

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
COORDENAÇÃO DE ENGENHARIA ELETRONICA
CURSO DE ENGENHARIA ELETRÔNICA

JAVAN ATAÍDE DE OLIVEIRA JÚNIOR

**SINTONIA MULTIOBJETIVO DE CONTROLADOR PREDITIVO GENERALIZADO
POR NSGA-II**

TRABALHO DE CONCLUSÃO DE CURSO

TOLEDO

2018

JAVAN ATAÍDE DE OLIVEIRA JÚNIOR

**SINTONIA MULTI OBJETIVO DE CONTROLADOR PREDITIVO GENERALIZADO
POR NSGA-II**

Trabalho de conclusão de curso apresentado à disciplina Trabalho de Conclusão de Curso 2, do Curso de Engenharia Eletrônica da Coordenação de Engenharia Eletrônica – COELE – da Universidade Tecnológica Federal do Paraná – UTFPR, Campus Toledo, como requisito parcial para obtenção do título de Engenheiro Eletrônico.

Orientador: Prof. Dr. Wesley Klewerton Guez Assunção

Coorientador: Prof. Dr. Daniel Cavalcanti Jeronymo

TOLEDO

2018



Ministério da Educação
Universidade Tecnológica Federal do Paraná
Campus Toledo
Coordenação do Curso de Engenharia Eletrônica



TERMO DE APROVAÇÃO

Trabalho de Conclusão de Curso número 75

**SINTONIA MULTIOBJETIVO DE CONTROLADOR PREDITIVO
GENERALIZADO POR NSGA-II**

por

JAVAN ATAÍDE DE OLIVEIRA JÚNIOR

Esse Trabalho de Conclusão de Curso foi apresentado às **13h30 do dia 13 de junho de 2018** como **requisito parcial** para a obtenção do título de **Engenheiro Eletrônico**. Após deliberação da Banca Examinadora, composta pelos professores abaixo assinados, o trabalho foi considerado **APROVADO**.

Prof Ma. Jaqueline Vargas
UTFPR

Prof Me. Willian Douglas Ferrari Mendonça
UTFPR

Prof. Dr. Wesley Klewerton Guez Assunção
UTFPR

Prof. Dr. Fábio Risental Coutinho
UTFPR

O termo de aprovação assinado encontra-se na coordenação do curso

TOLEDO, 13 de junho de 2018

*Este trabalho é dedicado a minha mãe Andréa, que atualmente luta pela sua saúde,
pelo exemplo de fé, dedicação e amor.
E aos meus avós, Duda e Marinaura, pela história de vida e superação.*

"A cruz foi o pupito onde Deus pregou o amor"
-Augustinho

RESUMO

Os sistemas de controle estão presentes na rotina da maioria das pessoas. Por exemplo, nos elevadores há dispositivos que controlam a velocidade, aceleração e a posição, em sua essência são responsáveis por controlar uma ou parte de um determinado processo, com o objetivo de obter uma saída e desempenho desejados baseado na entrada especificada. O funcionamento e desempenho desses dispositivos está definitivamente relacionado aos parâmetros característicos de cada tipo de controlador. A escolha destes parâmetros é conhecida como sintonia. Há vários métodos de sintonia, os mais utilizados são aqueles em que os parâmetros são previamente definidos e tornam-se fixo durante todo o processo, existem outros nos quais a sintonia é feita de modo automático, *on-line*, e são voltados para processos mais complexos. Esse trabalho consiste na utilização de algoritmos de otimização multiobjetivo aplicados à sintonia dos parâmetros de um controlador preditivo generalizado, de forma *off-line*. A solução proposta para este trabalho é baseada no algoritmo genético de ordenação por não dominância (*Non-dominated Sorting Genetic Algorithm II* - NSGA-II), que é caracterizado por possuir mecanismos que realizam a otimização de múltiplos objetivos simultaneamente. A partir dos resultados pode-se observar que o NSGA-II atinge soluções satisfatórias que permitem uma melhor tomada de decisão.

Palavras-chave: *Model Predictive Control*. Parametrização. Algoritmos evolutivos .

ABSTRACT

Control systems are present in routine of most people. For example, in a lift there are a devices which control speed, acceleration, and position; where based on input they must obtain a proper output. The function and performance of such devices are strictly related to the parameters of each kind of controllers. The choosing of these parameters is known as tuning. There are several tuning methods. On one hand, the most common are those where the parameters are defined beforehand and keep fixed allong all the process. On the other hand, there are automatic parameter tuning, called on-line tuning, that are common for complex processes. This work focuses on the use of multi-objective optimization applied to the parameter tuning on a generalized predictive control. The proposed solution is based on the Non-dominated Sorting Genetic Algorithm II – NSGA-II, which is characterized by dealing with the optimization of multiple objectives simultaneously. From the results we can observe that the NSGA-II algorithm reaches promising solutions that allow a better decision making.

Keywords: Model Predictive Control. Parameters Tuning. Genetic Algorithm.

LISTA DE ILUSTRAÇÕES

Figura 1 – Conceito de controle preditivo.	21
Figura 2 – Função Quadrática $f(x) = x^2$	26
Figura 3 – Exemplo de fronteira de Pareto	26
Figura 4 – Problema de otimização multiobjetivo com dois objetivos	27
Figura 5 – Estrutura básica de um Algoritmo Genético	30
Figura 6 – Esquemática dos procedimentos do NSGA-II.	32
Figura 7 – Diagrama de blocos do algoritmo NSGA-II, descrevendo o funcionamento da etapa de seleção.	33
Figura 8 – Exemplo da cubóide formada pelo <i>Crowding Distance</i> . Os Pontos marcados em círculos preenchidos são soluções da mesma fronteira	34
Figura 9 – Sistema de controle em malha fechada	36
Figura 10 – Contorno Γ_c	37
Figura 11 – Diagrama de Nyquist	39
Figura 12 – Diagrama de blocos do algoritmo NSGA-II utilizado para a Otimização dos parâmetros do GPC.	41
Figura 13 – Entrada e Saída da Função Simulação GPC	41
Figura 14 – Fluxograma da Função Simulação GPC	42
Figura 15 – Diagrama de Blocos RST	43
Figura 16 – Entrada e Saída da Função Robustez	44
Figura 17 – Fluxograma Função Robustez	45
Figura 18 – Resposta ao degrau unitário.	49
Figura 19 – Conjunto de soluções dos experimentos multiobjetivo não ponderado	53
Figura 20 – Número de soluções localizadas na fronteira de Pareto por experimento multiobjetivo não ponderado.	54
Figura 21 – (<i>ISE vs TVC</i>) multiobjetivo não ponderado.	55
Figura 22 – (<i>ISE vs Ms</i>) multiobjetivo não ponderado.	55
Figura 23 – (<i>TVC vs Ms</i>) multiobjetivo não ponderado.	56
Figura 24 – Menor <i>ISE</i> por experimento multiobjetivo não ponderado.	57
Figura 25 – Menor <i>TVC</i> por experimento multiobjetivo não ponderado.	57
Figura 26 – Menor <i>ED</i> por experimento multiobjetivo não ponderado.	58
Figura 27 – Menor <i>Ms</i> por experimento multiobjetivo não ponderado.	58
Figura 28 – BoxPlot <i>ISE</i> multiobjetivo não ponderado.	60
Figura 29 – <i>ISE</i> : Simulação multiobjetivo não ponderado.	60
Figura 30 – BoxPlot <i>TVC</i> multiobjetivo não ponderado.	61
Figura 31 – <i>TVC</i> : Simulação multiobjetivo não ponderado	62
Figura 32 – BoxPlot <i>Ms</i> multiobjetivo não ponderado.	62

Figura 33 – M_s : Simulação multiobjetivo não ponderado.	63
Figura 34 – Conjunto de soluções dos experimentos multiobjetivo ponderados.	64
Figura 35 – Número de soluções ótimas por experimento multiobjetivo ponderado.	65
Figura 36 – (ISE vs TVC) multiobjetivo ponderado.	65
Figura 37 – (ISE vs M_s) multiobjetivo ponderado.	66
Figura 38 – (TVC vs M_s) multiobjetivo ponderado.	66
Figura 39 – Menor ISE por experimento multiobjetivo ponderado.	67
Figura 40 – Menor TVC por experimento multiobjetivo ponderado.	68
Figura 41 – Menor ED por experimento multiobjetivo ponderado.	68
Figura 42 – Menor M_s por experimento multiobjetivo ponderado.	69
Figura 43 – BoxPlot ISE multiobjetivo ponderado.	70
Figura 44 – ISE : Simulação multiobjetivo ponderado	70
Figura 45 – BoxPlot TVC multiobjetivo ponderado.	71
Figura 46 – BoxPlot M_s multiobjetivo ponderado.	72
Figura 47 – $Fitness$ mono-objetivo por experimento mono-objetivo não ponderado	73
Figura 48 – Valor de ISE por experimento mono-objetivo não ponderado	73
Figura 49 – Valor de TVC por experimento mono-objetivo não ponderado	74
Figura 50 – Valor de M_s por experimento por experimento mono-objetivo não ponderado	74
Figura 51 – $Fitness$ mono-objetivo por experimento mono-objetivo ponderado	75
Figura 52 – Valor de ISE por experimento mono-objetivo ponderado	76
Figura 53 – Valor de TVC por experimento mono-objetivo ponderado	76
Figura 54 – Valor de M_s por experimento mono-objetivo ponderado	77
Figura 55 – ISE vs TVC - multiobjetivo e mono-objetivo.	79
Figura 56 – ISE vs M_s - multiobjetivo e mono-objetivo.	80
Figura 57 – TVC vs M_s - multiobjetivo e mono-objetivo.	80
Figura 58 – ISE vs TVC - multiobjetivo e heurísticas de referência.	82
Figura 59 – ISE vs M_s - multiobjetivo e heurísticas de referência.	82
Figura 60 – TVC vs M_s - multiobjetivo e heurísticas de referência.	83
Figura 61 – Simulação das saídas do sistema, com GPC sintonizado pelas respectivas melhores soluções.	84
Figura 62 – Simulação das ações de controle do sistema, com GPC sintonizado pelas respectivas melhores soluções.	85

LISTA DE TABELAS

Tabela 1 – Exemplo de dominância	28
Tabela 2 – Limites dos Parâmetros do GPC	47
Tabela 3 – Parâmetros do NSGA-II	48
Tabela 4 – Limites dos Parâmetros do GPC para o sistema estudado	49
Tabela 5 – Limites dos Parâmetros do método <i>get_bound()</i>	50
Tabela 6 – Valores de ponderação	51
Tabela 7 – Média e desvio padrão das melhores soluções multiobjetivo não ponderada.	59
Tabela 8 – Avaliação qualitativa da robustez	63
Tabela 9 – Margens Exemplo	63
Tabela 10 – Média e desvio padrão das melhores soluções multiobjetivo ponderado	69
Tabela 11 – Média e desvio padrão das melhores soluções da otimização mono-objetivo não ponderada.	75
Tabela 12 – Média e desvio padrão das melhores soluções da otimização mono-objetivo ponderada.	77
Tabela 13 – Melhores soluções multiobjetivo sem ponderação	78
Tabela 14 – Melhores soluções multiobjetivo com ponderação	78
Tabela 15 – Melhores soluções mono-objetivo	78
Tabela 16 – Soluções das heurísticas de referência	81

LISTA DE ABREVIATURAS E SIGLAS

AG	Algoritmo Genético
CARMA	<i>Controlled Auto-Regressive and Moving-Average</i>
CARIMA	<i>Controlled Auto-Regressive Integrating Moving- Average</i>
ED	Distância Euclidiana
GPC	<i>Generalized Predictive Control</i>
IAE	<i>Integral Absolute Error</i>
ITAE	<i>Integral Time-weighted Absolute Error</i>
ISE	<i>Integral Squared Error</i>
MPC	<i>Model Predictive Control</i>
Ms	Sensibilidade Máxima
NSGA-II	<i>Non-dominated Sorting Genetic Algorithm</i>
SBX	<i>Simulated binary crossover</i>
TVC	<i>Total Variaton Control</i>

LISTA DE SÍMBOLOS

N_1	Horizonte de predição mínimo
N_y	Horizonte de predição
N_u	Horizonte de controle
λ	Fator de ponderação de controle
t_{80}	Tempo de assentamento para 80% do estado estacionário do processo
t_{90}	Tempo de assentamento para 90% do estado estacionário do processo
t_{95}	Tempo de assentamento para 95% do estado estacionário do processo

SUMÁRIO

1	INTRODUÇÃO	14
2	JUSTIFICATIVA	16
3	REFERENCIAL TEÓRICO	18
3.1	CONTROLE PREDITIVO	18
3.1.1	<i>GENERALIZED PREDICTIVE CONTROL – GPC</i>	19
3.1.1.1	GPC - LEI DE CONTROLE	20
3.1.1.2	PREDIÇÃO DA SAÍDA: VIA CÁLCULO ITERATIVO DO MODELO	22
3.1.1.3	OTIMIZAÇÃO DA LEI DE CONTROLE	23
3.2	OTIMIZAÇÃO DE PARÂMETROS DO GPC	24
3.2.1	CONCEITOS BÁSICOS	25
3.2.2	MONO-OBJETIVO E MULTI-OBJETIVO	25
3.2.2.1	OTIMIZAÇÃO MONO-OBJETIVO	25
3.2.2.2	OTIMIZAÇÃO MULTI-OBJETIVO	26
3.3	ALGORITMO GENÉTICO	28
3.3.1	OPERADORES GENÉTICOS	31
3.3.2	APLICAÇÕES DOS ALGORITMOS GÊNÉTICOS	31
3.3.3	NSGA-II	32
3.4	MÉTRICAS DE CUSTO	34
3.4.1	ERRO	34
3.4.2	CONTROLE	35
3.4.3	ROBUSTEZ	35
3.4.3.1	CRITÉRIO DE NYQUIST	36
3.4.3.1.1	DIAGRAMA DE NYQUIST	37
3.4.3.1.2	DEFINIÇÃO DO CRITÉRIO DE NYQUIST	37
3.4.3.1.3	ESTABILIDADE RELATIVA	38
4	ABORDAGEM PROPOSTA	40
4.1	NSGA-II APLICADO A SINTONIA DO GPC	40
4.2	SIMULAÇÃO DO GPC	41
4.2.1	FUNÇÃO ROBUSTEZ	44
5	AVALIAÇÃO EXPERIMENTAL	46
5.1	ASPECTOS DE IMPLEMENTAÇÃO	46
5.2	PARÂMETROS DO NSGA-II	48
5.3	ESTUDO DE CASO	48

5.4	OTIMIZAÇÃO COM PONDERAÇÃO	50
6	RESULTADOS	52
6.1	OTIMIZAÇÃO MULTIOBJETIVO	52
6.1.1	ANÁLISE GERAL DA POPULAÇÃO: OTIMIZAÇÃO NÃO PONDE- RADA	52
6.1.2	ANÁLISE GERAL DA POPULAÇÃO: OTIMIZAÇÃO PONDERADA .	63
6.2	OTIMIZAÇÃO MONO-OBJETIVO	72
6.2.1	ANÁLISE GERAL DA POPULAÇÃO: OTIMIZAÇÃO NÃO PONDE- RADA	72
6.2.2	ANÁLISE GERAL DA POPULAÇÃO: OTIMIZAÇÃO PONDERADA .	75
6.3	ANÁLISE DAS MELHORES SOLUÇÕES	77
6.3.1	SOLUÇÕES MULTIOBJETIVO	78
6.3.2	SOLUÇÕES MONO-OBJETIVO	78
6.3.3	COMPARAÇÃO DAS SOLUÇÕES MONO-OBJETIVO E MULTIOJ- BETIVO	79
6.3.4	COMPARAÇÃO DAS SOLUÇÕES MULTIOBJETIVO E AS HEURÍS- TICAS DE REFERÊNCIA	81
6.4	SIMULAÇÃO DO SISTEMA DESCRITO NO ESTUDO DE CASO . .	83
7	CONCLUSÃO	86
A	DETERMINAÇÃO DA MATRIZ F	87
B	MATRIZ G	90
C	MATRIZ H	93
	REFERÊNCIAS	96

1 INTRODUÇÃO

Na sociedade moderna os sistemas de controle estão presentes de inúmeras maneiras, tanto de forma direta ou indireta. São encontrados nos mais diversos lugares, seja em automóveis, dispositivos mecânicos, na indústria, entre outros. Baseado nesses fatores a demanda crescente por esse tipo de sistema acarretou o desenvolvimento de métodos de controle cada vez mais avançados, principalmente no meio industrial com o objetivo de otimizar seus processos produtivos, conciliando competitividade e rentabilidade (DUTRA, 2003).

Os sistemas de controle podem ser definidos como um conjunto de subsistemas e processos construídos com objetivo de obter uma saída desejada (NISE, 2013). Os mesmos podem ser do tipo: controle realimentado (*feedback*) e controle antecipatório (*feedforward*), ou também uma combinação dos dois. O controlador do tipo realimentado é o mais utilizado devido a sua simplicidade, porém apresenta desvantagem devido ao fato de só atuar após verificar que há erro entre o *set-point* e a variável controlada. No entanto em um sistema controlado tem-se o intuito de evitar a ocorrência desses erros, e o controlador antecipatório é baseado nessa ideia (BAYER; ARAÚJO, 2011).

Entre as poderosas técnicas de controle que surgiram nas últimas décadas destaca-se o MPC (*Model Predictive Control*). O MPC foi uma das poucas técnicas que obtiveram impacto significativo no controle de processos industriais, a principal razão dar-se ao fato do mesmo poder ser aplicado a sistemas SISO (*Single Input, Single Output*), MIMO (*Multi-Input, Multi-Output*) e por possuir ações (*feedforward*) e (*feedback*) de forma direta (NORMEY-RICO; CAMACHO, 2007).

O MPC utiliza previsões futuras da saída do processo controlado na sua lei de controle (RIBEIRO, 2003), e é caracterizado por possuir forte dependência do modelo matemático do processo a ser utilizado na predição da saída (CAVALCANTI, 2003). Nesse tipo de controlador há componentes básicos que são: modelo de processo, tipo de perturbação, horizonte de modelo, horizonte de controle e ponderação do sinal de controle (MOREIRA, 2002). O mesmo é composto por quatro principais parâmetros: N_1 , N_y , N_u , λ que representam respectivamente horizonte mínimo de predição, horizonte de predição, horizonte de controle e fator de ponderação de controle (PEREIRA, 1997).

A sintonia de um controlador é um processo complexo, tanto para um controlador clássico como o PID (*Proportional, Integral and Derivative*), quanto para um controlador preditivo. A quantidade de parâmetros de um controlador está em consonância com a sua complexidade de sintonia, portanto quanto maior o número de parâmetros, maior a dificuldade de sintonia para alcançar certo desempenho desejado (MAITELLI; CAVALCANTI, 2006). Baseado nessa dificuldade grande parte dos controladores preditivos são sintonizados de maneira

heurística (TIZZO; LOPES, 2009). O desenvolvimento de metodologias práticas para a sintonia do MPC que não seja heurística é relativamente escassa na literatura (TIZZO; LOPES, 2009), algumas das metodologias já desenvolvidas para a sintonia são encontradas em (MAITELLI; CAVALCANTI, 2006; TIZZO; LOPES, 2009). Em (TIZZO; LOPES, 2009) foram utilizados os algoritmos de otimização de evolução diferencial com intuito de maximizar o desempenho do controlador. Já em (MAITELLI; CAVALCANTI, 2006) foram utilizado algoritmos de otimização genéticos aplicando o controle preditivo a uma coluna de destilação. Entretanto nenhum dos trabalhos citados consideram uma perspectiva multiobjetivo na sintonia dos parâmetros, em (MAITELLI; CAVALCANTI, 2006) a otimização foi realizada voltada apenas ao desempenho do controlador já em (TIZZO; LOPES, 2009) a sintonia é voltada a robustez e desempenho porém as métricas de avaliação são diferentes das utilizadas neste trabalho.

Diante das limitações apresentadas, o objetivo deste trabalho é encontrar um conjunto de parâmetros para a sintonia de um MPC, mais especificamente o GPC (*Generalized Predictive Control*), considerando o seu desempenho e robustez. Dado que o desempenho e robustez são métricas conflitantes, é adotada uma perspectiva de otimização multiobjetivo, possibilitando que sejam obtidas várias soluções que apresentam um bom compromisso entre as duas métricas. Desta maneira, o engenheiro de controle tem a liberdade de decidir qual solução é mais adequada para o processo em questão.

Com base no objetivo mencionado, utilizou-se o ISE (*Integral Squared Error*) e TVC (*Total Variaton Control*) como métricas de desempenho e MS (Sensibilidade Máxima) como métrica de robustez. Implementou-se o algoritmo NSGA-II (*Non-dominated Sorting Genetic Algorithm, NSGA-II*) como estratégia de otimização multiobjetivo. Este algoritmo que adota os conceitos de fronteira de pareto para avaliar independentemente as três métricas citadas anteriormente.

Para avaliar a solução proposta, utilizou-se como estudo de caso um sistema de segunda ordem, o qual representa um motor DC (*Direct Current*). Os resultados obtidos foram comparados com heurísticas de referência encontradas na literatura e com uma estratégia mono-objetivo que considera uma agregação das três métricas em questão. A base para a comparação dos resultados foram os valores absolutos obtidos para cada uma das métricas, a distância das soluções para um ponto ideal, além da distribuição das soluções no espaço de busca.

A partir dos resultados, observou-se que em valores absolutos o conjunto de soluções encontradas pela estratégia multiobjetivo geralmente contém as soluções das heurísticas de referência e da estratégia mono-objetivo. Além disso, em alguns casos as soluções encontradas pelo NSGA-II são melhores. Analisando-se a distância do ponto ideal, em todas as situações a estratégia proposta encontra uma solução melhor em relação as outras estratégias. Como esperado, a estratégia multi-objetivo também explora melhor o espaço de busca.

2 JUSTIFICATIVA

Os sistemas de controles atuais utilizam na sua maioria controladores monovariáveis, que por sua vez são majoritariamente do tipo PID ou uma combinação deles (estima-se que 90 % dos existentes são desse tipo) (ASTROM; HAGGLUND, 1995). A vasta utilização desse tipo de controlador se dá ao fato de que o mesmo apresenta uma implementação simples, barata e rápida. Contudo quando se deseja maximizar o desempenho de alguns tipos de sistemas, como por exemplo, sistemas multivariáveis, sistemas de ordem elevada, sistemas instáveis em malha aberta, sistemas de fase não-mínima ou sistemas com grandes atrasos de transporte, os controladores PID deixam de ser vantajosos, visto que nesses casos sua utilização torna-se complexa e/ou ineficiente (SUNAN; KIONG; HENG, 2002; JÚNIOR, 2015).

Devido às limitações presentes nos controladores clássicos, houve aumento significativo nas pesquisas por estratégias de controle avançado. Entre essas novas técnicas de controle desenvolvidas, encontram-se os controladores do tipo MPC, que vem ganhando espaço principalmente no meio industrial devido a facilidade de utilizá-lo em sistemas multivariáveis e em sistemas que apresentem restrições de processo (JÚNIOR, 2015). Uma característica importante dos MPC se dá ao fato de que podem lidar com as restrições de processo e o fato de existirem sistemas operando com *set-points* próximos das restrições de controle, pode acarretar no aumento da eficiência, na segurança e até ganhos financeiros (MACIEJOWSKI, 2000 apud JÚNIOR, 2015).

Apesar das vantagens, os MPCs são complexos de serem utilizados devido a dificuldade de implementação e sintonia de vários parâmetros. Atualmente existem na literatura várias técnicas de sintonia como em (LEE; YU, 1994 apud JÚNIOR, 2015) que assume que o sistema a ser controlado é estável em malha aberta e que a sintonia deve ser na seguinte ordem: primeiro a mesma deve voltar-se a estabilidade do sistema e depois para sua robustez, o mesmo ajusta o maior valor possível para o horizonte de controle de forma a manter a estabilidade nominal do sistema, a posteriori outros parâmetros são sintonizados mantendo-se fixos os valores do horizonte de controle e o do horizonte de previsão, de forma a alcançar a sintonia voltada a robustez, entretanto esse método está restrito a sistemas de malha aberta.

Em (SARHADI; SALAHSHOOR; KHAKI-SEDIGH, 2012 apud JÚNIOR, 2015) é apresentado um método de sintonia para GPC monovariável e sem restrições com a sintonia voltada a robustez. O mesmo realiza a sintonia do controlador observando o efeito do ajuste de cada parâmetro no sistema separadamente, os ajustes continuam até que a robustez desejada seja atingida, entretanto pode-se observar que este método não tem muita aplicabilidade, principalmente no meio industrial o qual a maioria dos sistemas são do tipo multivariável que por sua vez é o meio em que o MPC é mais utilizado.

Há métodos também de sintonia que delimitam valores máximos e mínimos para os

parâmetros do GPC, como mostra a Tabela 2, todavia grande parte deles buscam apenas um desempenho razoável para o mesmo, não buscando necessariamente otimizar determinados objetivos.

Diante desta dificuldade entram os métodos de sintonia ótima que buscam os parâmetros de forma a otimizar determinados objetivos pré-determinados do sistema de controle, assegurando dessa maneira seu funcionamento. Uma das vantagens desses tipos de métodos é que os mesmos necessitam de um conhecimento menor do sistema, além de serem mais rápidos (JÚNIOR, 2015). Todavia os objetivos são representados por funções matemáticas que são na maioria dos casos de difícil resolução analítica, por isso esse tipo de método utiliza algoritmos que otimizem estas funções, os mesmos podem ser multiobjetivo, ou mono-objetivo. O algoritmo utilizado neste trabalho será o NSGA-II o qual é um algoritmo de otimização multiobjetivo.

O NSGA-II apresenta inúmeras vantagens na sua utilização dentre elas a vasta gama de problemas no qual pode ser utilizado devido a sua robustez. Possui fácil implementação e uma das suas principais vantagens se dá ao fato de que não usam apenas informações locais, evitando assim, em grande parte das vezes, convergir para um único ótimo local como grande parte dos algoritmos de otimização (ZINI, 2009).

3 REFERENCIAL TEÓRICO

Neste capítulo será explicado o conceito do GPC, assim como é demonstrado a dedução matemática para encontrar sua lei de controle. Em sequência introduz-se o conceito de otimização. Será apresentado também os métodos de solução dos problemas de otimização e o conceito de algoritmo genético e suas derivações, posteriormente é apresentado o funcionamento do NSGA-II, algoritmo utilizado neste trabalho.

3.1 CONTROLE PREDITIVO

Os controladores preditivos são a classe de controladores na qual se utilizam previsões do sistema em sua lei de controle, de forma que quanto mais representativo for o preditor, melhores serão os valores preditos e, por consequência, melhor será o desempenho do controlador. Esses controladores geralmente são utilizados quando há necessidade de obter-se um desempenho melhor que os controladores não preditivos, como por exemplo em sistemas com atraso de transporte muito grande e sistemas multivariáveis além de serem mais robustos aos erros de modelagem (SUNAN; KIONG; HENG, 2002).

Dentro da classe de controladores preditivos destacam-se os controladores preditivos baseados no modelo (MPC). O termo MPC não é uma estratégia de controle específica, mas sim uma gama ampla de métodos de controle que utilizam um modelo do processo para obter o sinal de controle de forma a minimizar a função custo (CAMACHO; BORDONS, 2004 apud NORMEY-RICO; CAMACHO, 2007). Os controladores preditivos tem como base os seguintes critérios:

- O uso explícito de um modelo que será utilizado para prever a saída do processo em futuros instantes de tempo;
- O cálculo de uma sequência de controle que minimiza a função custo;
- A estratégia de horizonte deslizante, que é responsável por deslocar o horizonte de previsão para o futuro, a cada instante de tempo.

Os controladores do tipo MPC apresentam uma série de vantagens sobre os outros métodos, tais como (NORMEY-RICO; CAMACHO, 2007) :

- É baseado em princípios básicos possibilitando futuras extensões;
- É extremamente útil quando as referências futuras são conhecidas (referência preditiva);
- Sistema multivariável é facilmente resolvido, além de eliminar os atrasos de transporte de forma intrínseca.

A maior diferença entre os algoritmos existentes MPC são os modelos utilizado para representar o sistema (NORMEY-RICO; CAMACHO, 2007). Um dos controladores preditivos baseados em modelo mais difundidos é o GPC, que surgiu em 1987 proposto por D.W. Clarke, o mesmo utiliza o modelo matemático do processo de forma a minimizar uma função custo quadrática dos erros futuros para encontrar a ação do sinal de controle a ser utilizada (PEREIRA, 1997). Esse método é geralmente aplicado em sistemas de alto desempenho e é bastante eficaz em sistemas de fase não-mínima e instáveis em malha aberta (CLARK, 1987).

3.1.1 GENERALIZED PREDICTIVE CONTROL – GPC

O GPC é caracterizado por possuir como modelo de sistema o CARIMA (*Controlled Auto-Regressive Integrating Moving- Average*) (RIBEIRO, 2003). Uma das principais vantagens desse método é que pode ser aplicado desde sistemas mais simples à sistemas mais complexos por possuírem uma forma padrão de operação distinguindo-se apenas do tempo computacional demandado em consonância com a complexidade do sistema a ser tratado (CLARK, 1987). Esse tipo de controlador pode-se dizer que é composto basicamente por três itens (SUNAN; KIONG; HENG, 2002):

- O modelo de sistema a ser controlado;
- Formulação da função objetivo;
- Minimização da função objetivo de forma a produzir o horizonte de ações incrementais ótimas de controle.

