

UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO DE JOINVILLE
CURSO DE ENGENHARIA MECATRÔNICA

GABRIEL CÉSAR SILVEIRA

DESENVOLVIMENTO DE CONTROLADORES ADAPTATIVOS PARA TRATAMENTO
INTENSIVO DE PACIENTES HIPERTENSOS

Joinville
2023

GABRIEL CÉSAR SILVEIRA

DESENVOLVIMENTO DE CONTROLADORES ADAPTATIVOS PARA TRATAMENTO
INTENSIVO DE PACIENTES HIPERTENSOS

Trabalho de Conclusão de Curso apresentado como requisito parcial para obtenção do título de bacharel em Engenharia Mecatrônica no curso de Engenharia Mecatrônica, da Universidade Federal de Santa Catarina, Centro Tecnológico de Joinville.

Orientador: Dr. Alexandro Garro Brito

Joinville
2023

Dedico este trabalho às instituições públicas de ensino e a todas as formas de democratização da cultura, conhecimento e tecnologia.

AGRADECIMENTOS

Agradeço ao meu professor e orientador Alexandro Garro Brito pelos anos em que dividimos salas e laboratório. Graças ao senhor pude começar a entender meu lugar na Engenharia Mecatrônica.

Agradeço aos membros do grupo de robótica Botcem, nele sempre encontrei abrigo e histórias que valem a pena ser contadas.

Agradeço aos meus amigos e colegas, principalmente Guilherme Turatto, Gabriel Leal, Antônio Nascimento e Leonardo Pinzon, por estarem presentes em tantos momentos essenciais da minha graduação. Também agradeço aos meus amigos André Mahel, Denis Pereira, Lucas Zilz e Luiza Schulz, que permaneceram constantes em minha vida mesmo quando estive distante.

Agradeço de forma especial meu querido amigo Lucas Cristiano Bernardoni por carregar minha amizade nestes longos anos de Engenharia Mecatrônica, pois risadas e boas conversas transformaram anos difíceis em momentos incrivelmente simples. Graças a você pude continuar.

Agradeço as minhas irmãs Isabel Cristina Silveira e Fabiana Rafaela Silveira, pelo carinho e por serem meus grandes exemplos de dedicação e competência. Assim serão para sempre.

Por fim, agradeço aos meus pais pelo imenso amor, paciência e esforço que investiram durante todos os momentos da minha vida. Não há trabalhos acadêmicos que possam expressar a gratidão de um filho pelos pais, mas esse é a minha tentativa.

Obrigado a todos.

"If we wait until we're ready we'll be waiting for the rest of our lives"
- Lemony Snicket .

RESUMO

Crises hipertensas exigem intervenção médica prolongada para que morbidade e mortalidade sejam evitadas. Nas intervenções médicas, a adoção de controladores automáticos pode reduzir procedimentos manuais. Dessa forma, tratamentos automatizados para pacientes hipertensos auxiliam a atuação de profissionais experientes, embora controladores com ganhos fixos possam resultar em comportamentos indesejados quando usados em indivíduos com características imprevistas. A degradação do comportamento dos controladores clássicos pode ser causada por imprecisões do modelo matemático considerado no projeto. O modelo matemático da pressão arterial em resposta ao medicamento contém efeitos de componentes estocásticos, atrasos e variações temporais. Para contornar esses problemas, diversas formas de controle adaptativo são propostas na literatura e, com intuito inovador, esse trabalho propõe a adoção do controlador PID direta e explicitamente adaptado. Portanto, este trabalho pretende desenvolver controladores adaptativos via simulação numérica, considerando conceitos de aprendizado do algoritmo e aspectos dos procedimentos clínicos, tal que a ação de controle seja otimizada ao perfil dos pacientes.

Palavras-chave: Controle Adaptativo. Adaptação Direta. Simulação. Hipertensão.

ABSTRACT

Hypertensive crises demands prolonged medical intervention to prevent morbidity and mortality. The adoption of automatic controllers can reduce manual procedures, thereby facilitating the performance of experienced professionals in providing automated treatments for hypertensive patients. Nonetheless, the use of fixed-gain controllers in individuals with unforeseen characteristics like hypersensitivity can result in undesired behaviors. The degradation of classical controller behavior may stem from inaccuracies in the mathematical model used in the design step, which incorporates stochastic components, delays, and temporal variations in the response of blood pressure to medication. To address these challenges, various forms of adaptive control have been proposed in the literature. In this study, we propose an innovative approach by adopting the DIRAC controller in PID format. Our objective is to develop PID controllers that are directly adapted through numerical simulation, taking into account algorithm learning concepts and clinical procedure aspects.

Keywords: Adaptive Control. Direct Adaptive. Simulation. Hypertension.

LISTA DE FIGURAS

Figura 1 – Diagrama de blocos do sistema malha fechada	20
Figura 2 – Procedimento para escolha de controladores	24
Figura 3 – Diagrama de blocos do controladores adaptativos indiretos	26
Figura 4 – Diagrama de blocos do controladores adaptativos diretos	26
Figura 5 – Diagrama de blocos da filtragem dos sinais para estimação de $C(q^{-1})$	31
Figura 6 – Procedimento de identificação em malha aberta para design do controlador	33
Figura 7 – Estrutura integrativa comum aos PAA	34
Figura 8 – Exemplo de convergência da estimação conforme sinal de entrada	42
Figura 9 – Escolha de período máximo de um pulso no sinal PRBS	44
Figura 10 – Fluxograma de atendimento da crise hipertensiva	46
Figura 11 – Diagrama resumido dos possíveis mecanismos no desenvolvimento de hipertensão na diabetes mellitus	48
Figura 12 – Diagrama de blocos do controle malha fechada de administração de medicamentos	50
Figura 13 – Diagrama de blocos do modelo de pressão arterial	56
Figura 14 – Fluxograma do algoritmo RLS	61
Figura 15 – Indicadores de desempenho de controladores	63
Figura 16 – Distribuição Normal	64
Figura 17 – Exemplo dos Indicadores para Avaliação de Robustez	65
Figura 18 – Comparação entre Quatro Perfis de Paciente em Resposta ao degrau	68
Figura 19 – Comparação visual da resposta dinâmica ao sinal escada	70
Figura 20 – Comparação visual da estimação gerada pelo sinal escada	71
Figura 21 – Comparação visual da estimação convertida em formato PID gerada pelo sinal escada	71
Figura 22 – Dispersão de 1000 Amostras em Comparação à Média, MAP e Região Ideal do DIRAC com Estimador 2	73
Figura 23 – Diagrama Principal do Ambiente de Simulação	86
Figura 24 – Diagrama do Modelo de Pressão Arterial	86
Figura 25 – Diagrama da Atividade de Background do Modelo de Pressão Arterial	87
Figura 26 – Diagrama da Resposta ao Medicamento do Modelo de Pressão Arterial	87
Figura 27 – Diagrama Principal do Controlador	88
Figura 28 – Diagrama Expandido do PID Dirac	88
Figura 29 – Diagrama do Mecanismos de Identificação de Acomodação e Ativação do Controlador	89

Figura 30 – Diagrama Principal do Adaptador	89
Figura 31 – Diagrama da Filtragem de Sinais para Estimação	90
Figura 32 – Diagrama do Estimador 1	90
Figura 33 – Diagrama do Estimador 2	91
Figura 34 – Diagrama do Estimador 3	91
Figura 35 – Diagrama da Operação Recursiva do PID com Coeficientes Estimados	92

LISTA DE QUADROS

Quadro 1 – Efeitos da variação nos parâmetros do controlador PID	22
Quadro 2 – Estruturas de modelo	27
Quadro 3 – Efeitos da Adrenalina, Noradrenalina, Glicocorticoide, Aldosterona e Angiotensina 2	49
Quadro 4 – Resumo do desempenho dos controladores de PA	51
Quadro 5 – Distinções entre os algoritmos de estimação	61
Quadro 6 – Perfis de Pacientes	66

LISTA DE TABELAS

Tabela 1 – Valores esperados do modelo contínuo	54
Tabela 2 – Valores esperados do modelo discreto	57
Tabela 3 – Classificação da pressão arterial de acordo com a medição no consultório a partir de 18 anos de idade	67
Tabela 4 – Avaliação Preliminar dos Sobressinais Máximos (<i>mmHg</i>)	69
Tabela 5 – Avaliação Preliminar dos Tempos de Acomodação Máximos (<i>s</i>)	69
Tabela 6 – Avaliação Preliminar dos Erros de Estado Estacionário Máximos (<i>mmHg</i>)	69
Tabela 7 – Avaliação dos Sobressinais Máximos (<i>mmHg</i>)	72
Tabela 8 – Avaliação dos Tempos de Acomodação Máximos (<i>s</i>)	72
Tabela 9 – Avaliação dos Erros de Estado Estacionário Máximos (<i>mmHg</i>)	72
Tabela 10 – Avaliação dos Percentuais Residuais da Perturbação (%)	75
Tabela 11 – Avaliação dos Tempos de Resposta à Perturbação (<i>s</i>)	76
Tabela 12 – Avaliação do nível de infusão em Resposta à Perturbação (<i>mL/h</i>)	76

LISTA DE SÍMBOLOS

$u(t)$	Sinal de comando na saída do controlador
$e(t)$	Sinal de erro comparativo entre a saída e referência do processo
$y(t)$	Sinal de saída do processo
$w(t)$	Sinal de referência do processo
K_p	Ganho proporcional do controlador PID
K_i	Ganho integrativo do controlador PID
K_d	Ganho derivativo do controlador PID
K_c	Ganho proporcional do controlador PID reformulado
T_i	Constante de tempo de integração no controlador PID reformulado
T_d	Constante de tempo de derivação no controlador PID reformulado
$\psi(t)$	Sinal de saída verdadeiro do processo estimado
$\hat{\psi}$	Sinal de saída do processo estimado
ϕ	Vetor regressor composto de sinais conhecidos
θ^*	Vetor verdadeiro de parâmetros desconhecidos
$\hat{\theta}$	Vetor estimado de parâmetros desconhecidos
n_θ	Dimensão do modelo paramétrico
$r(t)$	Perturbação ou ruído no modelo paramétrico generalizado
$x(t)$	Entrada do modelo no modelo paramétrico generalizado
q	Operador de deslocamento temporal
$C(q^{-1})$	Controlador recursivo
c_n	Coeficientes do polinômio $C(q^{-1})$
$\Delta u(t)$	Incremento no comando de controle
T_s	Tempo de amostragem do controlador

$P(q^{-1})$	Função de transferência verdadeira da planta desconhecida
$R(q^{-1})$	Modelo de referência
u_f	Entrada filtrada para estimação
y_f	Saída filtrada para estimação
U_f	Filtro de entrada para estimação
Y_f	Filtro de saída para estimação
$\varepsilon(t)$	Erro residual de estimação
d_r	Parâmetro de atraso temporal do modelo de referência
a_r	Parâmetro de velocidade do modelo de referência
T_r	Tempo de acomodação do modelo de referência
$\hat{\theta}(k)$	Estimativa dos parâmetros desconhecidos
$\bar{e}(k)$	Erro genérico presente em estimadores online
$V(\hat{\theta}, t)$	Função custo do algoritmo estimador
ε	Erro de predição da estimação
$F(t)$	Matriz covariância
$\hat{\varepsilon}_0$	Erro de predição da estimação a priori
$\hat{\psi}_0$	Sinal de saída do processo estimado predito a priori
λ	Fator de esquecimento simplificado
$\lambda_1(t)$	Fator incrementador de esquecimento
$\lambda_2(2)$	Fator decrementador de esquecimento
N_λ	Horizonte de memória
T_{ff}	Constante de tempo de esquecimento exponencial
T	Tempo de amostragem do estimador
ϱ_1, ϱ_2	Valores reais maiores que zero
n	Dimensão do vetor regressor

m_{PE}	Inteiro condicional para que o estimador seja permanentemente excitado
p	Quantidade de componentes senoidais em $x(t)$ para que o estimador seja permanentemente excitado
ω_i	Frequências distintas que compõe o sinal $x(t)$
$\eta(t)$	Vetor de elementos com variação estocástica gaussiana
ν	Constante de escala
$\Upsilon(t)$	Vetor de elementos com variação vagarosa
p_a	Pressão arterial
p_{ba}	Atividade de background da pressão arterial
Δp_d	Varição da pressão arterial causada pelo medicamento
\bar{p}_a	Pressão arterial média vista pelo monitorador cateter-transdutor
$S(s)$	Filtro passa-baixa do equipamento monitorização da pressão arterial
$G_d(s)$	Função de transferência da ação do medicamento
$I(s)$	Taxa de infusão de medicamento
τ_i	Atraso de tempo para transporte após injeção
τ_c	Atraso de tempo devido à recirculação
K	Sensibilidade ao medicamento
α	Fração de recirculação
τ	Constante de tempo da absorção de medicamento
p_o	Componente de offset da pressão arterial
p_δ	Componente de reflexo da pressão arterial
p_s	Componente senoidal da pressão arterial
p_η	Componente estocástica da pressão arterial
$F_\eta(s)$	Filtro passa-baixa que caracteriza a componente estocástica da pressão arterial
K_η	Ganho ajustado para o filtro $F_\eta(s)$

$F_\delta(s)$	Filtro passa-baixa atraso temporalmente que caracteriza a ação de reflexo da pressão arterial
τ_δ	Atraso de tempo da ação de reflexo da pressão arterial
d	Atraso discreto relativo ao atraso de transporte da pressão arterial
m	Atraso discreto relativo ao atraso de recirculação da pressão arterial
b_0, b_m, a_1	Coefficientes do modelo discreto de pressão arterial

SUMÁRIO

1	INTRODUÇÃO	17
1.1	Objetivos	19
1.1.1	Objetivo Geral	19
1.1.2	Objetivos Específicos	19
2	FUNDAMENTAÇÃO TEÓRICA	20
2.1	Sistemas de controle clássicos	20
2.2	Alternativas ao controle clássico	23
2.3	Controladores adaptativos	24
2.3.1	Estrutura lógica do controlador adaptativo	25
2.3.2	Controlador PID adaptativo	28
2.3.3	Estimação de parâmetros	32
2.3.3.1	Algoritmo Recursive Least Squares	35
2.3.3.2	Convergência do algoritmo estimador	41
2.4	Hipertensão	45
2.4.1	Contexto Clínico da Hipertensão	45
2.4.2	Controle da pressão arterial	50
2.4.3	O modelo matemático da pressão arterial	53
3	MÉTODO	59
3.1	Detalhes de implementação	59
3.1.1	Controlador DIRAC	59
3.1.2	Estimadores	60
3.2	Indicadores de desempenho	63
3.3	Verificação do DIRAC para Pressão Arterial	66
4	CONCLUSÕES	78
	REFERÊNCIAS	81
	APÊNDICE A	86

1 INTRODUÇÃO

Crises hipertensas podem surgir em pacientes submetidos a tratamento intensivo, exigindo o reconhecimento de sintomas e intervenção médica a fim de evitar a morbidade e mortalidade causadas pelos danos da lesão progressiva do órgão-alvo (CARDOSO, 2013). O controle da pressão arterial é realizado pelo ajuste da bomba de difusão ou vaporizador de medicamento, procedimento que, devido ao longo período de tempo necessário, pode se tornar tedioso para humanos, favorecendo a distração do profissional responsável (MILLARD et al., 1987).

A adoção de controle automático tem mostrado seu potencial ao longo das décadas e, atualmente, está disseminado em artefatos de alta tecnologia, como veículos fabris e espaciais, e em processos de manufatura moderna (NISE; SILVA, 2004). A ampla gama de aplicações de controladores justifica-se pela possibilidade de operar sobre diferentes aspectos da natureza como pressão, temperatura, umidade, viscosidade, vazão e, com isso, incrementar a produtividade com a redução de operações manuais repetitivas (OGATA, 1998).

O estudo comparativo indica que, em alguns casos, controladores automatizados para tratamento da hipertensão podem resultar em performance semelhante à atuação de profissionais anestesistas experientes dedicando-se inteiramente à tarefa de controlar a pressão do paciente. A interferência de um profissional anestesista pode proporcionar piores resultados em relação à rapidez e sobressinal quando comparados ao do controle com realimentação em casos de pressão sanguínea com variações pequenas e lentas (WESTENSKOW, 1987).

Sendo assim, emprega-se o controle automático de pressão sanguínea em milhares de pacientes desde os anos 1980, porém, alguns pacientes podem ser hipersensíveis ao medicamento e, portanto, requerem atenção particular no uso de controladores de ganhos estáticos. Nesses pacientes, as formas clássicas de controle com ganhos fixos podem levar à instabilidade, gerando grandes oscilações. Para contornar essa situação de risco, é interessante o uso de métodos adaptáveis à sensibilidade do indivíduo e outras variações do processo (WESTENSKOW, 1987).

De acordo com Åström e Wittenmark (2013), o projeto da maioria dos controladores assume que o modelo proposto seja linear ou próximo disso, sendo que o procedimento padrão bem sucedido implica em desenvolver um modelo linearizado em torno de algum ponto de operação e prosseguir o design do controlador com parâmetros constantes tomando o modelo como verdadeiro. Entretanto, essa consideração de projeto pode comprometer os resultados e, nessa perspectiva, Landau et al. (2011) afirmam que o comportamento de variações desconhecidas e não mensuradas nos

parâmetros do processo degradam o resultado do sistema de controle.

