estipulados, pois estes apresentam ao sistema condições de funcionamento distintas.

Assim, as análises apresentadas a seguir estão relacionadas com o melhor controlador de cada operação de estudo, sendo que para o ponto de operação 1 a menor norma resultante é de 2,7575 e seu controlador é apresentado em (31).

$$\begin{aligned} \dot{x}_c &= \begin{bmatrix} -124,6111 & -28,0893 \\ 118,4817 & -82,7889 \end{bmatrix} x_c + \begin{bmatrix} -0,1243 \\ -3,2503 \end{bmatrix} y \\ u &= [11,4880 \quad 1,1488] x_c + [-0,0338] y \end{aligned} \quad (31)$$

O segundo ponto de operação analisado possui a menor norma igual 0,1669 e seu controlador para estas condições é expresso por (32).

$$\begin{aligned} \dot{x}_c &= \begin{bmatrix} -84,4568 & -217,8057 \\ 12,8400 & -51,6973 \end{bmatrix} x_c + \begin{bmatrix} 2,4296 \\ -0,1242 \end{bmatrix} y \\ u &= [0,9276 \quad 7,1141] x_c + [-0,0411] y \end{aligned} \quad (32)$$

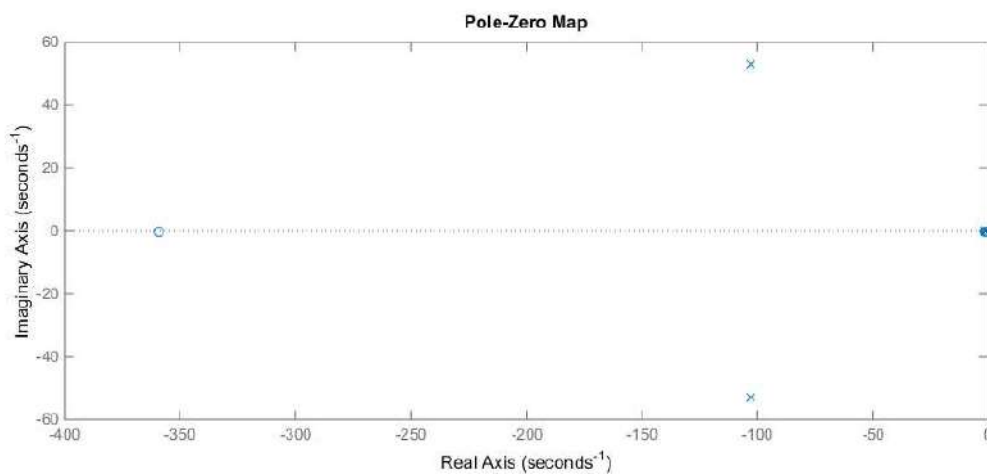
Por fim, o terceiro e último ponto de operação analisado, apresenta um controlador otimizado, dentro dos parâmetros apresentados, conforme (33). A norma apresentada pelo sistema nestas condições é 0,3276.

$$\begin{aligned} \dot{x}_c &= \begin{bmatrix} -73,8624 & 94,6685 \\ 60,1210 & -120,6508 \end{bmatrix} x_c + \begin{bmatrix} 0,6432 \\ -0,4499 \end{bmatrix} y \\ u &= [0,5016 \quad -0,3759] x_c + [-0,0053] y \end{aligned} \quad (33)$$

Conforme o ocorrido com os controladores sintetizados pelo algoritmo genético, os controladores resultantes do algoritmo de evolução diferencial também não acionaram o atuador do sistema torcional de estudo. Como era de se esperar, tais controladores também possuem polos com a parte real muito negativa e, como é possível observar, os valores de suas normas são ainda menores, ou seja, a energia de controle é muito baixa para tirar o disco da inércia.

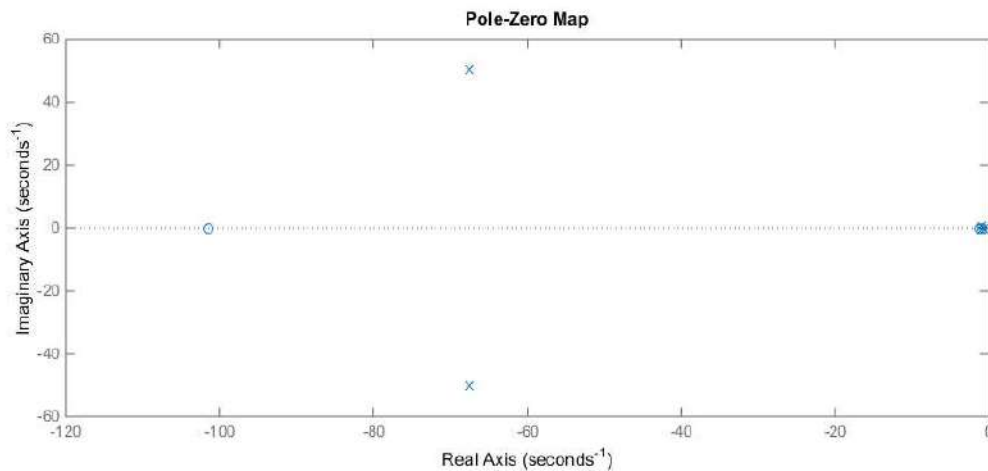
Os zeros e polos dos controladores (31), (32) e (33) são ilustrados pelas Figuras 30, 31 e 32, respectivamente.