Supondo um modelo linear descrito pela Equação 3.1 (CLARK, 1987):

$$A(z^{-1})y(t) = z^{-d}B(z^{-1})u(t-1) + x(t). \quad (3.1)$$

No qual $y(t)$ representa a saída, $u(t)$ a entrada e $x(t)$ representa uma perturbação do sistema. z^{-1} representa o operador de atraso unitário de forma que $z^{-1}y(k) = y(k-1)$, z^{-d} representa o atraso de transporte e $A(z^{-1})$ e $B(z^{-1})$ são polinômios dados pelas Equações (3.2) e (3.3), a perturbação deste sistema é considerada do tipo média móvel e pode ser descrita pela Equação (3.4). Visando facilitar a compreensão, o atraso de transporte será considerado suprimido *a priori*. Vale ressaltar que qualquer atraso de tempo deverá ser representado por coeficiente nulos no polinômio $B(z^{-1})$ (CLARK, 1987):

$$A(z^{-1}) = 1 + a_1z^{-1} + \dots + a_naz^{-na}, \quad (3.2)$$

$$B(z^{-1}) = b_0 + b_1z^{-1} + \dots + b_nbz^{-nb}, \quad (3.3)$$

$$x(t) = C(z^{-1})\xi(t). \quad (3.4)$$

Na Equação (3.4) o termo $C(z^{-1})$ representa um polinômio que pode ser descrito como $C(z^{-1}) = 1 + c_1 z^{-1} + \dots + c_{N_c} z^{-N_c}$, e $\xi(t)$ representa uma sequência aleatória não correlacionada, ruído branco. Substituindo a Equação (3.4) em (3.1) obtém-se o modelo que é denominado de CARMA (*Controlled Auto-Regressive and Moving-Average*) (CLARK, 1987):

$$A(z^{-1})y(t) = B(z^{-1})u(t-1) + C(z^{-1})\xi(t). \quad (3.5)$$

Mesmo que muita teoria de auto-ajuste seja baseada nesse tipo de modelo descrito pela Equação (3.5), o mesmo torna-se não adequado quando a perturbação é não estacionária. Tendo em vista que na prática as perturbações são do tipo passo aleatório: que ocorre em horários aleatórios, como por exemplo quando há mudança na qualidade do material, podendo acarretar na alteração dos pólos do sistema; ou são do tipo movimento browniano, comumente encontrado em sistemas que trabalham com balanceamento de energia. Os sistemas com esses tipos de perturbações o modelo mais apropriado a utilizar é descrito pela Equação (3.6) (CLARK, 1987):

$$x(t) = \frac{C(z^{-1})\xi(t)}{\Delta}, \quad (3.6)$$

$$\Delta = 1 - z^{-1}. \quad (3.7)$$

Substituindo a Equação (3.6) na Equação (3.1), resulta na Equação (3.8) que representa o modelo a qual denomina-se CARIMA (CLARK, 1987) :

$$A(z^{-1})y(t) = B(z^{-1})u(t-1) + \frac{C(z^{-1})\xi(t)}{\Delta}. \quad (3.8)$$

Por simplicidade adota-se $C(z^{-1}) = 1$ de forma a facilitar as deduções, resultando na Equação (3.9) (CLARK, 1987) :

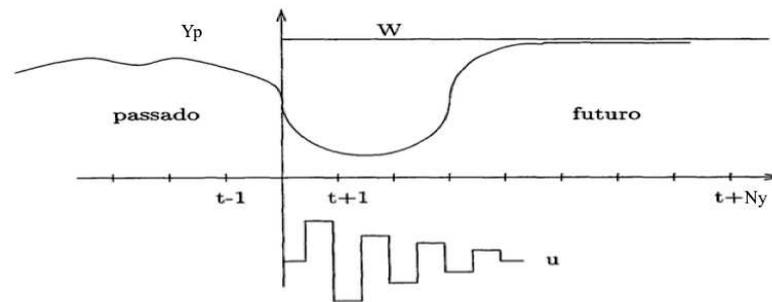
$$A(z^{-1})y(t) = B(z^{-1})u(t-1) + \frac{\xi(t)}{\Delta}. \quad (3.9)$$

Os GPC são caracterizados por calcularem um conjunto de previsões de saída, o qual o termo j varia de um valor mínimo à um valor máximo e esse intervalo é denominado de horizonte de predição. Considerando d como o tempo morto do sistema, quando $d \geq j$ a predição $\hat{y}(t+j | t)$ depende unicamente dos dados já disponíveis no sistema, entretanto quando $d < j$ é necessário utilizar ações de controles futuras, e esse pressuposto é a chave do funcionamento do GPC (CLARK, 1987).

3.1.1.1 GPC - LEI DE CONTROLE

O objetivo da lei de controle do GPC é conduzir as saídas preditas $\hat{y}(t+j)$ ao valor de referência futuro $w(t+j)$, onde $[w(t+j); j = 1, 2, \dots]$ normalmente permanece constante (referencia não preditiva), como mostra a Figura 1, todavia em alguns sistemas as referencias futuras são conhecidas como o controle de processos robóticos e a isso denomina-se referencia preditiva.

Figura 1 – Conceito de controle preditivo.



Fonte: Adaptado de (PEREIRA, 1997)

Esse tipo de controlador calcula toda uma sequência de ações incrementais de controle, utilizando o preditor e dados já obtidos do sistema, de forma a minimizar a função objetivo. Após obtida a sequência de ações incrementais de controle, apenas o $u(t)$ é utilizado e aplicado no sistema e os demais descartados. No instante seguinte todo processo é repetido, e a isso dá-se o nome de horizonte deslizante, mostrado na Figura 1 (SUNAN; KIONG; HENG, 2002). Portanto nos controladores preditivos generalizados segue-se os seguintes passos para obter-se a ação de controle $u(t)$ (CLARK, 1987) :

- Calcula-se a sequência de predição de saída;
- Encontra-se o valor da sequência de ações incrementais futuras de controle, de forma a minimizar uma função custo;
- Utiliza-se apenas o primeiro elemento da sequência, o elemento $u(t)$ e os vetores de dados são deslocados de forma que possa ser utilizado na próxima amostra.

A função custo utilizada por um controlador preditivo generalizado é dada pela Equação (3.10):

$$J = \sum_{N1}^{Ny} (\hat{y}(t+j) - w(t+j))^2 + \sum_1^{Nu} \lambda (\Delta u(t+j-1))^2 . \quad (3.10)$$

$N1$ Horizonte mínimo de predição

Ny Horizonte de predição

Nu Horizonte de controle

λ Fator de ponderação de controle

A escolha dos parâmetros desse tipo de controlador não são triviais, tendo em vista que há vários métodos para sua determinação, e que é o objetivo deste trabalho. Entretanto podem

ser feitas algumas considerações na prática, como por exemplo, o horizonte de predição, N_y , normalmente é escolhido de forma a abranger toda resposta que é significativamente atingida pelo controle atual, mas tipicamente N_y é escolhido de forma que seja maior que o grau de $B(z^{-1})$, ou então que seja maior que o tempo de subida da planta (CLARK, 1987).

Normalmente N_1 é adotado como 1, todavia quando adota-se, N_1 grande é porque não é importante se houver erros nos primeiros instantes, o que acarreta uma resposta suave. Do mesmo modo, não faz sentido adotar N_1 menor que o tempo de atraso de transporte do sistema, já que o sistema só começa a evoluir após esse tempo, por isso costuma-se adotar N_1 como valor maior que tempo do atraso de forma a minimizar os cálculos (CLARK, 1987).

No caso do horizonte de controle N_u há uma regra básica para o mesmo, que é configurá-lo igual ao número de polos instáveis do sistema e para sistema estáveis o mesmo é adotado como 1, apesar de ser uma opção viável a mesma deve ser evitada tendo em vista que geralmente apresenta um desempenho razoável. Valores pequenos de N_u correspondem a resposta mais lentas e ações de controle suave, em contra partida valores grande para N_u acarreta em ações mais ativas e menos estáveis (CLARK, 1987).

3.1.1.2 PREDIÇÃO DA SAÍDA: VIA CÁLCULO ITERATIVO DO MODELO

Um dos passos mais importantes para determinar a ação de controle ótima para um dado instante t consiste na predição das saídas futuras do sistema em um horizonte de tempo. Utilizando o modelo descrito pela Equação (3.9) é possível determinar uma expressão para as saídas futuras desse tipo de controlador. Vale a ressalva de que até no presente instante t , dispõe-se de todas as informações do sistema mais particularmente no que se diz respeito a saída do sistema. Com o intuito de isolar-se o termo $y(t)$ da Equação (3.9), multiplica-a por Δ , resultando em :

$$A(z^{-1})\Delta y(t) = B(z^{-1})\Delta u(t-1) + \xi(t), \quad (3.11)$$

O termo $A\Delta(z^{-1})$ pode ser representado por $\tilde{A}(z^{-1}) = 1 + \tilde{a}_1q^{-1} + \dots + \tilde{a}_{n\tilde{a}}q^{-n\tilde{a}}$, sendo $\tilde{a}_j = a_j - a_{j-1}$ e $n\tilde{a} = na + 1$. Substituindo estes termos na Equação 3.11 e desconsiderando a perturbação, resulta em :

$$\begin{aligned} \tilde{A}(z^{-1})y(t) &= B(z^{-1})\Delta u(t-1) \\ (1 + \tilde{a}_1q^{-1} + \dots + \tilde{a}_{n\tilde{a}}q^{-n\tilde{a}})y(t) &= B(z^{-1})\Delta u(t-1) \\ y(t) &= B(z^{-1})\Delta u(t-1) - (\tilde{a}_1q^{-1} + \dots + \tilde{a}_{n\tilde{a}}q^{-n\tilde{a}})y(t-1). \end{aligned} \quad (3.12)$$

A equação do preditor é obtida através da Equação (3.12) e por meio dela pode-se determinar previsões da saída:

$$\hat{y}(t+j) = B(z^{-1})\Delta u(t+j-1) - (\tilde{a}_1q^{-1} + \dots + \tilde{a}_{n\tilde{a}}q^{-n\tilde{a}})\hat{y}(t+j-1). \quad (3.13)$$

A Equação (3.13) representa de forma genérica a previsão da saída em um instante j do GPC, via cálculo iterativo, visto que as previsões do sistema poderiam ser calculadas utilizando

o método Diophantine ¹. Utilizando a Equação (3.13), as saídas futuras são composta por sinais incrementais presentes e futuros do sinal de controle, que são desconhecidos, e de valores anteriores aquele momento, permitindo que a mesma seja reescrita na forma (RIBEIRO, 2003):

$$\hat{y} = \hat{y}_l + \hat{y}_f. \quad (3.14)$$

O termo \hat{y}_l é denominado resposta livre do sistema, pois é composto apenas por valores incrementais passados, e valores passados e presente da saída do sistema, e o termo \hat{y}_f é denominado de resposta forçada e é composto por valores incrementais futuros e valores incrementais presentes do sinal de controle (RIBEIRO, 2003).

A Equação (3.14) também pode ser reescrita na forma vetorial e é expressa por:

$$\begin{aligned} \hat{Y} &= \hat{Y}_f + \hat{Y}_l, \\ \hat{Y}_f &= G\Delta U, \\ \hat{Y}_l &= F\hat{Y}_p + H\Delta U_p. \\ \hat{Y} &= G\Delta U + F\hat{Y}_p + H\Delta U_p. \end{aligned} \quad (3.15)$$

onde:

$$\begin{aligned} \hat{Y} &= [\hat{y}(t+1|t)\dots\hat{y}(t+Ny|t)]^T, \\ \hat{Y}_p &= [y(t)\dots y(t-na)]^T, \\ \Delta U_p &= [\Delta u(t-1)\dots\Delta u(t-(nb+d))]^T, \\ \Delta U &= [\Delta u(t)\dots\Delta u(t+Nu-1)]^T. \end{aligned}$$

A expressão para as Matrizes G , F e H foram deduzidas e demonstradas nos Apêndices A,B e C.

3.1.1.3 OTIMIZAÇÃO DA LEI DE CONTROLE

Após o cálculo da predição das saídas futuras do sistema o próximo passo do GPC para determinação da ação de controle consiste na minimização da função custo, expressa pela Equação (3.10) em busca de encontrar a sequência ótima incremental de ações de controle. No critério de minimização considera-se que as ações incrementais de controle fora do horizonte de controle são nulas, ou seja (RIBEIRO, 2003):

$$\Delta u(k) = 0, k \in [t+Nu, t+Ny+1].$$

¹ Equações Diophantine são equações com duas ou mais incógnitas onde são aceitas apenas soluções inteiras (BISPO, 2013)

Substituindo a expressão obtida pela Equação (3.15) em (3.16), definindo a sequência de referência como $\hat{W} = [w(t+1)...w(t+N_y|t)]^T$, e considerando o fator de ponderação de controle constante, obtêm-se:

$$J = (\hat{Y} - \hat{W})^T (\hat{Y} - \hat{W}) + \lambda \Delta U^T \Delta U \quad (3.16)$$

$$J = (G\Delta U + F\hat{Y}_p + H\Delta U_p - \hat{W})^T (G\Delta U + F\hat{Y}_p + H\Delta U_p - \hat{W}) + \lambda \Delta U^T \Delta U. \quad (3.17)$$

Para determinar os valores da sequência incrementais das ações de controle deriva-se a Equação (3.17) em função das ações incrementais de controle e iguala a zero. Portanto tem-se :

$$J = (\Delta U^T G^T G \Delta U) + (\Delta U^T G^T (\hat{Y}_l - \hat{W})) + ((\hat{Y}_l - \hat{W})^T G \Delta U) + ((\hat{Y}_l - \hat{W})^T (\hat{Y}_l - \hat{W})) + (\lambda \Delta U^T \Delta U)$$

$$\frac{\partial J}{\partial \Delta u} = 2G^T G \Delta U + 2G^T (\hat{Y}_l - \hat{W}) + 2\lambda \Delta U$$

$$0 = (G^T G + \lambda I) \Delta U - G^T (\hat{W} - \hat{Y}_l)$$

$$(G^T G + \lambda I) \Delta U = G^T (\hat{W} - \hat{Y}_l)$$

$$(G^T G + \lambda I)^{-1} (G^T G + \lambda I) \Delta U = (G^T G + \lambda I)^{-1} G^T (\hat{W} - \hat{Y}_l)$$

$$I \Delta U = (G^T G + \lambda I)^{-1} G^T (\hat{W} - \hat{Y}_l)$$

$$\Delta U = (G^T G + \lambda I)^{-1} G^T (\hat{W} - \hat{Y}_l). \quad (3.18)$$

A Equação (3.18) representa a lei de controle ótima do GPC. Como dito anteriormente o GPC utiliza o conceito de horizonte deslizante o qual a cada instante de t calcula-se a lei de controle dada pela Equação (3.18) e utiliza-se apenas o primeiro termo desta sequência (RIBEIRO, 2003). Sabe-se que $\Delta u(t) = u(t) - u(t-1)$, portanto o último passo para determinar a ação de controle que será utilizada pelo controlador no instante t, consiste em isolar o termo $u(t)$ o que resulta em:

$$u(t) = \Delta u(t) + u(t-1) \quad (3.19)$$

Dada a Equação (3.19) é importante perceber que a partir da mesma, insere automaticamente um integrador na malha desse sistema. O mesmo permite que os erros em regime permanente com relação ao sinal de referência sejam reduzidos, de forma semelhante ao integrador nos controladores PID (RIBEIRO, 2003).

3.2 OTIMIZAÇÃO DE PARÂMETROS DO GPC

De acordo com a definição matemática, otimização é o estudo que busca maximizar ou minimizar uma dada função através da escolha de suas variáveis (NARIÑO, 2014). Dentro da otimização vale ressaltar cinco conceitos básicos, que são: função objetivo, variáveis de projeto, restrições, espaço de busca e ponto ótimo.

3.2.1 CONCEITOS BÁSICOS

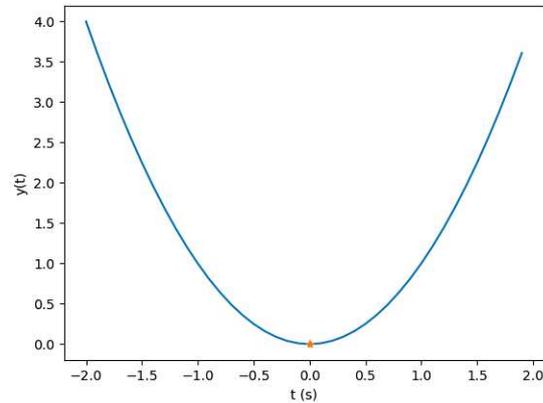
A função objetivo, ou função *fitness* é a representação matemática do problema a ser otimizado, a mesma sofre influência das variáveis de projeto. O problema pode ser formulado do tipo mono-objetivo quando só há uma função objetivo e multiobjetivo quando há várias funções objetivos. As variáveis de projeto representam os parâmetros variáveis do processo de otimização a que se pretende determinar de forma com que os mesmos otimizem um determinado problema. Geralmente essas variáveis são limitadas em um determinado intervalo com o intuito de viabilizar o processo de otimização, essas limitações ou restrições são as condições de projeto responsável por limitar o espaço de busca, e podem ocorrer devido a viabilidade de variáveis atingirem certos valores, ou para convergirem de forma mais rápida ao ponto ótimo. O espaço de busca consiste no domínio da função objetivo de forma a satisfazer as restrições impostas a ele pelo intervalo das variáveis de projeto, no entanto é possível haver soluções ótimas fora do espaço de busca, todavia as mesmas são inviáveis. A solução que leva a função objetivo aos seus valores extremos é denominada ponto-ótimo, podendo ser máximo ou mínimo (NARIÑO, 2014).

3.2.2 MONO-OBJETIVO E MULTI-OBJETIVO

Em caso de problema de otimização mono-objetivo, o mesmo contém apenas um objetivo a ser otimizado. Nesse tipo de problema busca-se encontrar o máximo ou o mínimo da função objetivo, de forma a atender suas restrições. Diferentemente, o multiobjetivo contém um conjunto de funções objetivos a serem otimizadas de forma simultânea, ressaltando-se o fato de que os vários objetivos dessas funções podem ser conflitantes, ou seja, os valores das variáveis que otimizam um objetivo não necessariamente vão otimizar os demais (NARIÑO, 2014).

3.2.2.1 OTIMIZAÇÃO MONO-OBJETIVO

Supondo que um problema de otimização mono-objetivo é dado pela função $f(x) = x^2$ e que deseja-se otimizar de forma a minimizar esta função, que tem como variável de projeto x e espaço de busca limitado na faixa $-2 < x < 2$. De acordo com gráfico mostrado no Figura 2, pode-se observar que o ponto ótimo dessa função objetivo é o ponto $x = 0$, o mesmo pode ser obtido derivando a função e achando seu ponto mínimo, que é onde a função objetivo atinge seu menor valor.

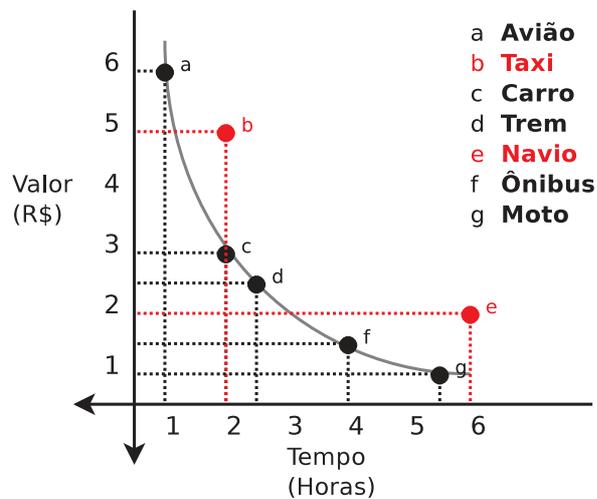
Figura 2 – Função Quadrática $f(x) = x^2$ 

Fonte: Autoria Própria

3.2.2.2 OTIMIZAÇÃO MULTIOBJETIVO

Por sua vez um problema de otimização multiobjetivo pode ser descrito como por exemplo conciliar o gasto de tempo para realizar uma viagem com o valor do meio de transporte utilizado para o mesmo. Normalmente, quanto mais rápido for o meio de transporte maior será o preço da passagem, como ilustra a Figura 3, portanto é possível perceber que não há uma única solução ótima, como no caso mono-objetivo e sim um conjunto de soluções ótimas as quais podem otimizar um objetivo e obter de forma não tão satisfatória os outros.

Figura 3 – Exemplo de fronteira de Pareto



Fonte: Autoria própria

Os problemas de otimização multiobjetivos podem ser definidos matematicamente da seguinte forma (ARROYO, 2002 apud HASHIMOTO, 2004):

$$z = (f(x_1), f(x_2), \dots, f(x_r)), \text{ Função Objetivo}$$

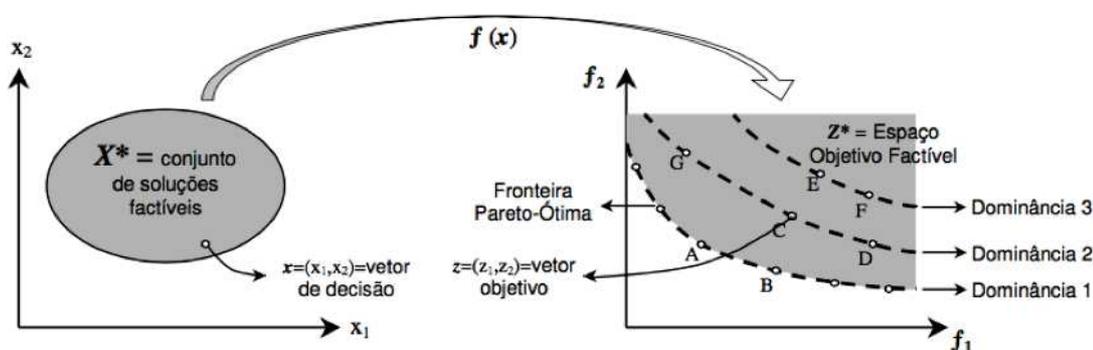
$$x = (x_1, x_2, \dots, x_r), \text{ Vetor de Decisão}$$

$$z = (z_1, z_2, \dots, z_r), \text{ Vetor Objetivo}$$

$$X^* = \{x \in X: g(x) \leq b\}, \text{ Conjunto de Soluções Factíveis}$$

De acordo com as definições acima, pode-se verificar o espaço de decisões representado por X no qual $x \in X$, ao restringir este espaço obtém-se X^* que representa o conjunto de soluções factíveis, ou seja, valores nos quais as variáveis de projeto podem de fato assumir, no qual $g(x)$ representa as restrições e b representa um vetor de dimensões p . Z representa o Espaço Objetivo no qual $z \in Z$, e do mesmo modo que o domínio da função objetivo é restrito sua imagem também torna-se restrita como mostra a Figura 4. O espaço objetivo quando restrito é denominado como espaço objetivo factível, denotado por Z^* .

Figura 4 – Problema de otimização multiobjetivo com dois objetivos



Fonte:(RIBEIRO, 2003)

Por meio da Figura 4 é possível observar a definição de problema multiobjetivo, a mesma representa a otimização de um problema multiobjetivo, o qual nesse caso possui dois objetivos, $f_1(x)$ e $f_2(x)$, e que no ponto A $f_1(x)$ tem valor menor que no ponto B, em contrapartida $f_2(x)$ detem valor menor no ponto B quando comparado ao ponto A (HASHIMOTO, 2004). Para resolver esse tipo de conflito utiliza-se o conceito de Dominância de Pareto.

O conceito de Dominância de Pareto é baseado na forma de como um vetor de soluções factíveis $x_1 \in X^*$ pode ser classificado com relação a todo conjunto de soluções factíveis, o mesmo acontece de forma conjunta com os vetores objetivo factível. Esses podem ser classificados em três categorias com relação a todo conjunto de soluções factíveis, ou conjunto objetivo factível, que são: dominados, dominantes e indiferentes.

Dado dois vetores decisão x_1 e $x_2 \in X^*$, diz-se que o vetor objetivo $f(x_1)$ domina $f(x_2)$ se $f_i(x_1) \leq f_i(x_2) \forall_i$ e que x_1 domina x_2 . Da mesma forma pode-se dizer que x_2 é dominado por x_1 assim como $f(x_2)$ é dominado por $f(x_1)$. Todavia quando $f_i(x_1) \not\leq f_i(x_2) \forall_i$ diz-se que o vetor de decisão x_1 é indiferente a x_2 e que o vetor objetivo $f(x_1)$ é indiferente a $f(x_2)$ (HASHIMOTO, 2004).

Analisando a Tabela 1 resultante da Figura 3 pode-se entender melhor o conceito de dominância. Observa-se que o avião, carro, trem, ônibus e moto são soluções não dominadas por nenhuma outra e também são indiferentes entre si. O táxi e navio são ditas soluções dominadas, pois há soluções que são melhores que elas, por exemplo no caso do táxi, a solução de carro possui o mesmo valor para o objetivo tempo, porém no objetivo valor o carro é mais barato, por isso diz-se que a solução de táxi é dominada.

Tabela 1 – Exemplo de dominância

Veículo	Valor (R\$)	Tempo(hora)
Avião	6	1
Táxi	5	2
Carro	3	2
Trem	≈ 2,5	≈ 1,9
Navio	2	≈ 6
Ônibus	≈ 1,5	4
Moto	≈ 0.5	≈ 5,5

Fonte:Autoria Própria

Baseado na Figura 3, diz-se que as soluções avião, carro, trem, ônibus e moto são soluções Pareto-ótimo, essa definição dá-se a todo vetor objetivo e vetor decisão o qual atenda o seguinte critério: x^* é dito solução Pareto-ótimo se não houver qualquer solução $x^* \in a X^*$ que domine x^* , da mesma forma $f(x^*)$ é dito Ponto Pareto-ótimo. O conjunto de todos os pontos Pareto-ótimo é denominado de fronteira de Pareto, como é mostrado na Figura 2 (HASHIMOTO, 2004).

3.3 ALGORITMO GENÉTICO

O algoritmo genético (AG), é um método meta-heurístico de otimização mono-objetivo que compõem um conjunto de técnicas baseadas na teoria da Seleção Natural de Charles Darwin. Darwin defendia em sua teoria que a sobrevivência das espécies está relacionada a seleção natural, para ele os indivíduos mais bem adaptados de uma população eram os que possuíam melhores condições de sobrevivência, desse modo deixando uma maior quantidade de indivíduos descendentes, com isso as características desses indivíduos mais bem adaptados tornam-se mais comuns nas gerações das populações seguintes do que as dos indivíduos menos adaptados (COSTA, 2009).

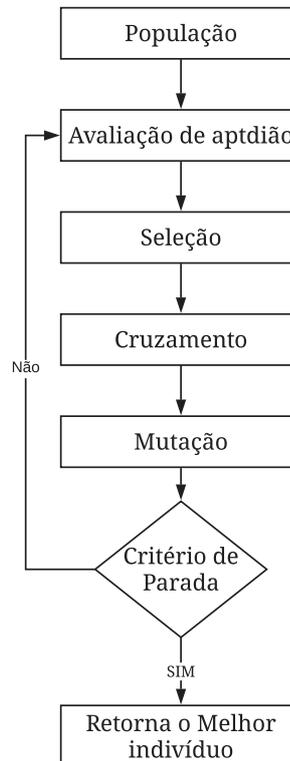
Embasados nesses conceitos, no AG, cria-se uma população de indivíduos que vão se reproduzir e competir por sua sobrevivência e os indivíduos mais adaptados da população, ou seja, os indivíduos da população que melhor otimizam o problema passam suas características para as populações subsequentes (POZO et al., 20-?). Para a implementação dos AG é necessário que o problema atenda alguns critérios para o funcionamento (POZO et al., 20-?) :

1. Encontrar uma forma de representar a população em formato de código (POZO et al., 20-?), ou seja, suponha que deseja-se otimizar um problema que tem como função objetivo $f(x, y) = (x^2, y^2)$, pode-se por exemplo assumir que os indivíduos desta população sejam números do tipo float;
2. Criar uma população que tenha diversidade entre os indivíduos capaz de permitir que AG consiga ter uma variedade de características e produzir novas soluções (POZO et al., 20-?). Geralmente quando os indivíduos são semelhantes entre si pode ocorrer do algoritmo convergir em um máximo ou mínimo local;
3. Desenvolver um método capaz de combinar os indivíduos para gerar novos indivíduos, de medir sua qualidade de solução e um critério de escolha dos indivíduos que deverão permanecer na próxima geração e os que deverão ser retirados (POZO et al., 20-?);
4. Desenvolver um método que introduz mudanças periódicas em alguns indivíduos da população, para com isso haja uma mudança na diversidade da população e até produzir indivíduos melhores adaptados possibilitando sair da uma convergência à um máximo ou mínimo local (POZO et al., 20-?);

Na Figura 5 pode-se observar o funcionamento dos AG. *A priori* cria-se uma população de indivíduos que respeitam o critério 1, os mesmos são criados estocasticamente. Após esse passo, avalia-se toda população de indivíduos, baseado numa função *fitness*, em seguida os indivíduos passam pelo operador seleção que escolhe os melhores indivíduos baseado nos valores dados pela função *fitness* e estes indivíduos serão utilizados para criar uma nova população. Após a escolha destes indivíduos são aplicados sobre eles os operadores de cruzamento e de mutação, que irão gerar uma nova população. Esses passos são repetidos, com exceção do primeiro, até que o código convirja para uma solução desejada ou até o número de passos pré-determinado seja atingido (POZO et al., 20-?).

Quando utiliza-se o termo população dentro dos AG, refere-se ao conjunto de indivíduos que podem tornar-se possíveis soluções do problema, entretanto nesses algoritmos é preciso haver consonância em detrimento do tamanho da população. Uma população demasiadamente grande pode apresentar uma maior diversidade e pode varrer um maior espaço de busca, todavia a mesma pode tornar o algoritmo demasiadamente lento visto que terá um exacerbado número de indivíduos a serem avaliados pela função *fitness*. Entretanto, se a população for muito pequena

Figura 5 – Estrutura básica de um Algoritmo Genético



Fonte: Adaptado de (POZO et al., 20-?).

pode ocorrer pouca diversidade entre si impedindo que a mesma possa convergir para uma boa solução (POZO et al., 20-?).