Para Ljung (1999), ao observar um sistema e estabelecer as relações entre seus comportamentos, pode-se propor modelos matemáticos, possibilitando análises e artifícios de engenharia, como simulações numéricas e previsões. Slate et al. (1979) apresentam o procedimento de modelagem matemática da pressão sanguínea em resposta à infusão de medicamentos ao longo do tempo, incluindo os efeitos das formas de ruído e a variação temporal das características fisiológicas, como atrasos de transporte e recirculação, absorção, distribuição e metabolização do medicamento.

Baseada nessa modelagem, a comunidade acadêmica tem proposto diferentes abordagens de controle adaptativo da pressão arterial para tratamentos intensivos; dentre essas, pode-se encontrar controladores Proporcional Integral (PI) (ZHU; ZHENG; LAVANYA, 2006), estocásticos de variação mínima (ARNSPARGER et al., 1983; MILLARD et al., 1987), fuzzy neural (GAO; ER, 2005), por modelo de referência (ISAKA; SEBALD, 1993) e multi modelos (HE; KAUFMAN; ROY, 1986).

Com intuito inovador, o presente trabalho introduz um controlador explicita e diretamente adaptado, conforme formulado por Keyser (2000), à problemática do tratamento de hipertensos. Assim, o algoritmo computa iterativamente o controlador tal que solucione a otimização do sinal de controle em relação ao modelo de referência. O algoritmo de controle é baseado em formas alternativas do Recursive Least Squares, sendo ajustado em projeto tal que lide satisfatoriamente com as incertezas e dinâmicas do processo.

Bailey e Haddad (2005) ressaltam que o controle de dosagem com realimentação não é amplamente adotado devido ao número de variáveis envolvidas e suas interações dinâmicas, mas que o avanço no estudo de técnicas adaptativas pode levar a maior adesão. Nessa perspectiva, propõe-se o desenvolvimento de um controlador PID diretamente adaptado de pressão arterial, via simulação numérica, tal que a influência dos parâmetros de projeto seja avaliada na intenção de diminuir a divergência em relação à resposta do modelo referência na presença de diferentes situações de variação, incertezas e ruído.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

Propor controladores PID explicita e diretamente adaptados para tratamento intensivo de pacientes hipertensos e verificar sua operação via simulação numérica.

1.1.2 Objetivos Específicos

- Contextualizar a problemática da hipertensão arterial;
- Descrever os fundamentos e a estrutura lógica do algoritmo estimador, bem como lei de controle adaptativo;
- Implementar os controladores adaptativos para o modelo de pressão arterial em ambiente de simulação;
- Avaliar o desempenho das variantes do algoritmo em diferentes situações de hipertensão arterial.

2 FUNDAMENTAÇÃO TEÓRICA

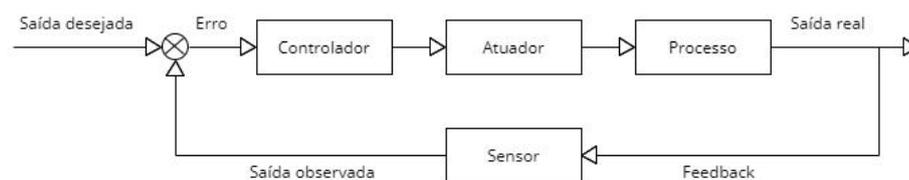
Este capítulo é dedicado aos conceitos necessários para o entendimento da problemática abordada e da proposta de controle. São apresentados tópicos como contextualizações da hipertensão, algoritmos de estimação e controladores adaptativos.

2.1 SISTEMAS DE CONTROLE CLÁSSICOS

Os sistemas de controle idealizados por humanos têm registro de origem próximo a 300 a.C, com a a criação do relógio de água e da lâmpada de óleo. Outro avanço significativo nesse estudo foi a invenção da válvula de segurança em meados de 1691, que efetua a regulação da pressão do vapor. A teoria moderna de sistemas de controle iniciou em 1868 com os critérios de estabilidade formulados por James Clerk Maxwell e, desde então, sua teoria e número de aplicações tem sido expandida (NISE; SILVA, 2004).

No estudo de controladores pode-se definir os sistemas de malha fechada (Figura 1) como aqueles em que a saída observada é comparada ao comportamento desejado fazendo com que o controlador seja excitado de tal forma que seu comando atue no sistema diminuindo o erro observado.

Figura 1 – Diagrama de blocos do sistema malha fechada



Fonte: Adaptado de Dorf e Bishop (2010, p. 3)

Existem também os sistemas de malha aberta, em que a saída gerada não é medida e, portanto, não afeta as ações de controle, ou seja, não possui realimentação (OGATA, 1998).

De acordo com Ogata (1998), os sistemas de malha aberta são usados quando a relação entre entrada e saída do processo é conhecida e não há distúrbios relevantes. O projeto de sistemas de malha fechada é preferível quando há características problemáticas, como sensibilidade a perturbações externas e variação interna dos parâmetros do sistema, tornando-o essencial em algumas dessas aplicações pois culmina em resultados que não seriam possíveis de outra maneira.

Para Nise e Silva (2004), a medição da saída em malhas fechadas permite que o controlador corrija sua atuação tal que compense efeitos de perturbações, levando a sistemas com maior exatidão, ou seja, menos sensíveis a ruído e alterações do ambiente. Dada a diferença de demandas, Ogata (1998) justifica que a seletividade das aplicações do controle em malha fechada surge do incremento de complexidade, pois são mais difíceis de construir e têm estabilidade mais problemática.

Na literatura, pode-se encontrar métodos para analisar o comportamento de sistemas, possibilitando o projeto de seus controladores para malha fechada. Artifícios de tempo contínuo averiguam a estabilidade e dão origem a técnicas de design de controladores, como métodos de lugar das raízes, métodos de resposta na frequência e formulações algébricas (DORF; BISHOP, 2010; NISE; SILVA, 2004; OGATA, 1998); outros autores, como Phillips e Nagle (1995), tratam dos mesmos temas dando enfoque à controladores digitais, também chamados de *microprocessados*.

Para Dorf e Bishop (2010), o controlador Proporcional Integral Derivativo (PID) tem sucesso em manter seu desempenho mesmo em condições operacionais distintas, sendo amplamente utilizado no contexto industrial graças à formulação com apelo intuitivo, tendo em vista que seus parâmetros possuem tradução temporal de fácil compressão, possibilitando ajustes de desempenho de maneira direta. Åström e Hagglund (1995) destacam a presença dos controladores PID no cotidiano da engenharia de controle, sendo usados na maioria dos processos industriais e, também, em aplicações peculiares.

A expressão temporal do comando $u(t)$ na saída do controlador PID é dada pela Equação 1

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt} \quad (1)$$

onde a expressão do erro temporal $e(t)$ (Equação 2) é a comparação entre o sinal saída e referência

$$e(t) = w(t) - y(t) \quad (2)$$

sendo $y(t)$ o sinal de saída e $w(t)$ o de referência (DORF; BISHOP, 2010).

Quando aplicada a transformada de Laplace, a ação do controlador mostrada na Equação 1 é traduzida para o domínio da frequência por uma função de transferência (Equação 3)

$$\begin{aligned} G_c(s) &= \frac{U(s)}{E(s)} \\ &= K_p + \frac{K_i}{s} + K_d s \end{aligned} \quad (3)$$

onde $U(s)$ é o sinal de saída do controlador e $E(s)$ é o sinal de erro. Fica clara a origem do nome, uma vez observado que K_p é um ganho proporcional ao erro, K_i é o ganho da ação integral do erro e K_d é o ganho da ação derivativa temporal do sinal de erro (DORF; BISHOP, 2010).

Dorf e Bishop (2010) apresentam o Quadro 1 com os efeitos típicos da variação de cada um dos parâmetros do controlador PID.

Quadro 1 – Efeitos da variação nos parâmetros do controlador PID

Variação do Parâmetro	Sobressinal	Tempo de Acomodação	Erro de Estado Estacionário
Aumento de K_p	Aumenta	Impacto mínimo	Diminui
Aumento de K_i	Aumenta	Aumenta	Não possui
Aumento de K_d	Diminui	Diminui	Não interfere

Fonte: Adaptado de Dorf e Bishop (2010, p. 483).

Dessa forma, os efeitos de ajuste do controlador podem ser estimados pelo operador de maneira intuitiva, exemplificando um dos motivos do sucesso industrial dos controladores PID.

O termo integrativo é comumente usado para que a diferença entre saída e referência do processo se anule em regime permanente; tal característica é vista no Quadro 1. A parcela derivativa do controlador PID pode ser associada ao aumento de estabilidade e predição da saída (ÅSTRÖM; HAGGLUND, 1995).

A partir de pequenas modificações nas Equações 1 e 3 obtém-se formas alternativas do controlador PID, como o controlador Proporcional Integral (PI) expresso na Equação 4.

$$G_c(s) = K_p + \frac{K_i}{s} \quad (4)$$

Nota-se que a modificação é a atribuição do ganho derivativo como nulo, concentrando as ações de controle e escolhas de projeto aos ganhos proporcional e integrativo (DORF; BISHOP, 2010).

Segundo Åström e Hagglund (1995), 95% dos controladores em malha fechada usados em processos industriais são PID e, desses, a maioria na versão PI. Portanto, negligenciar o termo derivativo é uma escolha comum e justificada pelo desempenho do controlador integral puro em processos aproximadamente de primeira ordem em relação à estabilidade de Nyquist. Além disso, afirma-se que em plantas compostas por longos períodos mortos o termo derivativo toma pouca ação (ÅSTRÖM; HAGGLUND, 1995).

Autores como Dorf e Bishop (2010), Ogata (1998) e Nise e Silva (2004) apontam a existência de diferentes formas para ajustar controladores PID, como tentativa e erro, critério de estabilidade de Routh-Hurwitz, métodos de Ziegler-Nichols, resposta na frequência e lugar das raízes. As propostas de Ziegler-Nichols e lugar das raízes são métodos amplamente usados na indústria e mais sistemáticos comparados à tentativa e erro, porém são apenas para regulagens offline e suscetíveis às limitações da teoria de controle clássico (DORF; BISHOP, 2010).

Segundo Åström e Hagglund (1995) levantamentos estatísticos demonstram que apenas 20% dos controladores em malha fechada mantêm seu desempenho na presença de variações. Pode-se inferir que, dos sistemas de controles afetados negativamente, 30% sofreram com má sintonização, 30% com problemas nos atuadores e 20% são afetados por falhas diversas, como problemas nos sensores e tempos de amostragem escolhidos inadequadamente (ÅSTRÖM; HAGGLUND, 1995).

2.2 ALTERNATIVAS AO CONTROLE CLÁSSICO

Para Dorf e Bishop (2010), o projeto da maioria dos controles tem como premissa o conhecimento exato do modelo do processo e de seus parâmetros, desconsiderando a ação de dinâmicas e atrasos não modelados, mudanças de ponto de operação, ruídos na leitura do sensor, perturbações imprevistas e variações nos parâmetros do modelo. Åström e Wittenmark (2013) exemplificam como controladores podem lidar inapropriadamente com as variações do processo ou perturbações, dando origem a maiores sobressinais, lentidões na resposta e oscilações prolongadas.

A adoção do controle robusto idealmente assegura um desempenho aceitável do sistema na presença de imprecisões e alterações significativas do modelo. A análise diferencial da mudança percentual relativa entre função de transferência do sistema e do processo é usada para pequenas e conhecidas variações; já para considerar perturbações multiplicativas no projeto de controle robusto emprega-se o critério de estabilidade robusta (DORF; BISHOP, 2010).

Aponta-se em estudos comparativos que controladores robustos respondem mais rápido a variações do processo do que métodos adaptativos, porém os bons resultados dos controladores robustos limitam-se aos casos previamente especificados e considerados no projeto. Dessa forma, controles adaptativos são preferíveis, mesmo que lentos, para variações como o atrasos temporais desconhecidos ao modelo do projeto (ÅSTRÖM; NEUMANN; GUTMAN, 1987).

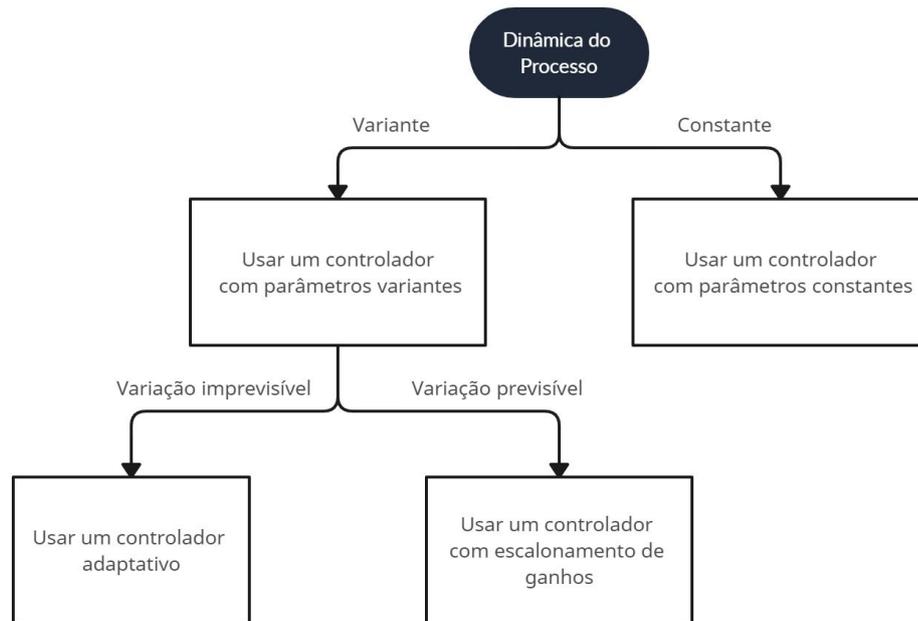
Segundo Åström e Hagglund (1995), modelos são precisos no momento em que o experimento de modelagem é realizado. Entretanto, se as características do processo mudam ao longo do tempo de forma imprevisível os esquemas adaptativos tornam-se uma solução para manter a sintonização do controlador. Para Landau et al. (2011), controles adaptativos podem assegurar o desempenho do sistema mesmo quando os parâmetros do processo são desconhecidos ou variam significativamente devido a mudanças no ambiente e simplificações na etapa de modelagem.

Tendo em vista as consequências e dificuldades de projeto dos controladores adaptativos, Åström e Wittenmark (2013) sugerem prudência, aconselhando sua utilização apenas quando necessários, evitando excessos. Para Ioannou e Fidan (2006), quando há informações prévias suficientes sobre os parâmetros do processo

a aprendizagem online torna-se desnecessária e outras formas de controle são apropriadas.

Nesse sentido, Åström e Wittenmark (2013) apresenta o fluxograma da Figura 2 estabelecendo um breve procedimento para identificar condições inviáveis ao controle adaptativo.

Figura 2 – Procedimento para escolha de controladores



Fonte: Adaptado de Åström e Wittenmark (2013, p. 30)

A Figura 2 mostra também que controladores adaptativos são indicados para lidar com variações imprevisíveis, como o caso da variação de pressão arterial abordado neste estudo, e aponta alternativas para os demais casos com parâmetros invariantes ou variações previsíveis.

2.3 CONTROLADORES ADAPTATIVOS

A origem do esforço acadêmico no tópico de controladores adaptativos nos anos da década de 1950 é atribuído ao estudo de pilotagem automática em sistemas aviônicos, pois esses sistemas sofrem de grandes variações e demandam controladores mais sofisticados do que ganhos constantes atribuídos a apenas um ponto de sua operação. Por essa razão, o escalonamento de ganhos foi criado como uma forma primitiva, mas ainda usual, de controladores adaptativos capazes de lidarem com a problemática do sistemas aviônicos (ÅSTRÖM; WITTENMARK, 2013).

Após o desenvolvimento inicial de sistemas adaptativos analógicos, o estudo

teórico de controladores adaptativos foi fomentado pela inviabilidade de implementação prática nos sistemas de tempo real disponíveis nos anos de 1960 (BOBÁL et al., 2006). Os controladores adaptativos ganharam relevância novamente no final dos anos 1980 com o avanço dos estudos baseados em noções de robustez, estabilidade e ciências da computação (ÅSTRÖM; WITTENMARK, 2013).

Uma definição possível de controle adaptativo parte do termo *adaptar*, que usualmente significa mudar o comportamento conforme novas circunstâncias, da mesma forma que um controlador adaptativo modifica seu comportamento conforme detecta mudanças na dinâmica do processo ou perturbações externas. Logo, um controlador adaptativo é a integração entre certa estrutura de controlador com parâmetros ajustáveis e mecanismo de ajuste (ÅSTRÖM; WITTENMARK, 2013).