Figura 30 – Zeros e polos operação 1 - ED



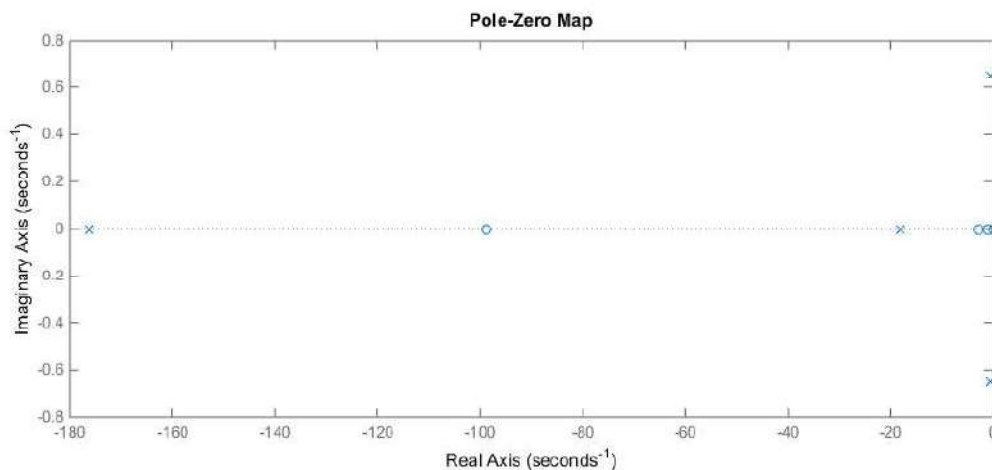
Fonte: Autoria própria.

Figura 31 – Zeros e polos operação 2 - ED



Fonte: Autoria própria.

Figura 32 – Zeros e polos operação 3 - ED



Fonte: Autoria própria.

Apesar do algoritmo de evolução diferencial se mostrar mais eficiente do que o algoritmo genético, pois apresenta controladores que resultam em menores normas, este algoritmo também precisa de restrições para que seus resultados possam ser fisicamente implementados. A alteração atribuída ao algoritmo de evolução diferencial, após sucessivos testes, também foi sua função objetivo da mesma forma que apresentada em (27).

Todos os testes do algoritmo de evolução diferencial são refeitos, agora com a nova função objetivo. Os valores da FO dos dez ciclos de trabalho nos três pontos de operação analisados estão disponíveis na Tabela 17.

Tabela 17 – *Fitness* ED com restrição

Ciclo	Operação 1	Operação2	Operação 3
1	56,1406	9,8849	25,6131
2	61,0014	8,0925	44,6956
3	43,5720	8,3572	30,3831
4	53,2591	6,2671	32,4253
5	40,2840	14,0937	26,3522
6	60,6776	5,7430	44,1091
7	45,4087	10,3917	48,3613
8	76,8870	5,8554	50,0333
9	39,1924	5,1162	59,1019
10	59,7548	6,0642	52,3335
Média	53,6178	7,9866	41,3408
Variância	138,2400	7,9223	139,0708

Fonte: Autoria própria.

Com a nova função objetivo expressa por (27) e seus valores apresentados na Tabela 17, nota-se que o controlador ótimo do sistema em malha fechada aplicado ao ponto de operação 1 possui *FO* igual a 39,1924 e apresenta seus coeficientes conforme (34).

$$\begin{aligned} \dot{x}_c &= \begin{bmatrix} -34,7649 & -0,3702 \\ -73,2970 & -11,9914 \end{bmatrix} x_c + \begin{bmatrix} -1,4291 \\ -0,2840 \end{bmatrix} y \\ u &= [-6,5510 \quad -0,1389] x_c + [-0,3501] y \end{aligned} \quad (34)$$

Já para o ponto de operação 2, o menor *fitness* do algoritmo é igual e 5,1162, sendo o controlador correspondente expresso por (35).

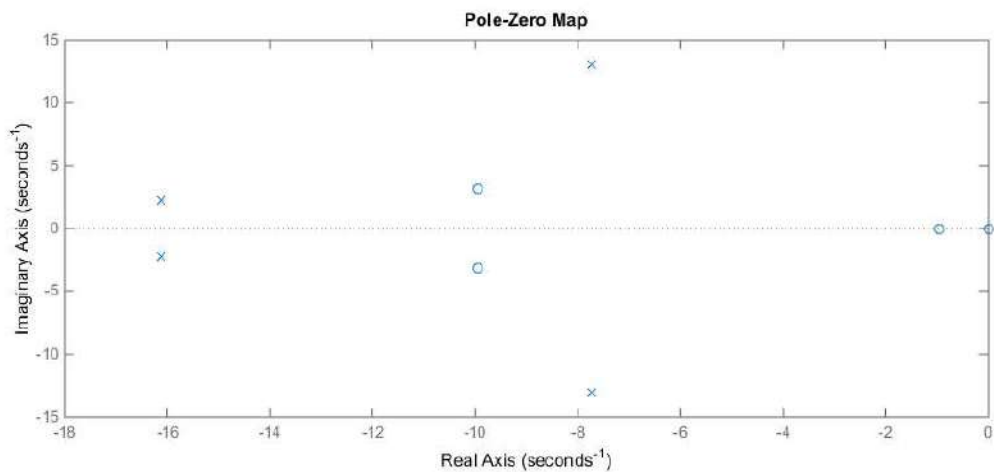
$$\begin{aligned} \dot{x}_c &= \begin{bmatrix} -30,2376 & 7,9814 \\ -34,2891 & -39,2987 \end{bmatrix} x_c + \begin{bmatrix} -1,0005 \\ 3,8490 \end{bmatrix} y \\ u &= [5,4532 \quad 3,1185] x_c + [-0,3906] y \end{aligned} \quad (35)$$

Por fim, o terceiro e último ponto de operação, apresenta um controlador otimizado conforme (36), cuja função objetivo é igual 25,6131.

$$\begin{aligned} \dot{x}_c &= \begin{bmatrix} -18,5071 & -27,6530 \\ 12,5730 & -29,4228 \end{bmatrix} x_c + \begin{bmatrix} 0,8634 \\ 2,1820 \end{bmatrix} y \\ u &= [-2,2125 \quad 0,3104] x_c + [-0,1884] y \end{aligned} \quad (36)$$