Os indivíduos de uma população representam as variáveis de projeto anteriormente citadas. Nos AG busca-se *a priori* uma forma de representar essas variáveis em forma de código, muitas vezes os mesmo são representados por um conjunto de números binários, porém podem ser representados também em formas de letras, códigos, números reais, etc. A escolha dessa representação varia de acordo com o tipo de problema (POZO et al., 20-?).

A função *fitness* ou também conhecida como função aptidão nos AG tem como dever avaliar cada indivíduo da população e determinar um valor de *fitness* para cada um deles. Esse valor indica o quão próximo da solução ótima ou desejada tal indivíduo se encontra, tornando-se assim um dos componentes primordiais dos AG, entretanto a mesma deve conseguir diferenciar de forma proporcional cada indivíduo, para tornar o código mais eficiente visando desprezar soluções ruins e com isso evitar que boas soluções sejam descartadas impedindo a perda de tempo com soluções ruins (POZO et al., 20-?).

3.3.1 OPERADORES GENÉTICOS

Assim como na teoria evolucionária cujo os indivíduos se reproduzem de forma que as características hereditárias mais favoráveis ao meio tendem a tornarem-se comum nas gerações futuras e as menos favoráveis desaparecerem, os algoritmos genéticos utilizam com um alto nível de abstração artifícios que tentem imitar a reprodução dos seres vivos, os mesmos utilizam três operadores genéticos: seleção, cruzamento, também conhecido como *crossover* e a mutação. A presença deles é de grande importância nos AG de forma que graças aos mesmos é possível transformar a população por meio de sucessivas gerações em busca de soluções cada vez mais satisfatórias de forma a manter as características que melhor se adaptam a cada geração.

Um passo importante do AG é a seleção de indivíduos a partir dos valores de aptidão atribuídos. Para o operador de seleção utilizam-se métodos de escolha para determinar em quais indivíduos serão aplicados os operadores genéticos de cruzamento e mutação.

O operador de cruzamento imita o cruzamento entre dois indivíduos. O mesmo atua de forma a trocar características dos indivíduos pais para criar novos indivíduos com possivelmente melhores características do que seus pais.

O operador de mutação é utilizado em uma menor escala dentro da população. O mesmo garante que haja uma diversificação nas características dos indivíduos e que novas características possam surgir. A mutação mantém a diversidade genética e exclui a hipótese de que não seja possível chegar a qualquer ponto do espaço de busca.

3.3.2 APLICAÇÕES DOS ALGORITMOS GÉNÉTICOS

Os algoritmos genéticos podem ser aplicados em uma vasta gama de problemas, no entanto os problemas devem possuir determinadas características, para seu melhor funcionamento (NARIÑO, 2014). Características essas como: espaço de busca bem delimitado dentro de um intervalo, possibilidade de criar uma função aptidão de forma que avalie de forma correta o quão bom é determinado indivíduo e os indivíduos devem ser transcritos de forma simples no formato de código para facilitar sua implementação (NARIÑO, 2014).

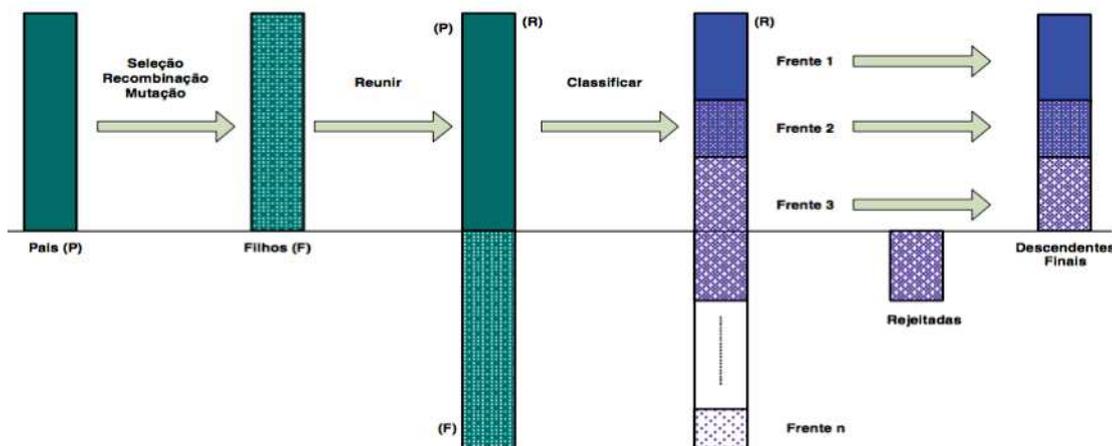
Os AG apresentam algumas vantagens e desvantagens com relação aos demais algoritmos de otimização. Grande parte dos algoritmos de otimização dependem de conhecimento do problema específico para realizar buscas tais como derivadas e integrais do problema, no entanto os AG não necessitam desses tipos de informação para realizar busca. Os outros algoritmos utilizam a abordagem de busca com um único ponto, enquanto os AG usam de forma simultânea uma população de pontos isso implica que os AG podem realizar uma série de projetos ao mesmo tempo, são capazes de sair das soluções locais enquanto os outros métodos geralmente ficam presos em soluções locais, porém os mesmos possuem desvantagens como grande custo computacional para o mesmo convergir e convergência prematura, apesar dessas desvantagens os AG continuam sendo bastante utilizados (NARIÑO, 2014).

O avanço tecnológico levou a criação de várias modificações sobre o algoritmo genético que são utilizados de acordo com o problema de otimização a ser tratado. Dentro desses AG, pode-se destacar alguns: MOGA (*multi-objective genetic algorithm*), NPGA (*niched Pareto genetic algorithm*), NSGA (*non-dominated genetic algorithm*), NSGA-II (NARIÑO, 2014). Baseado em algumas aplicações bem sucedidas como em (MAITELLI; CAVALCANTI, 2006; LEITE, 2016; FILHO PAULO A; POPPI, 2002) foi escolhido o método NSGA-II, para a esse trabalho, o qual será utilizado para obtenção dos parâmetros do GPC.

3.3.3 NSGA-II

O método NSGA-II é um algoritmo de otimização multiobjetivo que tem como característica a utilização do conceito de dominância e elitismo em sua implementação. O NSGA-II classifica a população de indivíduos do problema de acordo com seu grau de dominância e os separa em fronteiras, como mostra a Figura 6 (MARINHO, 2009).

Figura 6 – Esquemática dos procedimentos do NSGA-II.

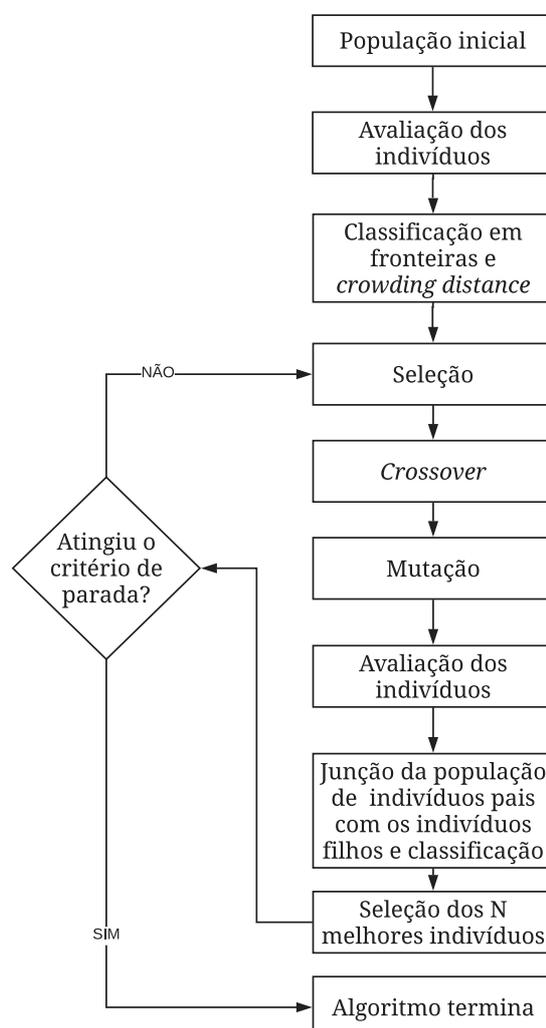


Fonte: (ZINI, 2009)

Em consonância com o critério de dominância este método inclui o conceito de elitismo o qual classifica a população em diferentes grupos ao invés de tratá-las como um só.

A priori todos os indivíduos passam por um processo os quais recebem seus valores de aptidão, após receberem esses valores passam pelo critério de dominância pelos quais serão classificados em fronteiras, *a posteriori* os indivíduos da mesma fronteira, serão ordenados pelo operador de diversidade denominado *Crowding Distance*. A Figura 7 mostra o diagrama de funcionamento do NSGA-II na etapa de seleção (NARIÑO, 2014; MARINHO, 2009).

Figura 7 – Diagrama de blocos do algoritmo NSGA-II, descrevendo o funcionamento da etapa de seleção.

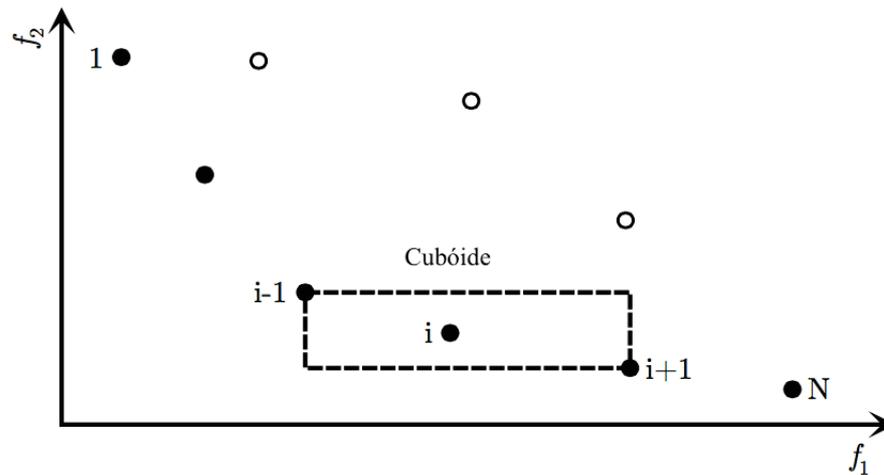


Fonte: Adaptado de (MARINHO, 2009)

O *Crowding Distance* utiliza como métrica a distância de um ponto i com relação a outros pontos na mesma fronteira, quanto mais distante dos outros pontos maior a chance de ser selecionado, com isso o mesmo permite que haja uma diversidade nas soluções impedindo que as mesmas converjam para um único ponto (NARIÑO, 2014; MARINHO, 2009). A Figura 8 ilustra a distância de um ponto i até os pontos localizados em sua extremidade, que formam uma cubóide (MARINHO, 2009), as soluções localizadas nos extremos dos objetivos recebem valor infinito (FORTIN; PARIZEAU, 2013).

O NSGA-II pode ser resumido em alguns passos, *a priori* é criada uma população de tamanho n , os indivíduos dessa população são classificados em fronteiras e distâncias de aglomeração, são aplicados os operadores genéticos de forma a criar uma população de tamanho n , os indivíduos criados são avaliados e unidos aos indivíduos pais de modo a formar uma

Figura 8 – Exemplo da cubóide formada pelo *Crowding Distance*. Os Pontos marcados em círculos preenchidos são soluções da mesma fronteira



Fonte: Adaptado de (FORTIN; PARIZEAU, 2013)

população de tamanho $2n$, novamente são separados em fronteiras e por suas distâncias de aglomeração, os com maior grau de dominância serão selecionados, caso ocorra de ultrapassar o tamanho n os indivíduos com maior distância de aglomeração serão selecionados. Caso o critério de parada seja atingido o código terminara, senão repete-se os passos até a quantidade de vezes pré-determinada, como mostra a Figura 6 (NARIÑO, 2014).

3.4 MÉTRICAS DE CUSTO

As métricas de custo são medidores de desempenho e robustez que podem ser utilizados de forma a qualificar o quão boa a sintonia de um controlador é para determinado objetivo. Os medidores selecionados para este trabalho irão compor a função *fitness* do NSGA-II.

3.4.1 ERRO

Os três medidores de desempenho mais utilizado para o erro são: ISE, IAE (*Integral of the Absolute magnitude of the Error*) e ITAE (*Integral of Time multiplied by Absolute of the Error*).

O ISE tem como característica penalizar erros maiores do que menores (MARLIN, 1995). Os sistemas que adotam este medidor tendem a ter uma resposta transitória mais rápida e uma resposta permanente com uma pequena oscilação.

$$ISE = \sum \varepsilon(t)^2. \quad (3.20)$$

O IAE não pondera os erros em nenhum momento, por isso os sistemas que adotam esse medidor tendem a ter uma resposta transitória mais lenta, entretanto com uma menor oscilação em regime permanente quando comparada com o ISE (MARLIN, 1995).

$$IAE = \sum |\varepsilon(t)|. \quad (3.21)$$

O ITAE tem como característica ponderar os erros que existem mesmo depois de muito tempo, mais que os do começo (MARLIN, 1995). Por isso sistemas que adotam essa métrica possuem resposta inicial lenta que são necessárias para evitar oscilações contínuas em regime permanente, mas que se ajustam muito mais rapidamente que o IAE e o ISE.

$$ITAE = \sum t |\varepsilon(t)|. \quad (3.22)$$

3.4.2 CONTROLE

O TVC é um medidor de desempenho análogo ao ISE. O TVC tem como característica penalizar grandes variações de controle mais do que as pequenas. Ou seja, os sistemas que adotam este medidor como parâmetro para sua sintonia tendem a encontrar soluções com pequenas variações das ações de controle.

$$TVC = \sum \Delta u(t)^2. \quad (3.23)$$

3.4.3 ROBUSTEZ

A característica mais importante de um sistema é a estabilidade e pode ser definida de várias formas dependendo do tipo de sistema a ser analisado (NISE, 2013). Este trabalho a análise se restringe a sistemas lineares e invariantes no tempo.

Um sistema é dito estável se, e somente se, toda sequência limitada de entrada gerar uma sequência limitada de saída, esta definição de estabilidade é conhecida como BIBO (*bounded-input, bounded-output*) (OPPENHEIM; SCHAFER, 2013).

Uma entrada $x(t)$ é dita limitada se houver um valor fixo positivo finito B_x de modo que (OPPENHEIM; SCHAFER, 2013):

$$|x(t)| \leq B_x < \infty.$$

A estabilidade é definida de modo que para toda entrada limitada haja um valor fixo positivo finito B_y de modo que (OPPENHEIM; SCHAFER, 2013):

$$|y(t)| \leq B_y < \infty.$$

Em grande parte das aplicações de controladores, a planta do sistema é instável em malha aberta e o controlador tem o dever de estabilizá-la em malha fechada. Um sistema de controle deve ser projetado de modo que a estabilidade seja preservada mesmo diante a várias incertezas da planta e a isto denomina-se robustez (PEREIRA, 2008).

Determinar a estabilidade de sistemas realimentados nem sempre é simples sem a necessidade de fatoração ou calcular de forma explícita as raízes do denominador da equação de malha fechada do sistema (NISE, 2013). Por isso foram desenvolvidos métodos capazes de determinar a estabilidade sem a necessidade de se calcular os polos do sistema em malha fechada.

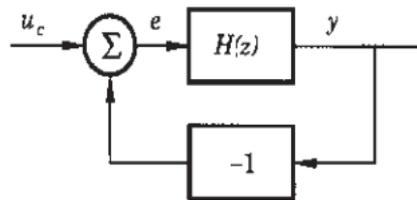
Um dos métodos mais conhecidos é o critério de Nyquist, que é um método semigráfico capaz de determinar a estabilidade dos sistemas em malha fechada apenas investigando as propriedades do gráfico no domínio da frequência, através do diagrama de Nyquist da função de transferência de malha.

3.4.3.1 CRITÉRIO DE NYQUIST

O critério de Nyquist baseia-se no princípio do argumento, da teoria das variáveis complexas, o mesmo é um teste de estabilidade para sistemas de tempo contínuo, mas pode ser reformulado para lidar com sistemas de tempo discreto (ASTROM; WITTENMARK, 1997).

Considere o sistema descrito na Figura 9, o critério de Nyquist pode dizer quantos polos em malha fechada estão dentro do círculo unitário, e a equação de malha fechada desse sistema é dada pela Equação (3.24), e a Equação (3.25) representa a equação de malha desse sistema .

Figura 9 – Sistema de controle em malha fechada



Fonte:(ASTROM; WITTENMARK, 1997)

$$T(z) = \frac{H(z)}{1 + H(z)} \quad (3.24)$$

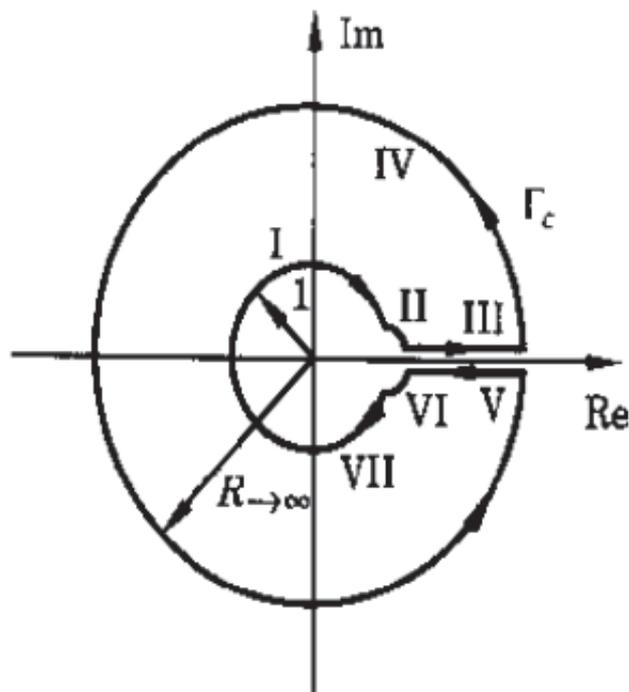
$$L(z) = H(z) \quad (3.25)$$

3.4.3.1.1 DIAGRAMA DE NYQUIST

O diagrama Nyquist é obtido por meio do mapeamento de um contorno Γ_c através da equação de malha. Diferentemente de sistemas de tempo contínuo os quais a estabilidade consiste em conter todos os polos no semi-plano esquerdo, para os sistemas discreto a estabilidade ocorre quando todos os polos da equação de malha fechada estiverem na área interior ao círculo unitário, de modo que o contorno Γ_c escolhido para determina o diagrama de Nyquist é mostrado na Figura 10.

Caso haja polos no círculo unitário o contorno Γ_c deve ser modificado de modo que os mesmo devem ser excluídos, o ponto $z = 1$ também é excluído como mostra a Figura 10 de modo a excluir os integradores em malha aberta (ASTROM; WITTENMARK, 1997).

Figura 10 – Contorno Γ_c



Fonte: (ASTROM; WITTENMARK, 1997)

3.4.3.1.2 DEFINIÇÃO DO CRITÉRIO DE NYQUIST

O princípio do argumento afirma que o número N de circunscrições na direção positiva ao redor do ponto $(-1, j0)$ no diagrama de Nyquist é igual a :

$$N = Z - P \quad (3.26)$$

- N = Número de circunSCRIÇÕES na direção positiva ao redor do ponto $(-1, 0j)$;
- Z = Número de zeros de $1 + H(z)$ externos a trajetória de Nyquist;
- P = Número de pólos de $1 + H(z)$ externos a trajetória de Nyquist.

A condição de estabilidade de acordo com o critério de Nyquist é definida como: $N = -P$. Isto quer dizer que um sistema em malha fechada será estável se o diagrama de Nyquist da função $H(z)$ circunscrever o ponto $(-1, 0j)$ um número de vezes igual ao número de polos de $H(z)$ que estão externos a trajetória de Nyquist, e caso exista esta circunSCRIÇÃO, deve ser no sentido contrário ao da trajetória adotada (GOLNARAGHI; KUO, 2012).

3.4.3.1.3 ESTABILIDADE RELATIVA

Na maioria das vezes, saber apenas se o sistema é estável não é suficiente, mas sim o quão estável o mesmo é, e essa condição comumente é chamada de estabilidade relativa (GOLNARAGHI; KUO, 2012). A quantificação dessa estabilidade é de suma importância, tendo em vista que um sistema de controle deve ser projetado de modo que a estabilidade seja preservada mesmo diante a varias incertezas da planta, e esse grau de liberdade é o que considera-se como robustez (PEREIRA, 2008).

Além de fornecer as informações sobre estabilidade absoluta o diagrama de Nyquist consegue fornecer informações sobre a estabilidade relativa, uma forma de mensurar esta medida é verificar através do diagrama de Nyquist da função $H(z)$, o quão próximo está do ponto $(-1, 0j)$, as métricas mais utilizadas para fornecer informações sobre a estabilidade relativa do sistema são a margem de ganho, margem de fase e sensibilidade máxima.

A margem de ganho, M_g , é o módulo da distância entre o ponto $(-1, 0j)$ e o ponto onde o diagrama cruza o eixo real negativo, o ponto desse cruzamento é conhecido como cruzamento de fase, e a frequência que ele ocorre é denominada de frequência de cruzamento de fase, ω_f e é mostrada na Figura 11. E pode ser interpretada como o ganho que pode ser adicionado a malha antes do sistema em malha fechada se tornar instável (GOLNARAGHI; KUO, 2012).

A margem de fase, M_f é definida como o ângulo, em graus, que o diagrama $H(z)$ deve ser girado em relação a origem de modo o cruzamento de ganho passe pelo ponto $(-1, 0j)$ (GOLNARAGHI; KUO, 2012). Cruzamento de ganho é definido como o ponto o qual a magnitude dose diagrama de $H(z)$ é igual a 1, e a frequência em que ocorre é definida como frequência de cruzamento de ganho, ω_g e é mostrada na Figura 11.

A função de sensibilidade mostra o efeito da atenuação das perturbações na saída do sistema. A sensibilidade máxima, M_s , mostrada na Figura 11, é medida pelo inverso da distancia entre o ponto mais próximo do diagrama de Nyquist ao ponto $(-1, 0j)$, e é uma variável importante pois é onde ocorre a maior amplificação das perturbações, de forma que quanto

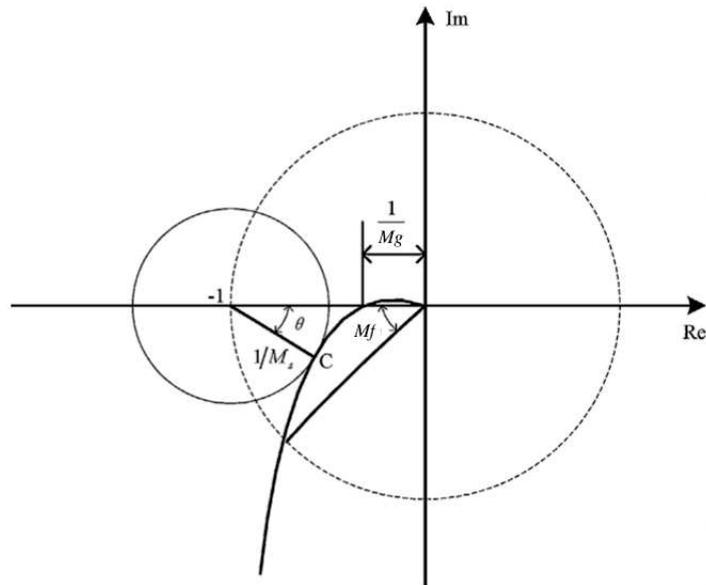
mais distante do $(-1, 0j)$, menor é a sensibilidade máxima do sistema e menos suscetível a perturbações (ASTROM; RICHARD, 2008).

A margem de ganho e a margem de fase estão relacionadas com a sensibilidade máxima, a relação entre elas é dada pelas Equações (3.27) e (3.28) (LI et al., 2015).

$$M_g > \frac{M_s}{M_s - 1}, \quad (3.27)$$

$$M_f > 2 \operatorname{sen}^{-1} \left(\frac{1}{2M_s} \right). \quad (3.28)$$

Figura 11 – Diagrama de Nyquist



Fonte: Adaptado de (LI et al., 2015)

4 ABORDAGEM PROPOSTA

O presente trabalho visa encontrar os valores dos parâmetros de um controlador GPC, aplicado a um sistema monovariável e estável em malha aberta de forma a otimizar os objetivos pré-definidos, que são: diminuição da variação das ações de controle, diminuição do erro e maior robustez do controlador. A determinação desses parâmetros foi feita utilizando o algoritmo evolutivo multiobjetivo NSGA-II.

4.1 NSGA-II APLICADO A SINTONIA DO GPC

Um dos passos mais importantes da otimização utilizando o NSGA-II consiste em definir uma forma de representar os indivíduos da população, assim como definir as funções de *fitness*. Os parâmetros a serem determinados nesse trabalho serão: N_y, N_u, λ ¹. Os mesmos são representados como um vetor de três elementos, como mostra a Equação (4.1). Cada indivíduo será avaliado pelas funções de *fitness* conforme apresentado pela Equação (4.2) e (4.3). Na Equação (4.2) cada métrica é tratada independentemente apresentando a perspectiva multiobjetivo. Na Equação (4.3) as métricas são somadas e consideradas como um objetivo único, representando uma perspectiva mono-objetivo.

$$X = [\lambda, N_y, N_u] \quad (4.1)$$

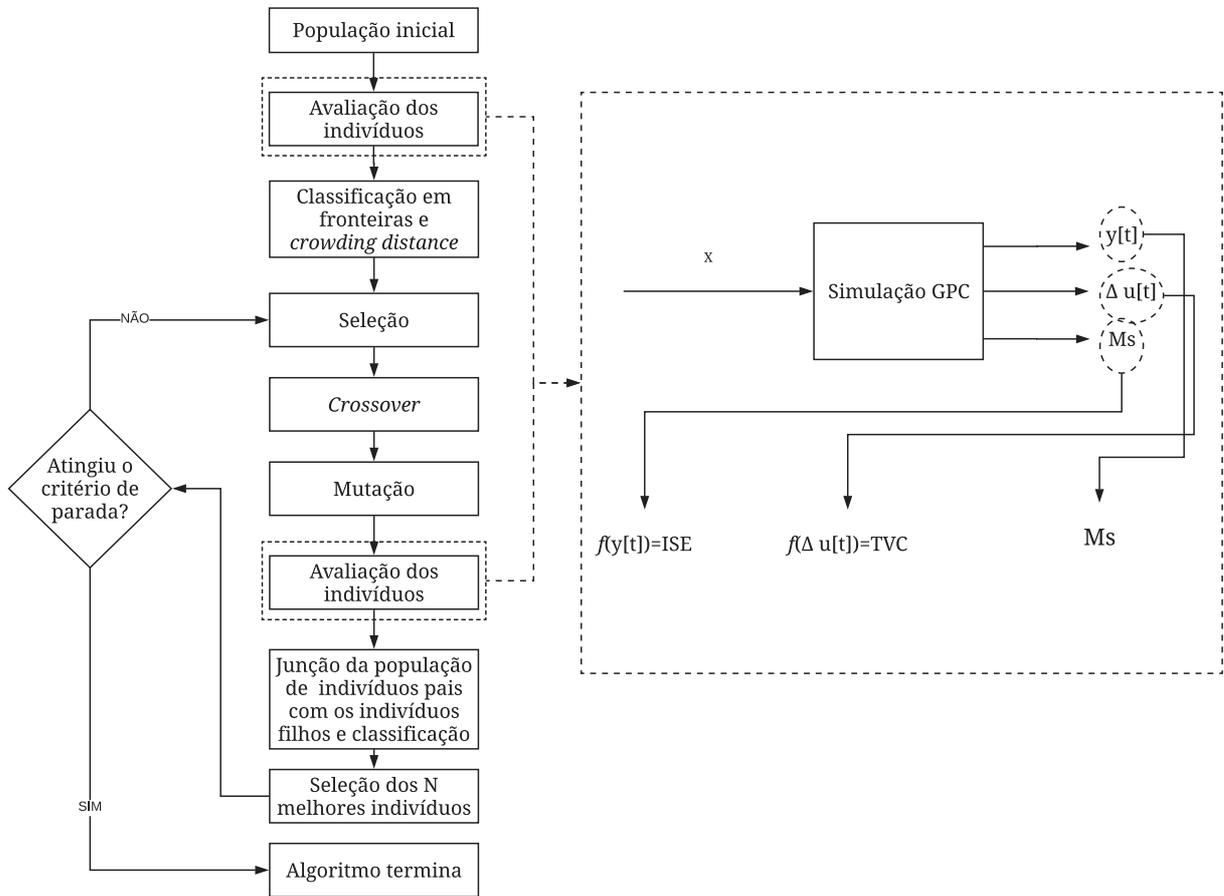
$$f(\lambda, N_y, N_u) = [ISE, TVC, Ms] \quad (4.2)$$

$$f(\lambda, N_y, N_u) = [ISE + TVC + Ms] \quad (4.3)$$

A obtenção dos valores das funções de *fitness* de cada indivíduo é obtida realizando a simulação do sistema controlado pelo GPC, que tem seus parâmetros sintonizado por cada indivíduo da população. Após a simulação é possível obter os os valores valores da saída, da variação das ações de controle e da Ms que são os valores utilizados para calcular as funções *fitness*. A Figura 12 ilustra como é realizado o processo de otimização dos parâmetros GPC.

¹ Devido a utilização da Métrica de desempenho ISE, adotou-se $N1 = d + 1$, visto que a saída do sistema no regime transitório é importante

Figura 12 – Diagrama de blocos do algoritmo NSGA-II utilizado para a Otimização dos parâmetros do GPC.

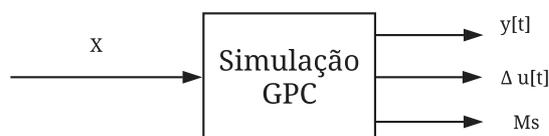


Fonte: Adaptado de (MARINHO, 2009)

4.2 SIMULAÇÃO DO GPC

A simulação de um determinado sistema controlado pelo GPC é realizada pela função Simulação GPC, a mesma foi implementada de forma que recebe como entrada um conjunto de parâmetros que é aplicado ao GPC, e retorna o vetor de saída, o vetor das variações das ações de controle e da M_s do sistema controlado, como ilustrado na Figura 13.