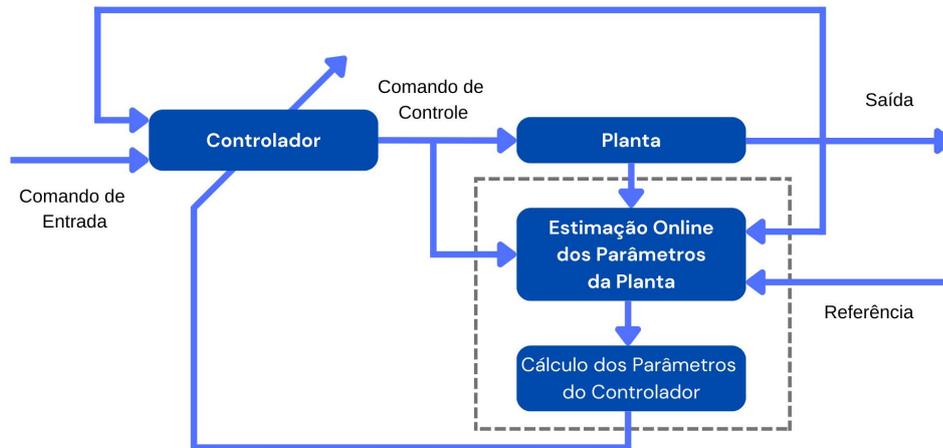
Para Landau et al. (2011), os diversos controladores adaptativos são caracterizados conforme suas técnicas de processar informação a fim de alterar o controlador para que atinja o objetivo performático. Portanto, controle adaptativo é o conjunto de técnicas sistemáticas para ajuste automático do controlador em tempo real, mantendo um nível desejado de desempenho mesmo quando há mudanças nos parâmetros da planta ou esses são, ou se tornam, desconhecidos (LANDAU et al., 2011).

2.3.1 Estrutura lógica do controlador adaptativo

As formas de adaptação podem ser classificadas entre explícitas ou implícitas, sendo as explícitas aquelas em que a lei de atualização dos parâmetros é confiada a um algoritmo estimador, já as implícitas define sua adaptação por meio de uma expressão diferencial que garanta estabilidade e convergência. Assim, as formas explícitas podem proporcionar vantagens, como combinar as estruturas de controlador com diferentes estimadores, programação modular e, portanto, avaliações dos resultados intermediários (ISERMANN, 1991).

As estruturas lógicas dos controladores adaptativos também podem ser divididas entre indiretas e diretas. O controlador é dito indireto (Figura 3) quando necessita da estimação do processo para atualizar o controlador

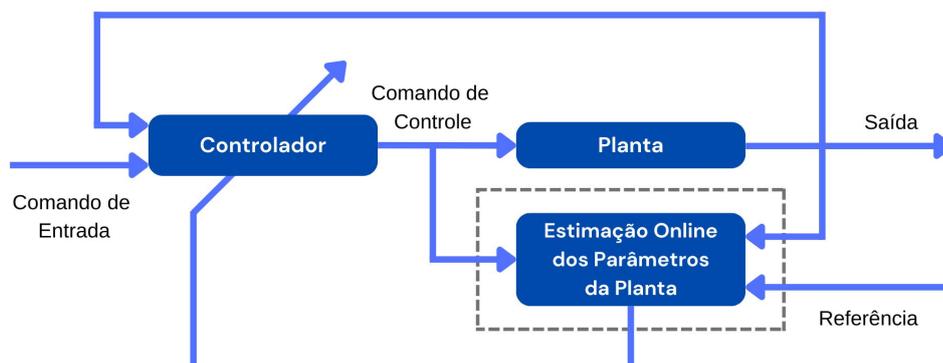
Figura 3 – Diagrama de blocos do controladores adaptativos indiretos



Fonte: Adaptado de Ioannou e Sun (2012, p. 10)

enquanto métodos diretos (Figura 4) parametrizam o sistema em função dos parâmetros do controlador, podendo atualizar-lo diretamente com o algoritmo estimador (ISERMANN, 1991).

Figura 4 – Diagrama de blocos do controladores adaptativos diretos



Fonte: Adaptado de Ioannou e Sun (2012, p. 11)

Åström e Wittenmark (2013) desenvolvem métodos de controle adaptativo diretos e indiretos, mas destacam que os indiretos podem se tornar mal condicionados e, portanto, as formas diretas costumam demandar menos tempo de projeto. Além disso, as considerações de estabilidade dos controladores adaptativos diretos são

assumidas implicitamente confiando no design do estimador.

O comportamento do sistema em relação aos critérios como tempo de subida e percentual de sobressinal pode ser expresso como uma função de transferência; desse modo, o design do controlador pode ser baseado em um modelo de referência. Métodos com modelo de referência são desenvolvidos tal que o erro inicial do sistema desapareça com certa dinâmica e inicialmente não haja diferença entre a saída da planta e do modelo referência (LANDAU et al., 2011).

Portanto, controladores adaptativos não são *projetos caixa preta*, ou seja, requerem conhecimentos prévios sobre a planta em que atuarão para que o comportamento esperado seja atingido (LANDAU et al., 2011). Nos casos em que é possível separar sinais conhecidos e parâmetros desconhecidos, sistemas podem ser descritos com modelos parametrizados linearmente, como na Equação 5

$$\psi(t) = \phi(t)^T \cdot \theta^* \quad (5)$$

(onde $\theta^* \in R^{n_\theta}$ é o vetor de parâmetros desconhecidos, $\psi(t) \in R$ é um sinal conhecido, $\phi(t) \in R^{n_\theta}$ é o vetor regressor composto de sinais conhecidos e n_θ é a dimensão do modelo paramétrico, isto é, a quantidade de parâmetros desconhecidos (TAO, 2003)).

Guidi et al. (2009) mostra que, dada a forma generalizada (Equação 6)

$$A(q)\psi(t) = \frac{C(q)}{F(q)}x(t) + \frac{B(q)}{D(q)}r(t) \quad (6)$$

onde $r(t)$ é a perturbação ou ruído, $x(t)$ é a entrada do modelo e $A(q)$, $B(q)$, $C(q)$ e $D(q)$ são polinômios do operador de deslocamento temporal q , diferentes estruturas podem ser consideradas na parametrização de modelos à medida que são compostas, ou não, por cada um dos polinômios, conforme o Quadro 2.

Quadro 2 – Estruturas de modelo

Polinômios na Equação 6	Nome e sigla dada a estrutura do modelo
C	Finite Impulse Response (FIR)
AC	Autoregressive with Extra Input (ARX)
ABC	Autoregressive Moving Average eXogenous (ARMAX)
AB	Autoregressive Moving Average Model (ARMA)
CF	Output-Error (OE)
CFBD	Box-Jenkins (BJ)

Fonte: Adaptado de Guidi et al. (2009, p. 20).

Segundo Åström e Wittenmark (2013), o modelo exato da resposta impulsiva do processo pode ter dimensão infinita; entretanto, se o processo for estável a resposta decai para zero exponencialmente, possibilitando truncar finitamente o número de termos do modelo e originando a parametrização Finite-Impulse Response (FIR) (Equação 7).

$$\psi(t) = c_0x(k) + c_1x(k-1) + \dots + c_nx(k-n) \quad (7)$$

O modelo FIR (Equação 7) pode ser parametrizado conforme a Equação 5 considerando θ^* e $\phi(t)$ definidos, respectivamente, pelas Equações 8 e 9.

$$\theta^* = [c_0, c_1, \dots, c_n]^T \quad (8)$$

$$\phi(t) = [x(k), x(k-1), \dots, x(k-n)]^T \quad (9)$$

Logo, pode-se parametrizar o modelo FIR em função os valores desconhecidos θ^* e, portanto, utilizar estimadores paramétricos com entrada dos sinais $\psi(t)$ e $\phi(t)$ conhecidos.

2.3.2 Controlador PID adaptativo

Åström e Hagglund (1995) afirmam que os controladores PID mantiveram-se presentes mesmo após as mudanças recentes na engenharia de controle e hoje são quase unânimes no uso de microprocessadores, possibilitando esquemas adaptativos de sintonização digital. Para Bányász e Keviczky (1982), apesar da vasta aplicação de controladores PID, ajustes que necessitam ciclos completos de transitório ou completamente offlines podem não ser triviais e até mesmo inviáveis. Assim, controladores PID tornam-se interessantes aos estudos de adaptações recursivas.

Nessa perspectiva, Keyser e De (1989) e Keyser (2000) desenvolvem o equacionamento e apresentam a simulação de uma classe de controladores adaptativos diretos que usam modelo de referência e, ao final, adaptam-os ao formato PID. Os estudos de Keyser e De (1989) e Keyser (2000) comparam o desempenho dos controladores adaptativos desenvolvidos a controladores de ganhos fixos na presença de ruído e avaliam processos com atrasos no tempo de alta e baixa ordem.

Dada a expressão discreta de um controlador PID genérico (Equação 10)

$$u(t) = u(t-1) + c_0e(t) + c_1e(t-1) + c_2e(t-2) \quad (10)$$

onde $u(t)$ é o comando de controle enviado para a planta e $e(t)$ é o erro da saída em relação à referência $w(t)$ (Equação 2), o equacionamento do controlador $C(q^{-1})$ com saída incremental é elaborada no formato de Finite Impulse Response conforme a Equação 11

$$\begin{aligned} C(q^{-1}) &= \frac{\Delta u(t)}{e(t)} \\ &= c_0 + c_1q^{-1} + c_2q^{-2} \end{aligned} \quad (11)$$

onde c_0 , c_1 e c_2 são os coeficientes do controlador $C(q^{-1})$ estipulado e $\Delta u(t)$ é o comando incremental (Equação 12).

$$\Delta u(t) = u(t) - u(t-1) \quad (12)$$

Conforme Keyser e De (1989), o comando incremental de controle pode ser reescrito no formato PID (Equação 14) tomando a discretização retangular da Equação 13, isto é, substituindo as derivadas temporais na equação por diferenças finitas regressivas

$$u(t) = K_c(e(t) + \frac{1}{T_i} \int e(t)dt + T_d \frac{de(t)}{dt}) \quad (13)$$

$$\Delta u(t) = K_c((1 - q^{-1}) + \frac{T_s}{T_i} + \frac{T_d}{T_s}(1 - 2q^{-1} + q^{-2}))e(t) \quad (14)$$

onde K_c , T_i e T_d são os parâmetros do controlador de ganho proporcional, constante de tempo de integração e derivação, respectivamente, do controlador PID. Na Equação 15 manipula-se a Equação 14 para indicar a relação do controlador PID com o formato FIR (Equação 11) usado na etapa de estimação.

$$\Delta u(t) = K_c \left(\left(1 + \frac{T_s}{T_i} + \frac{T_d}{T_s}\right) - \left(1 + 2\frac{T_d}{T_s}\right)q^{-1} + \frac{T_d}{T_s}q^{-2} \right) e(t) \quad (15)$$

Nota-se que os termos desconhecidos foram agrupados em torno dos operadores de diferença regressiva q^{-1} e q^{-2} .

Na estrutura do controlador PID, os parâmetros c_0 , c_1 e c_2 resultantes da estimação no formato FIR podem ser interpretados como:

$$c_0 = K_c \left(1 + \frac{T_s}{T_i} + \frac{T_d}{T_s}\right) \quad (16)$$

$$c_1 = -K_c \left(1 + 2\frac{T_d}{T_s}\right) \quad (17)$$

$$c_2 = K_c \frac{T_d}{T_s} \quad (18)$$

No mesmo contexto, as equações acima podem ser usadas para obter as expressões dos parâmetros K_c , T_i e T_d em relação aos coeficientes resultantes da estimação, conforme as Equações 19 a 21 (KEYSER; DE, 1989).

$$K_c = -(c_1 + 2c_2) \quad (19)$$

$$T_i = -T_s \frac{c_1 + 2c_2}{c_0 + c_1 + c_2} \quad (20)$$

$$T_d = -\frac{T_s c_2}{c_1 + 2c_2} \quad (21)$$

Dada a função de transferência da planta $P(q^{-1})$ (Equação 22),

$$y(t) = P(q^{-1})u(t) \quad (22)$$

sem prejuízo de generalização, pode-se assumir que a resposta em malha fechada da planta ao comando de referência $w(t)$ é dada pela Equação 23.

$$y(t) = \frac{C(q^{-1})P(q^{-1})}{(1 - q^{-1}) + C(q^{-1})P(q^{-1})}w(t) \quad (23)$$

Do mesmo modo, por obviedade, a relação do controlador e planta com o modelo de referência é dado pela Equação 24.

$$R(q^{-1}) = \frac{C(q^{-1})P(q^{-1})}{(1 - q^{-1}) + C(q^{-1})P(q^{-1})} \quad (24)$$

Dessa maneira, o controlador poderia ser projetado a partir da Equação 25 (KEYSER, 2000).

$$C(q^{-1}) = \frac{(1 - q^{-1})R(q^{-1})}{(1 - R(q^{-1}))P(q^{-1})} \quad (25)$$

Porém, uma vez que a natureza do problema abordado em estudos de controle adaptativo estipula o desconhecimento de $P(q^{-1})$ ao longo do tempo, impossibilita-se a determinação analítica de $C(q^{-1})$ diretamente pela Equação 25. O objetivo do desenvolvimento do controlador DIRAC proposto por Keyser (2000) é, portanto, contornar essa informação faltante no projeto do controlador. Para isso, manipula a igualdade da Equação 25 aplicando o comando $u(t)$ (Equação 26) tal que resulte na Equação 27 independente de $P(q^{-1})$.

$$C(q^{-1})(1 - R(q^{-1}))P(q^{-1})u(t) \approx (1 - q^{-1})R(q^{-1})u(t) \quad (26)$$

$$C(q^{-1})(1 - R(q^{-1}))y(t) \approx (1 - q^{-1})R(q^{-1})u(t) \quad (27)$$

O processamento envolvendo $y(t)$ e $u(t)$ (Equação 27) pode ser expresso como a filtragem do sinais, conforme a Equação 28

$$\begin{cases} u_f(t) = U_f(q^{-1})u(t) \\ y_f(t) = Y_f(q^{-1})y(t) \end{cases} \quad (28)$$

onde

$$U_f(q^{-1}) = (1 - q^{-1})R(q^{-1}) \quad (29)$$

$$Y_f(q^{-1}) = (1 - R(q^{-1})) \quad (30)$$

sendo $U_f(q^{-1})$ e $Y_f(q^{-1})$ dois filtros discretos causais.

Finalmente, graças a forma filtrada dos sinais (Equação 28), define-se a parametrização linear da Equação 27 compensando a aproximação com um erro residual $\varepsilon(t)$ (Equação 31).

$$u_f(t) = C(q^{-1})y_f(t) + \varepsilon(t) \quad (31)$$

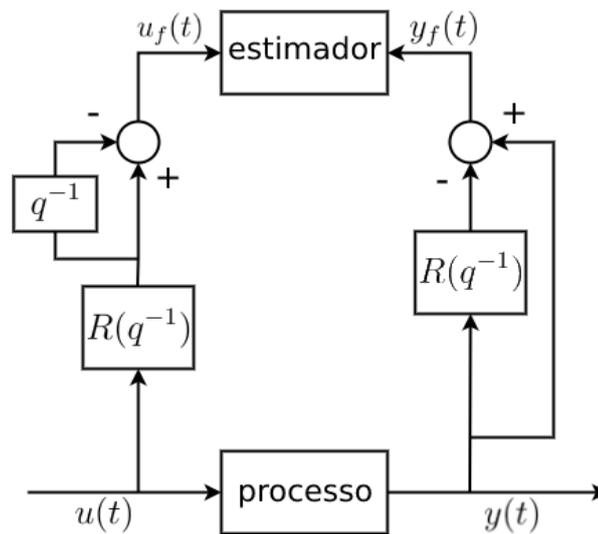
Assim, a Equação 31 formula um problema de otimização para que estime-se $C(q^{-1})$ minimizando o erro de predição de $u_f(t)$ baseado em $y_f(t)$, ou seja, soluciona computacionalmente o problema de determinação do controlador. O problema de otimização da Equação 31 relativo ao controlador PID pode ser linearmente parametrizado pela Equação 5, onde o vetor regressor é dado na Equação 33 e os coeficientes otimizados são contidos na Equação 32

$$\theta = [c_0, c_1, c_2]^T \quad (32)$$

$$\phi(t) = [y_f(t), y_f(t-1), y_f(t-2)]^T \quad (33)$$

A simplicidade lógica da construção dessa proposta é notada na Figura 5 (KEYSER, 2000).

Figura 5 – Diagrama de blocos da filtragem dos sinais para estimação de $C(q^{-1})$



Fonte: Adaptado de Keyser (2000, p. 175)

Conforme apontado por Keyser (2000), a função $R(q^{-1})$ é uma escolha de design no projeto de controle e permite ajustar, por exemplo, a velocidade de resposta esperada. Esse modelo deve refletir o comportamento da planta e por isso estipula-se as seguintes instruções:

- $R(q^{-1})$ deve ser tal que $R(1) \equiv 1$, pois assim o erro de estado estacionário é nulo e a saída do processo tende a atingir o nível definido pelo comando, assim como a planta submetida ao controlador com ação integral;
- Se a planta $P(q^{-1})$ possui um tempo morto conhecido a priori, esse deve ser aproximadamente incluído em $R(q^{-1})$, preferindo estimar sua magnitude em excesso;
- Se a planta $P(q^{-1})$ possui um zero não mínimo conhecido a priori, esse deve ser aproximadamente incluído em $R(q^{-1})$.

O modelo de referência adotado por Keyser e De (1989) (Equação 34) possui um polo real, atraso temporal e ganho estático unitário

$$\begin{aligned} R(q^{-1}) &= \frac{y(t)}{w(t)} \\ &= q^{-d_r} \frac{1 - a_r}{1 - a_r q^{-1}} \end{aligned} \quad (34)$$

onde d_r é o atraso temporal de referência e a_r é descrito como um parâmetro de projeto relacionado ao aumento de velocidade de resposta do sistema conforme é aproximado positivamente de zero.