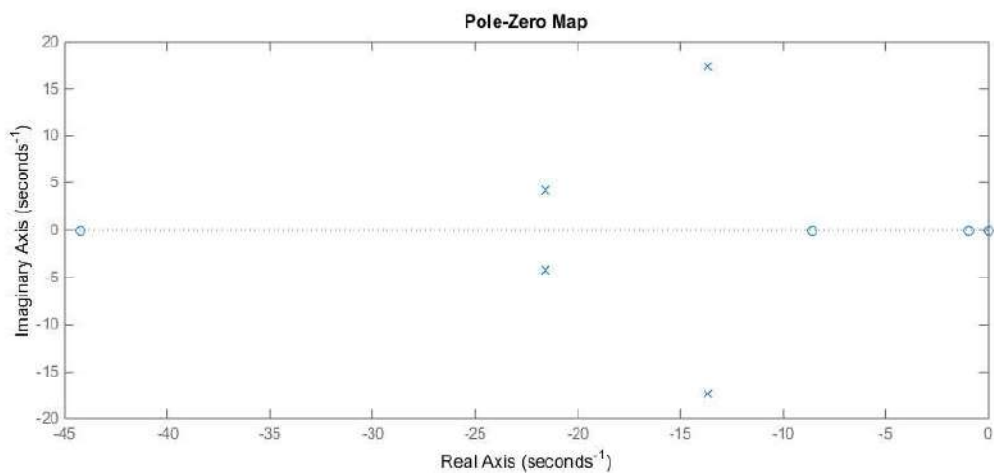
Com a função objetivo apresentada, sabe-se que os polos do sistema são mais próximos entre si, ou seja, não possuem valores reais muito negativos, mas também não são próximos de zero. Isso dá ao sistema um equilíbrio entre o tempo de resposta dos estados e sua energia de controle, permitindo que o sistema de controle seja implementado. As Figuras 33, 34 e 35 mostram os zeros e polos do sistema em malha fechada quando são aplicados os controladores (34), (35) e (36), respectivamente.

Figura 33 – Zeros e polos operação 1 – ED com restrição



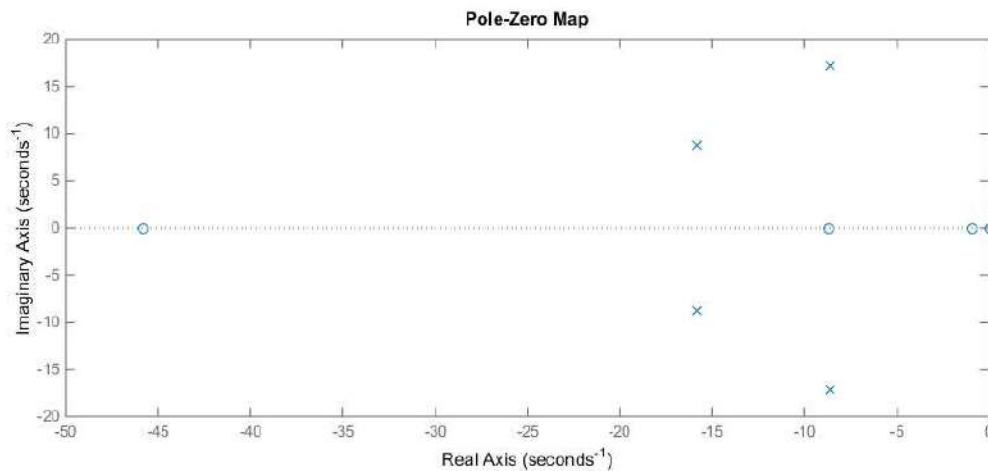
Fonte: Autoria própria.

Figura 34 – Zeros e polos operação 2 – ED com restrição



Fonte: Autoria própria.

Figura 35 – Zeros e polos operação 3 – ED com restrição



Fonte: Autoria própria.

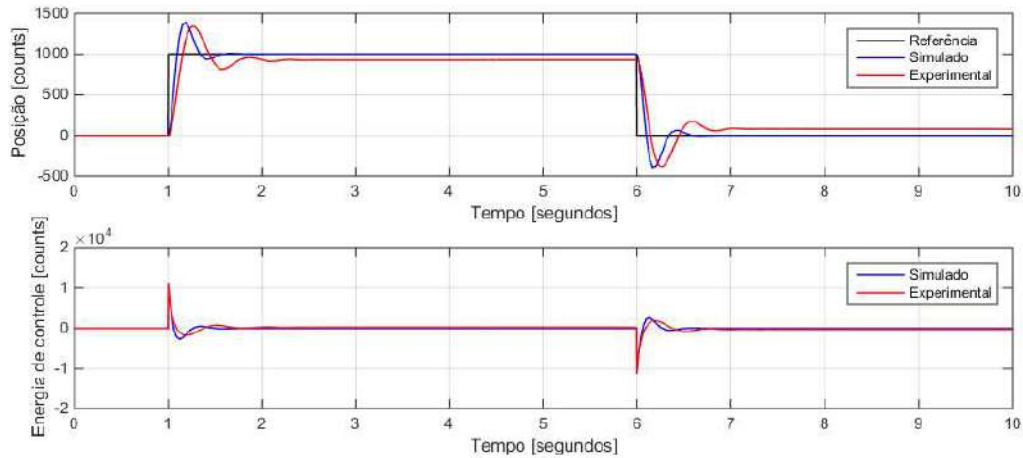
Ao se comparar os gráficos de zeros e polos apresentados nas Figuras 30, 31 e 32 com os das Figuras 33, 34 e 35 nota-se que os polos críticos do sistema em malha fechada estão menos negativos e todos possuem parte imaginária diferente de zero. Tais características geram sistemas com respostas sub-amortecidas e são as mesmas características apresentadas pelos controladores obtidos no algoritmo genético.

Os métodos computacionais e experimentais utilizados aqui são os mesmos já apresentados anteriormente, sendo que o controlador que resulta em menor *fitness* em cada ponto de operação será analisado a fim de validar os resultados obtidos pelo algoritmo de evolução diferencial.

Da mesma forma, os sinais de entrada degrau e senoidal foram aplicados ao sistema em malha fechada, primeiramente obteve-se seu sinal de saída simulado pelo *software Simulink* e posteriormente obteve-se a saída em malha fechada do sistema físico, sendo que o *software Simulink* também é utilizado nesta etapa, porém apenas como interface de comunicação.