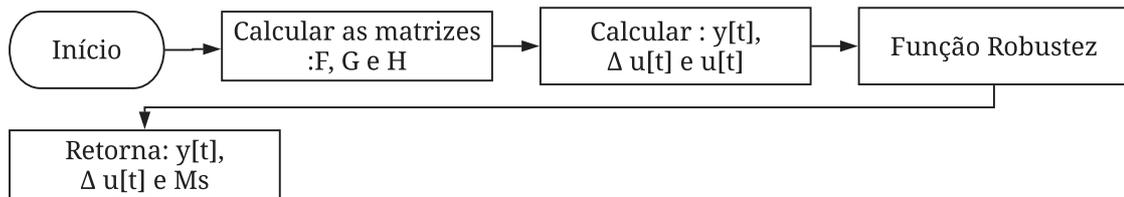
Figura 13 – Entrada e Saída da Função Simulação GPC



Fonte: Autoria Própria

A simulação do GPC pode ser dividida em duas partes: (I) calcular das matrizes F, G e H , que são utilizadas para determinar o valor da variação da ação de controle ótima para cada instante de tempo descrita pela Equação (3.18); (II) calcular o valor da sensibilidade máxima e das margens de ganho e de fase. A Figura 14 mostra o fluxograma da função simulação GPC.

Figura 14 – Fluxograma da Função Simulação GPC



Fonte: Autoria Própria

A primeira parte consiste em calcular as matrizes F, G e H do GPC, utilizando as expressões encontradas nos APÊNDICES A, B e C. Após o cálculo das matrizes é possível determinar as variações ótimas de controle para cada instante utilizando a Equação (3.18).

Após a determinação da variação ótima da ação de controle é possível determinar a ação ótima de controle através da Equação (3.19) e o valor da saída para o instante de tempo seguinte é determinado utilizando a Equação (3.1), a cada instante de tempo o valor calculado da ação de controle ótima, da variação de controle ótima e da saída são salvos em vetores distintos.

A segunda parte da função simulação GPC consiste em encontrar a sensibilidade máxima do sistema, para isso é implementada a função robustez que tem como entrada os coeficientes da função de transferência de malha e retorna a sensibilidade máxima e as margens de fase e de ganho.

Todavia para o cálculo desta métrica é necessário encontrar a função de transferência de malha para esboçar o diagrama de Nyquist. Portanto para encontrar a equação de malha para o GPC utiliza-se o projeto de controlador RST, que é representado em diagrama de blocos mostrado na Figura 15.

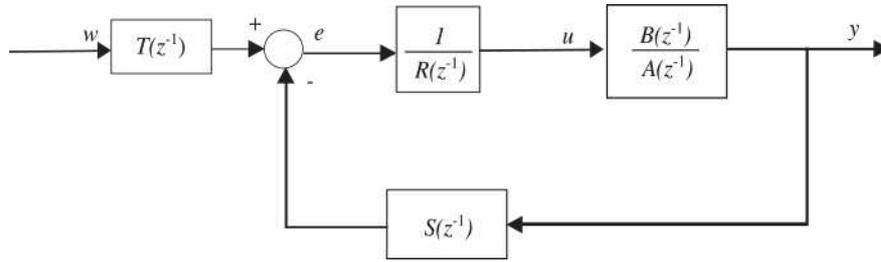
O projeto RST é válido para controladores que tem uma saída $u[t]$, e duas entradas : $w[t]$ e $y[t]$. A expressão pra esse projeto de controlador é dada pela Equação (4.4) (ASTROM; WITTENMARK, 1997).

$$R(z^{-1})u(t) = T(z^{-1})w(t) - S(z^{-1})y(t). \quad (4.4)$$

Pode-se perceber a similaridade da Equação (3.18) com a Equação (4.4). Portanto visando encontrar a expressão para R, S e T a Equação (3.18) pode ser reescrita da seguinte forma:

$$\Delta U = (G^T G + \lambda I)^{-1} G^T (\hat{W} - \hat{Y}_l),$$

Figura 15 – Diagrama de Blocos RST



Fonte: Autoria Propria

$$K_{GPC} = (G^T G + \lambda I)^{-1} G^T.$$

Considerando K_{GPC} como a primeira linha da matriz K_{GPC} , o termo $\Delta u[t]$ pode ser isolado:

$$\Delta u(t) = K_{GPC}(\hat{W} - \hat{Y}_l)$$

$$\Delta u(t) = K_{GPC}\hat{W} - K_{GPC}\hat{Y}_l$$

$$\Delta u(t) = K_{GPC}\hat{W} - K_{GPC}(F\hat{Y}_p + H\Delta U_p)$$

$$\Delta u(t) = K_{GPC}\hat{W} - K_{GPC}F\hat{Y}_p - K_{GPC}H\Delta U_p$$

$$(1 + K_{GPC}H)\Delta u(t) = K_{GPC}\hat{W} - K_{GPC}F\hat{Y}_p$$

Logo comparando a Equação (3.18) com a Equação (4.4) obtêm-se que:

$$R(z^{-1}) = (1 + K_{GPC}H)\Delta, \quad (4.5)$$

$$T(z^{-1}) = K_{GPC}, \quad (4.6)$$

$$S(z^{-1}) = K_{GPC}F. \quad (4.7)$$

A equação de malha fechada do projeto é dada por:

$$\frac{Y(z^{-1})}{W(z^{-1})} = \frac{BT}{1 + \frac{BS}{AR}}, \quad (4.8)$$

Substituindo as Equações (4.5), (4.6) e (4.7) na Equação (4.8) obtêm-se a equação de malha fechada para o GPC que é descrita como :

$$\frac{Y(z^{-1})}{W(z^{-1})} = \frac{K_{GPC}T}{1 + \frac{BK_{GPC}F}{A((1+K_{GPC}H)\Delta)}}. \quad (4.9)$$

A equação de malha pode ser obtida por meio da Equação (4.9) e é definida como:

$$\frac{BK_{GPC}F}{A((1 + K_{GPC}H)\Delta)} \quad (4.10)$$

Os coeficientes da Equação de malha fechada são passados como parâmetros para a função de robustez e a mesma retorna a sensibilidade máxima, e os ganho de margem e de fase. Após isso a função simulação retorna os vetores de saída e de variação de controle juntamente com a sensibilidade máxima que é a métrica de robustez utilizada neste trabalho.

4.2.1 FUNÇÃO ROBUSTEZ

A função robustez recebe como parâmetro os coeficientes da função de transferência de ganho de malha do sistema e retorna os valores da sensibilidade máxima e das margens de ganho e margem de fase. A Figura 16 mostra o diagrama de entrada e saída da função robustez.

Figura 16 – Entrada e Saída da Função Robustez



Fonte: Autoria Própria

O primeiro passo da implementação da função robustez consiste em criar um vetor que represente o caminho Γ_c , o mesmo varia de 0 a 2π (ASTROM; WITTENMARK, 1997), cada ponto espaçado de 0,0001 o que totaliza 62832 pontos, que são utilizados para calcular o diagrama de Nyquist. Em seguida é retirado do vetor de caminho todos os polos da função de ganho de localizados no círculo unitário.

Para o cálculo dos pontos do diagrama de Nyquist é utilizado uma função capaz de calcular a resposta frequência de um filtro digital para um dado conjunto de frequências.

Após o cálculo da resposta frequencial calcula-se as margem de ganho e a margem de fase. Nesta etapa da implementação verificou-se que há sistemas que possuem múltiplos cruzamentos no eixo real negativo, gerando múltiplas margens de ganho, assim como pôde-se observar sistemas em que o diagrama cortava várias vezes o círculo unitário gerando múltiplas margens de fase, tornando incompleta a definição tradicional de margem de ganho e margem de fase descrita na maioria das literaturas.

A definição tradicional de M_g pode ser entendida como a quantidade de ganho que pode ser aumentado ou diminuído antes que o sistema mude sua condição de estabilidade, todavia esta definição contem apenas uma direção e apenas um M_g é definido para cada sistema, esta definição não é completa, pois não aborda a questão da mudança na condição de estabilidade por variação de ganho (BAVAFA-TOOSI, 2015).

Uma definição mais abrangente para o M_g é reformulada e é descrita como: um sistema seja ele fase mínima ou fase não mínima possui um (M_g^+) e um (M_g^-). O M_g principal é o que

possuir o menor valor absoluto entre os dois, pois o sistema torna-se mais sensível a perturbação no ganho dessa direção (BAVAFA-TOOSI, 2015).

No caso da margem de fase, não faz sentido, e de fato é impossível que um sistema se torne instável em ambas as direções, tanto crescente quanto decrescente, ou seja, apenas uma margem é definida para cada sistema (BAVAFA-TOOSI, 2015).

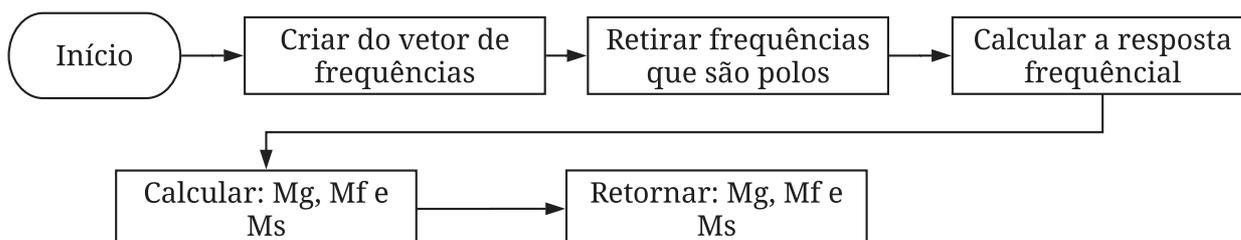
A margem de fase é definida como a quantidade de fase que pode ser aumentada ou diminuída (na frequência de cruzamento de ganho) antes que o sistema se torne instável, quando associada há sistema de fase mínima ou sistema de fase não mínima a margem de ganho pode ser negativa ou positiva (BAVAFA-TOOSI, 2015). A margem de fase no caso de múltiplos cruzamentos é aquela que tem o menor M_f em valor absoluto, pois é a que torna o sistema mais sensível a perturbações em fase (BAVAFA-TOOSI, 2015).

Para o cálculo da margem de ganho verifica-se todos os pontos em que houve cruzamento no eixo real negativo de diagrama de Nyquist tanto no sentido positivo quanto negativo, esses pontos de cruzamento são salvos em um vetor, após encontrar todos pontos de cruzamento é obtido um novo vetor com o valor absoluto da resposta frequencial do sistema para todas as frequências de cruzamento e a que possuir o menor valor absoluto é considerada a margem de ganho.

Para o cálculo da margem de fase verifica-se todos os pontos em que houve cruzamento no círculo unitário da curva do diagrama de Nyquist, esses pontos de cruzamento são salvos em um vetor, após encontrar todos pontos de cruzamento é obtido um novo vetor com as margens de fase. O vetor de margens de fase é obtido subtraindo 180 da fase da resposta frequencial do sistema para todas as frequências de cruzamento. A margem de fase que possuir o menor ângulo em modulo é considerada a margem de fase do sistema.

A sensibilidade máxima é obtida calculando a distância Euclidiana de todos os pontos da resposta frequencial ao ponto $(-1, 0j)$, após o cálculo dessas distâncias a sensibilidade máxima é escolhida pelo ponto que apresentar a menor distância. A Figura 17 mostra o fluxograma de forma resumida da função robustez.

Figura 17 – Fluxograma Função Robustez



Fonte: Autoria Própria

5 AVALIAÇÃO EXPERIMENTAL

A seguir são apresentados detalhes adotados para a implementação e avaliação da abordagem proposta.

5.1 ASPECTOS DE IMPLEMENTAÇÃO

Para a sintonia do GPC com o NSGA-II utilizou-se o *Framework* Pygmo. Neste *Framework* é necessário realizar a implementação de três classes : *Algorithm*, *Problems* e *Population*. A classe *Algorithm* é uma representação genérica de um algoritmo de otimização, portanto faz-se necessário a implementação de uma classe que descreve o funcionamento do algoritmo que então é passado para *Algorithm* que permite a interface entre as demais classes, todavia o Pygmo já fornece algumas classes de algoritmos de otimização, dentre eles o NSGA-II.

Assim como *Algorithm* é uma representação genérica, *Problems* também é, sendo necessário criar uma classe que descreva o problema a ser otimizado, que será utilizada para construir um *Problem* que por sua vez é o que um *Algorithm* pode resolver. Nesta classe que descreve o problema é obrigado serem implementados dois métodos.

O método *fitness()* o qual conterà as funções *fitness* do problema a ser otimizado, neste método será chamada função simulação do GPC. O indivíduo que é composto pelos parâmetros do GPC será passado por parâmetro que retornará os valores $y(t)$, $\Delta u(t)$ e Ms com esses valores as funções de *fitness* são calculadas.

O *get_bound()* define os limites dos indivíduos da população, ou seja, é o método responsável por definir o espaço de busca. Neste trabalho o espaço de busca foi definido pelo valor máximo da união dos conjuntos dos valores de referência descritos na Tabela 2. Além desses dois métodos outros dois foram implementados, o método *get_nix()* e o método *get_nobj()*.

O primeiro é responsável por restringir que o valor dos dois últimos termos do vetor de indivíduos sejam inteiros, pois os mesmos representam Ny e Nu . O segundo é responsável por definir a quantidade de objetivo do problema.

A classe *Population* cria uma população baseada em um *Problem* e a evolui utilizando um *Algorithm*. Por meio da implementação apresentada a otimização multiobjetivo é realizada de acordo com os seguintes passos:

- Calcular os tempos de assentamento do sistema de forma a calcular os valores da Tabela 2;
- Definir como limite máximo o maior valor de cada parâmetro e como mínimo o menor o valor de cada parâmetro e aplica-los no método *get_bound()*;
- Definir como algoritmo o NSGA-II para a classe *Algorithm* ;

Tabela 2 – Limites dos Parâmetros do GPC

	N_y	N_u	λ
[1]	$\frac{t_{80}}{T_s}$	$N_y/4$	1
[2]	$\frac{t_{80}}{T_s} \leq N_y \leq \frac{t_{90}}{T_s}$	$3 \leq N_u \leq 5$	1
[3]	10	1	0
[4]	$N_u + d + 1 \leq N_y \leq \frac{t_{95}}{T_s}$	$na + 1$	tentativa e erro
[4]	$d + 1 \leq N_y \leq \frac{t_{95}}{T_s}$	1	tentativa e erro

[1] (TRIERWEILER; FARINA; DURAIKI, 2001)

[2] (YAMASHITA; ZANIN; ODLOAK, 2016)

[3] (CLARKE; MOHTADI; TUFFS, 1987)

[4] (MCINTOSH, 1989; MCINTOSH, 1991)

- Definir o vetor de parâmetros como o primeiro termo sendo do tipo ponto flutuante e os dois últimos como inteiros pelo método *get_nix()*;
- Definir como um problema com 3 objetivos pelo método *get_nobj()*
- Definir como funções de *fitness* a Equação 4.2 no método *fitness()*;
- Aplicar o algoritmo genético de forma e obter uma população final de tamanho 100 X 6, onde os 3 primeiros termos da matriz são os parâmetros e os últimos os objetivos .

A otimização mono-objetiva é realizada conforme os passos demonstrados abaixo:

- Calcular os tempos de assentamento do sistema de forma a calcular os valores da Tabela 2;
- Definir como limite máximo o maior valor de cada parâmetro e como mínimo o menor o valor de cada parâmetro e aplica-los no método *get_bound()*;
- Determinar como o algoritmo de otimização um algoritmo genético simples para a classe *Algorithm* ;
- Define o vetor de parâmetros como o primeiro termo sendo do tipo ponto flutuante e os dois últimos como inteiros pelo método *get_nix()*;
- Define como problema com 1 objetivo pelo método *get_nobj()*
- Define como função de *fitness* a Equação 4.3;
- Aplicar o algoritmo genético de forma a obter uma população final de tamanho 100 X 4, onde os três primeiros termos da matriz são os parâmetros e os últimos os objetivos ;

5.2 PARÂMETROS DO NSGA-II

A Tabela 3 apresenta os valores definidos para os parâmetros do NSGA-II, além das estratégias adotadas para os operadores de *crossover*, mutação, seleção.

Tabela 3 – Parâmetros do NSGA-II

	População Características
Tamanho	100
Gerações	100
Taxa de Cruzamento	95%
Taxa de Mutação	1%
Método de Seleção	Torneio
Método de <i>Crossover</i>	SBX
Método de Mutação	Polinomial

Durante a avaliação foram realizados 30 execuções independentes, utilizando os mesmos parâmetros apresentados na Tabela 3. O objetivo de executar vários experimentos independentes é avaliar o comportamento do algoritmo.

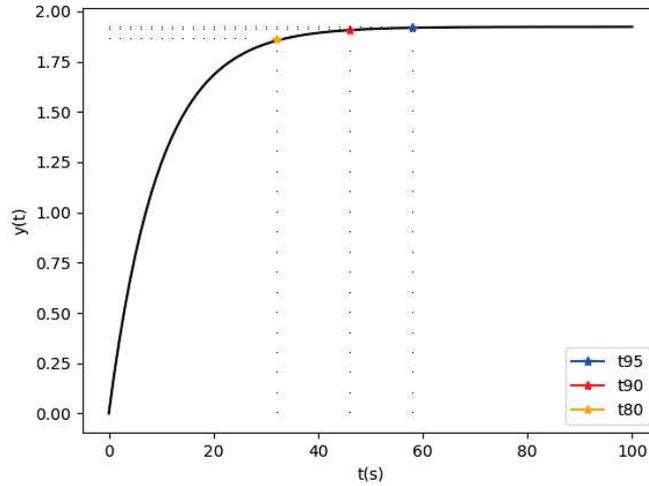
5.3 ESTUDO DE CASO

O sistema escolhido para este trabalho é descrito pela Equação (5.1) que representa um Motor DC (ABATE et al., 2017):

$$H(z^{-1}) = \frac{0,1898z^{-1} + 0,0002z^{-2}}{1 - 0,9012z^{-1} - 1,000610^{-16}z^{-2}} \quad (5.1)$$

A saída do sistema em malha aberta descrito na Equação 5.1 quando submetido ao sinal degrau unitário é mostrada na Figura 18, os valores obtidos pela resposta ao degrau unitário foram utilizados para calcular a Tabela 2, resultando na Tabela 4.

Figura 18 – Resposta ao degrau unitário.



Fonte: Autoria Própria

Tabela 4 – Limites dos Parâmetros do GPC para o sistema estudado

	N_y	N_u	λ
[1]	16	4	1
[2]	$16 \leq N_y \leq 24$	$3 \leq N_u \leq 5$	1
[3]	10	1	0
[4]	$5 \leq N_y \leq 29$	4	tentativa e erro
[4]	$1 \leq N_y \leq 29$	1	tentativa e erro

[1] (TRIERWEILER; FARINA; DURAIKI, 2001)

[2] (YAMASHITA; ZANIN; ODLOAK, 2016)

[3] (CLARKE; MOHTADI; TUFFS, 1987)

[4] (MCINTOSH, 1989; MCINTOSH, 1991)

Fonte: Autoria Própria

Por tanto os limites utilizados no método *get_bound()* de forma a restringir o espaço de busca estão descritos na tabela 5. O valor 10 para o λ foi escolhido de forma arbitrária visto que o limite superior para este parâmetro baseado nas referencias (MCINTOSH, 1989; MCINTOSH, 1991) não é especificado.

Tabela 5 – Limites dos Parâmetros do método *get_bound()*.

Motor DC	
N_y	1 † 29
N_u	1 † 5
λ	0 † 10

Fonte: A autoria Própria

5.4 OTIMIZAÇÃO COM PONDERAÇÃO

Tendo em vista que os valores do *ISE* e *TVC* normalmente são muito maiores que os de *Ms* realiza-se também as otimizações com as funções de *fitness* ponderadas pelos possíveis maiores valores de cada objetivo limitados no espaço de busca.

Para o valor de ponderação do *ISE* será obtido simulando o sistema para o caso que possivelmente é o que tem-se o maior *ISE*, que vai ter como os parâmetros : $N_y = 1$, $N_u = 1$ e $\lambda =$ Limite Máximo. O valor de N_y e N_u foram escolhidos como 1 pois nesse caso a otimização do *GPC* para obter-se a ação de controle ótima concentra-se em encontrar valor ótimo para aquele instante e não pensando em todo processo, e o valor de λ é escolhido de modo que na otimização do *GPC* as otimização da variações ações tenham fator de ponderação maior que o erro, o que acarreta em ações de controles melhores, porém com um maior erro.

Para o valor de ponderação do *TVC* será obtido simulando o sistema para o caso que possivelmente é o que tem o maior *TVC*, que vai ter como os parâmetros : $N_y =$ Limite Máximo, $N_u =$ Limite Máximo e $\lambda = 0$. O valor de N_y e N_u foram escolhidos como os valores máximos pois nesse caso a otimização do *GPC* o valor de N_y grande permite a lei de controle do *GPC* determinar a ação de controle ótima não visando apenas aquele instante, mas por todo um horizonte, e o valor de N_u grande permite que o *GPC* calcule a ação de controle mais efetiva para todo um horizonte de controle, o que resulta em maiores valor do *TVC*. O valor de $\lambda = 0$ aumenta ainda mais o valor do *TVC* tendo em vista que na otimização do *GPC* a variação das ações de controle são desprezadas e o mesmo visa apenas minimizar o erro, acarretando maiores variações das ações de controle.

O valor da ponderação para *Ms* é definido como três, este valor foi definido de forma empírica verificando que os valores da mesma normalmente não ultrapassam este valor.

$$f(\lambda, N_y, N_u) = \left[\frac{ISE}{\rho}, \frac{TVC}{\xi}, \frac{Ms}{3} \right] \quad (5.2)$$

$$f(\lambda, N_y, N_u) = \left[\frac{ISE}{\rho} + \frac{TVC}{\xi} + \frac{Ms}{3} \right] \quad (5.3)$$

Para o estudo de caso utilizado neste trabalho, obteve-se como fatores ponderação das funções de *fitness* os valores apresentados na Tabela 6. Os passos para a otimização ponderada dar-se de forma análoga a descrita na seção 5.1, modificando apenas as funções de *fitness*.

Tabela 6 – Valores de ponderação

Motor DC	
ρ	137,334
ξ	480,061

Fonte: Aatoria Propria

6 RESULTADOS

Neste capítulo são apresentados os resultados da otimização mono-objetivo e multiobjetivo ponderados e não ponderados.

A primeira parte consiste em apresentar os resultados das otimizações multiobjetivo, *a priori* uma visão mais geral dos resultados são apresentados como o número de soluções localizadas na fronteira de Pareto, as melhores soluções de o todo conjunto as melhores soluções por experimento e a autonomia do algoritmo de encontrar soluções ótimas em apenas um experimento. *A posteriori* uma análise mais aprofundada do conjunto de soluções é feita por meio do BoxPlot¹, visando analisar a capacidade da otimização em encontrar soluções próximas do ponto ótimo.

A segunda parte consiste em mostrar os resultados das otimizações mono-objetivo, a análise dos resultados para esse caso é feita de modo diferente da multiobjetivo, *a priori* é mostrado uma visão geral das soluções e *a posteriori* é mostrada a melhor soluções por experimento e comparadas com as demais.

A terceira parte consiste em comparar os métodos de otimização entre-si, por meio das melhores soluções de cada método e *a posteriori* com os métodos de sintonias de referência. Por fim, são apresentadas as simulação do sistema descrito no estudo de caso quando controlado pelo GPC sintonizado pelas melhores soluções obtidas das otimização multiobjetivo e mono-objetivo.

Vale ressaltar que as otimizações que tiveram a função de *fitness* ponderadas, tiveram aqui os valores reponderados após a otimização de modo tornar possível a análise dos dados quando comparados aos demais.

A distância Euclidiana, ED , utilizada neste capítulo refere-se a distância entre os objetivos de cada indivíduo ao ponto ótimo teórico (0,0,1). O mesmo corresponde ao ponto onde ISE é igual a zero, TVC igual a zero e Ms igual um, Ms igual a um representa o ponto onde a robustez do controlador é infinita.

6.1 OTIMIZAÇÃO MULTIOBJETIVO

Nesta são analisados os resultados das otimizações multiobjetivos.

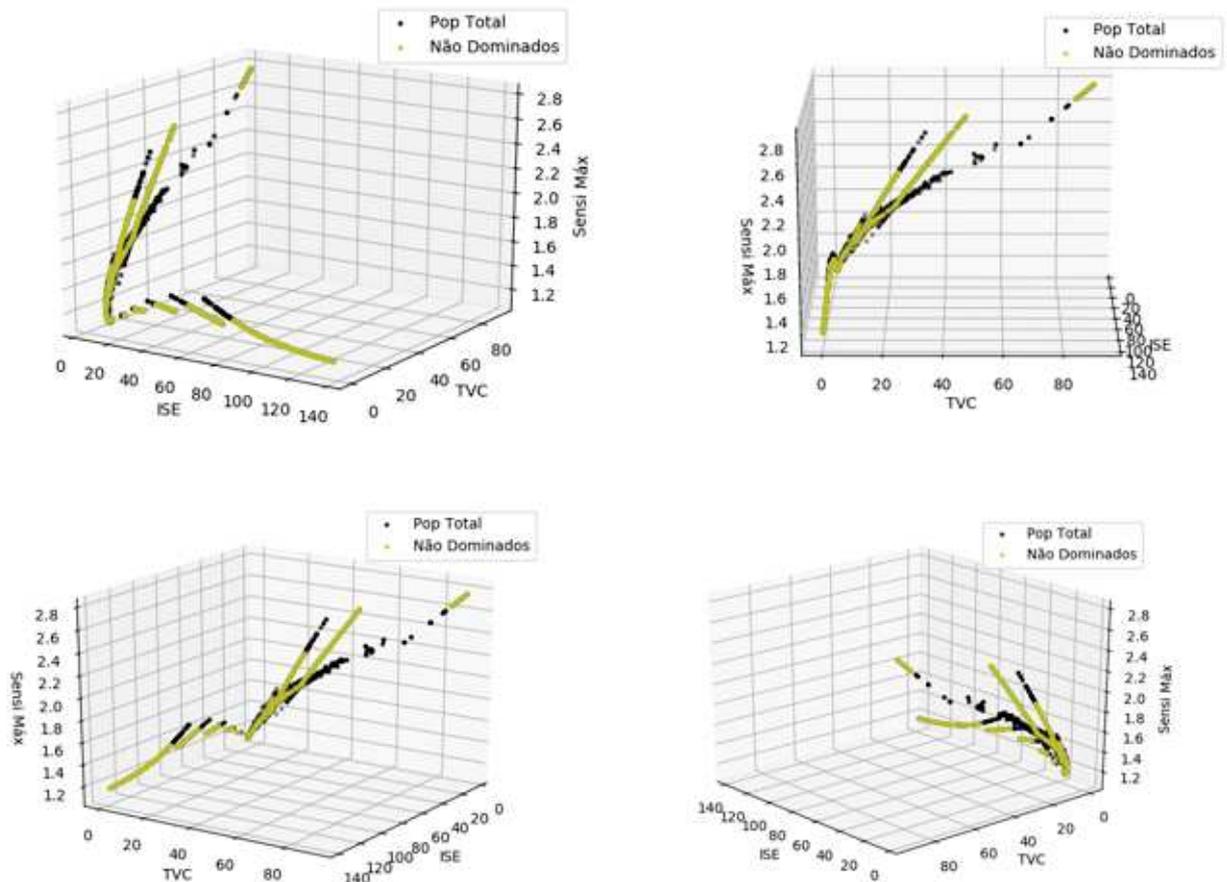
6.1.1 ANALISE GERAL DA POPULAÇÃO: OTIMIZAÇÃO NÃO PONDERADA

O conjunto de soluções resultantes após os 30 experimentos é mostrado na Figura 19 em várias perspectivas. Entre as 3000 soluções obtida, 1941 são soluções não dominadas, ou

¹ É um método gráfico utilizado para analisar um conjunto de dados, o mesmo divide a população em quartis e consegue fornecer informações como localização e dispersão dos dados.

seja, são as soluções localizadas na fronteira de Pareto, o que representa aproximadamente 65% do conjunto total.

Figura 19 – Conjunto de soluções dos experimentos multiobjetivo não ponderado

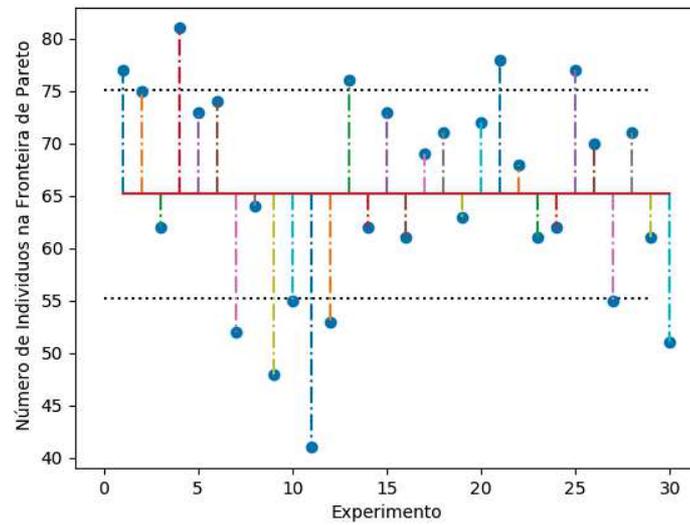


Fonte: Autoria Própria

Analisando o conjunto de solução por experimento de forma separada, pôde-se observar que os mesmos obtiveram uma taxa de 100% de suas soluções não dominadas, todavia a taxa de soluções localizadas na fronteira de Pareto por experimento quando analisado o conjunto total de soluções foi em média de 65% com um desvio padrão de 10% das soluções.