Segundo Keyser (2000), a definição de $R(q^1)$ é versátil e pode mudar até mesmo em tempo de execução conforme a etapa operacional do processo. Apesar do desenvolvimento original baseado em funções de transferências discretas, a versatilidade do algoritmo em relação a $R(q^1)$ possibilita usar diferentes modelos como redes neurais e modelos não lineares.

A versatilidade do controlador DIRAC proporcionada pela liberdade de definição do modelo de referência ou estrutura do controlador enseja possibilidades de aplicações, como data-drive tuning (VRANČIĆ et al., 2023) ou em áreas biomédicas como em respiradores artificiais (KEYSER; IONESCU, 2011) e tratamento de paralisção muscular (IONESCU; KEYSER, 2007). Contudo, salienta-se que não foram localizados estudos acadêmicos desse controlador em tratamento de crises hipertensas, o que reforça o intuito inovador das propostas deste trabalho.

2.3.3 Estimação de parâmetros

Algumas aplicações requerem que seus parâmetros sejam deduzidos pela observação do comportamento enquanto exercem resposta à certa entrada. Os esquemas de estimação certos para esses casos são chamados *estimadores online*, pois apresentam estimativas do modelo proposto de forma frequente e, para isso, processam em tempo de execução as informações de entrada e saída (IOANNOU; SUN, 2012).

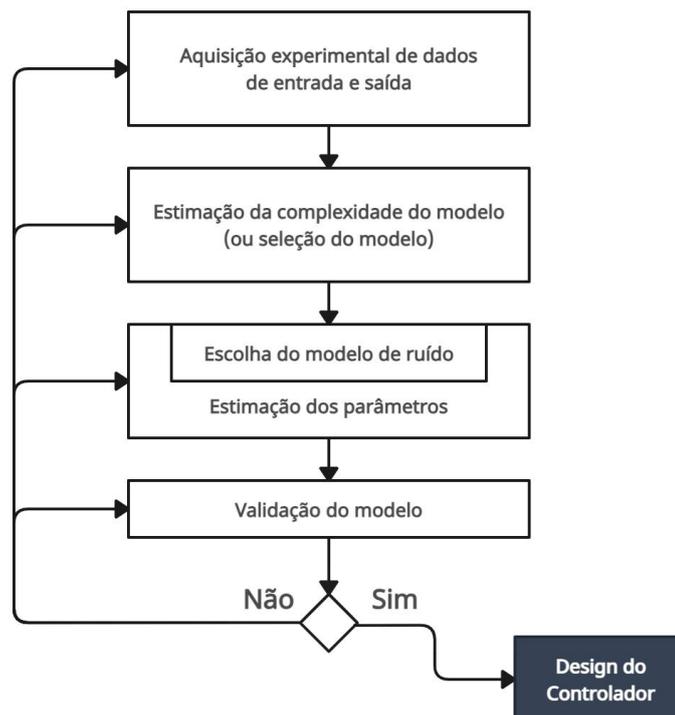
Um algoritmo de estimação pode ser qualificado conforme sua efetividade, robustez e simplicidade, estando a efetividade relacionada à capacidade de perseguir parâmetros constantes ou variantes no tempo; a robustez à geração de estimações sempre que dados forem coletados e a simplicidade à habilidade de manter a complexidade próxima ao de algoritmos não adaptativos (KULHAVÝ; ZARROP, 1993).

Kulhavý e Zarrop (1993) indicam que a falta de conhecimento exato em relação à forma de variação dos parâmetros, a vaga definição dos critérios de qualidade dos algoritmos de estimação e o número de soluções apresentadas pela comunidade acadêmica revelam que não há soluções padronizadas para problemas de estimação de parâmetros, motivando a comunidade acadêmica ao longo dos anos.

O estudo da estimação de parâmetros assemelha-se ao de identificação, sendo que estimadores podem ser tratados como uma classe específica de identificação na qual essa ocorre de maneira automática. Portanto, sugere-se atenção ao estudo dos elementos chave à identificação de sistemas: seleção e parametrização da estrutura do modelo, seleção do sinal de entrada, estimação de parâmetros e validação (ÅSTRÖM; WITTENMARK, 2013).

Para Guidi et al. (2009), o estudo de algoritmos de identificação podem ser motivados por prever o comportamento do processo e estratégias de controle, possibilitar simulações numéricas que evitem esforços experimentais ou auxiliar no diagnóstico de falhas. A identificação de sistemas descrita por Landau et al. (2011) é composta por quatro etapas, conforme a Figura 6, e tem seu método determinado pelas características do processo, sendo paramétrico ou não, e projetado em tempo contínuo ou discreto.

Figura 6 – Procedimento de identificação em malha aberta para design do controlador



Fonte: Adaptado de Landau et al. (2011, p. 154)

Já o uso de estimadores online, ou apenas *estimadores*, é motivado pelo seu formato sequencial com menor tempo de processamento, tornando viável serem computados em tempo de execução do sistema e, portanto, podendo auxiliar em tomadas de decisão como (LJUNG, 1999):

- Qual entrada deveria ser aplicada no próximo intervalo de tempo;

- Como deveria ser sintonizado os parâmetros de um filtro;
- Qual o melhor preditor para os próximos instantes.

Estimadores online com essas características também são chamados na literatura de recursive identification methods, real-time identification, adaptive parameter estimation ou sequential parameter estimation (LJUNG, 1999). O estudo de estimadores online para sistemas adaptativos é relevante no desenvolvimento de filtros, preditores e controladores, podendo ser aplicados em contextos como (GOODWIN; SIN, 2014):

- Recepção e discriminação de sinais de rádio;
- Transmissão digital de dados telefônicos;
- Detecção de sinais radares;
- Processamento de imagens;
- Prever demandas de produto;
- Prever crescimento populacional;
- Prever consumo de matéria prima;
- Prever saídas futuras de plantas industriais.

Para esses e outros fins, vários estimadores são propostos na literatura, como a integração de redes neurais artificiais que ganhou destaque nos últimos anos, gerando a classe dos controles neuro adaptativos ou, também chamados, controladores inteligentes. Apesar das suas comprovadas habilidades de aprendizado para um grande número de sistemas não lineares, estes estimadores sofrem com possíveis problemas de estabilidade e número de iterações para convergência (ESFANDIARI; ABDOLLAHI; TALEBI, 2022; IOANNOU; FIDAN, 2006; NGUYEN, 2018).

Entretanto, existem também os estimadores Parametric Adaptive Algorithm (PAA) que são recorrentes ao contexto de adaptação de controladores. Entre os PAA estão heurísticas, métodos do gradiente, métodos por análise de estabilidade, aproximações do Filtro de Kalman e minimizações por Least Squares. Em geral, tais algoritmos usam estruturas algorítmicas integrativas discretas (Figura 7) baseadas em dados coletados em cada período de amostragem para gerar os parâmetros do modelo preditor (LANDAU et al., 2011).

Figura 7 – Estrutura integrativa comum aos PAA

$$\begin{aligned} \begin{bmatrix} \text{New estimated} \\ \text{parameters} \\ \text{(vector)} \end{bmatrix} &= \begin{bmatrix} \text{Previous estimated} \\ \text{parameters} \\ \text{(vector)} \end{bmatrix} + \begin{bmatrix} \text{Adaptation} \\ \text{gain} \\ \text{(matrix)} \end{bmatrix} \\ &\quad \times \begin{bmatrix} \text{Measurement} \\ \text{function} \\ \text{(vector)} \end{bmatrix} \times \begin{bmatrix} \text{Prediction error} \\ \text{function} \\ \text{(scalar)} \end{bmatrix} \end{aligned}$$

Fonte: Landau et al. (2011, p. 57)

A estrutura mostrada na Figura 7 é comum para diferentes estimadores recursivos e pode ser utilizada para analisar as condições de estabilidade dessa classe de algoritmos, sendo o elemento *Measurement function* usualmente chamado de *Observation Vector* e o *Prediction error function* chamado de *Adaptation Error* (LANDAU et al., 2011).

De maneira semelhante, Goodwin e Sin (2014) apresenta uma forma de equacionar estimadores online (Equação 35) análoga ao formato indicado na Figura 7

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \mathbf{M}(k-1)\phi(k-d)\bar{e}(k) \quad (35)$$

onde $\hat{\theta}(k-1)$ é a estimativa anterior, $\mathbf{M}(k-1)$ representa os ganhos do algoritmo de estimação e $\bar{e}(k)$ é o erro originado pelo uso do modelo e $\hat{\theta}(k-1)$ inexatos. O equacionamento dos diferentes algoritmos estimadores são derivados a partir da função custo que minimizam, fazendo com que variem entre si nas definições de $\hat{\theta}(k)$, $\mathbf{M}(k-1)$, $\phi(k)$ e $\bar{e}(k)$ que empregam ao problema (GOODWIN; SIN, 2014).

2.3.3.1 Algoritmo Recursive Least Squares

O princípio de Least Squares é empregado para identificação em diferentes problemas matemáticos e sua criação é atribuída a Karl Friedrich Gauss (1777-1855) para determinação de órbitas dos corpos celestes (HOSTETTER, 1987; LANDAU et al., 2011). Atualmente esse princípio é aplicado, por exemplo, aos contextos de sistemas de comunicação, sistemas de navegação, processamento de sinais, processamento de imagem e controle adaptativo (HOSTETTER, 1987); em sistemas lineares ou não lineares (IOANNOU; SUN, 2012).

O principal conceito do Least Squares é a busca pelos valores do modelo matemático que minimizem a diferença entre os dados preditos e a sequência de informações observadas. Portanto, a formulação do Least Squares consiste em definir os valores de $\hat{\theta}$ baseado nos valores experimentais do vetor regressor ϕ tal que as saídas do modelo $\hat{\psi}$ sejam de erro quadrado mínimo em relação as saídas do processo real, conforme a função custo $V(\theta, t)$ dada pelas Equações 36-37

$$V(\hat{\theta}, t) = \sum_{i=1}^t \varepsilon^2(i+1) \quad (36)$$

$$V(\hat{\theta}, t) = \sum_{i=1}^t [\psi(i) - \hat{\theta}(t)^T \cdot \phi(i-1)]^2 \quad (37)$$

onde $\varepsilon(t)$ é o erro dado pela Equação 38 e $\hat{\psi}$ é a saída estimada pelo modelo conforme a Equação 39.

$$\varepsilon(t+1) = \psi(t+1) - \hat{\psi}(t+1) \quad (38)$$

$$\hat{\psi}(t+1) = \hat{\theta}(t+1)^T \cdot \phi(t) \quad (39)$$

Logo, o método é aplicável aos modelos paramétricos que seguem a estrutura linear da Equação 5 (ÅSTRÖM; WITTENMARK, 2013; LANDAU et al., 2011).

A solução analítica garante que a função custo $V(\hat{\theta}, t)$ do Least Squares possua mínimo em relação a $\hat{\theta}$ (Equação 37), porém, a forma sequencial de computação é necessária para aplicações em tempo real de controle adaptativo, tal que a computação da estimativa anterior seja reaproveitada no momento de cálculo da nova estimativa (ÅSTRÖM; WITTENMARK, 2013). Propõe-se obter a formulação recursiva para obter $\hat{\theta}$ tal que no instante t haja minimização via anulação da Equação 37 no horizonte de t medidas sequenciais (Equação 40).

$$\begin{aligned} \frac{\partial V(\hat{\theta}, t)}{\partial \hat{\theta}} &= -2 \sum_{i=1}^t [\psi(i) - \hat{\theta}(t)^T \cdot \phi(i-1)] \phi(i-1) \\ &= 0 \end{aligned} \quad (40)$$

Portanto, dada a equivalência indicada pela Equação 41

$$[\hat{\theta}(t)^T \cdot \phi(i-1)] \phi(i-1) = \phi(i-1) \cdot \phi(i-1)^T \hat{\theta}(t) \quad (41)$$

desenvolve-se a Equação 40 a fim de equacionar $\hat{\theta}(t)$ em função de $\psi(i)$ e $\phi(i-1)$ (Equações 42-46).

$$\sum_{i=1}^t [\psi(i) - \hat{\theta}(t)^T \cdot \phi(i-1)] \phi(i-1) = 0 \quad (42)$$

$$\sum_{i=1}^t [\psi(i) \cdot \phi(i-1) - \hat{\theta}(t)^T \cdot \phi(i-1) \cdot \phi(i-1)] = 0 \quad (43)$$

$$\sum_{i=1}^t [\psi(i) \cdot \phi(i-1) - \phi(i-1) \cdot \phi(i-1)^T \cdot \hat{\theta}(t)] = 0 \quad (44)$$

$$\sum_{i=1}^t [\psi(i) \cdot \phi(i-1)] = \sum_{i=1}^t [\phi(i-1) \cdot \phi(i-1)^T] \cdot \hat{\theta}(t) \quad (45)$$

$$\hat{\theta}(t) = \left\{ \sum_{i=1}^t [\phi(i-1) \cdot \phi(i-1)^T] \right\}^{-1} \sum_{i=1}^t [\psi(i) \cdot \phi(i-1)] \quad (46)$$

Assim, a formulação de $\hat{\theta}(t)$ pode ser expressa pela Equação 47

$$\hat{\theta}(t) = F(t) \sum_{i=1}^t [\psi(i) \cdot \phi(i-1)] \quad (47)$$

onde $F(t)^{-1}$ é indicado pela Equação 48. Segundo Hunt (1986), $F(t)$ é proporcional à matriz covariância das estimações se o modelo do processo é verdadeiramente um regressor linear à da matriz $F(t + 1)$ e por esse motivo $F(t)$ é comumente chamada de matriz covariância.

$$F(t)^{-1} = \sum_{i=1}^t [\phi(i-1) \cdot \phi(i-1)^T] \quad (48)$$

Nesse desenvolvimento supõe-se que a matriz presente na Equação 40 indicada pela Equação 49 é inversível devido a fatores de excitação do sinal que compõe o vetor regressor (LANDAU et al., 2011), conforme a Seção 2.3.3.2.

$$\sum_{i=1}^t \phi(i-1)\phi(i-1) \quad (49)$$

Porém, para tornar recursiva a formulação da Equação 47, o instante $t + 1$ é avaliado na Equação 47, conforme as Equações 50 e 52.

$$\hat{\theta}(t+1) = F(t+1) \sum_{i=1}^{t+1} [\psi(i) \cdot \phi(i-1)] \quad (50)$$

$$\hat{\theta}(t+1) = F(t+1) \left\{ \sum_{i=1}^t [\psi(i) \cdot \phi(i-1)] + \psi(t+1) \cdot \phi(t) \right\} \quad (51)$$

$$\hat{\theta}(t+1) = F(t+1) [F(t)^{-1} \cdot \hat{\theta}(t) + \psi(t+1) \cdot \phi(t)] \quad (52)$$

Na mesma perspectiva, avalia-se o instante $t + 1$ na Equação 48 conforme a Equação 53

$$\begin{aligned} F(t+1)^{-1} &= \sum_{i=1}^{t+1} [\phi(i-1) \cdot \phi(i-1)^T] \\ &= F(t)^{-1} + \phi(t) \cdot \phi(t)^T \end{aligned} \quad (53)$$

portanto, analogamente, $F(t)^{-1}$ é dado por:

$$F(t)^{-1} = F(t+1)^{-1} - \phi(t) \cdot \phi(t)^T \quad (54)$$

Assim, a descrição de $F(t)^{-1}$ (Equação 54) pode ser substituída na Equação 52, conforme desenvolvido nas Equações 55 e 56.

$$\hat{\theta}(t+1) = F(t+1) \{ [F(t+1)^{-1} - \phi(t) \cdot \phi(t)^T] \cdot \hat{\theta}(t) + \psi(t+1) \cdot \phi(t) \} \quad (55)$$

$$\hat{\theta}(t+1) = F(t+1) \{ F(t+1)^{-1} \cdot \hat{\theta}(t) + \phi(t) [\psi(t+1) - \hat{\theta}(t)^T \cdot \phi(t)] \} \quad (56)$$

Pode-se definir o erro de predição a priori $\varepsilon_0(t+1)$ (Equação 57) como a avaliação da diferença entre a saída medida ψ no instante $t+1$ e a saída predita $\hat{\psi}_0$ do instante $t+1$ baseada na estimaco $\hat{\theta}$ do instante t (Equaco 58).

$$\hat{\varepsilon}_0(t+1) = \psi(t+1) - \hat{\psi}_0(t+1) \quad (57)$$

$$\hat{\psi}_0(t+1) = \hat{\theta}(t)^T \cdot \phi(t) \quad (58)$$

Escreve-se a estimaco de $\hat{\theta}(t+1)$, conforme a Equaco 59, substituindo as Equaces 58-57 na Equaco 56 e reorganizando seus termos.

$$\hat{\theta}(t+1) = \hat{\theta}(t) + [F(t+1) \cdot \phi(t)] \hat{\varepsilon}_0(t+1) \quad (59)$$

Nota-se que a estrutura é anloga ao formato PAA apresentado na Figura 7 e, alm disso, o ganho de adaptao é variante no tempo pois $F(t+1)$ depende das medidas de sada e, baseado no Matrix Inversion Lemma, formula-se $F(t+1)$ conforme a Equaco 60.

$$F(t+1) = F(t) - \frac{F(t) \cdot \phi(t) \cdot \phi(t)^T \cdot F(t)}{1 + \phi(t)^T \cdot F(t) \cdot \phi(t)} \quad (60)$$

Finalmente, os termos recursivos $\hat{\theta}(t+1)$ (Equaco 59), $F(t+1)$ (Equaco 60) e $\varepsilon_0(t+1)$ (Equaco 57) so organizados para formular o algoritmo RLS.