Assim, para o controlador apresentados em (34), o sinal de saída deste sistema à referência degrau, bem como, o sinal de sua energia de controle é apresentado pela Figura 36.

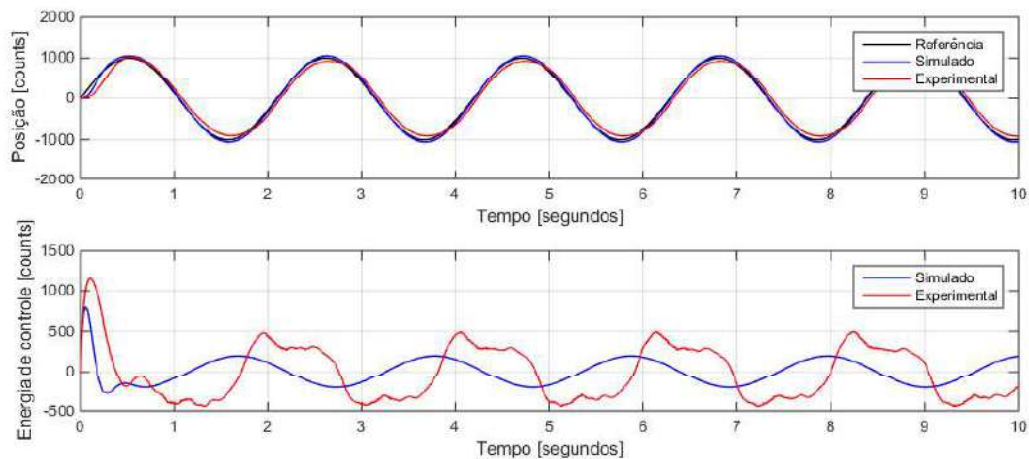
Figura 36 – Resposta ao degrau e energia de controle – operação 1 ED



Fonte: Autoria própria.

A resposta simulada não apresenta erro em regime permanente, contudo, o sinal de saída experimental mostra um erro considerável de aproximadamente 7,4% do seu valor em regime. Para o sinal de entrada senoidal, a saída do sistema é ilustrada conforme Figura 37.

Figura 37 – Resposta senoidal e energia de controle – operação 1 ED

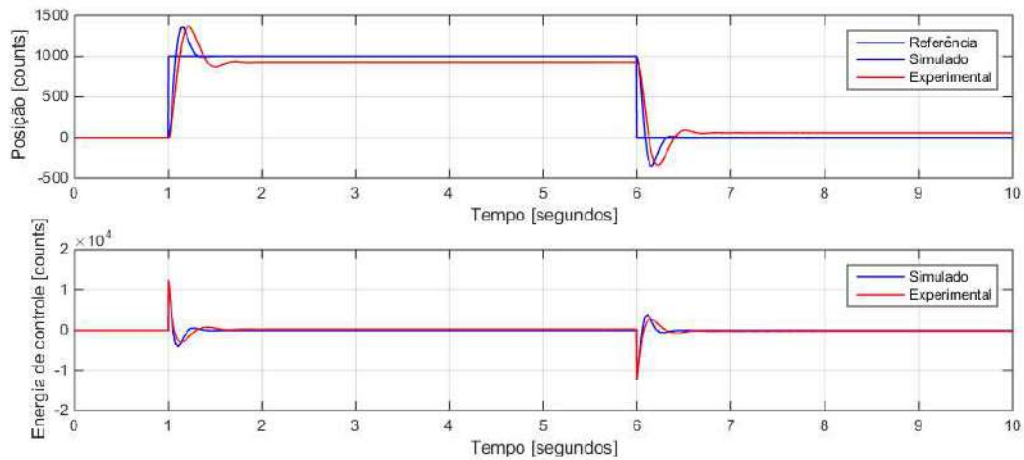


Fonte: Autoria própria.

O controle também conseguiu seguir o sinal de entrada senoidal de forma satisfatória. Percebe-se que a energia de controle experimental apresenta um atraso em relação a energia de controle simulada e também apresenta picos mais elevados.

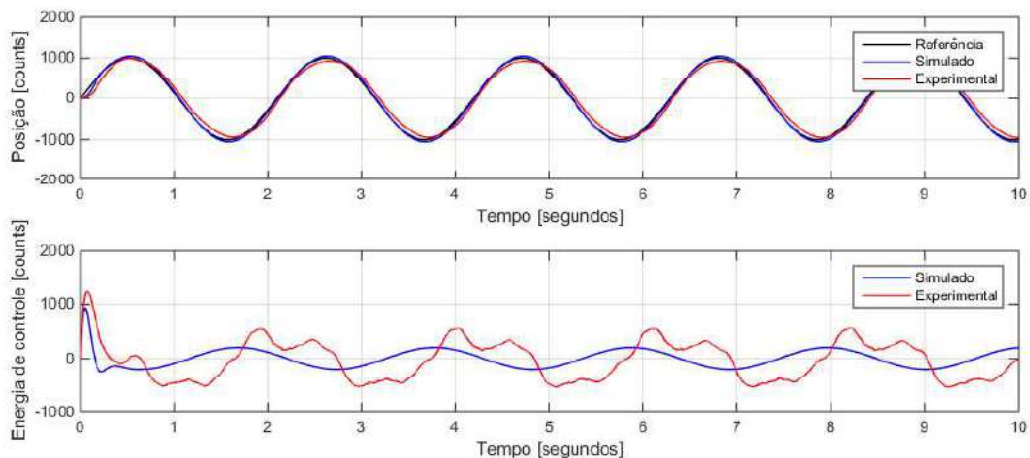
As Figura 38 e 39 mostram o sinal de saída do sistema aplicando-se o controlador (35) quando excitados por sinal de entrada degrau e senoidal, respectivamente.

Figura 38 – Resposta ao degrau e energia de controle – operação 2 ED



Fonte: Autoria própria.