Em outras palavras, cada experimento contribuiu com aproximadamente 65 ± 10 soluções para a fronteira de Pareto das soluções analisadas. A Figura 20 mostra o número de soluções localizadas na fronteira de Pareto por experimento realizado, a linha vermelha representa a média e a linha tracejada representa o desvio padrão.

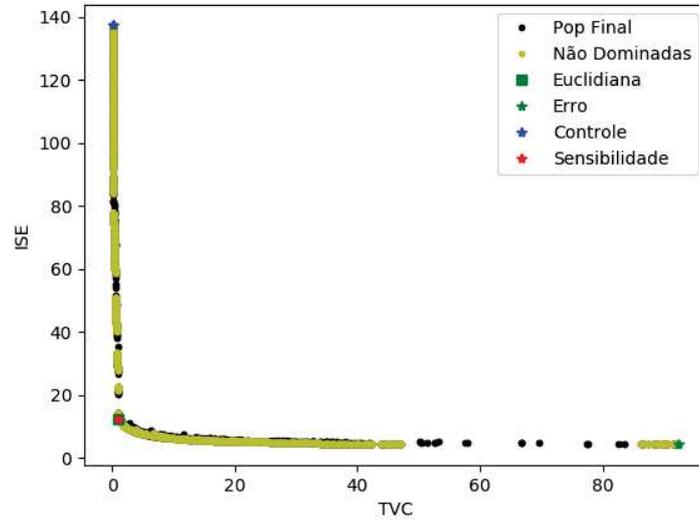
Figura 20 – Número de soluções localizadas na fronteira de Pareto por experimento multiobjetivo não ponderado.



Fonte: Autoria Própria

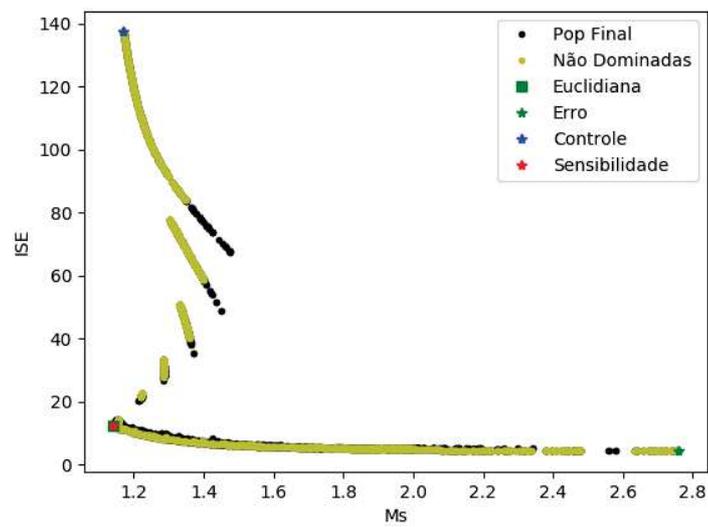
Entre o conjunto total de soluções, as melhores soluções para os critérios de menor *ISE*, menor *TVC*, menor *Ms* e menor *ED*, são apresentadas nas Figuras 21, 22 e 23. Além das melhores soluções, as figuras apresentam o conjunto total de soluções e o conjunto de soluções não dominadas ou soluções localizadas na fronteira de Pareto .

Figura 21 – (ISE vs TVC) multiobjetivo não ponderado.



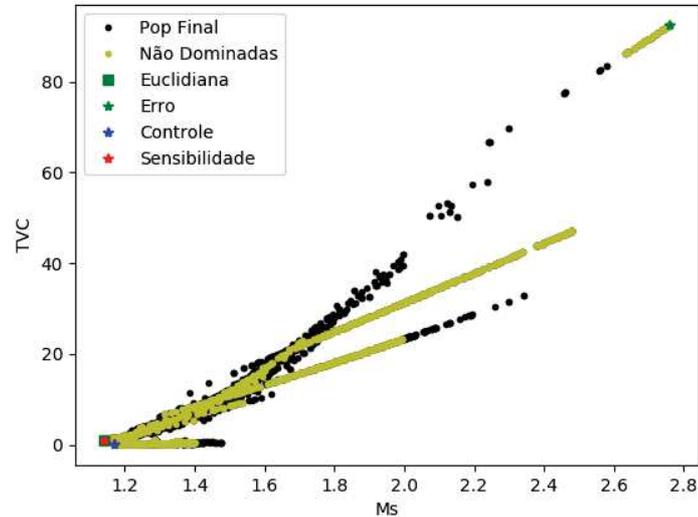
Fonte: Autoria Própria

Figura 22 – (ISE vs M_s) multiobjetivo não ponderado.



Fonte: Autoria Própria

Figura 23 – (TVC vs M_s) multiobjetivo não ponderado.



Fonte: Autoria Própria

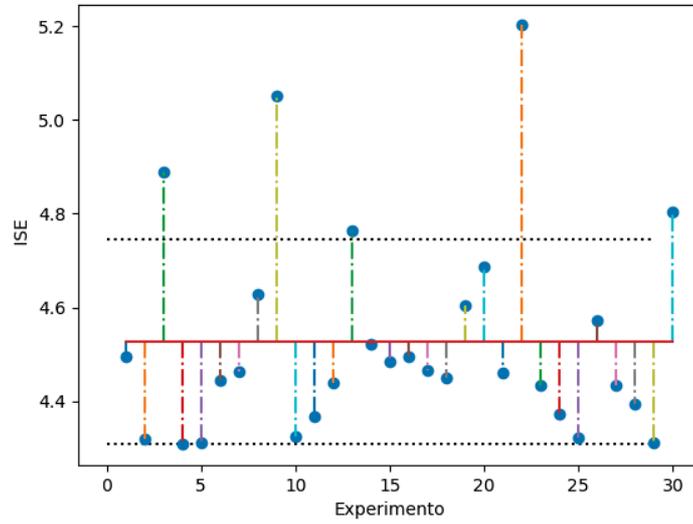
Pode-se observar que as soluções com menores ISE possuem maior TVC , o que implica que para obter o erro em regime transitório baixo, ações de controle efetivas tiveram que ser realizadas, acarretando grandes variações, essas ações de controle efetivas também impactam na robustez do controlador, tornando menos estável como pode-se observar na Figura 22.

De forma análoga as soluções com menores TVC , foram as soluções que possuíram maior ISE, isso implica que uma ação de controle praticamente constante resulta em um sistema com maior erro, porém acarreta em uma maior estabilidade como mostra a Figura 23.

Apesar de ter-se soluções com melhor resultado para cada objetivo separadamente a solução que melhor consegue se aproximar do valor ótimo de todos os objetivos é a solução da menor distância euclidiana, como ilustrado nas figuras.

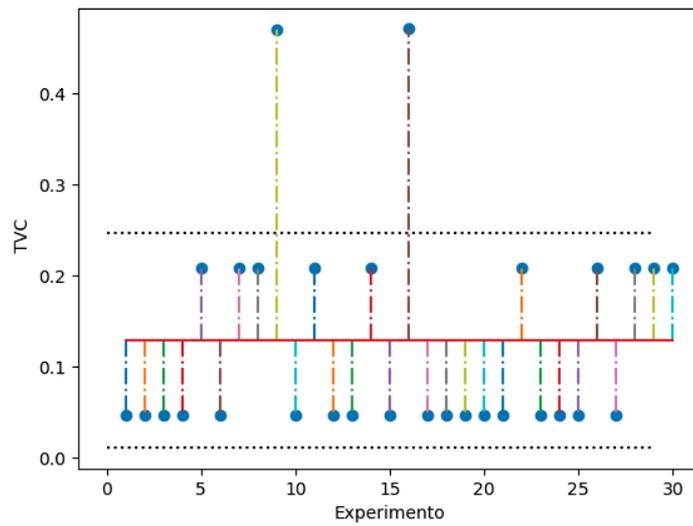
Além da análise das soluções com menor ISE , TVC , M_s e ED de toda população realizada por meio das Figuras 21, 22, 23, analisa-se a autonomia da otimização por experimento de encontrar soluções ótimas. A mesma pode ser feita por meio das Figuras 24,25,26, 27, que ilustram as melhores soluções por experimento.

Figura 24 – Menor ISE por experimento multiobjetivo não ponderado.



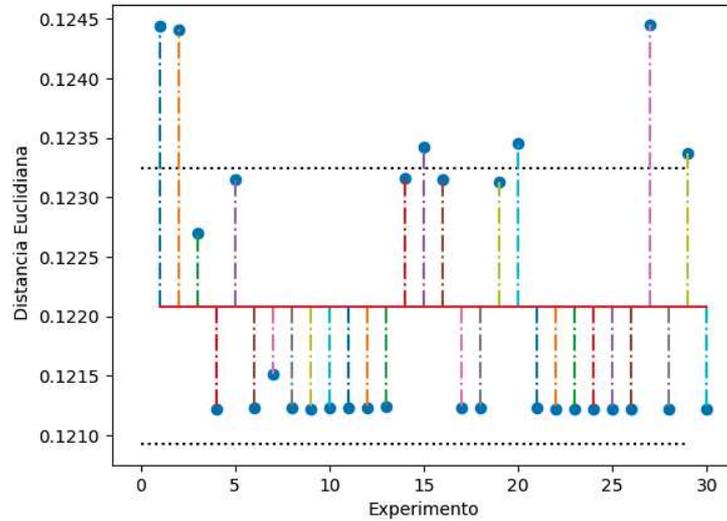
Fonte: Autoria Própria

Figura 25 – Menor TVC por experimento multiobjetivo não ponderado.



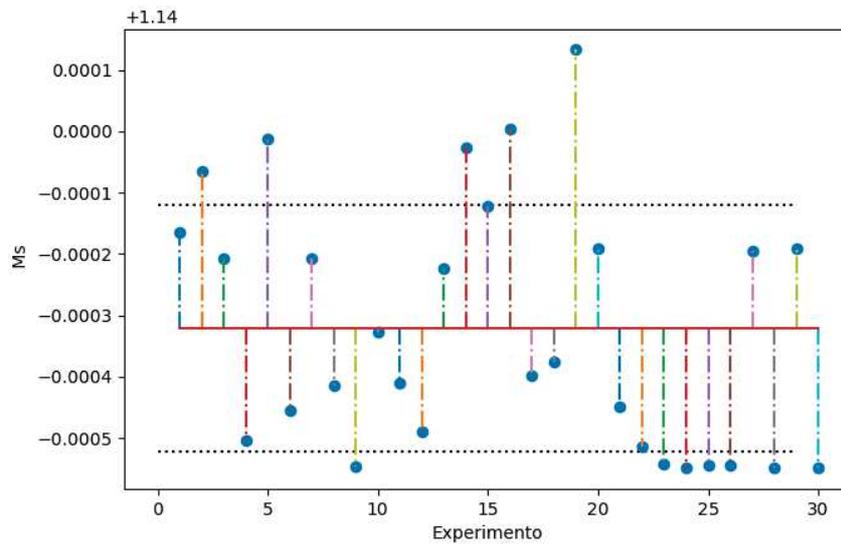
Fonte: Autoria Própria

Figura 26 – Menor ED por experimento multiobjetivo não ponderado.



Fonte: Autoria Própria

Figura 27 – Menor Ms por experimento multiobjetivo não ponderado.



Fonte: Autoria Própria

Por meio da Tabela 7 e das figuras é possível observar as melhores soluções para ISE , TVC , Ms e ED que em cada experimento realizado conseguiu-se chegar aproximadamente nas mesmas melhores soluções, tendo em vista que os desvios padrões são baixos, e

Tabela 7 – Média e desvio padrão das melhores soluções multiobjetivo não ponderada.

Soluções:	\bar{x}	σ^2
Menor <i>DE</i>	0,122	0,001
Menor <i>ISE</i>	4,527	0,2180
Menor <i>TVC</i>	0,129	0,118
Menor <i>Ms</i>	1,139	0,0002

\bar{x} = Média

σ^2 = Desvio Padrão

Fonte: Autoria Propria

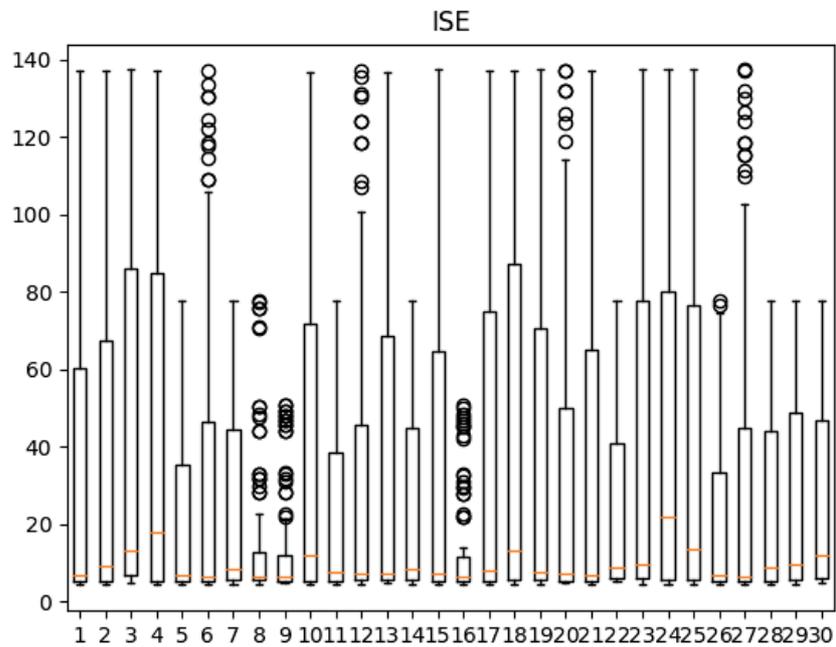
em alguns casos pode-se até desconsiderar, como na *ED* e *Ms*, de forma que em um único experimento consegue-se chegar nas mesmas melhores soluções do que todo o conjunto.

A análise da população de cada experimento com relação a aproximação do ponto ótimo é feita por meio das Figuras 28, 30, 32.

Analisando Figura 28 é possível observar que aproximadamente mais de 50% da população por experimento obteve o valor de *ISE* menor que 10 e que a maioria dos experimentos obteve aproximadamente 75% da população um valor de *ISE* menor que 60, pode-se concluir também que os menores 50% dos indivíduos da população apresentam uma menor do que os outros 50% posterior a mediana, para todos os experimentos.

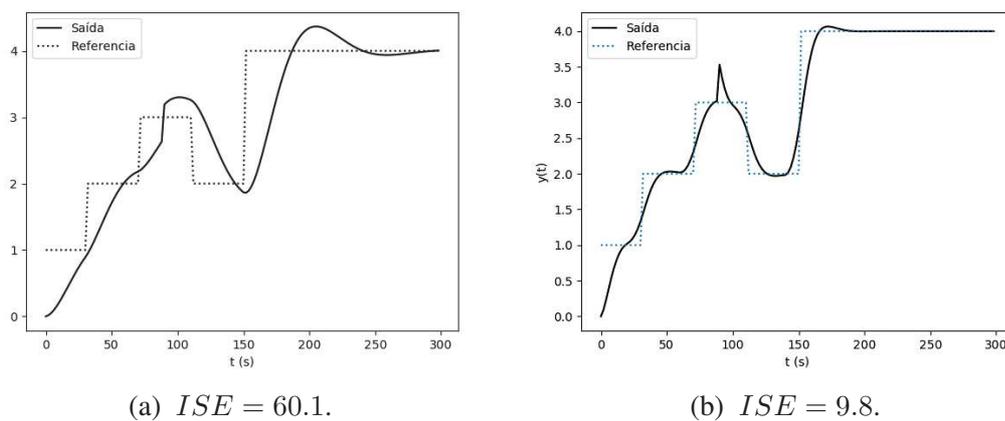
A Figura 29 ilustra a simulação da saída do sistema com o valor de *ISE* de aproximadamente 60 e outro de aproximante 10, assim como a referência a ser seguida pelo sistema, também é introduzida na saída do sistema uma perturbação externa de 0.5 a partir de 100s.

Figura 28 – BoxPlot ISE multiobjetivo não ponderado.



Fonte: Autoria Própria

Figura 29 – ISE : Simulação multiobjetivo não ponderado.



(a) $ISE = 60.1$.

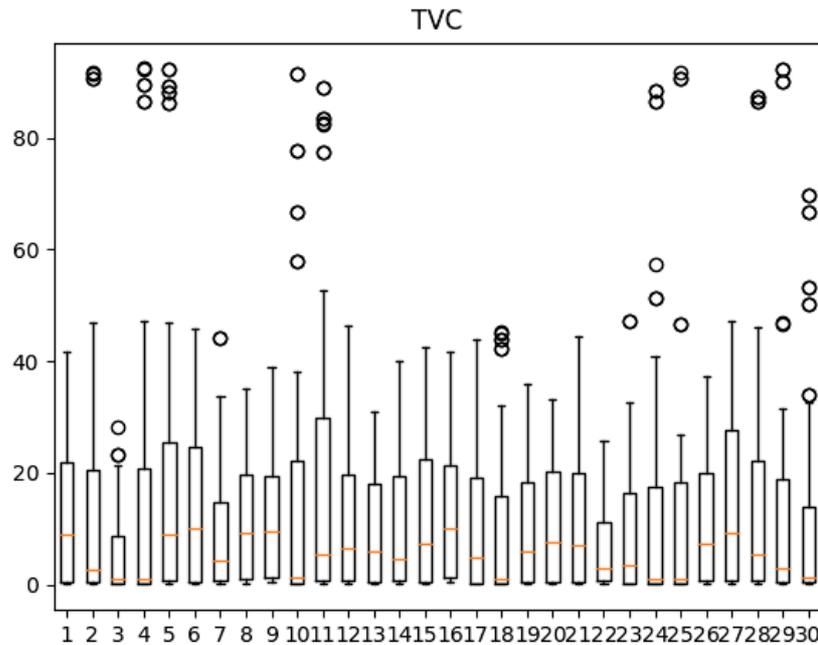
(b) $ISE = 9.8$.

Fonte: Autoria Própria

Para a análise das soluções com relação a variação das ações de controle, analisa-se a Figura 30, por meio desta é possível observar que 50% da população de cada experimento

obteve valor de TVC menor que 10 e que maioria dos experimentos obteve aproximadamente 75% da TVC menor que 20, do mesmo modo do ISE , o grau de dispersão de 50% das soluções foi menor que os 50% restante. De forma análoga ao ISE , a Figura 31 mostra dois exemplos de soluções com TVC de aproximadamente 10 e outra de aproximadamente 20.

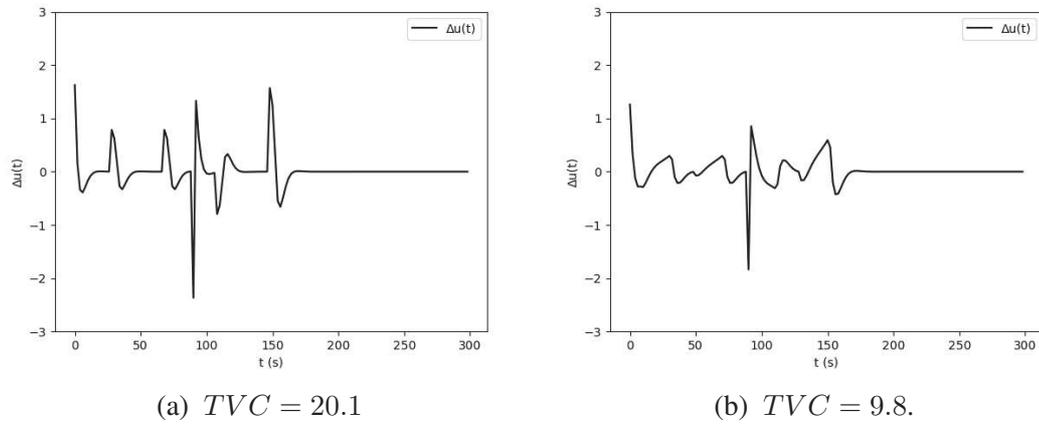
Figura 30 – BoxPlot TVC multiobjetivo não ponderado.



Fonte: Autoria Própria

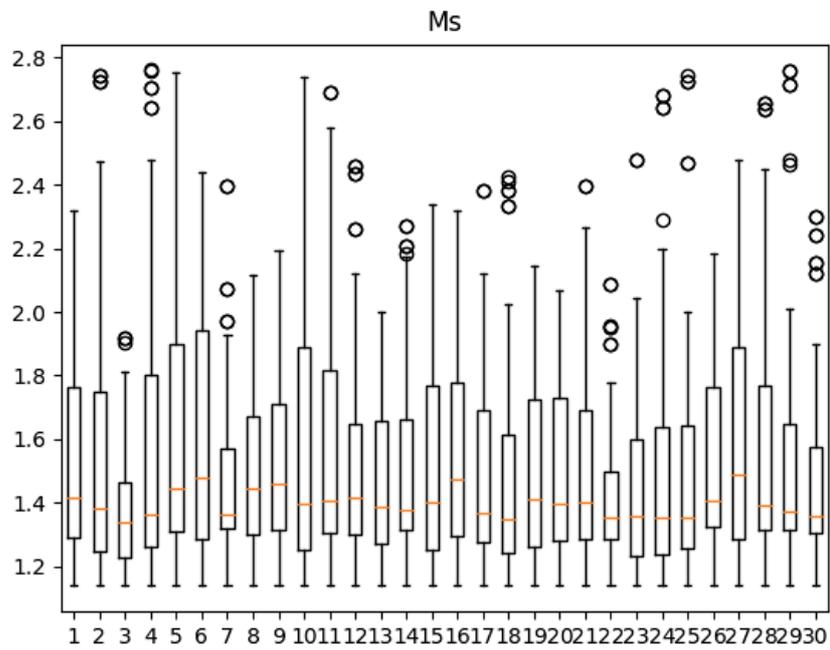
Para a análise da Ms , analisa-se por meio da Figura 32 que aproximadamente 50% da população apresentou valor inferior a 1.4 e que 75% da população de cada experimento apresentou uma Ms abaixo de 1.8, que ainda são valores bons de robustez baseado na Tabela 8. Pode-se observar também que diferentemente dos dois outros objetivos, a Ms foi a que apresentou um maior grau de dispersão tanto superior quanto inferior. A Figura 33 ilustra duas soluções com valores de Ms diferentes, quando analisado graficamente, a diferença é muito sutil, todavia substituindo nas Equações (3.27), (3.28) obtêm-se os valores da Tabela 9

Figura 31 – *TVC*: Simulação multiobjetivo não ponderado



Fonte: Autoria Própria

Figura 32 – BoxPlot M_s multiobjetivo não ponderado.

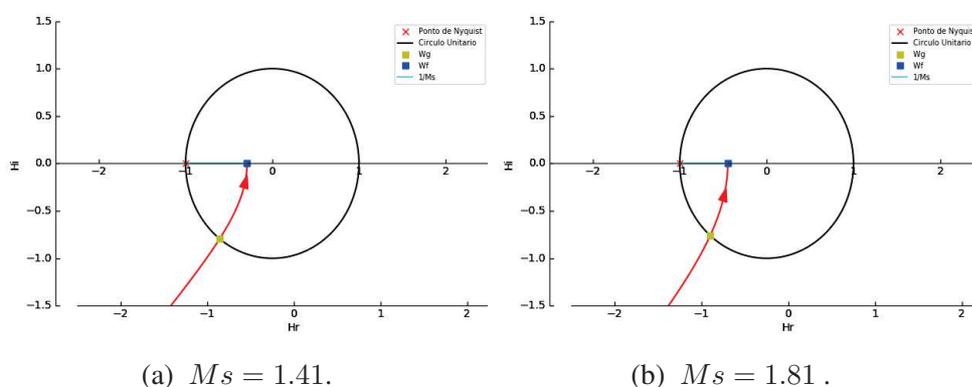


Fonte: Autoria Própria

Tabela 8 – Avaliação qualitativa da robustez

M_s	Nível de robustez
1,4	Alto
1,6	Médio-alto
1,8	Médio-baixo
2,0	Baixo

Fonte: Adaptado de (ARRIETA et al., 2016)

Figura 33 – M_s : Simulação multiobjetivo não ponderado.

Fonte: Autoria Própria

Tabela 9 – Margens Exemplo

M_s	M_g	M_f
1,41	3,437	52,612
1,81	2,23	49,448

Fonte: Autoria Propria

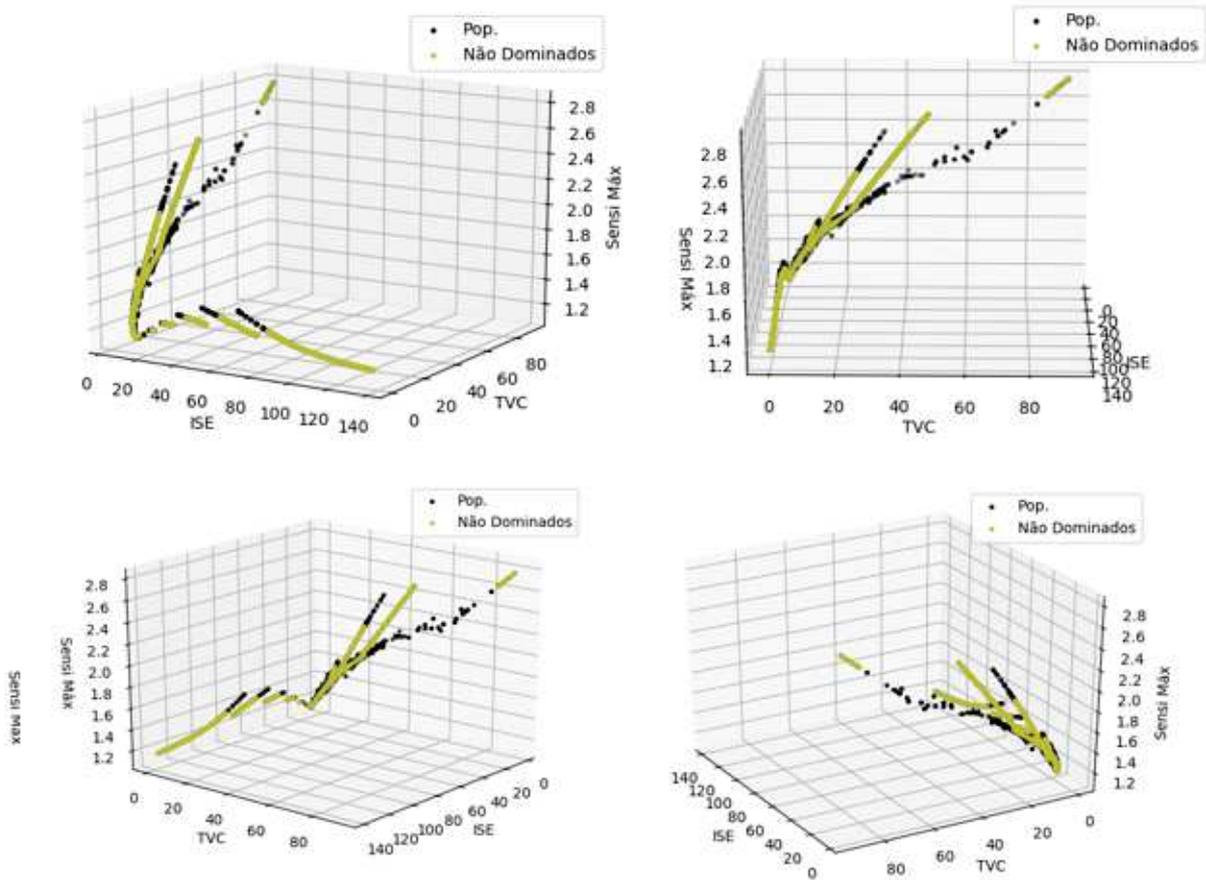
Outras análises podem a ser feitas analisando cada experimento separadamente, por exemplo o experimento 16 que apresenta um baixo grau de dispersão para ISE, contendo 100% da população com valor menor que 10, porém contra partida o TVC obteve um grau elevado de dispersão das soluções, e M_s também apresentou elevado grau de dispersão e com indivíduos da população chegando a valores de sensibilidade maior que 2.

6.1.2 ANÁLISE GERAL DA POPULAÇÃO: OTIMIZAÇÃO PONDERADA

De forma análoga ao conjunto de soluções da otimização não ponderada, o conjunto de soluções resultantes após os 30 experimentos é mostrado na Figura 34 em várias perspectivas.

Entre as 3000 soluções obtida, 2058 soluções estão localizadas na fronteira de Pareto, o que representa aproximadamente 68% do conjunto total.

Figura 34 – Conjunto de soluções dos experimentos multiobjetivo ponderados.

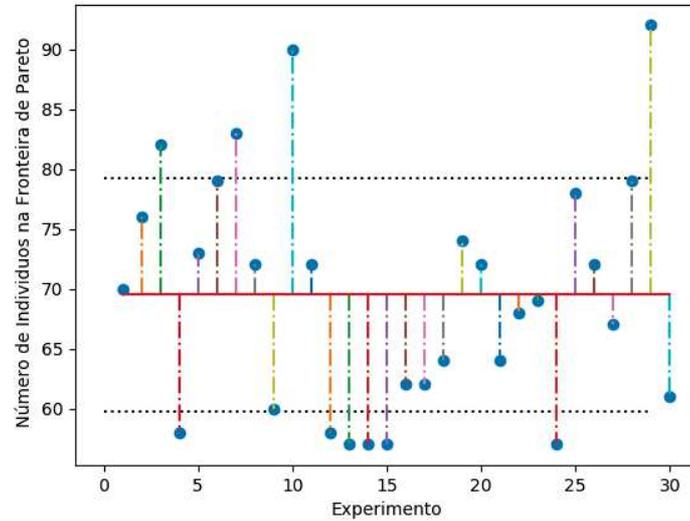


Fonte: Autoria Própria

Analisando o conjunto de solução por experimento de forma separada pôde-se observar que os mesmos obtiveram uma taxa de 100% de suas soluções não dominadas, todavia a taxa de soluções localizadas na fronteira de Pareto por experimento quando analisado o conjunto total de soluções foi em média de 69.5% com um desvio padrão de 10% das soluções.