O algoritmo RLS pode ser equivalente ao Least Squares no recursivo se for iniciado pela estimaco do instante $t = \dim(\phi(t))$ para assegurar a singularidade de $F(t)^{-1}$, porm, geralmente o RLS é iniciado em $t = 0$. A anlise de estabilidade feita por Landau et al. (2011) mostra que os efeitos do erro de estimaco inicial decaem ao longo do tempo para qualquer $F(0)$ definida positiva.

Analisa-se o comportamento temporal de $F(t+1)$ a partir do caso em que $\phi(t)$ é unidimensional e, conseqentemente, $F(t+1)$ so escalares (Equaco 61).

$$F(t+1) = \frac{F(t)}{1 + \phi(t)^2 F(t)} \leq F(t) \quad (61)$$

Dessa forma, nota-se que $F(t+1)$ decai, pois é sempre menor que seu antecessor $F(t)$ para todo $\phi(t)$, $F(t) \in \mathbb{R}^1$. Seguindo o mesmo raciocnio, se a matriz $\phi(t) \cdot \phi(t)^T$ for positiva definida em mdia $F(t)$ decai para zero ao longo do tempo. Assim, o peso aplicado pelo algoritmo RLS ao erro de predico é cada vez menor ao longo das iteraes, ficando claro que a abordagem apresentada no é totalmente satisfatria para sistemas variantes no tempo (LANDAU et al., 2011).

Na perspectiva desses casos, as formulaes modificadas do algoritmo RLS so apresentadas nas obras sobre controle adaptativo e identificao de sistemas como

Åström e Wittenmark (2013), Landau et al. (2011), Ioannou e Sun (2012), Alexander (2012). Divide-se as principais formas de variação temporal dos parâmetros em dois casos: aqueles que a variação é abrupta mas pouco frequente ou aqueles que variam vagarosa e continuamente (ÅSTRÖM; WITTENMARK, 2013).

Quando a variação é abrupta, recomenda-se abordagens de Resetting para que a matriz covariância $F(t+1)$ seja retornada aos valores de magnitude arbitrariamente maior ou inicial, fazendo com que o ganho de adaptação do algoritmo seja aumentado novamente e permitindo que a estimação sofra uma atualização maior. A ação de Resetting deve ocorrer periodicamente ou no momento em que algoritmos analisadores detectam mudanças dos parâmetros ou dos resíduos de predição. (ÅSTRÖM; WITTENMARK, 2013).

Por outro lado, para abordar os casos de variação vagarosa Åström e Wittenmark (2013) propõem alterar a função custo minimizada para que pondere informações antigas e novas diferentemente de acordo com o fator de esquecimento λ (Equação 62)

$$V(\hat{\theta}, t) = \sum_{i=1}^t [\lambda^{t-i} \varepsilon^2(i+1)] \quad (62)$$

sendo λ o fator de esquecimento relacionado ao horizonte de memória N_λ - em número de iterações - aproximadamente dado pela Equação 63 (LJUNG, 2002).

$$N_\lambda = \frac{1}{1-\lambda} \quad (63)$$

O fator λ pode ser proposto conforme Equação 64, relacionando uma constante de tempo T_{ff} e o período de amostragem do algoritmo estimador T .

$$\lambda = e^{-\frac{T}{T_{ff}}} \quad (64)$$

Dessa maneira, λ atrela-se diretamente ao conhecimento prévio sobre a dinâmica do processo (ÅSTRÖM; WITTENMARK, 2013). O fator de esquecimento também pode ser ajustado na perspectiva de trade-off entre a capacidade de adaptação e a sensibilidade ao ruído conforme diminui-se λ (HUNT, 1986).

De forma análoga, pode-se introduzir fatores de esquecimento diretamente na definição de $F(t+1)^{-1}$ previamente estabelecida (Equação 54), conforme a Equação 65

$$F(t+1)^{-1} = \lambda_1(t)F(t)^{-1} + \lambda_2(t)\phi(t) \cdot \phi(t)^T \quad (65)$$

onde $0 < \lambda_1(t) \leq 1$ e $0 \leq \lambda_2(t) < 2$, tendendo a aumentar e diminuir, respectivamente, o ganho de adaptação. Assim, escreve-se a forma recursiva de $F(t+1)$ na Equação 66

$$F(t+1) = \frac{1}{\lambda_1(t)} \left\{ F(t) - \frac{F(t) \cdot \phi(t) \cdot \phi(t)^T \cdot F(t)}{\frac{\lambda_1(t)}{\lambda_2(t)} + \phi(t)^T \cdot F(t) \cdot \phi(t)} \right\} \quad (66)$$

onde optar por $\lambda_2(t) = 1$ refere-se ao desconto constante tal que $\lambda_1(t) = \lambda$, ao mesmo passo que optar por $\lambda = 1$ é equivalente ao formato original do RLS sem uso de fatores de esquecimento (LANDAU et al., 2011).

Para Sripada e Fisher (1987), aplicar fatores de esquecimento pode evitar que a escolha de estimativas iniciais ruins afetem penosamente os valores finais da estimação, pois evitam que a atualização da estimação seja desativada antes de convergir. Porém, a ação ininterrupta de esquecimento faz com que sejam ponderados dados antigos do processo enquanto, por sua vez, $\psi(t)$ não retém novas informações relevantes sobre o processo, isto é $F(t+1)\phi(t) = 0$, causando uma desvantagem para esta abordagem em processos que não são excitados adequadamente durante todo o período de operação (ÅSTRÖM; WITTENMARK, 2013).

Segundo analisado por Åström e Wittenmark (2013), esse fenômeno é chamado Estimator Windup e ocorre quando adota-se fatores de esquecimento na estimação de processos pouco excitados, sendo provocado pelo emprego de $\phi(t)$ aproximadamente constante ou nulo. Nesse fenômeno, a estimativa $\hat{\theta}(t)$ tende a permanecer contante, já $F(t+1)$ e o ganho de adaptação tendem a crescer exponencialmente aumentando a sensibilidade do algoritmo e, portanto, provocando mudanças bruscas de $\hat{\theta}(t)$ no momento em que $\phi(t)$ variar novamente (ÅSTRÖM; WITTENMARK, 2013).

Pode-se evitar o Estimator Windup empregando métodos de atualização especiais, como a atualização condicional tal que sejam esquecidos dados apenas durante intervalos do processo excitados apropriadamente. A detecção de excitação pode ser feita pelo monitoramento da covariância, ou seja, testando a magnitude de sinais como $\varepsilon(t)$, $\phi(t)^T \cdot F(t) \cdot \phi(t)$ e variação das entradas e saídas do processo.

Nesse sentido, propõe-se testes condicionais necessários para atualização da estimativa, como o apresentado por Åström e Wittenmark (2013) na Equação 67.

$$\phi(t)^T \cdot F(t) \cdot \phi(t) > 2(1 - \lambda) \quad (67)$$

As definições de testes condicionais visam impedir a ocorrência de Estimator Windup, entretanto, deve-se considerar que restrições rígidas demais resultam em atualizações demasiadamente infrequentes, prejudicando a aprendizagem do algoritmo.

Outra proposta para evitar o Estimator Windup é introduzida por Fortescue, Kershenbaum e Ydstie (1981), incorporando a estimativa de variância do ruído e o horizonte de memória no cálculo do fator de esquecimento. Porém, essa abordagem depende de relativa exatidão da estimativa de ruído para que garanta-se a eliminação dos problemas de Estimator Windup (MALIK; HOPE; CHENG, 1991).

A matriz covariância representa um indicador de confiabilidade do algoritmo em relação as medições da iteração anterior comparadas às atuais. Tendo isso em vista, propõe-se uma abordagem distinta para evitar o Estimator Windup: calcular o fator de esquecimento tal que a matriz covariância seja mantida constantemente próxima de seu valor inicial. Portanto, empregar um mesmo valor de matriz covariância ao longo

do tempo conduz o algoritmo a perceber novas informações com mesmo nível de confiabilidade que as informações passadas (MALIK; HOPE; CHENG, 1991).

Portanto, para manter $F(t + 1)$ com magnitude controlada, propõe-se usar o valor de *trace* da matriz covariância para atualizar λ conforme a Equação 68

$$\lambda(t + 1) = \frac{\text{tr}\{F(t + 1)\}}{\text{tr}\{F(0)\}} \quad (68)$$

sendo o valor de *trace* a soma dos elementos diagonais, garantindo que em menor velocidade o algoritmo não seja desligado e continue sua aprendizagem mesmo com poucas informações prévias das perturbações do processo (MALIK; HOPE; CHENG, 1991).

Åström e Wittenmark (2013) apresentam os problemas de implementação do RLS e a influência dos seus parâmetros de design na dinâmica de convergência da estimação do processo. Portanto, para ser incorporado na estrutura do controlador adaptativo é necessário dedicar atenção aos problemas de design do estimador, como selecionar o algoritmo, fator de esquecimento, matriz de covariância inicial, estimação inicial e frequência de amostragem do algoritmo (ÅSTRÖM; WITTENMARK, 2013).

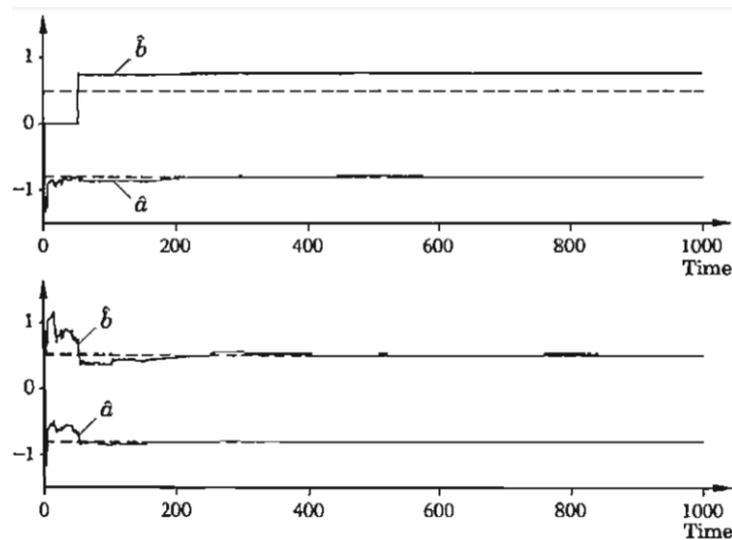
2.3.3.2 Convergência do algoritmo estimador

Segundo Ioannou e Sun (2012), a identificação é tratada como a garantia de convergência da estimação resultante aos valores desconhecidos verdadeiros. A garantia de convergência em processos de identificação é dada por meio da seleção da entrada aplicada a planta, tal que as medidas do sistema que compõem ϕ sejam Permanentemente Excitadas (PE).

De acordo com Ioannou e Fidan (2006), a adaptação consiste em aprendizagem e, portanto, depende da riqueza de informações usada. Para Åström e Wittenmark (2013), a noção de permanência da excitação no sinal é uma forma de caracterizar-lo tendo em vista que a entrada do sistema influencia diretamente na estimação dos seus parâmetros. Wittenmark e Åström (1984) relacionam as condições de excitação necessárias proporcionalmente ao nível de complexidade do modelo estimado, ou seja, quanto maior a ordem do modelo, maior a exigência imposta ao sinal de entrada.

Åström e Wittenmark (2013) ilustram alguns dos comportamentos esperados nos casos de baixa excitação do processo, como a convergência de apenas parte dos parâmetros, ou seja, alguns podem atingir o valor e outros não, conforme exemplificado na Figura 8.

Figura 8 – Exemplo de convergência da estimação conforme sinal de entrada



Fonte: Åström e Wittenmark (2013, p. 72)

Na Figura 8, as linhas sólidas indicam a estimação dos parâmetros \hat{a} e \hat{b} ao longo do tempo e as linhas tracejadas os valores reais. O gráfico na parte superior da Figura 8 mostra uma estimação usando um pulso unitário em $t = 50$ e na parte inferior o gráfico do mesmo processo de estimação mas submetido a uma entrada quadrada unitária periódica a cada 100 amostras.

Portanto, a Figura 8 sugere que a mudança no sinal de entrada pode levar a melhores resultados e um sinal que estimule frequentemente o comportamento transitório do processo auxilia na convergência do processo de estimação. Nessa perspectiva, Landau et al. (2011) propõe as condições para a convergência da estimação de um processo modelado linearmente por uma função de transferência como sendo:

- Usar um algoritmo de adaptação paramétrica (PAA) que garanta a anulação do resíduo de estimação (Equação 69);

$$\lim_{t \rightarrow \infty} \varepsilon(t+1) = \lim_{t \rightarrow \infty} [y(t+1) - \hat{y}(t+1)] = 0 \quad (69)$$

- A ordem dos polinômios numerador e denominador do modelo da planta são conhecidos exatamente;
- O modelo do processo que será identificado é uma função de transferência irredutível em z^{-1} , ou seja, com maior divisor comum entre numerador e denominador igual a um.
- A entrada é um sinal PE de ordem n igual a soma das ordens dos polinômios numerador e denominador do modelo da planta.

No contexto da análise da excitação do sinal para sistemas adaptativos, define-se matematicamente esse conceito como: o sinal $x(t)$ é dito PE de ordem n se para

todo t existir m_{PE} tal que a Equação 70 é satisfeita

$$\varrho_1 I > \sum_{k=t}^{t+m_{PE}} \phi(\mathbf{k}) \phi^T(\mathbf{k}) > \varrho_2 I \quad (70)$$

onde m_{PE} é um inteiro, $x(t)$ o sinal de entrada, ρ_1 e ρ_2 são maiores que zero e $\phi(t)$ é considerado conforme a Equação 70 (ÅSTRÖM; WITTENMARK, 2013).

$$\phi(\mathbf{k}) = [x(k-1), x(k-2), \dots, x(k-n)] \quad (71)$$

Conforme Landau et al. (2011), analogamente pode-se considerar que $x(t)$ é permanentemente excitado de ordem n se o limite abaixo existe e se a matriz C_n é positiva definida (Equação 72).

$$C_n = \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{k=1}^t \phi(\mathbf{k}) \phi^T(\mathbf{k}) > 0 \quad (72)$$

Compreende-se com a análise da Equação 72 apresentada por Landau et al. (2011) que um sinal composto pela soma de p senoides com frequências distintas entre si (Equação 73) é um sinal suficientemente excitado de ordem n e eficiente para estimar n parâmetros

$$x(t) = \sum_{i=1}^p \sin \omega_i \cdot T \cdot t \quad (73)$$

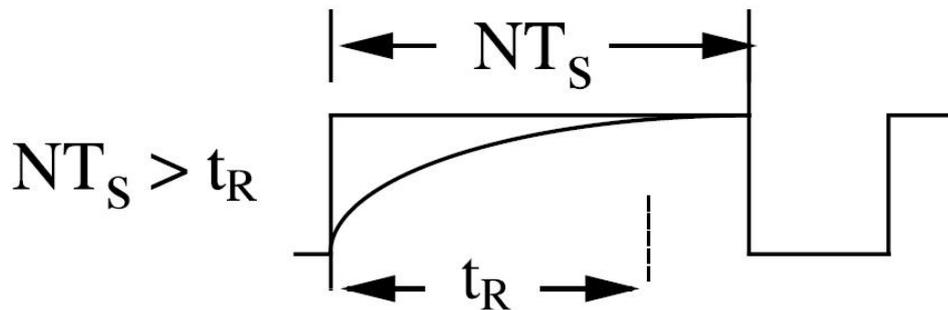
onde ω_i são as frequências distintas que compõe o sinal, T_s é o período de amostragem e p é definido pela Equação 74.

$$p = \begin{cases} \frac{n}{2} \forall n \text{ par} \\ \frac{n+1}{2} \forall n \text{ ímpar} \end{cases} \quad (74)$$

Na mesma perspectiva, Landau et al. (2011) apresentam o sinal de entrada Pseudo Random Binary Sequence (PRBS) para proporcionar eficiência na identificação de parâmetros. Esse sinal oferece uma larga faixa no espectro de frequências, ou seja, é composto por um grande número de senoides e possui magnitude constante. A magnitude constante é vantajosa para conhecer precisamente o comportamento e o stress da planta durante o processo de identificação.

O PRBS é uma sequência de pulsos retangulares modulados por largura definida pseudoaleatoriamente via software ou hardware. A duração de pelo menos um pulso da sequência deve ser maior que o tempo de acomodação da planta para identificar corretamente o estado estacionário da planta, conforme exemplificado na Figura 9.

Figura 9 – Escolha de período máximo de um pulso no sinal PRBS



Fonte: Landau et al. (2011, p. 180)

O experimento de identificação apresenta maior dificuldade de aprendizagem quando embutido em sistemas de controle, pois o sinal de entrada não é escolhido previamente e sim gerado pela realimentação da malha fechada. Portanto, a entrada do estimador baseada na realimentação do sistema faz com que os parâmetros verdadeiros não sejam identificados (ÅSTRÖM; WITTENMARK, 2013).

A possível correlação entre o ruído em malha fechada resulta em falhas na identificação e, além disso, a redução da perturbação causada pelo controlador diminui também a excitação do sinal (GUIDI et al., 2009). A dificuldade expressa-se em fenômenos como o surgimento de dependências lineares entre as colunas do termo $\phi^T \phi$ presente no equacionamento do estimador, levando à não unicidade das soluções (ÅSTRÖM; WITTENMARK, 2013). Logo, a não convergência aos valores reais é justificada pela falta de garantia que o sinal de realimentação seja PE (ÅSTRÖM; WITTENMARK, 2013).