Figura 39 – Resposta senoidal e energia de controle – operação 2 ED

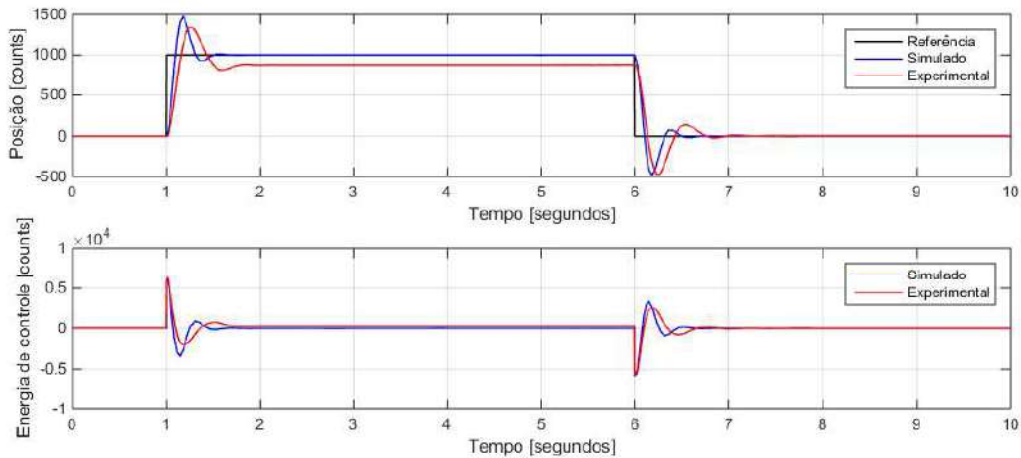


Fonte: Autoria própria.

Para ambos os sinais de entrada pode-se aferir que o controlador foi eficiente, porém, apresenta erro em regime maior do que o apresentado no ponto de operação anterior, sendo aproximadamente 8,2% de seu valor em regime.

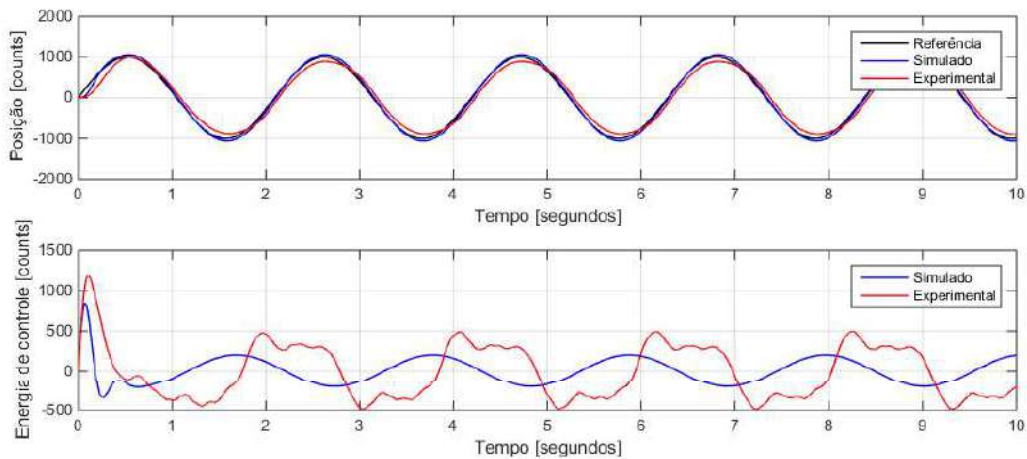
Por fim, os sinais de saída do terceiro ponto de operação obtido pelo controlador (36) quando aplicados à entrada degrau e senoidal são apresentados pelas Figura 40 e 41, respectivamente.

Figura 40 – Resposta ao degrau e energia de controle – operação 3 ED



Fonte: Autoria própria.

Figura 41 – Resposta senoidal e energia de controle – operação 3 ED



Fonte: Autoria própria.

Conforme as Figuras 40 e 41, tal controlador apresenta erro em regime maior do que os controladores apresentados nos pontos de operação anteriores, em torno de 12,5%, contudo os sinais se mantêm próximos da referência esperada.

7.3 COMPARAÇÕES ENTRE AG E ED

Alguns quesitos podem ser utilizados para comparar os algoritmos genético e de evolução diferencial, entre eles o tempo computacional, valor da função objetivo, erro em regime.

Tais quesitos são apresentados na Tabela 18 para ambos algoritmos evolutivos desenvolvidos neste trabalho, é importante observar que os dois

apresentam suas vantagens e desvantagens, porém ambos se mostram boas ferramentas para a síntese de controladores.

Tabela 18 – Comparações entre AG e ED

	Algoritmo genético	Algoritmo de evolução diferencial
Função objetivo – Operação 1	49,9246	39,1924
Função objetivo – Operação 2	8,6465	5,1162
Função objetivo – Operação 3	27,9722	25,6131
Erro em regime – Operação 1	7,3%	7,4%
Erro em regime – Operação 2	2,5%	8,2%
Erro em regime – Operação 3	5,4%	12,5%
Tempo computacional	18,7 minutos	6,2 minutos

Fonte: Autoria própria.