Em outras palavras, cada experimento contribuiu com aproximadamente 70 ± 10 soluções para a fronteira de Pareto das soluções analisadas. A Figura 35 mostra o número de soluções localizadas na fronteira de Pareto por experimento realizado.

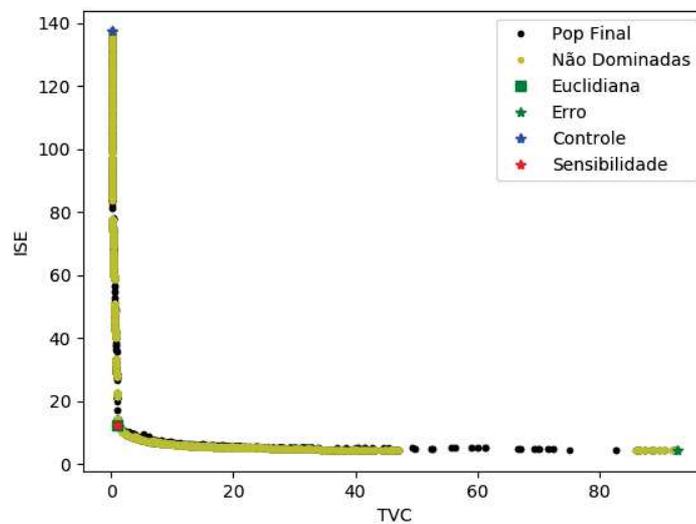
Figura 35 – Número de soluções ótimas por experimento multiobjetivo ponderado.



Fonte: Autoria Própria

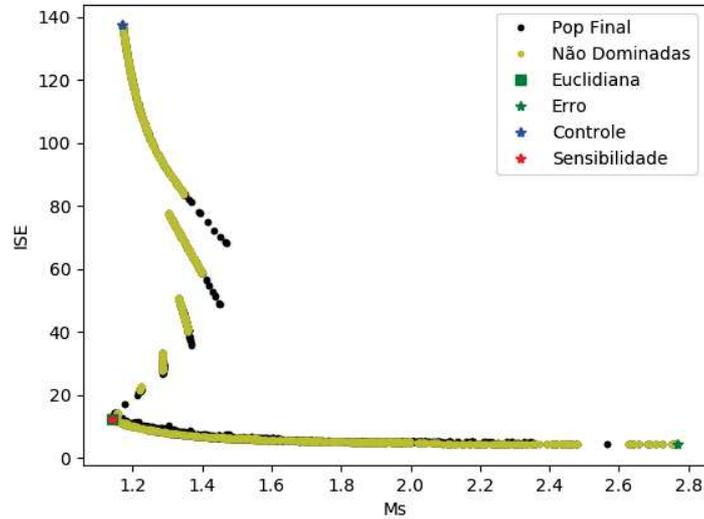
Entre o conjunto total de soluções, as melhores soluções para os critérios de menor *ISE*, menor *TVC*, menor *Ms* e menor *ED* são apresentadas nas Figuras 36, 37 e 38. Além das melhores soluções, as figuras apresentam o conjunto total de soluções e o conjunto de soluções localizadas na fronteira de Pareto.

Figura 36 – (*ISE* vs *TVC*) multiobjetivo ponderado.



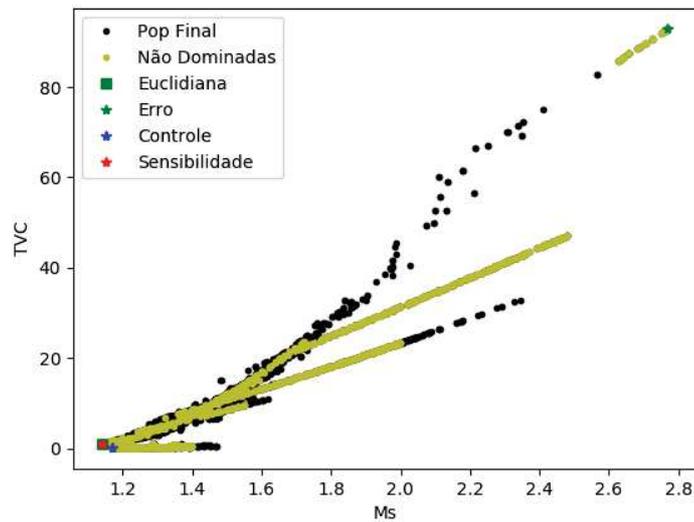
Fonte: Autoria Própria

Figura 37 – (ISE vs M_s) multiobjetivo ponderado.



Fonte: Autoria Própria

Figura 38 – (TVC vs M_s) multiobjetivo ponderado.



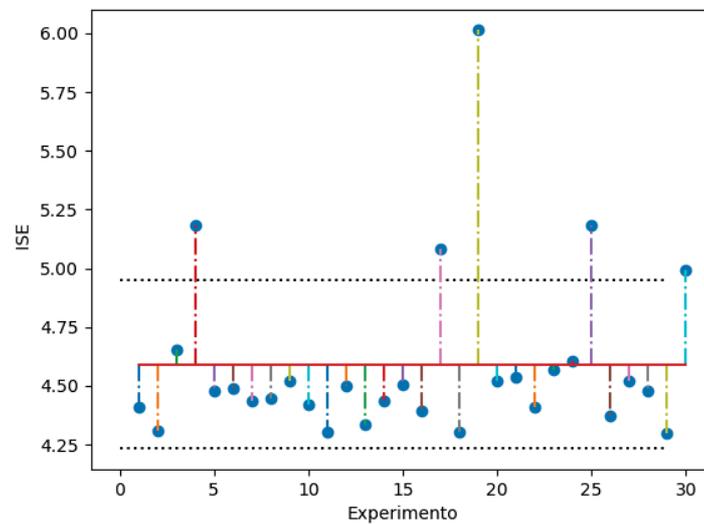
Fonte: Autoria Própria

De forma análoga a otimização não ponderada, as soluções que obtiveram menores ISE possuíram maiores TVC , assim como as soluções com menores TVC possuíram os menores

valores de Ms e novamente a solução que melhor conseguiu aproximar-se do ponto ótimo foi a solução com menor ED .

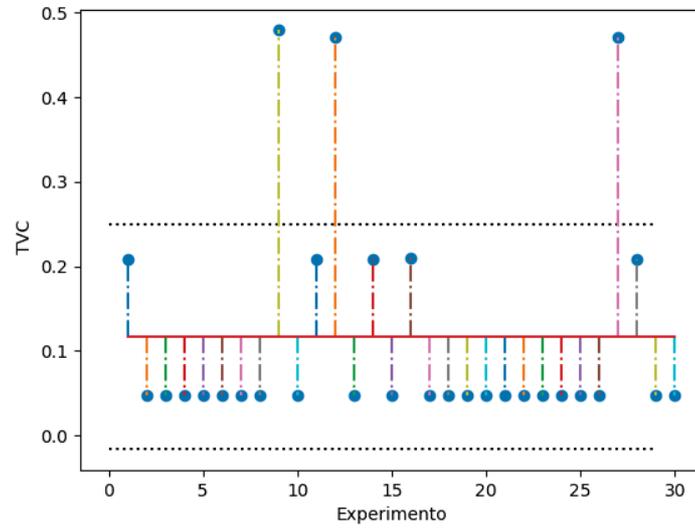
Além da análise das soluções com menor ISE , TVC , Ms e ED de toda população realizada por meio das Figuras 36, 37, 38, analise-se a autonomia da otimização por experimento em encontrar soluções ótimas, a mesma pode ser feita por meio das Figuras 39,40,41, 42, que ilustram as melhores soluções por experimento.

Figura 39 – Menor ISE por experimento multiobjetivo ponderado.



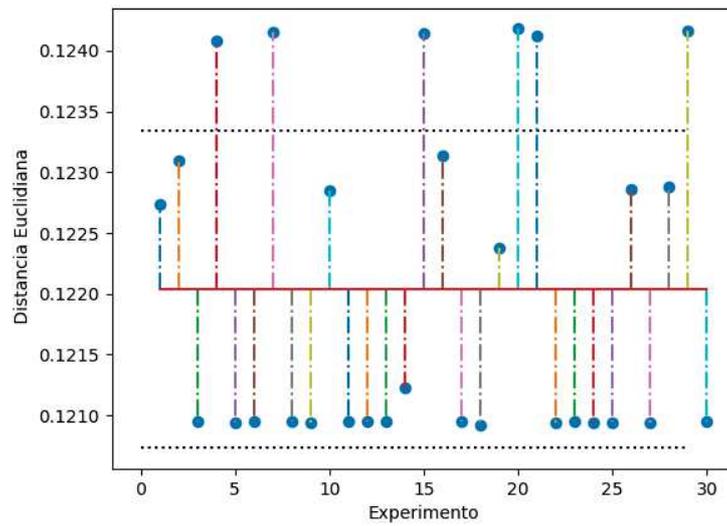
Fonte: Autoria Própria

Figura 40 – Menor TVC por experimento multiobjetivo ponderado.

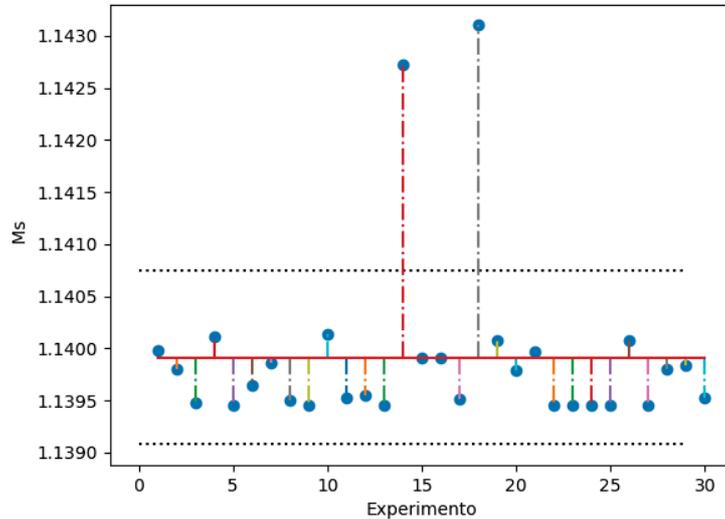


Fonte: Autoria Própria

Figura 41 – Menor ED por experimento multiobjetivo ponderado.



Fonte: Autoria Própria

Figura 42 – Menor M_s por experimento multiobjetivo ponderado.

Fonte: Autoria Própria

Tabela 10 – Média e desvio padrão das melhores soluções multiobjetivo ponderado

Soluções :	\bar{x}	σ^2
Menor DE	0,122	0,001
Menor ISE	4,590	0,358
Menor TVC	0,116	0,133
Menor M_s	1,139	0,0008

\bar{x} = Média

σ^2 = Desvio Padrão

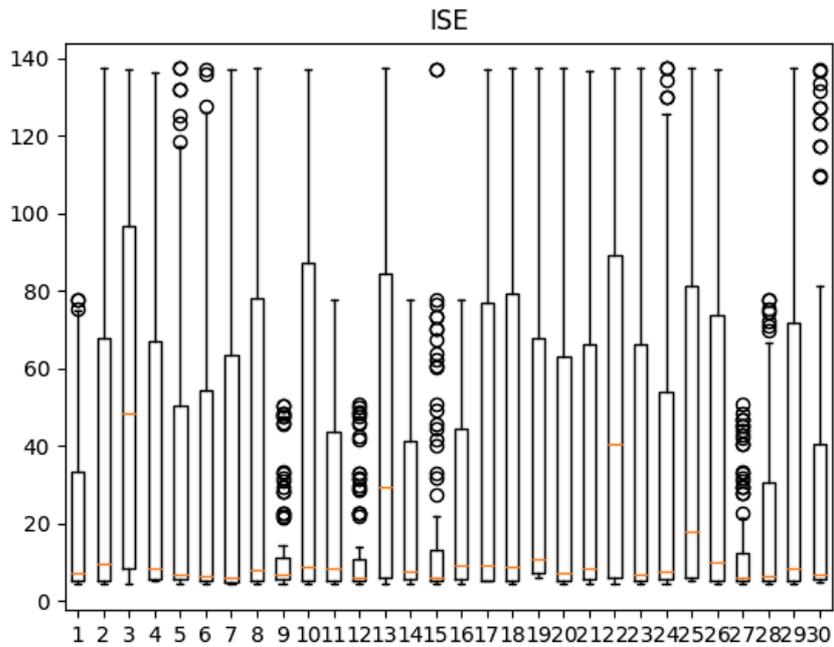
Por meio da Tabela 10 e das figuras é possível observar que em cada experimento realizado conseguiu chegar aproximadamente nas mesmas melhores soluções, visto que os desvios padrões são baixos, e em alguns casos pode-se até desconsiderar, de forma semelhante ao que ocorreu na otimização não ponderada.

A análise da população de cada experimento com relação a aproximação do ponto ótimo pode ser feita por meio das Figuras 43, 45, 46.

Por meio da Figura 43 é possível que a maioria dos experimentos obtiveram mais de 50% da população com o valor de ISE menor que 10 e que a maioria dos experimentos obteve aproximadamente 75% da população um valor de ISE menor que 70, pode-se concluir também que o grau de dispersão dos 50% dos indivíduos com menor ISE é menor do que os outros 50% posterior a mediana, para todos os experimentos.

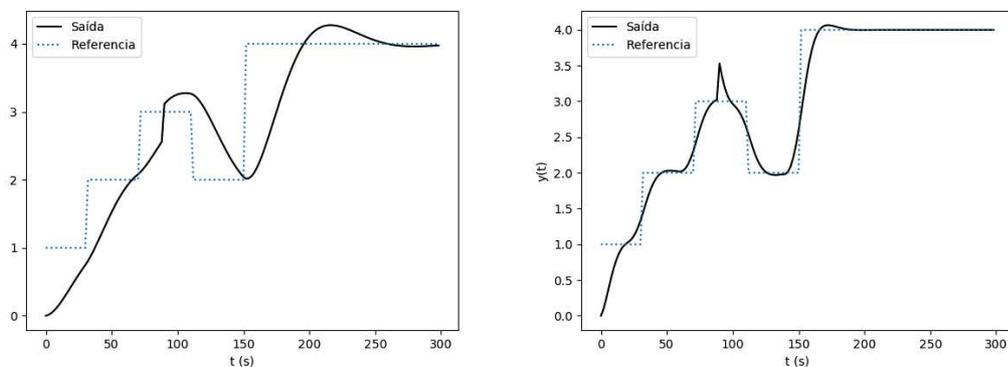
A Figura 44 ilustra a simulação da saída do sistema com o valor de ISE de aproximadamente 69,9 e outro de aproximante 10, assim como a referência a ser seguida pelo sistema, também é introduzida na saída do sistema uma perturbação externa de 0.5 a partir de 100s.

Figura 43 – BoxPlot ISE multiobjetivo ponderado.



Fonte: Autoria Própria

Figura 44 – ISE : Simulação multiobjetivo ponderado



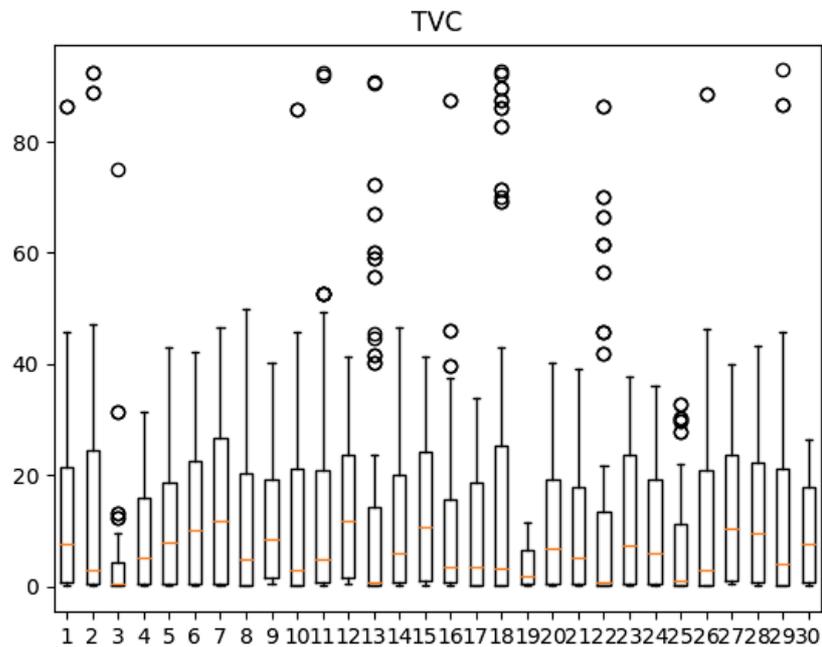
(a) $ISE = 69.9$.

(b) $ISE = 9.8$.

Fonte: Autoria Própria

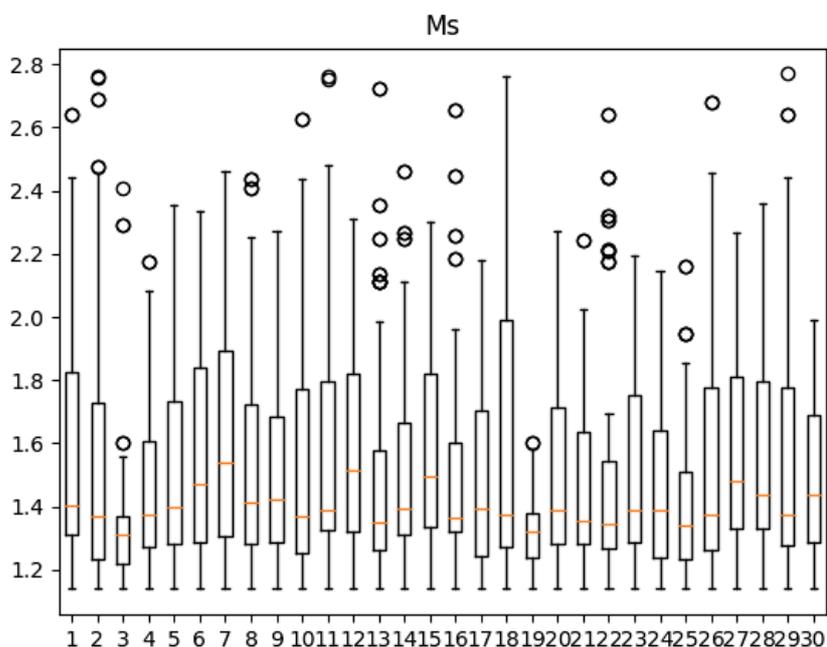
Para a análise das soluções com relação a variação das ações de controle, analisa-se a Figura 45, por meio desta é possível observar que 50% da população de cada experimento obteve valor de TVC menor que 10 e que maioria dos experimentos obteve aproximadamente 75% da população com TVC menor que 20, de modo análogo ao que ocorreu na otimização não ponderada. Do mesmo modo do ISE , o grau de dispersão dos 50% das soluções com menor TVC foi menor que os 50% restante.

Figura 45 – BoxPlot TVC multiobjetivo ponderado.



Fonte: Autoria Própria

Para a análise da M_s , verifica-se por meio da Figura 46 que aproximadamente 50% da população por experimento apresentou valor inferior a 1,4 e que 75% da população de cada experimento apresentou uma M_s abaixo de 1,8, resultando em valores relativamente bons de M_s de forma semelhante ao que ocorreu otimização não ponderada.

Figura 46 – BoxPlot M_s multiobjetivo ponderado.

Fonte: Autoria Própria

6.2 OTIMIZAÇÃO MONO-OBJETIVO

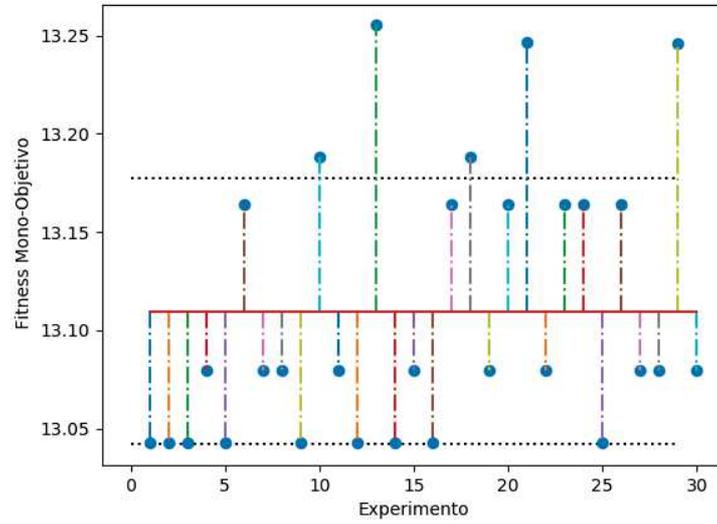
Esta seção destina-se a análise dos dados das otimizações mono-objetivo ponderada e não ponderada.

6.2.1 ANÁLISE GERAL DA POPULAÇÃO: OTIMIZAÇÃO NÃO PONDERADA

Entre as 3000 soluções obtidas pela otimização mono-objetivo, apenas 30 delas são distintas entre-si, cada experimento convergiu para um único ponto ótimo, ou seja, toda a população por experimento tem o mesmo valor.

O valor de *fitness* mono-objetivo por experimento é apresentado na Figura 47, pode-se observar que o valor de *fitness* por experimento são relativamente próximos.

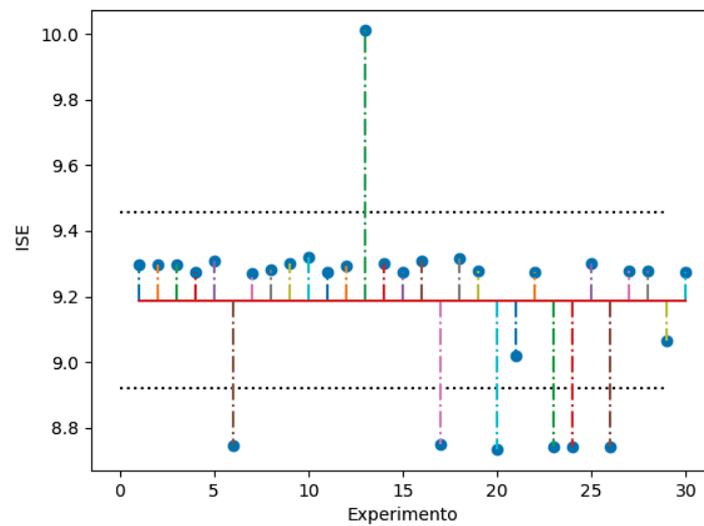
Figura 47 – *Fitness* mono-objetivo por experimento mono-objetivo não ponderado



Fonte: Autoria Própria

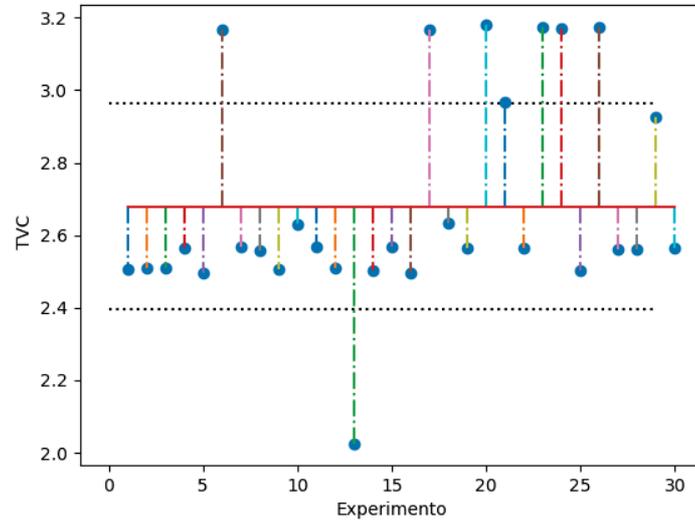
A análise apenas do valor de *fitness* mono-objetivo não é capaz inferir nada com relação aos objetivos otimizados, portanto os objetivos são separados e ilustrados nas Figuras 48, 49, 50.

Figura 48 – Valor de *ISE* por experimento mono-objetivo não ponderado



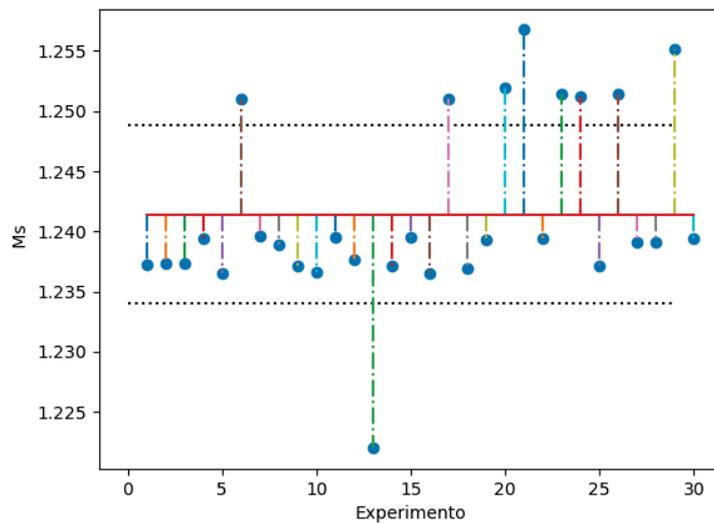
Fonte: Autoria Própria

Figura 49 – Valor de TVC por experimento mono-objetivo não ponderado



Fonte: Autoria Própria

Figura 50 – Valor de M_s por experimento por experimento mono-objetivo não ponderado



Fonte: Autoria Própria

Por meio da Tabela 11 e das figuras observa-se que as soluções tendem a convergir para a mesma solução. A análise por BoxPlot não é feita devido ao fato em que as soluções por experimento convergiram em uma única solução.

Tabela 11 – Média e desvio padrão das melhores soluções da otimização mono-objetivo não ponderada.

Soluções:	\bar{x}	σ^2
<i>Fitness Mono-Objetivo</i>	13,109	0,067
Menor <i>ISE</i>	9,188	0,266
Menor <i>TVC</i>	2,679	0,284
Menor <i>Ms</i>	1,241	0,007

\bar{x} = Média

σ^2 = Desvio Padrão

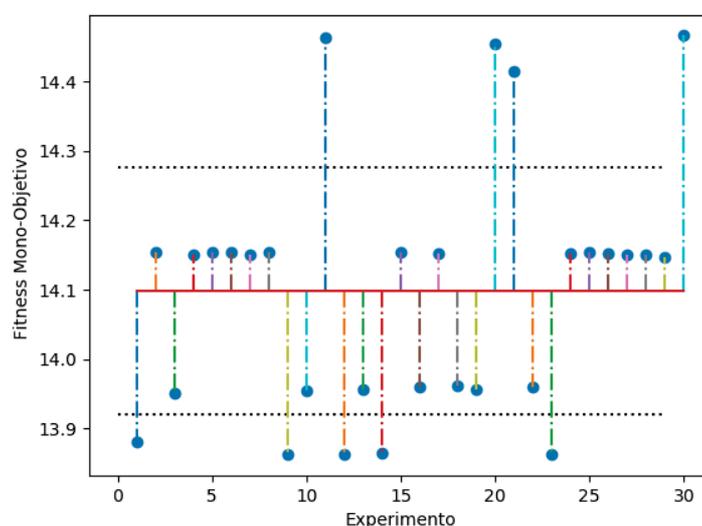
Fonte:Autoria Propria

6.2.2 ANÁLISE GERAL DA POPULAÇÃO: OTIMIZAÇÃO PONDERADA

Da mesmo modo que ocorreu na otimização não ponderada, entre as 3000 soluções obtidas pela otimização mono-objetivo ponderada, apenas 30 delas são distintas entre-si, cada experimento convergiu para um único ponto ótimo, ou seja, toda a população por experimento tem o mesmo valor.

O valor de *fitness* mono-objetivo por experimento é ilustrado na Figura 51, pode-se observar que o valor de *fitness* por experimento são relativamente próximos, uns aos outros.

Figura 51 – *Fitness* mono-objetivo por experimento mono-objetivo ponderado

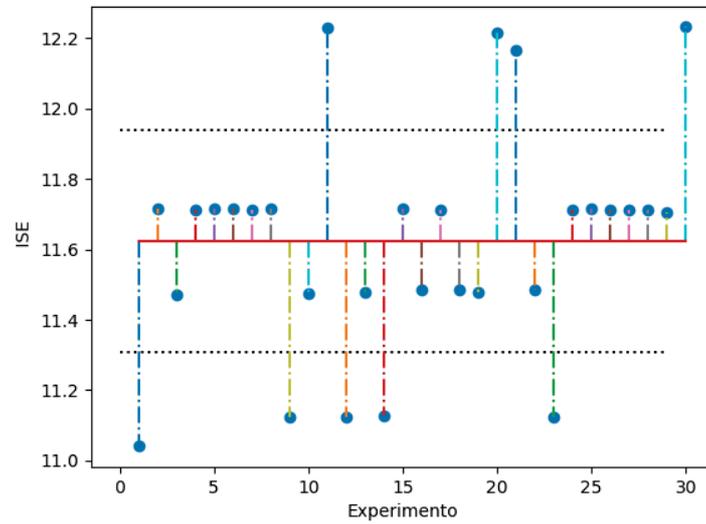


Fonte: Autoria Própria

A análise apenas do valor de *fitness* mono-objetivo não é capaz inferir nada com relação

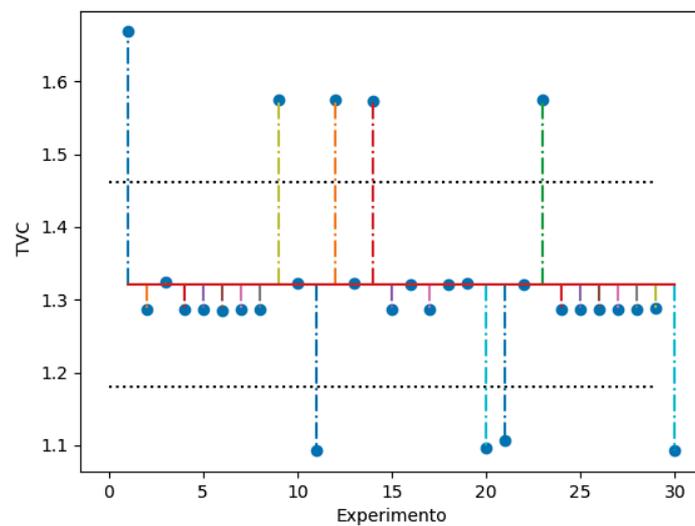
aos objetivos otimizados, portanto os objetivos são separados e mostrados nas Figuras 52, 53, 54.