Segundo Goodwin e Sin (2014), outro fator relevante para o desempenho de algoritmos estimadores é a utilização de conhecimento prévio sobre o processo. Explorar e empregar o conhecimento de características do processo tipicamente melhora o resultado do estimador, pois gera restrições estruturais, valores e intervalos permitidos aos parâmetros.

Os valores iniciais da estimativa dos parâmetros são arbitrários ao processo de estimação, porém se mal selecionados podem provocar maior demora de aprendizagem do algoritmo (GOODWIN; SIN, 2014). Assim, empregar o conhecimento dos valores iniciais é vantajoso e pode ser obtido por modelagens físicas, experimento de identificação prévio ou usando o algoritmo de estimação inicialmente atrelado a uma lei de controle fixa (GOODWIN; SIN, 2014). Tendo em vista as consequências no desempenho de controle, Wittenmark e Åström (1984) indicam três alternativas para escolha da estimação inicial: valores nulos, relacionados as características conhecidas do processo ou tal que resultem em um controlador de baixo ganho ou ação integral.

A literatura apresenta diferentes critérios para a frequência de amostragem, como Goodwin e Sin (2014) sugerindo que o período de amostragem deve ser escolhido em relação à dinâmica do processo, sendo um múltiplo par do atraso temporal e próximo a um quinto da constante de tempo mais rápida. Já em vista dos cálculos numéricos, Malik, Hope e Cheng (1991) estabelecem que para descrever o comportamento do sistemas com exatidão o período de amostragem não deve ser longo demais, porém, pode ser sensível aos erros de medição se demasiadamente curto.

Wittenmark e Åström (1984) indicam por experiência que são necessárias de 10 à 50 amostras para que a estimação seja razoável em aplicações de controle. Ainda na perspectiva dos controladores, propõe-se uma relação para amostragem em controle PID tal que seja executada 6 a 15 vezes durante o intervalo para acomodação de 95% do valor de estado estacionário (ISERMANN, 1981).

Acerca dos efeitos de perturbações em estimadores PAA, conclui-se que o valor médio quadrado do erro de adaptação a posteriori tende a ser limitado pelo limite superior da perturbação (LANDAU et al., 2011). A estimação tende a piorar quando as condições experimentais favorecem que a matriz covariância aumente e o vetor regressor seja composto majoritariamente por ruído, portanto, a excitação causada por perturbações nas estradas e saídas do sistema pode auxiliar na convergência do algoritmo e auto corrigir o comportamento do sistema (SRIPADA; FISHER, 1987).

Em relação à estabilidade do estimador, justifica-se aplicar métodos de aprimoramento de robustez ao algoritmo nos casos que o modelo real do processo é significativamente diferente do modelo estimado, as perturbações não tem seus limites completamente estabelecidos ou quando é necessário melhorar certa faixa de frequência dos dados de entrada e saída (LANDAU et al., 2011).

2.4 HIPERTENSÃO

2.4.1 Contexto Clínico da Hipertensão

Conforme descrito nas Diretrizes Brasileiras de Hipertensão de 2020 (BARROSO et al., 2021), a hipertensão arterial (HA) é uma doença crônica e não transmissível, que eleva persistentemente a pressão arterial sistólica acima de 140 *mmHg* ou pressão arterial diastólica acima de 90 *mmHg* na ausência de medicação anti-hipertensiva. Dado seu perfil assintomático, a HA costuma evoluir para alterações estruturais ou funcionais do coração, cérebro, rins e vasos; e associando-se continuamente à doenças cardiovasculares, renais crônicas e mortes prematuras.

Lopes (2019) alerta que a hipertensão afeta em torno de doze milhões de pessoas no mundo, sendo a principal causa de mortes prematura em 2015 e, em países como Portugal, pode causar cerca de 32% da totalidade de óbitos. No Brasil foram estimadas cerca de 667.184 mortes relacionáveis a HA entre 2008 e 2017, já no

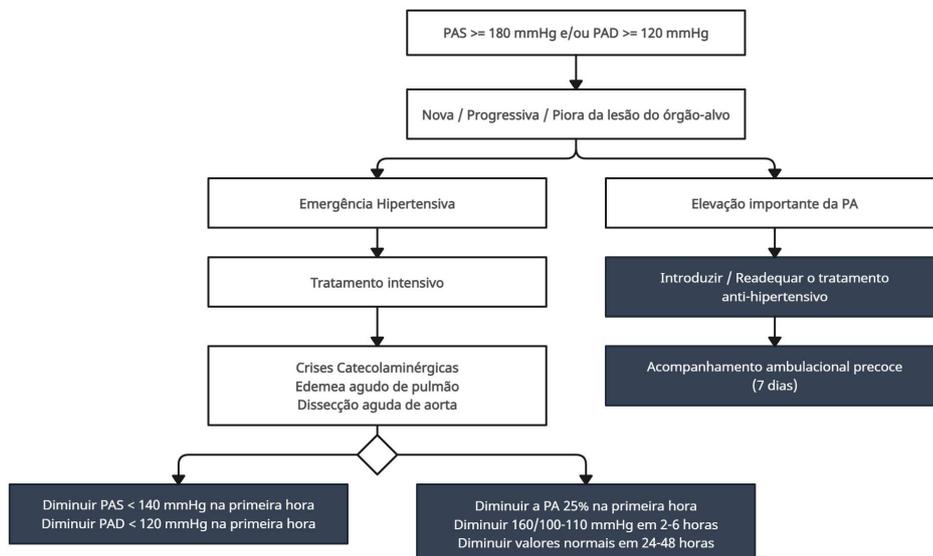
intervalo de 2008 a 2018 observou-se a tendência de mortalidade da HA aumentar em 128% (BARROSO et al., 2021).

Em relação aos gastos, ao longo da década de 2000 dedicaram-se 77% dos investimentos em hospitalizações, equivalente a 1.6 bilhões de reais, no Sistema Único de Saúde (SUS) com doenças cardiovasculares relacionadas à HA e entre 2010 e 2019 esse custo aumentou 32%, passando à monta de 2.2 bilhões no mesmo período. Assim, os investimentos com a HA tem sido maiores do que doenças como obesidade e diabetes melito (BARROSO et al., 2021).

Segundo Basha, Vivekanandan e Parthasarathy (2018), as crises de hipertensão podem surgir de complicações pós cirúrgicas, especialmente em operações cardíacas. Barroso et al. (2021) divide empiricamente crises hipertensivas em urgências e emergências hipertensivas, sendo ambas situações clínicas sintomáticas de elevação acentuada da PA sistólica acima, ou igual, a 180 *mmHg* ou diastólica acima, ou igual, a 120 *mmHg*.

Entretanto, na emergência hipertensiva, ao contrário da urgência, há risco iminente de morte causada pela lesão aguda e progressiva do órgão-alvo. A diferença também é notada uma vez que a urgência conduz a uma sobrevida de até cinco anos maior que nos casos de emergência hipertensiva (BARROSO et al., 2021). A Figura 10 ilustra os procedimentos indicados de decisão e tratamento para crises hipertensivas,

Figura 10 – Fluxograma de atendimento da crise hipertensiva



Fonte: Adaptado de Barroso et al. (2021, p. 602)

devendo ser realizado após um período de observação do paciente em ambiente calmo a fim de afastar casos de pseudocrise hipertensa causados por eventos emocionais dolorosos, enxaqueca, tontura rotatória, cefaleias vasculares, desconfortos

musculoesqueléticos ou síndrome do pânico.

As Diretrizes Brasileiras de Hipertensão sugerem que problemas de hipertensão aumentam com o envelhecimento e estão presentes em cerca de 65% dos indivíduos acima de 60 anos (BARROSO et al., 2021). Entretanto, estudos indicam que parte significativa dos pacientes com crises hipertensas tendem a ser homens, jovens, de baixa escolaridade e nível socioeconômico e com conhecimento de seu diagnóstico mas baixo nível de adesão aos medicamentos (CARDOSO, 2013; FEITOSA-FILHO et al., 2008) .

A ocorrência de crise pode ser manifestação de qualquer doença hipertensiva, sendo que em 90% dos pacientes a hipertensão tem causa desconhecida, entretanto, apesar de apenas 1% dos pacientes hipertensos desenvolverem quadros acelerados ou malignos, situações de urgência hipertensa acarretam na maior probabilidade de novos eventos cardiovasculares (CARDOSO, 2013).

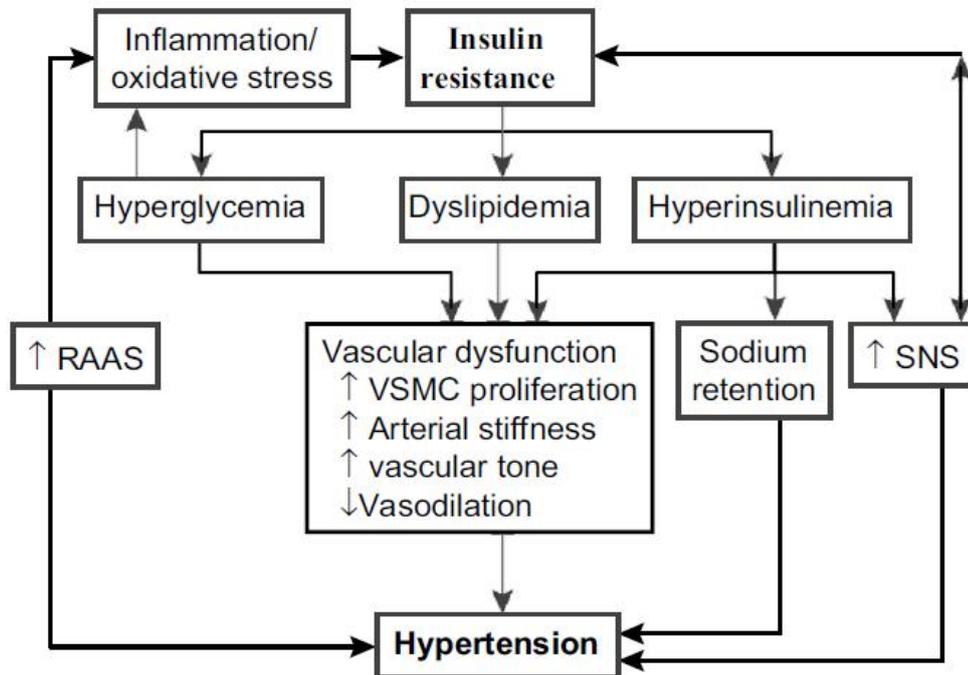
Indivíduos podem ser pré dispostos a desenvolver hipertensão devido a fatores genéticos, anomalias celulares, desarranjos metabólicos e hemodinâmicos (BASHA; VIVEKANANDAN; PARTHASARATHY, 2018). Além disso, considera-se que a hipertensão arterial envolve fatores ambientais e comportamentais, tais como uso de tabaco, obesidade, alcoolismo, sedentarismo e excesso alimentar de sódio (LOPES, 2019).

Nishimura e Saito (1992) alertam que o problema da hipertensão é agravado ao longo do tempo, pois se não tratado, pode acarretar doenças cardiovasculares, cerebrovascular e circulatórias. Na mesma perspectiva, Barroso et al. (2021) indicam que, se não tratadas, a mortalidade de emergências hipertensivas pode chegar a 80% ao decorrer de um ano.

Para Basha, Vivekanandan e Parthasarathy (2018), casos de hipertensão também têm sido recentemente vinculados a ocorrências de outras doenças, como o aumento da resistência à insulina e diabetes. A hipertensão e a diabetes também compartilham entre si os mecanismos nos quais se expressam, como estresse, estresse oxidativo e inflamações.

A Figura 11 resume a interação entre os principais mecanismos comuns da diabetes mellitus e hipertensão

Figura 11 – Diagrama resumido dos possíveis mecanismos no desenvolvimento de hipertensão na diabetes mellitus



Fonte: Basha, Vivekanandan e Parthasarathy (2018, p. 5)

onde RAAS é a sigla para Renin-Angiotensin-Aldosterone System (RAHIMI; MORADI; NASRI, 2014), SNS é a sigla para Sympathetic Nervous System (SIDDQUI et al., 2015) e VSMC é a sigla para Vascular Smooth Muscle Cells (YUAN et al., 2019). Vê-se, portanto, como esses mecanismos interagem e influenciam um ao outro, ocasionando ciclos malignos.

Segundo Nishimura e Saito (1992), a regulação da pressão sanguínea é dada pelas glândulas endócrinas e pelo sistema nervoso autônomo, variando conforme o ciclo circadiano e condições de respiração. O Quadro 3 é apresentado para melhor compreensão dos químicos atuantes na regulagem de pressão.

Quadro 3 – Efeitos da Adrenalina, Noradrenalina, Glicocorticoide, Aldosterona e Angiotensina 2

Químicos corporais	Efeitos
Adrenalina	1) Aumenta frequência cardíaca 2) Dilata vasos periféricos
Noradrenalina	1) Contraí vasos periféricos 2) Diminui a frequência cardíaca por reflexo baroreceptor
Glicocorticoide	Tranforma noradrenalina em adrenalina
Aldosterona	1) Promove a reabsorção do íon de sódio e água, aumentando o retorno venoso e a pressão arterial média
Angiotensina 2	1) Contraí os vasos periféricos 2) Estimula o córtex e medula adrenal, promovendo a secreção de hormônios da medula adrenal e adrenocorticais

Fonte: Adaptado de Nishimura e Saito (1992, p. 2).

Nota-se a complexidade do sistema pelo número de agentes químicos envolvidos na regulagem de pressão e as várias interações cruzadas entre os efeitos.

A combinação das ações de monitoramento e ajuste da taxa de infusão do medicamento torna possível rejeitar perturbações na PA e lidar com mudanças nas condições do paciente, mantendo o nível de pressão próximo ao desejado. Por sua vez, a ministração do medicamento consiste em seguir adequadamente as normas de dosagem, elegibilidade de pacientes, tempo de ministração e seleção do medicamento (SHARMA et al., 2021).

De acordo com Rodrigues (2002), o consenso internacional indique que emergências hipertensas requerem monitoramento frequente da PA e drogas injetáveis por bombas de infusão em um acesso venoso. Para Basha, Vivekanandan e Parthasarathy (2018), em geral, o tratamento emergencial para regulagem da pressão arterial é baseado na infusão de nitroprussiato de sódio, pois tem a capacidade de agir em até 2 minutos.

O nitroprussiato de sódio é uma droga com capacidade de vasodilatar diretamente áreas da musculatura lisa arteriolar e venular, sendo usada principalmente em crises relacionadas à insuficiência cardíaca congestiva. Esse medicamento diminui o retorno venoso, ao mesmo tempo que diminui a resistência periférica enfrentada pelo sangue no momento que é expelido do coração (RODRIGUES, 2002).

Rodrigues (2002) sinaliza que uma das possíveis adversidades do nitroprussiato de sódio é o aumento da pressão intracraniana, entretanto a queda da PA que promove tem bônus maior que o risco de hiperfluxo cerebral e, por isso, considera-se um efeito não proibitivo. Outras decorrências comuns da ministração de nitroprussiato

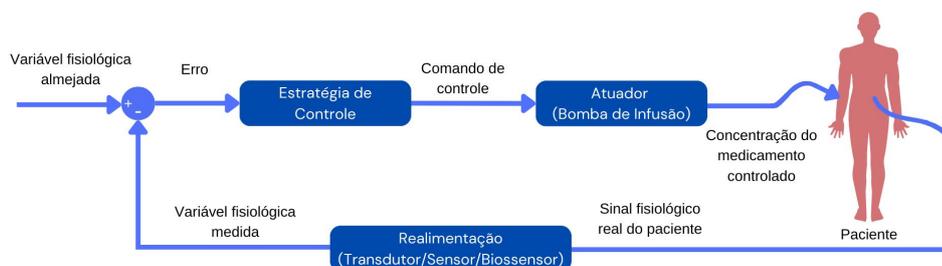
de sódio são náuseas, vômitos, taquicardia, retenção hídrica e fasciculações. Além disso, é alarmantemente o risco letal de intoxicação por tiocianato, principalmente mediante uso prolongado de altas doses.

2.4.2 Controle da pressão arterial

Para Sharma et al. (2021), situações emergenciais acarretam em condições não ideais de trabalho para os médicos, como a dificuldade de dedicar atenção a todos os pacientes, assim, sistemas automatizados se tornaram parte integral no ambiente hospitalar. Segundo Westenskow (1987), o tratamento manual de pacientes hipertensos via nitroprussiato de sódio exige alto nível de atenção pois tem ação rápida e necessita variações na dosagem, assim, controladores automáticos são menos suscetíveis a fadiga e distração dos operadores.

Nessa perspectiva, sistemas de malha fechada são estudados pois, se bem projetados, podem manter variáveis fisiológicas constantes mesmo que sejam submetidas a perturbações ou que certas características variem entre cada paciente (SHARMA et al., 2021). A estrutura lógica de controles para sistemas de dosagem de medicamento é apresentada pela Figura 12 de forma semelhante a Figura 1.