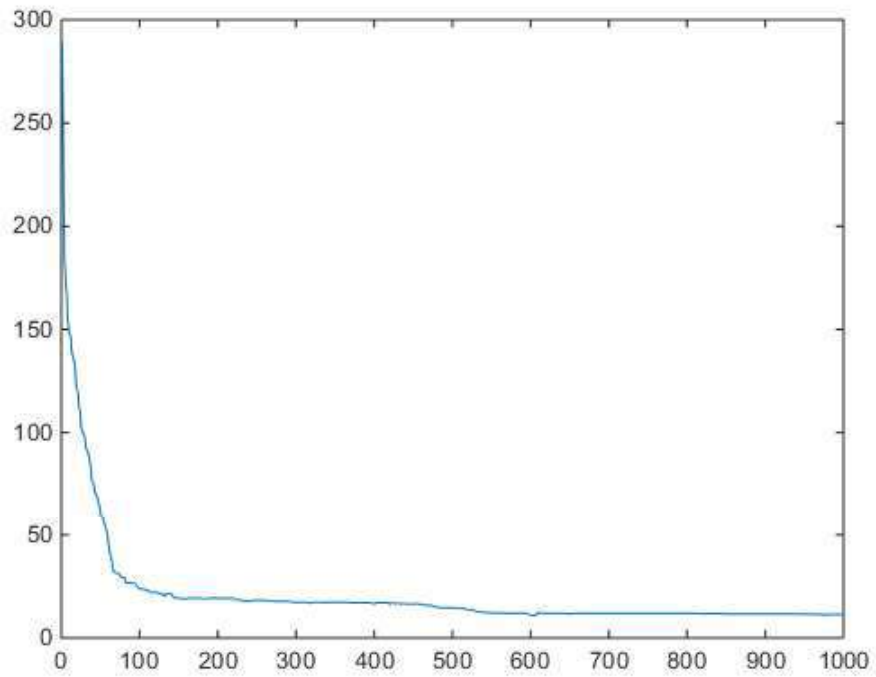
O erro em regime dos sistemas físicos apresentados pode estar sendo causado devido a inconsistências no modelo definido da planta torcional. Este tipo de planta pode ter seus parâmetros alterados com o tempo devido a desgastes e esforços.

Os valores das funções objetivo referentes aos controladores ótimos obtidos pelo algoritmo de evolução diferencial são menores do que as funções objetivo referentes ao algoritmo genético. Em contrapartida, os sinais de saída em malha fechada resultante dos controladores obtidos pelo ED apresentam erros de regime maiores do que os sistemas resultantes dos controladores obtidos pelo AG.

Outro ponto possível para a comparação entre os algoritmos evolutivos é o decaimento da sua função objetivo ao longo das gerações. Conforme mencionado anteriormente, o valor máximo de gerações atribuído foi 1000, porém, não há a necessidade de se executar as 1000 gerações para obter o valor de FO próximo ao ótimo.

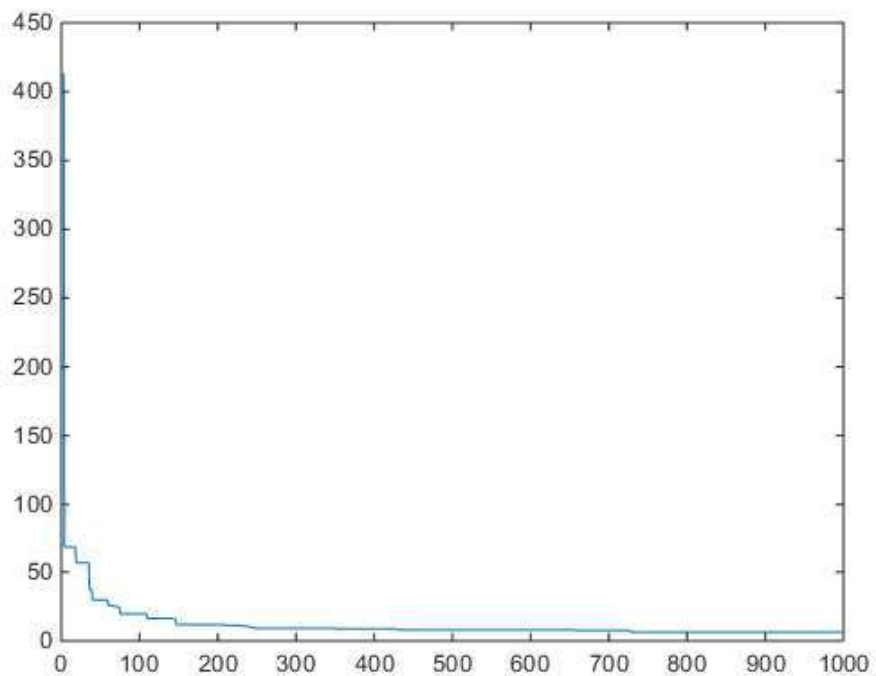
As Figuras 42 e 43 apresentam o decaimento da função objetivo apresentada em (27) para o algoritmo genético e para o algoritmo de evolução diferencial, respectivamente. Os decaimentos apresentados são referentes ao segundo ponto de operação, com as matrizes ponderadas de C_z e D_z .

Figura 42 – Decaimento do valor FO a cada iteração do AG – operação 2



Fonte: Autoria própria.

Figura 43 – Decaimento do valor FO a cada iteração do ED – operação 2



Fonte: Autoria própria.

Nota-se que o decaimento apresentado no algoritmo de evolução diferencial é mais rápido do que o apresentado no algoritmo genético, no AG são

necessárias aproximadamente 600 gerações para se obter um valor próximo ao ótimo, já no ED são necessárias em torno de 300 gerações. Este fato era esperado, devido as técnicas de seus operadores genéticos, especialmente a seleção que determina o melhor cromossomo entre o inicial e o já alterado geneticamente.

Por fim, o valor da função objetivo, o erro em regime do sistema físico e o tempo computacional, apresentados na Tabela 18, estão relacionados ao custo dos algoritmos, sendo que ao se observar desta forma o ED se mostra mais vantajoso do que o AG. Além de apresentar funções objetivo minimizadas e tempo computacional 3 vezes menor, o ED também possui menos parâmetros a serem sintonizados em comparação com o AG, o que também otimiza o processo de busca dos parâmetros.

7.3 CONCLUSÃO DO CAPÍTULO

Neste capítulo foram apresentados os resultados obtidos pelos algoritmos genético e de evolução diferencial, bem como suas restrições para que estes apresentem controladores que possam ser fisicamente implementados. Foram apresentados também os principais controladores e seus sinais de saída em malha fechada correspondentes à sinais de entrada degrau e senoidal.

Por fim, no Capítulo 8 são apresentadas as considerações a respeito dos resultados obtidos e também as propostas para trabalhos futuros.