Figura 52 – Valor de ISE por experimento mono-objetivo ponderado

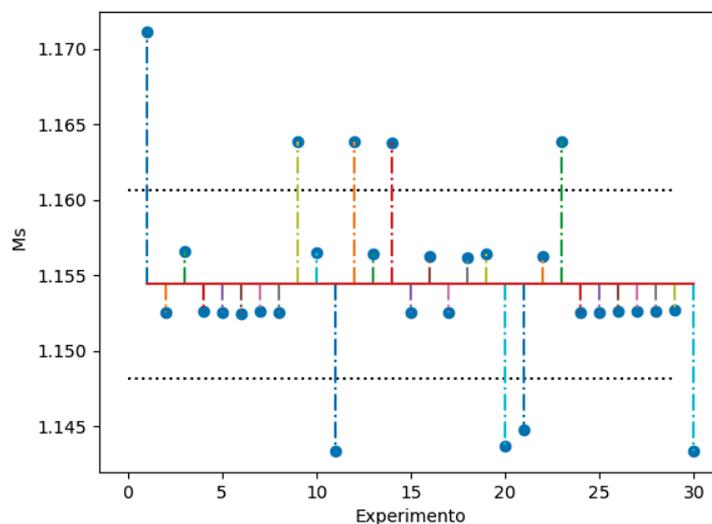


Fonte: Autoria Própria

Figura 53 – Valor de TVC por experimento mono-objetivo ponderado



Fonte: Autoria Própria

Figura 54 – Valor de M_s por experimento mono-objetivo ponderado

Fonte: Autoria Própria

Por meio da Tabela 12 e das figuras é possível observar que as soluções tendem a convergir para a mesma solução ótima de forma análoga a mono-objetivo não ponderada.

Tabela 12 – Média e desvio padrão das melhores soluções da otimização mono-objetivo ponderada.

Soluções:	\bar{x}	σ^2
<i>Fitness</i> Mono-Objetivo	14,098	0,177
Menor <i>ISE</i>	11,623	0,314
Menor <i>TVC</i>	1,320	0,140
Menor <i>M_s</i>	1,154	0,006

\bar{x} = Média

σ^2 = Desvio Padrão

6.3 ANÁLISE DAS MELHORES SOLUÇÕES

Esta seção destina-se a análise das melhores soluções de cada método, *a priori* a análise é feita comparando as ponderadas com as não ponderadas, visando verificar se a ponderação na função de *fitness* impactou de alguma forma no processo de otimização. *A posteriori* serão comparadas as soluções mono-objetivo e multiobjetivo e em seguida serão comparadas as soluções multiobjetivos com as soluções das heurísticas de referência.

6.3.1 SOLUÇÕES MULTIOBJETIVO

As Tabelas 13, 14 mostram as melhores soluções das otimizações multiobjetivo atendendo os seguintes critérios: solução com menor ED , solução com menor ISE , menor TVC e menor M_s .

Tabela 13 – Melhores soluções multiobjetivo sem ponderação

Melhores Soluções	Parâmetros GPC			Funções Objetivo			ED
	λ	N_y	N_u	ISE	TVC	M_s	
Menor ED	9,999	9	5	12,504	1,054	1,139	0,120
Menor ISE	0,00034	2	1	4,307	92,459	2,760	1,414
Menor TVC	9,999	1	1	137,333	0,004	1,171	1,004
Menor M_s	9,999	9	5	12,504	1,054	1,139	0,120

Tabela 14 – Melhores soluções multiobjetivo com ponderação

Melhores Soluções	Parâmetros GPC			Funções Objetivo			ED
	λ	N_y	N_u	ISE	TVC	M_s	
Menor ED	9,999	9	4	12,242	1,089	1,143	0,120
Menor ISE	0,00034	2	1	4,307	92,459	2,760	1,414
Menor TVC	9,999	1	1	137,333	0,004	1,171	1,004
Menor M_s	9,999	9	5	12,504	1,054	1,139	0,120

Analisando as tabela pode-se perceber que a ponderação na função de *fitness* não acarretou grandes mudanças nas soluções da otimização multiobjetivo, apenas as soluções com menor ED , diferenciaram-se com valores quase que desprezíveis as demais soluções tiveram seus valores diferentes a partir da sexta casa decimal, aqui omitidas. Por tanto, as otimizações multiobjetivos convergiram para as mesmas soluções ótimas, não sendo necessário para isso realizar uma ponderação dos valores, por mais discrepantes que sejam os valores dos objetivos otimizados.

6.3.2 SOLUÇÕES MONO-OBJETIVO

A Tabela 15 apresenta a melhor solução obtida da otimização mono-objetivo ponderada e não ponderada. Analisando a tabela pode-se perceber que o fator de ponderação no caso mono-objetivo teve impacto na otimização. Apesar da solução não ponderada obter um menor valor de ISE , a solução ponderada conseguiu obter valores melhores para o TVC e M_s .

Tabela 15 – Melhores soluções mono-objetivo

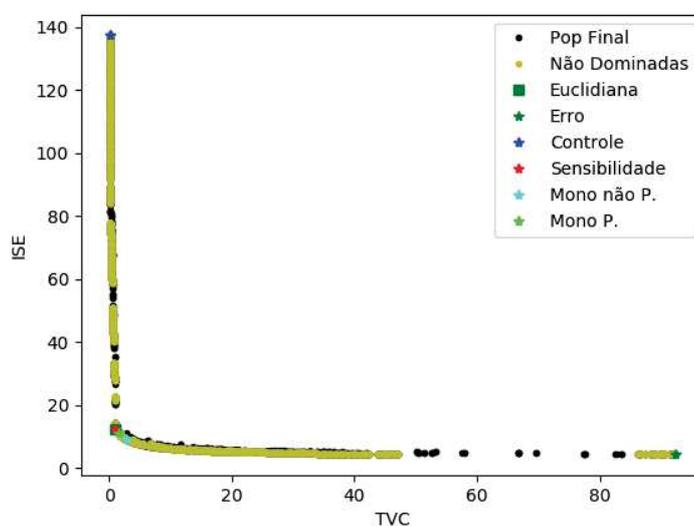
Melhores Soluções	Parâmetros GPC			Funções Objetivo		
	λ	N_y	N_u	ISE	TVC	M_s
Não Ponderada	5,916	7	1	9,297	2,507	1,237
Ponderada	9,372	7	1	11,123	1,574	1,163

Fonte: Autoria Propria

6.3.3 COMPARAÇÃO DAS SOLUÇÕES MONO-OBJETIVO E MULTIOBJETIVO

As Figuras 55, 56 e 57 mostram o conjunto de soluções gerais, assim como a fronteira de Pareto e as melhores soluções mono-objetivo e multiobjetivo ². Por meio das figuras é possível perceber que as soluções mono-objetivos estão contidas no conjunto de soluções localizadas na fronteira de Pareto da otimização multiobjetivo, além disso nas 3 figuras pode-se perceber que a otimização multi-objetivo obteve soluções mais próximas do ponto ótimo.

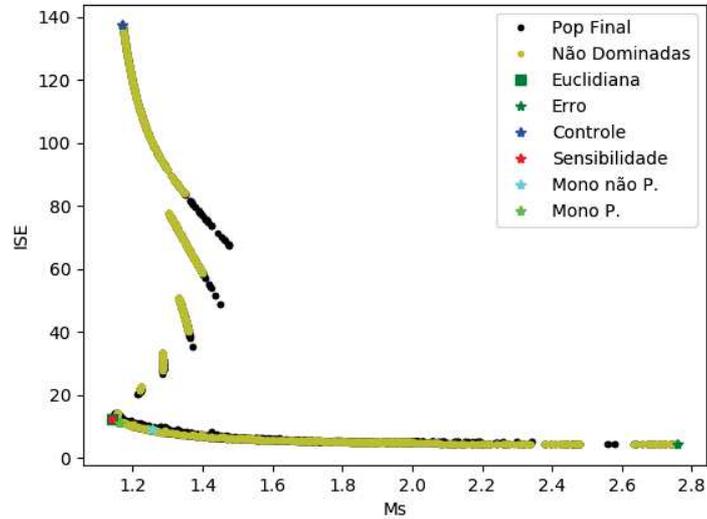
Figura 55 – *ISE* vs *TVC* - multiobjetivo e mono-objetivo.



Fonte: Autoria Própria

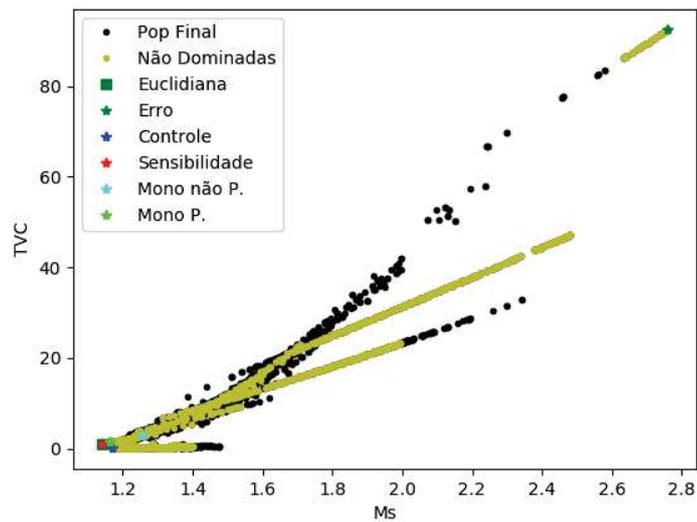
² Devido a similaridade dos gráficos das soluções multiobjetivo tanto ponderada, quanto não ponderada, apenas o conjunto de soluções não ponderado é mostrado nas figuras.

Figura 56 – ISE vs M_s - multiobjetivo e mono-objetivo.



Fonte: Autoria Própria

Figura 57 – TVC vs M_s - multiobjetivo e mono-objetivo.



Fonte: Autoria Própria

Por tanto, pode-se perceber que a sintonia multi-objetivo permite a escolha de uma solução entre o conjunto de soluções localizadas na fronteira de Pareto de acordo com que melhor atender as preferências controlador a ser sintonizado, seja o ISE , TVC , M_s ou o três

ao mesmo tempo. Diferentemente da otimização mono-objetivo que convergiu para soluções pontuais, restringindo assim a sintonia para uma única solução.

6.3.4 COMPARAÇÃO DAS SOLUÇÕES MULTIOBJETIVO E AS HEURÍSTICAS DE REFERÊNCIA

A Tabela 16 mostra os valores dos objetivos quando o GPC é sintonizado com os valores das heurísticas de referência.

Tabela 16 – Soluções das heurísticas de referência

	Parâmetros GPC			Funções Objetivo			ED
	λ	N_y	N_u	ISE	TVC	M_s	
[1]	1	16	4	8,086	4,410	1,318	0,196
[2]	1	24	5	10,096	4,379	1,331	0,206
[3]	0	10	1	11,902	17,414	2,114	0,666
[4]	10	29	1	37,601	9,360	1,764	0,523
[4]	10	29	4	16,880	1,039	1,175	0,158

[1] (TRIERWEILER; FARINA; DURAIKI, 2001)

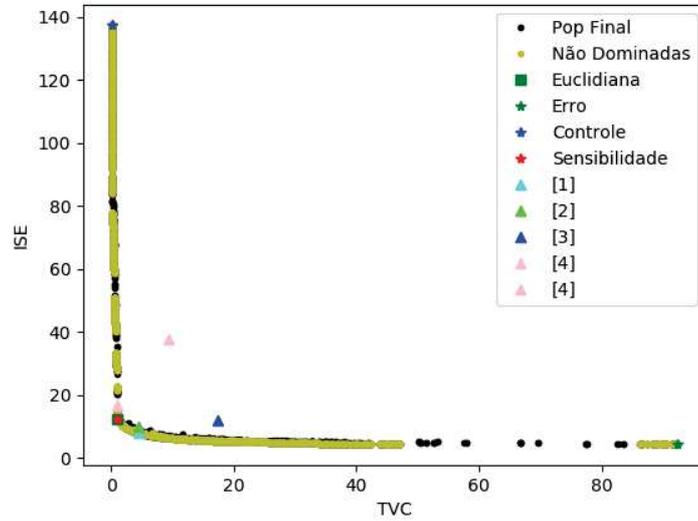
[2] (YAMASHITA; ZANIN; ODLOAK, 2016)

[3] (CLARKE; MOHTADI; TUFFS, 1987)

[4] (MCINTOSH, 1989; MCINTOSH, 1991)

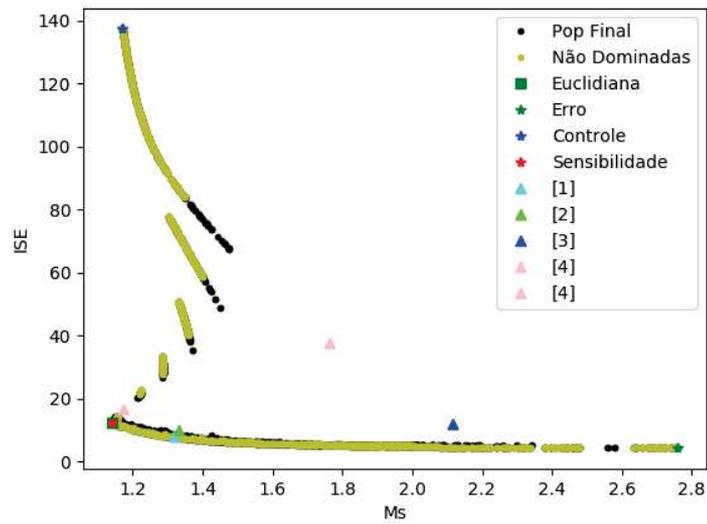
As Figuras 58, 59 e 60 ilustram a fronteira de Pareto obtida pela otimização multiobjetivo juntamente com o conjunto de soluções e as soluções obtidas pelas heurísticas de referência. É possível observar por meio das figuras que o conjunto de soluções ótimas obtidas pela otimização multiobjetivo consegue obter soluções melhores que as de referência para determinados objetivos assim como possui soluções mais próximas do ponto ótimo do que as de referência. Além disso, pode-se perceber que alguma das soluções de referência estão contidas no conjunto de soluções da otimização, como é o caso da referência [2].

Figura 58 – ISE vs TVC - multiobjetivo e heurísticas de referência.



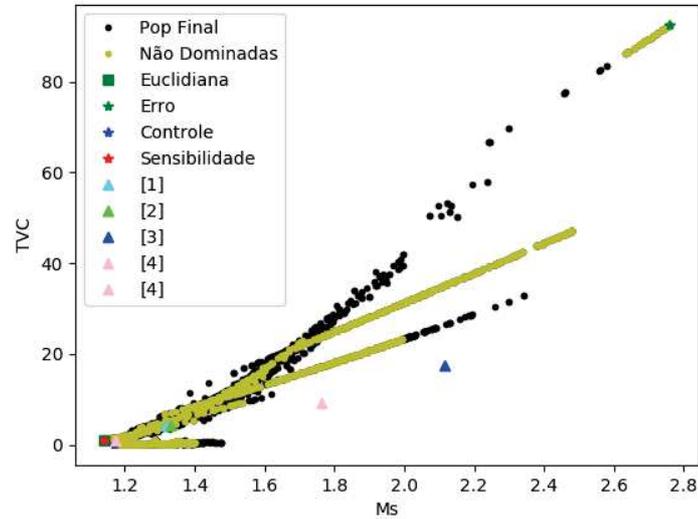
Fonte: Autoria Própria

Figura 59 – ISE vs M_s - multiobjetivo e heurísticas de referência.



Fonte: Autoria Própria

Figura 60 – TVC vs M_s - multiobjetivo e heurísticas de referência.



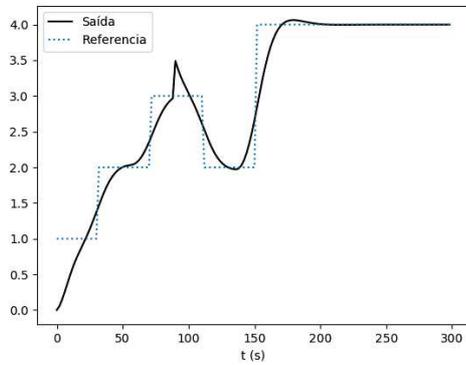
Fonte: Autoria Própria

6.4 SIMULAÇÃO DO SISTEMA DESCRITO NO ESTUDO DE CASO

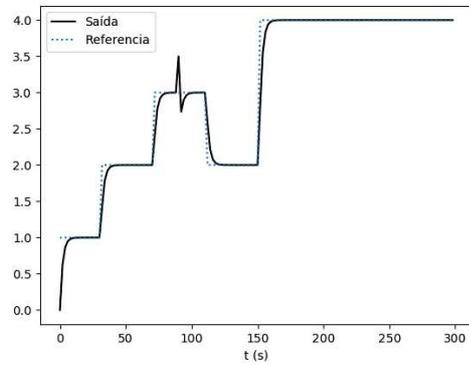
De forma a se ter uma melhor interpretação das soluções obtidas pelas otimizações as Figuras 61 e 62 ilustram o sinal de saída e o sinal da variação da ação de controle obtidos da simulação do sistema controlado quando sintonizado com as melhores soluções multiobjetivo e mono-objetivo.

Por meio delas pode-se perceber o impacto da escolha de diferentes tipos de soluções de acordo com o que melhor atender as necessidades do sistema controlado. Supondo que o sistema a ser controlado não tenha preocupação com a robustez, apenas com a precisão da saída do sistema, para esse caso pode-se escolher a solução com menor ISE que tem o sinal de saída semelhante ao de referência, porém com grande variações nas ações de controle. Ou um sistema que precisa conciliar robustez e desempenho, a solução com menor ED e as soluções mono-objetivo poderiam ser uma escolha.

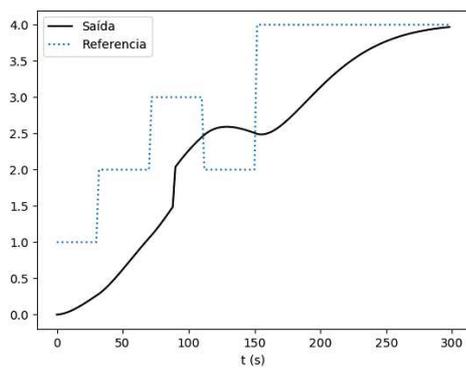
Figura 61 – Simulação das saídas do sistema, com GPC sintonizado pelas respectivas melhores soluções.



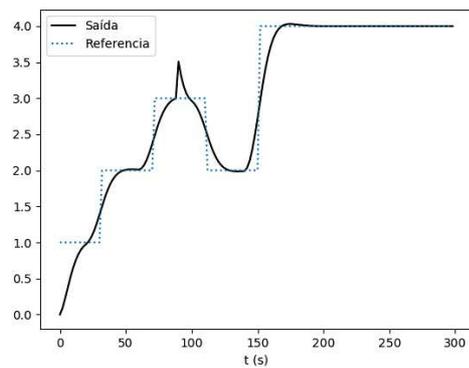
(a) Menor ED e Menor M_s .



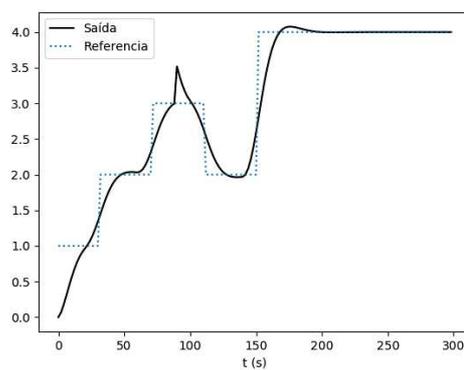
(b) Menor ISE .



(c) Menor TVC .



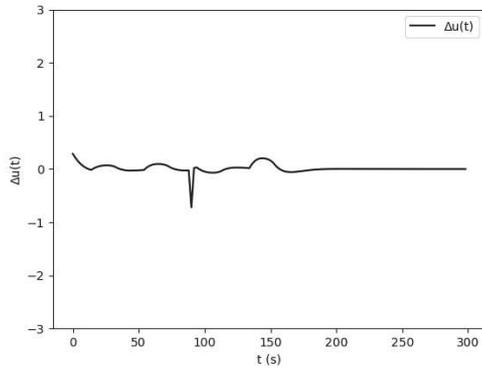
(d) Melhor solução mono-objetivo não ponderada .



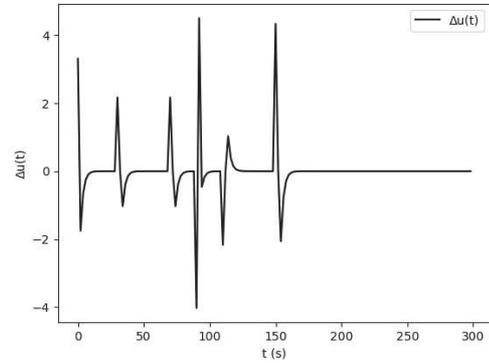
(e) Melhor solução mono-objetivo ponderada .

Fonte: Autoria Própria

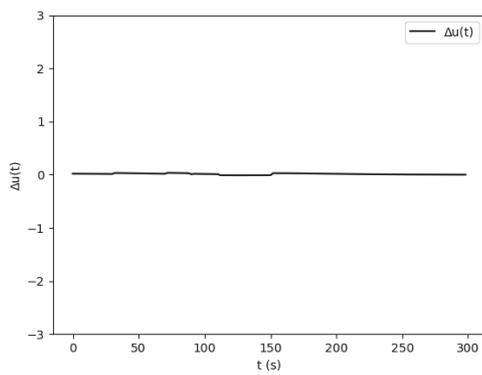
Figura 62 – Simulação das ações de controle do sistema, com GPC sintonizado pelas respectivas melhores soluções.



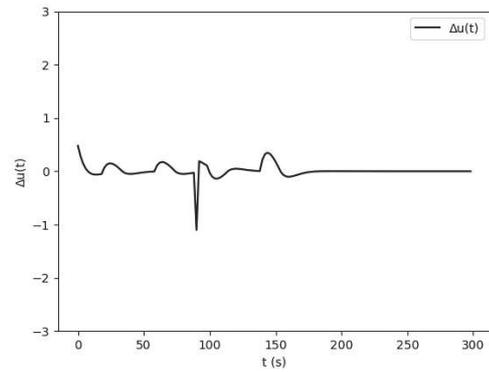
(a) Menor ED e Menor M_s .



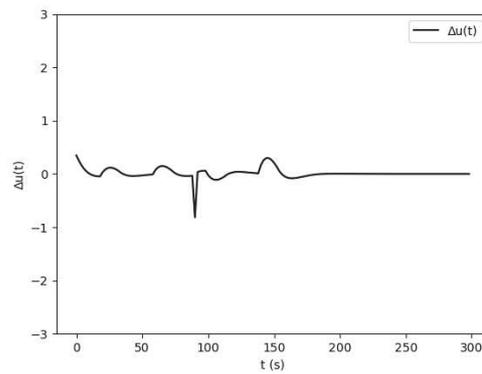
(b) Menor ISE .



(c) Menor TVC .



(d) Melhor solução mono-objetivo não ponderada .



(e) Melhor solução mono-objetivo ponderada .

Fonte: Autoria Própria

7 CONCLUSÃO

Pode-se observar que a solução proposta para a sintonia multiobjetivo dos parâmetros do GPC obteve-se resultados satisfatórios, capaz de conciliar o desempenho e a robustez do controlador. Também é possível observar que as soluções que melhor otimizaram o erro em regime permanente, tiveram que aplicar ações de controles mais efetivas comprometendo estabilidade do controlador implícita na robustez. Além disso percebe-se que por meio da sintonia multiobjetivo fica a disposição do programador a melhor solução que lhe convém localizada na fronteira de Pareto, diferentemente da solução mono-objetivo e das de referências, que fornecem apenas uma única solução sem a possibilidade de escolha qual objetivo deseja-se ser otimizado. Um fato interessante a ser destacado é que as soluções com menor ED foram as mesmas que obtiveram menor Ms em todos os experimentos para o caso multiobjetivo não ponderado, já para o caso multiobjetivo ponderado, o fato ocorreu para a maioria dos experimentos.

Além das contribuições em relação a sintonia multiobjetivo do GPC, para a implementação da abordagem proposta neste trabalho, foi necessário criar uma função robustez capaz de calcular o diagrama de Nyquist de tempo discreto, a mesma não foi encontrada em nenhuma biblioteca do *Python*, portanto deseja-se a disponibilizar a função implementada para alguma biblioteca ou ferramenta já existente.

O trabalho possuiu uma limitação em relação ao sistema escolhido, pelo fato de haver poucos limites de referências para os parâmetros do GPC, quando trata-se de sistemas instáveis em malha aberta. Devido a isso o trabalho restringiu-se apenas a sistemas estáveis em malha aberta, visto que os valores limitantes para os parâmetros do GPC são utilizados para restringir o espaço de busca.

Acredita-se que trabalhos futuros poderiam encontrar formas de delimitar esse espaço assim como um estudo do porquê as soluções com menor distância e sensibilidade coincidiram, se o mesmo ocorre apenas para esse sistema ou para uma classe de sistemas.

A DETERMINAÇÃO DA MATRIZ F

Considere um sistema descrito por :

$$A(z^{-1})y(t) = z^{-d}B(z^{-1})u(t-1)$$

Onde :

$$A(z^{-1}) = 1 + a_1z^{-1} + \dots + a_{N_a}z^{-N_a}$$

$$B(z^{-1}) = b_1z^{-1} + \dots + b_{N_b}z^{-N_b}$$

$$d = 0$$

Os vetores A e B podem ser reescritos como um vetor de coeficientes, representados por :

$$A = [1 \ a_1 \ \dots \ a_{N_a}]$$

$$A = [1 \ \bar{A}]$$

$$\bar{A} = [a_1 \ a_2 \ \dots \ a_{N_a}]$$

$$B = [b_1 \ b_2 \ \dots \ b_{N_b}]$$

$$\Delta = [1 \ -1]$$

Utilizando a Equação (3.11), reescrevendo na forma vetorial e desconsiderando o ruído, obtêm-se :

$$A[1 \ -1]y(t) = B\Delta u(t-1)$$

$$[1 \ \bar{A}][1 \ -1]y(t) = B\Delta u(t-1)$$

$$([1 \ 0] + [0 \ \bar{A}])[1 \ -1]y(t) = B\Delta u(t-1)$$

$$[1 \ \bar{A}](y(t) - y(t-1)) = B\Delta u(t-1)$$

$$[1 \ \bar{A}]y(t) = B\Delta u(t-1) + [1 \ \bar{A}]y(t-1)$$

isolando $y(t)$ da expressão acima obtém-se :

$$y(t) = B\Delta u(t-1) + [1 \ \bar{A}]y(t-1) - [0 \ \bar{A}]y(t)$$

O termo $[0 \ \bar{A}]y(t)$ pode ser reescrito como $[\bar{A} \ 0]y(t-1)$ resultando em :

$$y(t) = B\Delta u(t-1) - ([1 \ \bar{A}] - [\bar{A} \ 0])y(t-1)$$

Logo a equação do preditor pode ser escrita como:

$$\hat{y}(t+j) = B\Delta u(t-1+j) + ([1 \ \bar{A}] - [\bar{A} \ 0])y(t-1+j) \quad (\text{A.1})$$

Para $j = 1$

$$\begin{aligned}\hat{y}(t+1) &= B\Delta u(t) + ([1 \ \bar{A}] - [\bar{A} \ 0])y(t) \\ F_1 &= ([1 \ \bar{A}] - [\bar{A} \ 0]) = [\tilde{a}_1 \ \tilde{a}_2 \ \dots \tilde{a}_{N_a+1}]_{1 \times N_a+1} \\ \hat{y}(t+1) &= B\Delta u(t) + F_1 y(t)\end{aligned}$$

Para $j = 2$

$$\begin{aligned}\hat{y}(t+2) &= B\Delta u(t+1) + F_1 y(t+1) \\ \hat{y}(t+2) &= B\Delta u(t+1) + \tilde{a}_1 y(t+1) + [\tilde{a}_2 \ \dots \tilde{a}_{N_a+1} \ 0]y(t) \\ \hat{y}(t+2) &= B\Delta u(t+1) + \tilde{a}_1(B\Delta u(t) + F_1 y(t)) + [\tilde{a}_2 \ \dots \tilde{a}_{N_a+1} \ 0]y(t) \\ \hat{y}(t+2) &= B\Delta u(t+1) + \tilde{a}_1 B\Delta u(t) + \tilde{a}_1 F_1 y(t) + [\tilde{a}_2 \ \dots \tilde{a}_{N_a+1} \ 0]y(t) \\ F_2 &= \tilde{a}_1 F_1 + [\tilde{a}_2 \ \dots \tilde{a}_{N_a+1} \ 0] \\ \hat{y}(t+2) &= B\Delta u(t+1) + \tilde{a}_1 B\Delta u(t) + F_2 y(t)\end{aligned}$$

Para $j = 3$

$$\begin{aligned}\hat{y}(t+3) &= B\Delta u(t+2) + F_1 y(t+2) \\ \hat{y}(t+3) &= B\Delta u(t+2) + \tilde{a}_1 y(t+2) + \tilde{a}_2 y(t+1) + [\tilde{a}_3 \ \dots \tilde{a}_{N_a+1} \ 0 \ 0]y(t) \\ \hat{y}(t+3) &= B\Delta u(t+2) + \tilde{a}_1(B\Delta u(t+1) + \tilde{a}_1 B\Delta u(t) + F_2 y(t)) + \tilde{a}_2(B\Delta u(t) + F_1 y(t)) \\ &\quad + [\tilde{a}_3 \ \dots \tilde{a}_{N_a+1} \ 0 \ 0]y(t) \\ \hat{y}(t+3) &= B\Delta u(t+1) + \tilde{a}_1 B\Delta u(t+1) + (\tilde{a}_1)^2 B\Delta u(t) + \tilde{a}_2 B\Delta u(t+1) \\ &\quad + \tilde{a}_1 F_2 y(t) + \tilde{a}_2 F_1 y(t) + [\tilde{a}_3 \ \dots \tilde{a}_{N_a+1} \ 0 \ 0]y(t) \\ F_3 &= \tilde{a}_1 F_2 + \tilde{a}_2 F_1 + [\tilde{a}_3 \ \dots \tilde{a}_{N_a+1} \ 0 \ 0] \\ \hat{y}(t+2) &= B\Delta u(t+1) + \tilde{a}_1 B\Delta u(t) + F_3 y(t)\end{aligned}$$

Como o foco deste apêndice é determinar a matriz F , que é a matriz de coeficientes que multiplica o vetor de respostas passadas e presente como mostrada na equação 3.15, com base nas três predições assim é possível observar um padrão para as linhas da matriz F o que resulta na expressão :

Considerando n como $\min(j-1, N_a+1)$

Para $j-1 < N_a+1$

$$\begin{aligned}\hat{y}(t+j) &= B\Delta u(t-1+j) + F_1 y(t-1+j) \\ F_j &= \sum_{k=1}^n \tilde{a}_k F_{j-k} + [\tilde{a}_j \ \tilde{a}_{j+1} \ \dots \tilde{a}_{N_a+1} \ \text{zeros}(j-1)]\end{aligned}$$

Onde $\text{zeros}(j-1)$ representa um vetor de zeros de tamanho $j-1$.