Figura 12 – Diagrama de blocos do controle malha fechada de administração de medicamentos



Fonte: Adaptado de Sharma et al. (2021, p. 4)

Conforme apresentado na Seção 2.1, controlares PID são amplamente usados no contexto industrial, porém, não se limitam ao contexto manufatureiro. Segundo Sharma et al. (2021), devido à variedade de métodos de sintonização, desempenho aceitável e implementação simples, o formato PID é uma escolha comum para aplicações médicas de controle, como:

- ministração de insulina para pacientes diabéticos;
- ministração de medicamento para doenças cardíacas;
- ministração de medicamentos combativos ao câncer;
- ministração de medicamentos anestésicos;
- terapias de problemas neurológicos, como epilepsia, recuperação cognitiva pós lesão e supervisão de derrame.

Para Westenskow (1987), a presença de pacientes hipersensíveis ao medicamento torna o emprego de técnicas adaptativas no controle da pressão arterial especialmente vantajoso em relação aos controladores de ganhos fixos. Basha, Vivekanandan e Parthasarathy (2018) ilustram como os níveis de pressão resultantes da ação de controladores podem divergir conforme são usados em pacientes com diferentes níveis de sensibilidade.

É esperado que em crises hipertensivas a resposta ao medicamento inicie entre um a dois minutos, tal que diminua a PA progressivamente conforme necessário em no máximo 24 horas para casos de urgência hipertensiva (RODRIGUES, 2002). Na perspectiva de engenharia de controle, Silva et al. (2019) resume os objetivos da regulagem segura de PA como:

- Percentual de sobressinal (PSS) ≤ 10 mmHg;
- Tempo de acomodação (TS) ≤ 10 minutos (600 segundos);
- Erro de estado estacionário (ESS) de ± 5 mmHg.

Em vista desses critérios, o Quadro 4 compara o desempenho estimado de diferentes controladores adaptativos propostos na literatura em relação ao tempo de acomodação, percentual de sobressinal e erro de estado estacionário.

Quadro 4 – Resumo do desempenho dos controladores de PA

Artigo	Ano	Controlador	Desempenho do controlador		
			TS(s)	PSS(mmHg)	ESS(mmHg)
Slate et. al	1980	Nonlinear PID	<600		± 10
Arnsparger et al.	1983	STR	120	30	10
Meline et al.	1985	MV+PID	222	<20	± 5
Voss et al.	1987	CAMAC	78-438	0-22	-4 to 9.8
Hernandez et al.	1989	MRAC	90-580	0-16	± 5
Pajunen et al.	1990	MRAC	<300	<15	± 15
Yu et al.	1992	MRAC	180-630	0-12	± 5
Isaka et al.	1992	Fuzzy controller	<180	<10	± 5
Rao et al.	2001	MMPC	720		± 5
Kashihara et al.	2004	ANN	120		± 5
Zhu et al.	2007	Adaptive PI	<300	10-20	± 5
Rhida	2011	MPC	438-1190	<19	± 0.5
Malagutti et al.	2013	RMMAC			± 5 to ± 10
Cheriyachan et al.	2015	Fuzzy logic	282	<5	± 0.5
Sowpamika et al.	2017	PI/PID	380	42.9	
Sowpamika et al.	2017	Adaptive PI/PID	200	0	

Fonte: Adaptado de Silva et al. (2019, p. 158).

onde STR é a sigla para *Self-Tuning Regulator*, MV para *Minimum Variance control*,

CAMAC para *Control Advance Moving Average Controller*, MRAC para *Model Reference Adaptive Control*, MMAC para *Multiple Model Adaptive Control*, MMPC para *Multiple Model Predictive Control*, MPC para *Model Predictive Control*, RMMC para *Robust Multiple Model Control* e ANN para *Artificial Neural Network*.

Avalia-se por meio do Quadro 4 que parte dos controladores propostos não atingem o desempenho esperado nos critérios listados anteriormente, uma vez que: dois desses podem ter TS acima de 600 segundos, nove desses podem ter PSS acima de 10 mmHg e um desses pode ter ESS acima de 10 mmHg. Portanto, apesar da complexidade em comparar diretamente a literatura acadêmica, pode-se concluir que o percentual de sobressinal aparenta ser o critério mais crítico na avaliação realizada no Quadro 4.

2.4.3 O modelo matemático da pressão arterial

Na perspectiva da automação no tratamento terapêutico, Slate et al. (1979) identificam um modelo matemático Single-Input-Single-Output (SISO) com foco em aplicações de controle e, diferente de trabalhos anteriores, propõem considerar ações em segundo plano na pressão arterial e não negligenciar os atrasos nos transportes de fluido. O modelo apresentado por Slate et al. (1979) é válido apenas para pequenas taxas de infusão a fim de mantê-lo em região de operação linear e não considera interações com as outras variáveis ou estados, como a saída cardíaca do paciente.

As ações em segundo plano consideradas no modelo são proporcionadas pela observação da pressão média espontânea e dopada, além disso, essas se mostram variantes tanto no tempo quanto entre pacientes, sendo compostas por componentes determinísticas e estocásticas. Supõe-se que parte da complexidade observada no comportamento da pressão seja causada pelos mecanismos de controle neuro-humoral e transientes de redistribuição na dinâmica cardiovascular (SLATE et al., 1979).

Para Slate et al. (1979), a pressão arterial do paciente oscila em torno de uma média com variação entre dois a dez mmHg em pacientes ditos com baixo ruído e entre dez até 15 em pacientes ditos ruidosos, porém, essa variação da média é aproximadamente estacionária. A característica estacionária é validada para pequenas taxas de infusão de medicamento em pacientes com condições fisiológicas estáveis e para dentro do intervalo de cinco a 15 minutos (SLATE et al., 1979).

Tendo a vista a necessidade de descrever processos variantes no tempo, Ljung e Gunnarsson (1990) apresentam formulações matemáticas para variações de parâmetros. Ljung e Gunnarsson (1990) distinguem quatro formas de equacionar a variação temporal dos valores verdadeiros do processo θ^* : saltos, cadeias de Markov, baseadas em conhecimento do processo, caminhada aleatória e caminha aleatória com tendências globais e locais.

A caminhada aleatória, formulada na Equação 75, é a descrição básica de variação de parâmetros, pois quase não necessita informações prévias.

$$\theta^*(t) = \theta^*(t-1) + \nu\eta(t) \quad (75)$$

onde $\eta(t)$ é um vetor de elementos estocásticos gaussianos e ν uma constante

de escala. Em certos processo é necessário modelar a tendência das variações, então adapta-se a Equação 75 direcionando-a globalmente com um vetor de variação vagarosa $\Upsilon(t)$.

$$\theta^*(t) = \theta^*(t-1) + \nu\eta(t) + \Upsilon(t) \quad (76)$$

Em relação à identificação do processo, o comportamento estático do modelo de pressão foi determinado a partir da resposta do processo a entradas degrau e o comportamento dinâmico pôde ser obtido das respostas a entradas de taxas de infusão binárias pseudoaleatórias. Mudanças significativas no ganho estático e não linearidades do sistema foram constatadas quando aplicadas diferentes taxas de infusão em experimentos com cães (SLATE et al., 1979).

Finalmente, a pressão arterial p_a é modelada como a soma da atividade de background p_{ba} e a variação de pressão causada pelo medicamento Δp_d (Equação 77)

$$p_a = p_{ba} + \Delta p_d. \quad (77)$$

Já a pressão média \bar{p}_a é monitorada por um instrumento cateter-transdutor submetido a um filtro passa-baixa de primeira ordem com constante de tempo de dois segundos (Equação 78).

$$S(s) = \frac{1}{2s + 1} \quad (78)$$

No mesmo contexto, a variação de pressão é dita linearmente relacionada por $G_d(s)$ à taxa de infusão de medicamento $I(s)$ (Equação 79)

$$\begin{aligned} G_d(s) &= \frac{\Delta P_d(s)}{I(s)} \\ &= \frac{K e^{-\tau_i s} (1 + \alpha e^{-\tau_c s})}{1 + \tau s} \end{aligned} \quad (79)$$

onde τ_i é o atraso de tempo para transporte após injeção, τ_c é o tempo de recirculação, K é a sensibilidade para o medicamento, α é a fração de recirculação e a constante de tempo τ caracterizando a absorção, distribuição e metabolização do medicamento. A faixa de valores identificada para os parâmetros da Equação 79 é resumida na Tabela 1.

Tabela 1 – Valores esperados do modelo contínuo

Parâmetro	Mínimo	Máximo	Nominal
K	0.25	9.0	1.0
τ_i	20	60	30
τ_c	30	75	45
α	0	0.4	-
τ	30	60	40

Fonte: Adaptado de Steinmetz (1987).

A atividade de background, dada pela Equação 80, é composta pelos valores de offset p_o , elemento estocástico p_η , ação de reflexo p_δ e senoidal p_s .

$$p_{ba} = p_o - p_\delta + p_s + p_\eta \quad (80)$$

A componente senoidal é em fase com a respiração, podendo ser estimada em 12 impulsos por minuto, ou seja, 1.2566 rad/s e dispõe de baixa amplitude em relação à pressão média \bar{p}_a , variando entre dois e quatro mmHg.

Já o elemento estocástico é modelado como a saída de um filtro de segunda ordem com frequência de corte correspondente ao período de 30 segundos (Equação 81) em resposta ao ruído branco.

$$F_\eta(s) = \frac{K_\eta}{(30s + 1)^2} \quad (81)$$

O ganho K_η do elemento estocástico é tal que varie \bar{p}_a entre cinco e dez mmHg em torno da média.

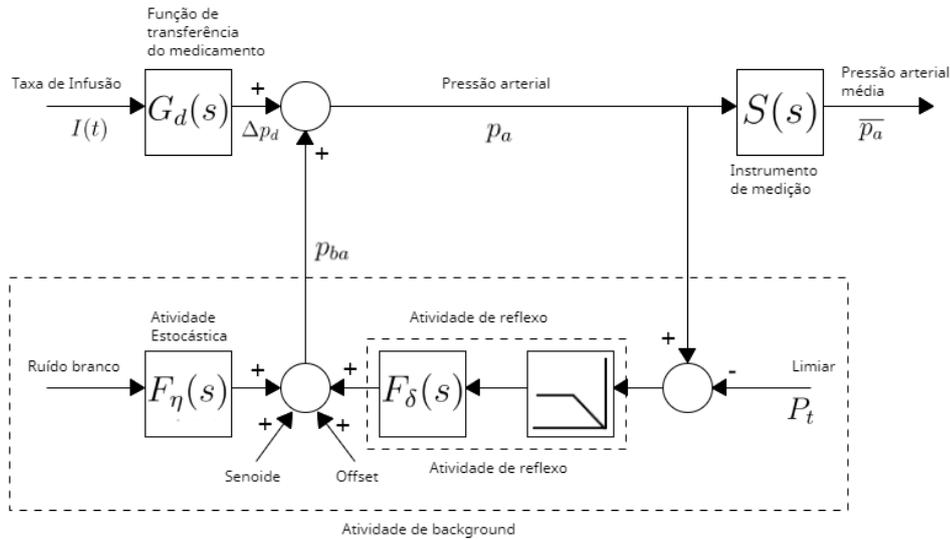
A atividade de reflexo responde ao decaimento de p_a abaixo do limiar P_t , sendo P_t entre 60 e 70 mmHg na maior parte dos casos mas potencialmente maior que 100 e variante no tempo para pacientes pós cirúrgicos. Nesse sentido, a atividade de reflexo é considerada um saturador unidirecional de 35 mmHg seguido por um filtro com constante de tempo de 4 minutos atrasada temporalmente em τ_δ segundos (Equação 82),

$$F_\delta(s) = \frac{e^{-\tau_\delta s}}{240s + 1} \quad (82)$$

sendo τ_δ um atraso estimado entre 30 e 60 segundos para transporte de agentes sanguíneos.

Portanto, a pressão arterial é influenciada não somente por dois atrasos de tempo e parâmetros desconhecidos mas também pela adição de componentes determinísticos e estocásticas variantes no tempo críticos ao desempenho seguro do sistema de controle, conforme a Figura 13.

Figura 13 – Diagrama de blocos do modelo de pressão arterial



Fonte: Adaptado de Slate et al. (1979, p.71)

Em vista da magnitude significativa da ação de background, pode-se considerar que a saída do processo é influenciada por caminhada aleatória dos parâmetros (Equações 75 e 76), reforçando a importância de métodos adaptativos. Para Bailey e Haddad (2005) a dinâmica proposta resultante sugere as dificuldades no controle, como as possíveis oscilações causadas pelo atraso temporal entre a aplicação do medicamento e o efeito clínico.

O modelo discreto da Equação 79 é proposto por Steinmetz (1987) com foco em sistemas autoregressivos de estimação, adotando o mesmo tempo de amostragem de 15 segundos proposto por Slate et al. (1979), porém, partindo de considerações da malha fechada esperada. Pelo intermédio do retentor de ordem zero, propõe-se o modelo discreto expresso pelas Equações 83 a 88

$$G_d(t) = \frac{\Delta P_d(t)}{I(t)} \quad (83)$$

$$= \frac{q^{-d}(b_0 + b_m q^{-m})}{1 - a_1 q^{-1}}$$

onde

$$b_0 = K(1 - e^{-\frac{T}{\tau}}) \quad (84)$$

$$b_m = \alpha K(1 - e^{-\frac{T}{\tau}}) \quad (85)$$

$$a_1 = e^{-\frac{T}{\tau}} \quad (86)$$

$$d = 1 + \frac{\tau_i}{T} \quad (87)$$

$$m = \frac{\tau_c}{T} \quad (88)$$

sendo q^{-1} o operador de atraso unitário e T o tempo de amostragem.

No modelo discreto apresentado os atrasos, τ_i e τ_c , são considerados como múltiplos inteiros do tempo de amostragem, ou seja, desconsiderando suas partes fracionárias, tal que d e m são números naturais (STEINMETZ, 1987). A Tabela 2 exibe a faixa de valores permitidos para cada parâmetro do modelo discreto de acordo com as limitações expostas na Tabela 1.

Tabela 2 – Valores esperados do modelo discreto

Parâmetro	Mínimo	Máximo	Nominal
b_0	0.053	3.546	0.187
b_m	0	1.418	0.075
a_1	0.606	0.779	0.741
d	2	5	3
m	2	5	3

Fonte: Adaptado de Pajunen, Steinmetz e Shankar (1990).

Outras formas de propor modelos discretizados da Equação 79 são possíveis, como expressar os atrasos de tempo via aproximação de Padé, conforme apresentado por Albertos, Salgado e Olivares (2000), dando origem a uma função de transferência discreta de segunda ordem sem atraso explícitos na equação. Discretizar modificações diretas do modelo (Equação 79), como apresentado por Martin, Schneider e Smith (1987), seria também uma alternativa. Jury (1964) investiga métodos estendidos da transformada z que poderiam ser usados para expressar os efeitos dos atrasos fracionários, fazendo com que o modelo contemplasse atrasos não múltiplos inteiros do tempo de amostragem.

Segundo Cardoso (2013), saturadores da taxa de injeção devem ser aplicados, pois doses excessivas causam a diminuição abrupta da pressão arterial e podem proporcionar estado de choque ao paciente. Conforme Westenskow (1987), o uso de controles adaptativos torna-se impraticável clinicamente caso sejam demasiadamente agressivos, indicando o cuidado necessário com o projeto das ações de controle para que se evitem as taxas de infusão indesejadas descritas por Slate et al. (1979).

Para evitar situações de toxicidade e overdose, Zhu, Zheng e Lavanya (2005) expressam as limitações impostas ao sistema amostrado a cada 15 segundos (Equações 89-93). Para a entrada de medicamento, o nível da taxa infusão deve ser:

$$0 \leq I(t) \leq 180\text{mL/h} \quad (89)$$

A mudança no nível da taxa de infusão é aceitável para pacientes com sensibilidade

normal ou baixa, ou seja, se:

$$-40 \leq I(t) - I(t - 1) \leq 7mL/h \quad (90)$$

Já para pacientes de alta sensibilidade, a mudança na taxa de infusão é aceitável se:

$$-40 \leq I(t) - I(t - 1) \leq 3mL/h \quad (91)$$

Em relação à pressão, a mudança máxima permitida de seu valor médio é dada por:

$$|\overline{p_a}(t) - \overline{p_a}(t - 1)| \leq 15mmHg \quad (92)$$

Além disso, a pressão deve ser mantida próxima ao nível desejado. A métrica adotada é não permitir estar abaixo de 20 mmHg abaixo da referência de 100 mmHg (Equação 93).

$$\min(\overline{p_a}(t)) \geq 80mmHg \quad (93)$$

O conjunto de inequações expresso acima indica o uso de saturadores no comando de infusão gerado pelo controlador e a definição de condições necessárias para validar o sistema de controle projetado.

3 MÉTODO

Este capítulo é dedicado a expor a metodologia adotada para ajustar os parâmetros de projeto e indicar os desempenhos dos controladores adaptativos. Para isso, este capítulo divide-se entre a descrição das variantes aderidas do algoritmo, a apresentação de indicadores de desempenho e, finalmente, o controle da pressão arterial proposto.

3.1 DETALHES DE IMPLEMENTAÇÃO

O modelo de pressão arterial e seu controlador DIRAC, incluindo os estimadores propostos, foram implementados em ambiente de simulação (Apêndice A e B) na plataforma Matlab e Simulink (MATLAB, 2020). Essa plataforma foi escolhida devido à proposta de interação entre scripts e programação em blocos para simulações de modelos dinâmicos, além da adesão encontrada na literatura.