8 CONSIDERAÇÕES FINAIS

Este trabalho propôs a utilização de dois métodos de otimização aplicados a sistemas de controle, sendo eles o algoritmo genético e o algoritmo de evolução diferencial. Ambos algoritmos objetivaram o cálculo de controladores que resultassem na minimização da norma H_2 de sistemas de controle.

Os algoritmos se mostraram ferramentas eficientes de otimização, contudo, seus controladores obtidos pela minimização da norma H_2 resultaram em sistemas que não foram passíveis de implementação física, pois a energia de controle resultante era tão pequena que não permitia que o sistema físico implementado saísse da inércia. Assim, mostrou-se necessário inserir aos algoritmos evolutivos restrições em sua função objetivo e, com isso, além de minimizar a norma H_2 , também foi necessário otimizar os polos do sistema em malha fechada.

Apesar das restrições inseridas e custos apresentados aos sistemas, os algoritmos genéticos e de evolução diferencial se mostram eficientes alternativas às soluções algébricas como LMIs e de Riccati para sistemas não-convexos.

Conforme mencionado neste trabalho, alguns pontos de operação foram analisados para diferentes configurações relacionando a resposta dos estados e a energia de controle. Com os resultados obtidos, notou-se que ao considerar a mesma ponderação entre a energia de controle e velocidade de reposta obteve-se o menor valor da função objetivo e também respostas com menor erro em regime. Assim, torna-se interessante aplicar as matrizes C_z e D_z de forma igualitária na otimização de controladores a partir de algoritmos evolutivos, conforme o disposto no ponto de operação descrito.

Algumas comparações são realizadas entre os algoritmos evolutivos analisados, como o tempo de convergência, o valor da função objetivo e o erro em regime dos sinais de saída em malha fechada.

O tempo de convergência entre ambos algoritmos são muito discrepantes, sendo que o algoritmo genético leva em média 18,7 minutos para sua convergência, enquanto que o algoritmo de evolução diferencial demora em média 6,2 minutos. Mesmo que os algoritmos sejam implementados de forma *offline*, o esforço computacional apresentado no AG é três vezes maior do que o apresentado pelo ED.

O algoritmo de evolução diferencial também apresenta menores valores médios de sua função objetivo, em comparação com o algoritmo genético, o que o caracteriza como uma melhor ferramenta de otimização para o sistema estudado, apenas os controladores obtidos pelo AG apresentam um sinal de saída com menor erro em regime do que os controladores obtidos pelo ED.

Portanto, pode-se concluir que os algoritmos genéticos e de evolução diferencial são eficientes ferramentas aplicadas na otimização de controladores, apesar das restrições necessárias. Cabe ao usuário conhecer as aplicações do sistema, a fim de definir qual dos dois algoritmos deve ser utilizado.

8.1 TRABALHOS FUTUROS

Propõe-se para trabalhos futuros a investigação de novas penalidades na modelagem dos algoritmos evolutivos a fim de anular o erro em regime dos sinais de saída do sistema em malha fechada. Há também a possibilidade de adaptar os algoritmos a fim de sintetizar controladores aplicados em sistemas afetados por incertezas politópicas.

8.2 PRODUÇÃO CIENTÍFICA

RODRIGUES, Ana Beatriz, AGULHARI, Cristiano Marcos. Algoritmos genéticos e de evolução diferencial aplicados à otimização de controladores utilizando a norma H_2 como critério de desempenho. 14º Simpósio Brasileiro de Automação Inteligente, 2019, Ouro Preto – Brasil. Trabalho submetido.

REFERÊNCIAS

- ALI, R. S., AL-THUWAINY, A. A. K. **Designing robust mixed H_2/H_∞ PID controllers based intelligent genetic algorithm**. 1st International Conference on Energy, power and Control, 2010, Basrah – Iraq.
- ANDRADE, L. H. S. de, AGULHARI, C. M. **Robust mixed H_2/H_∞ control of an nonlinear multivariable system**. 12th IEEE/IAS International Conference on Industry Application, 2016, Curitiba – Brasil.
- BOERS, Y., WEILAND, S., DAMEN, A. A. H. **Optimal expected H_2 performance controllers**. European Control Conference, 1997, Brussels – Belgium.
- BOSCO, M. C., GUEDES, J. J., CASTOLDI, M. F., GOEDEL, A., SILVA, E. R. P. da, BRUZACHERO, L. F. S. **Estimation of parameters and tuning of a speed PI of permanent magnet DC motor using differential evolution**. 12th IEEE/IAS International Conference on Industry Application, 2016, Curitiba – Brasil.
- BOYD, S., GHAOVI, L. E., FERON, E., BALAKRISHNAN, V. **Linear matrix inequalities in system and control theory**. Society for industrial and applied mathematics, Philadelphia, 1994.
- CHUN, S., CHOI, G., KIM, T. **Design of mixed static/dynamic output feedback controller using multi-objective distribute particle swarm optimization**. 11th Asian Control Conference, 2017, Gold Coast Convention Centre – Australia.
- DAS, S., SUGARATNAM, P. N. **Differential evolution: a survey of the state-of-the-art**. IEEE transactions on evolutionary computation, vol. 15, n. 1, pp. 4-31, 2011.
- DRAGAN, V., MOROZAN, T., STOICA, A. **H_2 optimal control for linear stochastic systems**. Automatica, vol. 40, pp. 1103-1113, 2004.
- EMAM, M., FRAKHARIAN, A. **Attitude tracking of quadrotor UAV via mixed H_2/H_∞ controller: an LMI based approach**. 24th Mediterranean Conference on Control and Automation, 2016, Athens – Greece.
- FARIAS, D. P. de, GEROMEL, J. C., VAL, J. B. R. do, COSTA, O. L. V. **Output feedback control of Markov jump linear systems in continuous-time**. IEEE transactions on Automatic Control, vol. 45, pp. 944-949, 2000.
- GÄMPELLE, R., MÜLLER, S. D., KOUMOUNTSAKOS, P. **A parameter study for differential evolution**. Advanced in Intelligent Systems, Fuzzy Systems, Evolutionary Computation. Switzerland: WSEAS Press, pp. 293-298, 2002.
- GHAOUI, L. E., NICULESCU, S.-L. **Advances in linear matrix inequality methods in control**. Society for industrial and applied mathematics, Philadelphia, 2000.
- GOMEZ, M. A., EGOROV, A. V., MONDIÉ, S., MICHIELS, W. **Optimization of the H_2 norm for single delay systems, with application to control design and model approximation**. Journal of LATEX class files, vol. 14, n. 8, pp. 1-8, 2015.