Para $j - 1 \geq N_a + 1$

$$\hat{y}(t + j) = B\Delta u(t - 1 + j) + F_1 y(t - 1 + j)$$

$$F_j = \sum_{k=1}^n \tilde{a}_k F_{j-k}$$

Portanto as linhas da matriz F podem ser descritas pelas seguinte expressão:

$$F_j = \begin{cases} ([1 \ \bar{A}] - [\bar{A} \ 0]), & \text{se } j = 1, \\ \sum_{k=1}^n \tilde{a}_k F_{j-k} + [\tilde{a}_j \ \tilde{a}_{j+1} \ \dots \\ \tilde{a}_{N_a+1} \ \text{zeros}(j-1)], & \text{se } j - 1 < N_a + 1, \\ \sum_{k=1}^n \tilde{a}_k F_{j-k}, & \text{se } j - 1 \geq N_a + 1. \end{cases} \quad (\text{A.2})$$

B MATRIZ G

Considere um sistema descrito por :

$$A(z^{-1})y(t) = z^{-d}B(z^{-1})u(t-1)$$

Onde :

$$A(z^{-1}) = 1 + a_1z^{-1} + \dots + a_{N_a}z^{-N_a}$$

$$B(z^{-1}) = b_1z^{-1} + \dots + b_{N_b}z^{-N_b}$$

$$d = 0$$

Os vetores A e B podem ser reescritos como um vetor de coeficientes, representados por :

$$A = [1 \ a_1 \ \dots \ a_{N_a}]$$

$$A = [1 \ \bar{A}]$$

$$\bar{A} = [a_1 \ a_2 \ \dots \ a_{N_a}]$$

$$B = [b_1 \ b_2 \ \dots \ b_{N_b}]$$

$$\Delta = [1 \ -1]$$

Utilizando a equação 3.11 , reescrevendo na forma vetorial e desconsiderando o ruído, obtêm-se :

$$A[1 \ -1]y(t) = B\Delta u(t-1)$$

$$[1 \ \bar{A}][1 \ -1]y(t) = B\Delta u(t-1)$$

$$([1 \ 0] + [0 \ \bar{A}])[1 \ -1]y(t) = B\Delta u(t-1)$$

$$[1 \ \bar{A}](y(t) - y(t-1)) = B\Delta u(t-1)$$

$$[1 \ \bar{A}]y(t) = B\Delta u(t-1) + [1 \ \bar{A}]y(t-1)$$

isolando $y(t)$:

$$y(t) = B\Delta u(t-1) + [1 \ \bar{A}]y(t-1) - [0 \ \bar{A}]y(t)$$

O termo $[0 \ \bar{A}]y(t)$ pode ser reescrito como $[\bar{A} \ 0]y(t-1)$ resultando em :

$$y(t) = B\Delta u(t-1) - ([1 \ \bar{A}] - [\bar{A} \ 0])y(t-1)$$

Logo a equação do preditor pode ser escrita como:

$$\hat{y}(t+j) = [b_1 \ b_2 \ \dots \ b_{N_b}]\Delta u(t-1+j) + ([1 \ \bar{A}] - [\bar{A} \ 0])y(t-1+j) \quad (\text{B.1})$$

Para $j = 1$

$$\hat{y}(t+1) = [b_1 \ b_2 \ \dots \ b_{N_b}]\Delta u(t) + ([1 \ \bar{A}] - [\bar{A} \ 0])y(t)$$

$$\hat{y}(t+1) = [b_1]u(t) + [b_2 \dots b_{N_b} 0]\Delta u(t-1) + F_1 y(t)$$

$$G_1 = [b_1 \text{ zeros}(Ny-1)]$$

$$\hat{y}(t+1) = G_1 u(t) + [b_2 \dots b_{N_b} 0]\Delta u(t-1) + F_1 y(t)$$

Para $j = 2$

$$\hat{y}(t+2) = [b_1 \ b_2 \ \dots \ b_{N_b}] \Delta u(t+1) + F_1 y(t+1)$$

$$\hat{y}(t+2) = [b_1 \ b_2 \ \dots \ b_{N_b}] \Delta u(t+1) + \tilde{a}_1 y(t+1) + [\tilde{a}_2 \ \dots \ \tilde{a}_{N_a+1} \ 0] y(t)$$

$$\hat{y}(t+2) = [b_1 \ b_2 \ \dots \ b_{N_b}] \Delta u(t+1) + \tilde{a}_1 (G_1 u(t) + [b_2 \ \dots \ b_{N_b} 0] \Delta u(t-1) + F_1 y(t)) + [\tilde{a}_2 \ \dots \ \tilde{a}_{N_a+1} \ 0] y(t)$$

$$\begin{aligned} \hat{y}(t+2) &= [b_1 \ b_2 \ \dots \ b_{N_b}] \Delta u(t+1) + \tilde{a}_1 (G_1 u(t) + [b_2 \ \dots \ b_{N_b} 0] \Delta u(t-1)) \Delta u(t) \\ &\quad + \tilde{a}_1 F_1 y(t) + [\tilde{a}_2 \ \dots \ \tilde{a}_{N_a+1} \ 0] y(t) \end{aligned}$$

$$G_2 = [b_2 \ b_1 \ \text{zeros}(Ny-2)] + \tilde{a}_1 G_1$$

$$\hat{y}(t+2) = G_2 \Delta u(t) + ([b_3 \ \dots \ b_{N_b} \ 0 \ 0] + \tilde{a}_1 [b_2 \ \dots \ b_{N_b} 0]) \Delta u(t-1) + F_2 y(t)$$

Para $j = 3$

$$\hat{y}(t+3) = [b_1 \ b_2 \ \dots \ b_{N_b}] \Delta u(t+2) + F_1 y(t+2)$$

$$\hat{y}(t+3) = [b_1 \ b_2 \ \dots \ b_{N_b}] \Delta u(t+2) + \tilde{a}_1 y(t+2) + \tilde{a}_2 y(t+1) + [\tilde{a}_3 \ \dots \ \tilde{a}_{N_a+1} \ 0 \ 0] y(t)$$

$$\begin{aligned} \hat{y}(t+3) &= [b_1 \ b_2 \ \dots \ b_{N_b}] \Delta u(t+2) + \tilde{a}_1 (G_2 \Delta u(t) + ([b_3 \ \dots \ b_{N_b} \ 0 \ 0] + \tilde{a}_1 [b_2 \ \dots \ b_{N_b} 0]) \Delta u(t-1) + F_2 y(t)) \\ &\quad + \tilde{a}_2 (G_1 u(t) + [b_2 \ \dots \ b_{N_b} 0] \Delta u(t-1) + F_1 y(t)) + [\tilde{a}_3 \ \dots \ \tilde{a}_{N_a+1} \ 0 \ 0] y(t) \end{aligned}$$

$$G_3 = [B_3 B_2 B_1 + \text{zeros}(Ny-3)] + \tilde{a}_1 G_2 + \tilde{a}_2 G_1$$

$$\hat{y}(t+3) = G_3 B \Delta u(t) + (\tilde{a}_1 ([b_3 \ \dots \ b_{N_b} \ 0 \ 0] + \tilde{a}_1 [b_2 \ \dots \ b_{N_b} 0]) + \tilde{a}_2 ([b_2 \ \dots \ b_{N_b} 0])) u(t-1) + F_3 y(t)$$

Como o foco deste apêndice é determinar a matriz G , que é a matriz de coeficientes que multiplica o vetor das variações incrementais de controle presentes e futuras como mostrada na equação 3.15, com base nas três previsões assim é possível observar um padrão para as linhas da matriz G

Considerando n como $\min(j-1, Na+1)$

Para $j > N_b$

$$\hat{y}(t+j) = B \Delta u(t-1+j) + F_1 y(t-1+j)$$

$$G_j = [\text{zeros}(j-N_b) \ b_{N_b} \ b_{N_b-1} \ \dots \ b_1 \ \text{zeros}(Ny-j)] + \tilde{a}_1 G_{j-1} + \tilde{a}_2 G_{j-2} + \dots + \tilde{a}_n G_n$$

Para $j \leq N_b$

$$\hat{y}(t+j) = B \Delta u(t-1+j) + F_1 y(t-1+j)$$

$$G_j = [b_j \ b_{j-1} \ \dots \ b_1 \ \text{zeros}(Ny-j)] + \tilde{a}_1 G_{j-1} + \tilde{a}_2 G_{j-2} + \dots + \tilde{a}_n G_n$$

Portanto as linhas da matriz G podem ser descritas pelas seguinte expressão:

$$G_j = \begin{cases} [\text{zeros}(j - Nb) \ b_{N_b} \ b_{N_b-1} \ \dots \ b_1 \ \text{zeros}(Ny - j)] \sum_{k=1}^n \tilde{a}_k G_{j-k}, & \text{se } j > N_b, \\ [b_j \ b_{j-1} \ \dots \ b_1 \ \text{zeros}(Ny - j)] + \sum_{k=1}^n \tilde{a}_k G_{j-k}, & \text{se } j \leq N_b, \\ [b_1 \ \text{zeros}(Ny - 1)], & \text{se } j = 1. \end{cases} \quad (\text{B.2})$$

C MATRIZ H

Considere um sistema descrito por :

$$A(z^{-1})y(t) = z^{-d}B(z^{-1})u(t-1)$$

Onde :

$$A(z^{-1}) = 1 + a_1z^{-1} + \dots + a_{N_a}z^{-N_a}$$

$$B(z^{-1}) = b_1z^{-1} + \dots + b_{N_b}z^{-N_b}$$

$$d = 0$$

Os vetores A e B podem ser reescritos como um vetor de coeficientes, representados por :

$$A = [1 \ a_1 \ \dots \ a_{N_a}]$$

$$A = [1 \ \bar{A}]$$

$$\bar{A} = [a_1 \ a_2 \ \dots \ a_{N_a}]$$

$$B = [b_1 \ b_2 \ \dots \ b_{N_b}]$$

$$\Delta = [1 \ -1]$$

Utilizando a equação 3.11 , reescrevendo na forma vetorial e desconsiderando o ruído, obtêm-se :

$$A[1 \ -1]y(t) = B\Delta u(t-1)$$

$$[1 \ \bar{A}][1 \ -1]y(t) = B\Delta u(t-1)$$

$$([1 \ 0] + [0 \ \bar{A}])[1 \ -1]y(t) = B\Delta u(t-1)$$

$$[1 \ \bar{A}](y(t) - y(t-1)) = B\Delta u(t-1)$$

$$[1 \ \bar{A}]y(t) = B\Delta u(t-1) + [1 \ \bar{A}]y(t-1)$$

isolando $y(t)$:

$$y(t) = B\Delta u(t-1) + [1 \ \bar{A}]y(t-1) - [0 \ \bar{A}]y(t)$$

O termo $[0 \ \bar{A}]y(t)$ pode ser reescrito como $[\bar{A} \ 0]y(t-1)$ resultando em :

$$y(t) = B\Delta u(t-1) - ([1 \ \bar{A}] - [\bar{A} \ 0])y(t-1)$$

Logo a equação do preditor pode ser escrita como:

$$\hat{y}(t+j) = [b_1 \ b_2 \ \dots \ b_{N_b}]\Delta u(t-1+j) + ([1 \ \bar{A}] - [\bar{A} \ 0])y(t-1+j) \quad (\text{C.1})$$

Para $j = 1$

$$\hat{y}(t+1) = [b_1 \ b_2 \ \dots \ b_{N_b}]\Delta u(t) + ([1 \ \bar{A}] - [\bar{A} \ 0])y(t)$$

$$\hat{y}(t+1) = [b_1]u(t) + [b_2 \dots b_{N_b} 0]\Delta u(t-1) + F_1 y(t)$$

$$H_1 = [b_2 \dots b_{N_b} 0]$$

$$\hat{y}(t+1) = G_1 \Delta u(t) + H_1 \Delta u(t-1) + F_1 y(t)$$

Para $j = 2$

$$\hat{y}(t+2) = [b_1 \ b_2 \ \dots \ b_{N_b}] \Delta u(t+1) + F_1 y(t+1)$$

$$\hat{y}(t+2) = [b_1 \ b_2 \ \dots \ b_{N_b}] \Delta u(t+1) + \tilde{a}_1 y(t+1) + [\tilde{a}_2 \ \dots \ \tilde{a}_{N_a+1} \ 0] y(t)$$

$$\hat{y}(t+2) = [b_1 \ b_2 \ \dots \ b_{N_b}] \Delta u(t+1) + \tilde{a}_1 (G_1 u(t) + H_1 \Delta u(t-1) + F_1 y(t)) + [\tilde{a}_2 \ \dots \ \tilde{a}_{N_a+1} \ 0] y(t)$$

$$\hat{y}(t+2) = [b_1 \ b_2 \ \dots \ b_{N_b}] \Delta u(t+1) + \tilde{a}_1 (G_1 u(t) + H_1 \Delta u(t-1))$$

$$+ \tilde{a}_1 F_1 y(t) + [\tilde{a}_2 \ \dots \ \tilde{a}_{N_a+1} \ 0] y(t)$$

$$H_2 = [b_3 \ b_4 \ \dots \ b_{N_b} \ 0] + \tilde{a}_1 H_1$$

$$\hat{y}(t+2) = G_2 \Delta u(t) + H_2 \Delta u(t-1) + F_2 y(t)$$

Para $j = 3$

$$\hat{y}(t+3) = [b_1 \ b_2 \ \dots \ b_{N_b}] \Delta u(t+2) + F_1 y(t+2)$$

$$\hat{y}(t+3) = [b_1 \ b_2 \ \dots \ b_{N_b}] \Delta u(t+2) + \tilde{a}_1 y(t+2) + \tilde{a}_2 y(t+1) + [\tilde{a}_3 \ \dots \ \tilde{a}_{N_a+1} \ 0 \ 0] y(t)$$

$$\hat{y}(t+3) = [b_1 \ b_2 \ \dots \ b_{N_b}] \Delta u(t+2) + \tilde{a}_1 (G_2 \Delta u(t) + H_2 \Delta u(t-1) + F_2 y(t))$$

$$+ \tilde{a}_2 (G_1 \Delta u(t) + H_1 \Delta u(t-1) + F_1 y(t)) + [\tilde{a}_3 \ \dots \ \tilde{a}_{N_a+1} \ 0 \ 0] y(t)$$

$$H_3 = [b_4 \ b_5 \ \dots \ b_{N_b} \ 0 \ 0] + \tilde{a}_1 H_2 + \tilde{a}_2 H_1$$

$$\hat{y}(t+3) = G_3 B \Delta u(t) + H_3 u(t-1) + F_3 y(t)$$

Como o foco deste apêndice é determinar a matriz H , que é a matriz de coeficientes que multiplica o vetor das variações incrementais de controle passadas como mostrada na equação 3.15, com base nas três predições assim é possível observar um padrão para as linhas da matriz H

Considerando n como $\min(j-1, N_a+1)$

Para $j \geq N_b$

$$\hat{y}(t+j) = B \Delta u(t-1+N_b) + F_1 y(t-1+N_b)$$

$$H_j = \sum_{k=1}^n \tilde{a}_k H_{j-k}$$

Para $j < N_b$

$$\hat{y}(t+j) = B \Delta u(t-1+j) + F_1 y(t-1+j)$$

$$B_j = [b_{j+1} \ b_{j+2} \ \dots \ b_{N_b} \ \text{zeros}(j-1)] + \sum_{k=1}^n \tilde{a}_k H_{j-k}$$

Portanto as linhas da matriz H podem ser descritas pelas seguinte expressão:

$$H_j = \begin{cases} \sum_{k=1}^n \tilde{a}_k H_{j-k}, & \text{se } j \geq N_b, \\ [b_{j+1} \ b_{j+2} \ \dots \ b_{N_b} \ \text{zeros}(j-1)] + \sum_{k=1}^n \tilde{a}_k H_{j-k}, & \text{se } j < N_b, \\ [b_2 \ b_3 \ \dots \ b_{N_b}], & \text{se } j = 1. \end{cases} \quad (\text{C.2})$$

REFERÊNCIAS

- ABATE, A. et al. Sound and automated synthesis of digital stabilizing controllers for continuous plants. In: *Proceedings of the 20th International Conference on Hybrid Systems: Computation and Control*. New York, NY, USA: ACM, 2017. (HSCC '17), p. 197–206. ISBN 978-1-4503-4590-3. Disponível em: <<http://doi.acm.org/10.1145/3049797.3049802>>. Citado na página 48.
- ARRIETA, O. et al. Improved pid controller tuning rules for performance degradation/robustness increase trade-off. v. 98, p. 12042–12047, 03 2016. Citado na página 63.
- ARROYO, J. *Heurísticas e metaheurísticas para otimização combinatória multiobjetivo*. 253 p. Tese (Doutorado) — Universidade Estadual de Campinas, Brasil, 2002. Citado na página 27.
- ASTROM, K. J.; RICHARD, M. M. *Feedback Systems*. [S.l.]: Princeton Univeristy Press, 2008. 396 p. Citado na página 39.
- ASTROM, K. J.; WITTENMARK, B. *Computer-controlled Systems*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1997. 557 p. Citado 4 vezes nas páginas 36, 37, 42 e 44.
- ASTROM, K. J. A.; HAGGLUND, T. *PID Controllers: Theory, Design, and Tuning*. 2. ed. [S.l.]: Instrument Society of America, 1995. 354 p. Citado na página 16.
- BAVAFA-TOOSI, Y. A note on the gain and phase margin concepts. *Journal of Control and Systems Engineering*, v. 3, p. 51 – 59, 2015. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0104-66322016000200333&nrm=iso>. Citado 2 vezes nas páginas 44 e 45.
- BAYER, F. M.; ARAÚJO, O. C. B. Apostila. *Controle Automático de Processos*. 3. ed. [s.n.], 2011. Disponível em: <http://estudio01.proj.ufsm.br/cadernos_automacao/quinta_etapa/control_e_automatizado_processos_2012>. Citado na página 14.
- BISPO, D. D. S. *Equações diofantinas lineares e suas aplicações*. 75 p. Monografia (Licenciatura em Matemática) — Universidade Estadual do Sudoeste da Bahia, 2013. Disponível em: <<http://www.uesb.br/mat/download/Trabamonografia/2013/Dinguiston.pdf>>. Citado na página 23.
- CAMACHO, E.; BORDONS, C. *Model Predictive Control*. [S.l.]: Springer, 2004. 405 p. Citado na página 18.
- CAVALCANTI, A. L. *Estudo e implementação de um controlador preditivo generalizado bilinear compensado*. 68 p. Dissertação (Mestrado em Engenharia Elétrica) — Universidade Federal do Rio Grande do Norte, 2003. Disponível em: <<https://repositorio.ufrn.br/jspui/bitstream/123456789/15425/1/AndersonLOC.pdf>>. Citado na página 14.
- CLARK, D. e. a. Generalized predictive control- part i. the basic algorithm. In: *Pergamon Journals Ltd*. [S.l.: s.n.], 1987. v. 23, n. 2, p. 137–148. Citado 4 vezes nas páginas 19, 20, 21 e 22.
- CLARKE, D.; MOHTADI, C.; TUFFS, P. Generalized predictive control—part ii extensions and interpretations. v. 23, p. 149–160, 03 1987. Citado 3 vezes nas páginas 47, 49 e 81.

COSTA, J. T. The darwinian revelation: Tracing the origin and evolution of an idea. *BioScience*, v. 59, n. 10, p. 886–894, 2009. Disponível em: <<http://dx.doi.org/10.1525/bio.2009.59.10.10>>. Citado na página 28.

DUTRA, C. B. S. *Estudo e implementação de um controlador preditivo generalizado bilinear compensado*. 127 p. Tese (Doutorado em Engenharia Elétrica) — Universidade Federal de Santa Catarina, Brasil, 2003. Citado na página 14.

FILHO PAULO A; POPPI, R. D. C. Aplicação de algoritmos genéticos na seleção de variáveis em espectroscopia no infravermelho médio. determinação simultânea de glicose, maltose e frutosos. *Quimi. Nova*, v. 25, n. 1, p. 46–52, 2002. Citado na página 32.

FORTIN, F.-A.; PARIZEAU, M. Revisiting the nsga-ii crowding-distance computation. In: *Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation*. New York, NY, USA: ACM, 2013. (GECCO '13), p. 623–630. ISBN 978-1-4503-1963-8. Disponível em: <<http://doi.acm.org/10.1145/2463372.2463456>>. Citado 2 vezes nas páginas 33 e 34.

GOLNARAGHI, F.; KUO, B. *Sistemas de Controle Automático*. 9. ed. [S.l.]: LTC, 2012. 716 p. Citado na página 38.

HASHIMOTO, K. *Métodos de otimização multiobjetivo in: Técnicas de otimização combinatória multiobjetivo aplicadas na estimação do desempenho elétrico de redes de distribuição*. 127 p. Tese (Doutorado em Engenharia) — cola Politécnica da Universidade de São Paulo, Universidade de São Paulo, Brasil, 2004. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/3/3143/tde-19112004-165342/pt-br.php>>. Citado 2 vezes nas páginas 27 e 28.

JÚNIOR, G. N. *Sintonia ótima de controladores MPC considerando incertezas de modelagem*. 87 p. Dissertação (Mestrado em Engenharia Industrial) — Universidade Federal da Bahia, 2015. Citado 2 vezes nas páginas 16 e 17.

LEE, J.; YU, Z. Tuning of model predictive controllers for robust performance. In: *Computers & Chemical Engineering*. [S.l.: s.n.], 1994. v. 18, n. 1, p. 15–37. Citado na página 16.

LEITE, P. T. e. a. Aplicação de algoritmos genéticos na determinação da operação ótima de sistemas hidrotérmicos de potência. *Revista Controle e Automação.*, v. 17, n. 1, 2016. Disponível em: <<http://www.scielo.br/pdf/ca/v17n1/a08v17n1>>. Citado na página 32.

LI, D. et al. Maximum sensitivity based fractional imc–pid controller design for non-integer order system with time delay. v. 31, 07 2015. Citado na página 39.

MACIEJOWSKI, J. M. *Predictive control: with constraints*. 1. ed. [S.l.]: Prentice Hall, 2000. 352 p. Citado na página 16.

MAITELLI, A. L.; CAVALCANTI, A. L. Sintonia de controladores preditivos baseada em algoritmos genéticos multi-objetivos. In: *XII Latin-American Congress on Automatic Control*. [S.l.: s.n.], 2006. p. 116–120. Citado 3 vezes nas páginas 14, 15 e 32.

MARINHO, D. *Uma aplicação do algoritmo genético multiobjetivo NSGA II para seleção de imagens de satélite de trechos de mata atlântica*. 68 p. Monografia (Bacharel em Engenharia da Computação) — Universidade de Pernambuco, 2009. Disponível em: <<http://tcc.ecomp.poli.br/20091/TCC%20Davi%20Marinhho%20NSGA%20II%20TCC%202009-1.pdf>>. Citado 3 vezes nas páginas 32, 33 e 41.

- MARLIN, T. E. *Process Control*. [S.l.]: McGraw-Hill, 1995. Citado 2 vezes nas páginas 34 e 35.
- MCINTOSH, A. R. e. a. Selection of tuning parameters for adaptive generalized predictive control. In: *Proceedings of the 1989 American Control Conference*. [S.l.: s.n.], 1989. p. 21–23. Citado 3 vezes nas páginas 47, 49 e 81.
- MCINTOSH, A. R. e. a. Analysis and tuning of adaptive generalized predictive control. v. 69, p. 97 – 110, 1991. Citado 3 vezes nas páginas 47, 49 e 81.
- MOREIRA, A. P. G. M. *O Controle Predictivo*. 2002. Disponível em: <https://paginas.fe.up.pt/~amoreira/documentospdf/controlo_preditivo.pdf>. Citado na página 14.
- NARIÑO, G. A. *Otimização in: Otimização de risers em catenária com amortecedor hidrodinâmicos*. 42-68 p. Dissertação (Mestrado em Engenharia Civil) — Pontifícia Universidade Católica do Rio de Janeiro, 2014. Disponível em: <https://www.maxwell.vrac.puc-rio.br/24617/24617_1.PDF>. Citado 6 vezes nas páginas 24, 25, 31, 32, 33 e 34.
- NISE, N. S. *Engenharia de Sistemas de Controle*. 6. ed. [S.l.]: LTC, 2013. Citado 3 vezes nas páginas 14, 35 e 36.
- NORMEY-RICO, J.; CAMACHO, E. *Control of Dead-Time Processes*. [S.l.]: Springer, 2007. 459 p. Citado 3 vezes nas páginas 14, 18 e 19.
- OPPENHEIM, A.; SCHAFFER, R. *Processamento Em Tempo Discreto De Sinais*. 3. ed. [S.l.]: PEARSON BRASIL, 2013. ISBN 9788581431024. Citado na página 35.
- PEREIRA, R. *Robustez e sensibilidade paramétrica de controladores com base na identificação bayesiana- abordagem contínua*. 123 p. Dissertação (Mestrado em Engenharia Elétrica) — Instituto Militar de Engenharia, 2008. Citado 2 vezes nas páginas 36 e 38.
- PEREIRA, W. F. *Projeto e avaliação de controlador preditivo generalizado sujeito a restrições via métodos*. 91 p. Dissertação (Mestrado em Engenharia Elétrica) — Universidade Federal de Santa Catarina, 1997. Disponível em: <<https://core.ac.uk/download/pdf/30390622.pdf>>. Citado 3 vezes nas páginas 14, 19 e 21.
- POZO, A. et al. *Computação Evolutiva*. [S.l.], 20–? Citado 2 vezes nas páginas 29 e 30.
- RIBEIRO, J. B. *Controlador Preditivo Generalizado (GPC)*. Campinas. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Norte, 2003. Disponível em: <<http://www.dt.fee.unicamp.br/~jbosco/ia703/capitulo6.pdf>>. Citado 5 vezes nas páginas 14, 19, 23, 24 e 27.
- SARHADI, P.; SALAHSHOOR, K.; KHAKI-SEDIGH, A. Robustness analysis and tuning of generalized predictive control using frequency domain approaches. In: *Mathematical Modelling*. [S.l.: s.n.], 2012. v. 36, n. 12, p. 6167–6185. Citado na página 16.
- SUNAN, H.; KIONG, T.; HENG, L. *Applied predictive control*. [S.l.]: Springer, 2002. Citado 4 vezes nas páginas 16, 18, 19 e 21.
- TIZZO, L. M.; LOPES, L. C. Otimização com multiobjetivo aplicada à sintonia de controlador preditivo. In: *VIII Congresso Brasileiro de Engenharia Química em Iniciação Científica*. [S.l.: s.n.], 2009. p. 27–30. Citado na página 15.

TRIERWEILER, J. O.; FARINA, L. A.; DURAIKI, R. G. Rpn tuning strateg model predective control. In: *DYCOPS*. [S.l.: s.n.], 2001. p. 283–289. Citado 3 vezes nas páginas 47, 49 e 81.

YAMASHITA, A. S.; ZANIN, A. C.; ODLOAK, D. TUNING OF MODEL PREDICTIVE CONTROL WITH MULTI-OBJECTIVE OPTIMIZATION. *Brazilian Journal of Chemical Engineering*, scielo, v. 33, p. 333 – 346, 06 2016. ISSN 0104-6632. Citado 3 vezes nas páginas 47, 49 e 81.

ZINI, E. *Algoritmo genético especializado na resolução de problema de variáveis contínuas e altamente restritos*. 144 p. Dissertação (Mestrado em Engenharia Elétrica) — Universidade Estadual Paulista, 2009. Disponível em: <http://www.feis.unesp.br/Home/departamentos/engenhariaeletrica/pos-graduacao/234-dissertacao_erico.pdf>. Citado 2 vezes nas páginas 17 e 32.