GONÇALVES, E. N., PALHARES, R. M., TAKAHASHI, R. H. C. **Multi-objective optimization applied to robust H_2/H_∞ state-feedback control synthesis**. American Control Conference, 2004, Boston – EUA.

GUIMARÃES, F. G., PALHARES, R. M., CAMPELO, F., IGARASHI, H. **Design of mixed H_2/H_∞ control systems using algorithms inspired by the immune system**. Information Sciences, vol. 177, pp. 4368-4386, 2007.

KAO, Y., ZAHARA, E. **A hybrid genetic algorithm and particle swarm optimization for multimodal functions**. Applied Soft Computing, vol. 8, pp. 849-857, 2007.

KHALIL, H. K. **Nonlinear Systems**. 2 ed. Prentice Hall, Upper Saddle River, 1996, New Jersey – USA.

KUCERA, V. **The H_2 control problem**: state-space and transfer-function solutions. 14th Mediterranean Conference on Control and Automation, 2006, Ancona – Italy.

LACERDA, E. G. M. de, CARVALHO, A. C. P. F. L. de. **Introdução aos algoritmos genéticos**. Editora da Universidade da UFRGS – ABRH, 1999, Porto Alegre – Brasil.

LEIBFRTZ, F. **An LMI-based algorithm for designing suboptimal static H_2/H_∞ output feedback controllers**. SIAM J. Control Optim, vol. 39, no. 6, pp. 1711-1735, 2001.

LINDEN, R. **Algoritmos genéticos**. 1 ed. Ei Edições Inteligentes, 2008, São Paulo – Brasil.

OGATA, K., MAYA, P. A., LEONARDI, F. **Engenharia de controle moderno**. 3 ed. Prentice Hall, 2010.

OUYANG, P., PANO, V. **Comparative study of DE, PSO and GA for position domain PID controller tuning**. Algorithms, Multidisciplinary Digital Publishing Institute, v. 8, n. 3, pp. 697-711, 2015.

PARKS, T. R. **Manual for Model 205/205a**: Torcional control system. Educational Control Products, 1999, Bell Canyon – EUA.

PEREIRA, G. J., ARAÚJO, H. X. de. **Robust output feedback controller design via genetic algorithms and LMIs**: the mixed H_2/H_∞ problem. American Control Conference, 2004, Boston – EUA.

PEZZELLA, F., MORGANTI, G., CIASCETTI, G. **A genetic algorithm for the flexible job-shop scheduling problem**. Computers & Operations Research, vol. 35, pp. 3202-3212, 2008.

PRICE, K., STORN, R. M., LAMPINEN, J. A. **Differential evolution**: a practical approach to global optimization. Springer Science & Business Media, 2006.

QIN, A. K., HUANG, V. L., SUGANTHAM, P. N. **Differential evolution algorithm with strategy adaptation for global numerical optimization**. IEEE transactions on evolutionary computation, vol. 13, n. 2, pp. 398-417, 2009.

ROBERGE, V., TARBOUCHI, M., LABONTÉ, G. **Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning**. IEEE transactions on industrial informatics, vol. 9, n. 1, pp. 132-141, 2013.

RONKKONEN, J., KUKKONEN, S., PRICE, K. **Real parameter optimization with differential evolution**. IEEE CEC, vol. 1, pp. 506-513, 2005.

SALIM, R., BETTAYEB, M. **H_2 and H_∞ optimal model reduction using genetic algorithms**. Journal of the Franklin Institute, vol. 348, n. 2011, pp. 1177-1191, 2009.

SATO, M., EBIHARA, Y., PEAUCELLE, D. **Gain-scheduled state-feedback controllers using inexact measured scheduling parameter: H_2 and H_∞ problems**. American Control Conference, 2010, Baltimore – EUA.

SCHERER, C. W. **Mixed H_2/H_∞ control for time-varying and linear parametrically-varying systems**. International Journal of Robust and Nonlinear Control, vol. 6, pp. 929-952, 1996.

STORN, R., PRICE, K. **Differential evolution: A simple and efficient adaptive scheme for global optimization over continuous spaces**. Journal of Global Optimization, vol. 11, pp. 341-359, 1997.

TROFINO, A., COUTINHO, D. F., BARBOSA, K. A. **Improved H_2 and H_∞ conditions for robust analysis and control synthesis of linear systems**. Revista Controle e Automação, vol. 16, n. 4, pp. 427-434, 2005.

WU, L., WANG, Y., SHOU, S., YUAN, X. **Design of mixed H_2/H_∞ optimal control systems using multi-objective differential evolution algorithm**. J Control Theory Appl, vol. 11, n. 3, pp. 521-528, 2013.

XIE, W., CHENG, Y. **Simultaneous H_2 design of structural and controller parameters with state-feedback**. 8th World Congress on Intelligent Control and Automation, 2010, Jian – China.

YANG, X.-S., CUI, Z., XIAO, R., GANDOMI, A. H., KARAMANOGLU, M. **Swarm Intelligence and bio-inspired computation**. 1st ed. Elsevier, 2013.

YU, J.-T. **A new static output feedback approach to the suboptimal mixed H_2/H_∞ problem**. International Journal of Robust and Nonlinear Control, vol. 14, pp. 1023-1034, 2004.

ZHANG, J., SANDERSON, A. C. **JADE: adaptive differential evolution with optimal external archive**. IEEE transactions on evolutionary computation, vol. 13, n. 5, pp. 945-958, 2009.

ZHOU, K., DOYLE, J. C., GLOVER, K. **Robust and optimal control**. Prentice Hall, Englewood Cliffs, New Jersey, 1996.