3.1.1 Controlador DIRAC

O algoritmo proposto por Keyser (2000) soluciona recursivamente o problema de controle estabelecido baseado na filtragem dos sinais de entrada e saída a planta, tal que otimize-se o sinal de controle diminuindo a diferença entre a planta e o modelo de referência por consequência. Nesse contexto, ressalta-se que este trabalho negligencia questões computacionais e de implementação em hardware como a complexidade computacional das operações, porém, mantém compromisso conceitual com essas questões ao seguir as considerações de projeto vistas anteriormente.

Assim, apesar do método DIRAC ser expansível a outras formas de modelo de referência, este trabalho adota a função de transferência discreta de primeira ordem proposta no trabalho de Keyser (2000), dada a semelhança em relação ao modelo previsto para a PA quando a fração de recirculação α é anulada na Equação 83. Estipula-se encontrar uma constante de tempo de referência τ_r , tal que o modelo atinja a acomodação respeitando os requisitos do problema e a latência usual para aprendizagem.

Logo, define-se o parâmetro a_r do modelo de referência conforme as Equações 94 e 95, tendo em vista sua forma de função de transferência discreta de primeira ordem.

$$\tau_r = 3 \tau \quad (94)$$

$$a_r = e^{-\frac{T_s}{\tau_r}} \quad (95)$$

onde τ é a constante de tempo da planta esperada no modelo. A latência de resposta é desejável levando em consideração o tempo para aprendizagem e em perspectiva das premissas apontadas por Keyser (2000) em relação à necessidade inicial de manter a proximidade entre a resposta da planta e do modelo de referência. Já o atraso temporal do modelo de referência d_r é estabelecido segundo o maior atraso dentro do intervalo de valores conhecidos do processo (Tabela 2), sendo superestimado conforme sugerido por Keyser (2000).

A estimativa $\hat{\theta}(0)$ é escolhida tal que proporcione um controlador PI com baixa ação proporcional; dessa forma, prioriza-se evitar efeitos de sobressinal que levam o paciente ao estado de choque. Para isso, utilizou-se do método de sintonização Lambda para controladores PID (ÅSTRÖM; HÄGGLUND, 2006), porém, reduzindo gradualmente o ganho proporcional K_c e mantendo o ganho integral $\frac{K_c}{T_i}$ resultantes baseado nas ações de ajuste do Quadro 1. As Equações 96 e 97 expressam o controlador PI inicial ajustado nesse procedimento experimental

$$K_c(0) = \frac{\tau_{max}}{1000 K_{max}(\tau_{i_{max}} + \tau_r)} \quad (96)$$

$$T_i(0) = \frac{\tau_{max}}{1000} \quad (97)$$

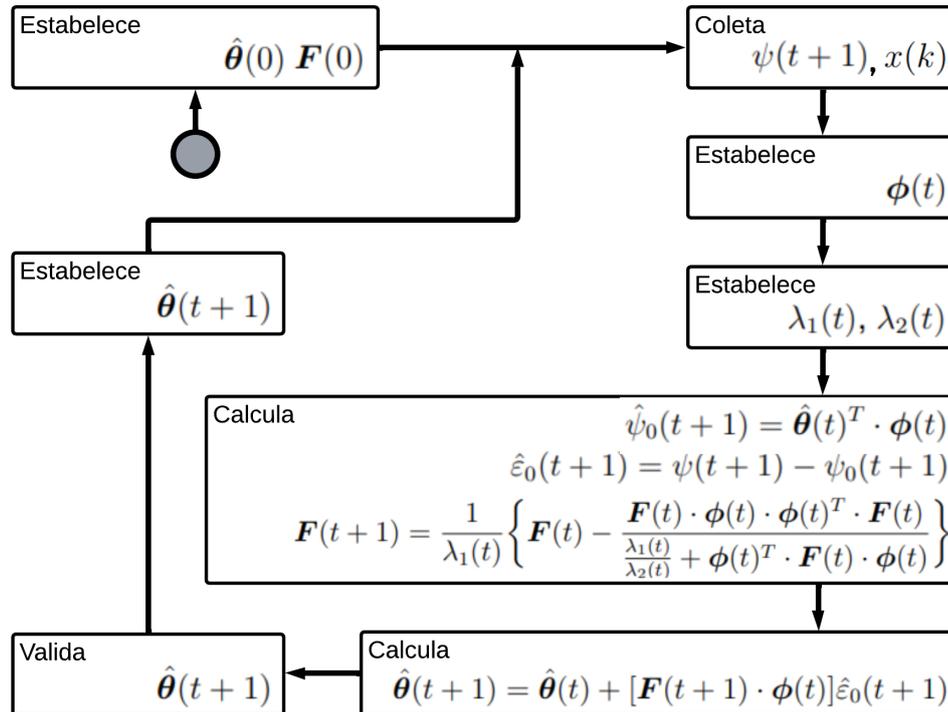
onde $K_{max} = 9$, $\tau_{max} = 75$ e $\tau_{i_{max}} = 60$ segundo o Quadro 6.

O uso e extrapolação do método de design Lambda visa encontrar um controlador PI inicial plausível aos pacientes mais sensíveis previstos no modelo, porém atuando com agressividade diminuída até o momento em que os valores adaptados dominem os comandos de controle. Essa estratégia segue as recomendações de inicialização de controles adaptativos e características do controle de pressão arterial discutidos no capítulo anterior.

3.1.2 Estimadores

O Capítulo anterior introduziu o algoritmo estimador RLS com fator de esquecimento e suas formas alternativas, incorporando atualizações condicionais ou compensações de excitação ou confiabilidade. O algoritmo RLS com fatores de esquecimento pode ser resumido conforme o diagrama da Figura 14

Figura 14 – Fluxograma do algoritmo RLS



Fonte: Autor (2023)

considerando, também, as etapas de validação da estimação segundo o contexto da aplicação ou condições de excitação, além do estabelecimento de λ_1 e λ_2 . Neste trabalho, $\lambda_2(t)$ é escolhido como unitário e, portanto, $\lambda_1(t)$ é reduzido a $\lambda(t)$.

O Quadro 5 apresenta as principais diferenças nas abordagens do algoritmo RLS e nomeia a identificação que terão neste trabalho

Quadro 5 – Distinções entre os algoritmos de estimação

Identificador do Algoritmo	Proposta conceitual	Principal distinção
Estimador 1	Ser influenciado por todas as informações coletadas	$\lambda = 1$
Estimador 2	Ser influenciado apenas pelo horizonte de tempo necessário	$\lambda = e^{-\frac{T_s}{T_{ff}}}; T_{ff} = 4\tau_r$
Estimador 3	Atualizar λ tal que mantenha a confiabilidade constante para novas informações	$\lambda(t+1) = \frac{tr\{F(t)\}}{tr\{F(0)\}}$

Fonte: Autor (2023).

O ajuste do Estimador 2 considera o horizonte de memória igual ao tempo de acomodação esperado pelo modelo de referência. Essa escolha é reforçada pois cumpre a expectativa de coletar entre 10 e 50 amostras para resultar em um controlador razoável, conforme discutido no capítulo anterior.

Nota-se que as variantes do RLS diferem em relação ao estabelecimento

do fator de esquecimento, estabelecendo uma comparação entre três diferentes abordagens para controle de pressão arterial:

- Manter o controlador sem esquecimento de dados aproxima da estimação ótima, apesar de prolongar o aprendizado;
- Estabelecer um curto horizonte de memória acelera o aprendizado, porém torna a estimação mais sensível ao ruído;
- O fator de esquecimento variante acelera o aprendizado e mantém compromisso com o ruído, mas não há garantia de sua ponderação entre as duas premissas.

Logo, os algoritmos adaptativos são conceitual e numericamente avaliados em função do desempenho de controle dadas suas formas de esquecimento.

A matriz covariância inicial $F(0)$ é definida com $F_{1,1}(0) = 0.01$, $F_{2,2}(0) = 0.001$, $F_{3,3}(0) = 1$ em sua diagonal principal e demais elementos nulos, tendo em vista a diferença inicial de escala entre c_0 e c_1 e o desconhecimento em relação à escala de c_2 inicialmente nulo. Já o ajuste do T_{ff} apontado no Quadro 5 foi definido com base nas características de excitação do sinal conforme discutido no Capítulo anterior, sendo definido igual ao tempo de acomodação esperado.

A etapa de validação da estimativa é executada a cada iteração para auxiliar a convergência do algoritmo e manter seu comando em uma região conhecida. Dada as características temporais do controlador PID e da problemática da pressão arterial, definem-se as seguintes condições necessárias para validar uma estimativa $\hat{\theta}(t+1)$:

$$\begin{cases} K_c < 0 \\ T_i > 0 \\ T_d \geq 0 \end{cases} \quad (98)$$

isto é

$$\begin{cases} -(c_1 + 2c_2) < 0 \\ -T_s \frac{c_1 + 2c_2}{c_0 + c_1 + c_2} > 0 \\ -T_s \frac{c_2}{c_1 + 2c_2} > 0 \end{cases} \quad (99)$$

visando manter valores plausíveis ao formato de controladores PID na perspectiva do domínio do tempo. Técnicas como essa ressaltam que, apesar de essencialmente livre de modelo, o DIRAC e demais formas de controle adaptativo usualmente necessitam de conhecimentos prévios da planta ou esforços de projeto.

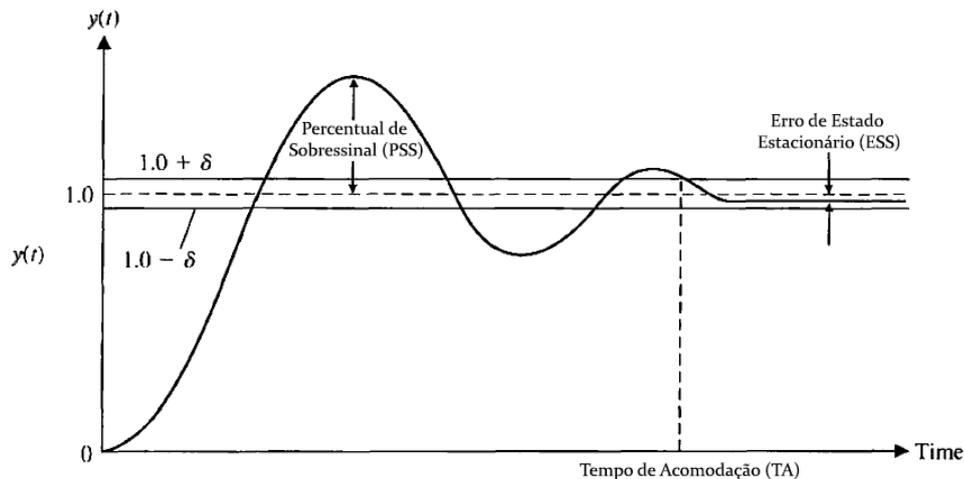
Dadas as características de um controlador robusto em seguir a referência, é esperado que o processo controlado acomode temporariamente sua saída em torno do referencial após certo intervalo de tempo, passando a exibir seu comportamento de estado estacionário durante o tempo restante em que não houver mudanças significativas. Assim, as informações de transitório utilizadas pelo algoritmo são gradativamente menos relevantes conforme novos dados de regime permanente são coletados ao longo do tempo.

Nessa perspectiva, propõe-se desligar a adaptação do controlador após a acomodação para rejeitar rapidamente possíveis perturbações e mudanças internas do processo. Logo, aplica-se o conceito de Conditional Update discutido no capítulo anterior, suprimindo momentos de baixa excitação, porém, visando a manutenção do controlador e não somente evitando Estimator Windup. Essa proposta mostrou-se eficaz em manter o controlador adaptado ao transitório e necessária para obter respostas em tempo viável, apesar de prejudicar parcialmente os resultados em relação ao erro estado estacionário.

3.2 INDICADORES DE DESEMPENHO

Para avaliar os resultados do controlador adaptativo da pressão arterial é necessário compreender os critérios de controle. A Figura 15 apresenta conceitualmente os critérios de Tempo de Acomodação (TA), Percentual de Sobressinal (PSS) e Erro de Estado Estacionário (ESS).

Figura 15 – Indicadores de desempenho de controladores



Fonte: Adaptado de Dorf e Bishop (2010, p. 311)

Os critérios na Figura 15 são escolhidos para avaliar o desempenho do controle da pressão arterial neste trabalho tal que contribuam em relação aos demais estudos na literatura, conforme expõe a Tabela 4.

Apesar de elucidarem a respeito das principais características da resposta temporal do processo, esses indicadores são introduzidos em casos lineares. Logo, as implementações dos indicadores de TA, PSS e ESS; requerem atenção e adaptações para que avaliem satisfatoriamente o controle de pressão arterial no ambiente de simulação numérica.

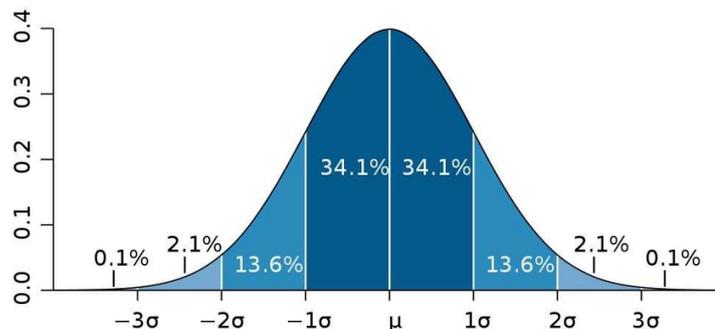
A atividade de estocástica atua continuamente no modelo e tende a oscilar a PA com amplitude maior que o intervalo entre 2 e 5 % comumente considerado na definição de TA. Além disso, dado que a atividade de estocástica e senoidal somadas podem oscilar cerca de $\pm 12 \text{ mmHg}$ do setpoint de 120 mmHg , por simplicidade, a faixa que determina o instante de acomodação foi definida como $\pm 15 \text{ mmHg}$ do setpoint para a análise numérica de desempenho.

O ESS foi medido numericamente tomando o valor máximo de erro na porção final da simulação, ou seja, considerando apenas os últimos 10% do intervalo de tempo no qual assume-se que o processo está acomodado. Uma abordagem semelhante é introduzida por Slate e Sheppard (1982), considerando o ESS a mínima faixa de erro em que o processo se encontra 90% do tempo.

É necessário considerar as características não lineares de aprendizado e esquecimento do algoritmo de controle adaptativo em que, diferente do caso linear, não há garantia que as oscilações sejam gradualmente menores. À vista disso, o sobressinal (SS) foi medido como o maior pico inferior ou superior em relação ao setpoint até o final da simulação. Em vista da presença de ação estocástica no modelo de pressão arterial e a característica não linear dos controladores adaptativos, avaliam-se os indicadores de TA, ESS e SS de diferentes amostras e tomam-se aproximações probabilísticas.

A adoção da curva normal como palpite inicial da distribuição é comum em projetos, pois a constatação de alta probabilidade baseada nessa costuma refletir também em alta probabilidade de ocorrência nas demais formas de distribuição possíveis (OPEX, 2013). Portanto, por simplicidade, assume-se uma distribuição normal das ocorrências e aproxima-se probabilisticamente os valores máximos para cada indicador de desempenho. Esses são computados com cerca de 97.8% de confiabilidade segundo a Figura 16

Figura 16 – Distribuição Normal

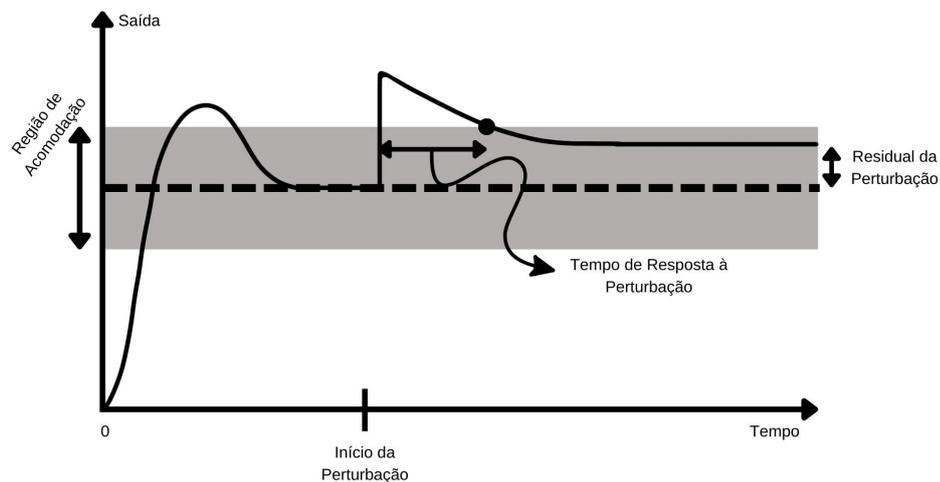


Fonte: Chandler (2012)

ou seja, a magnitude máxima de cada indicador é dita ser igual ao valor médio somado a dobro do desvio padrão (CHANDLER, 2012), chamados neste trabalho como Máximos Aproximados Probabilisticamente (MAP). Assim, analisa-se com simplicidade o desempenho de cada controlador com base nas diversas simulações, examinando com determinada confiabilidade se os valores MAP estão na região de interesse.

Após indicar que o algoritmo é capaz de conjuntamente se adaptar ao processo e perseguir o setpoint de maneira eficiente e segura, investiga-se um cenário hipotético que exija robustez dos controladores. Dado o esforço extensivo necessário para conclusões analíticas acerca da robustez de controladores adaptativos, estipula-se dois indicadores para avaliar numericamente a robustez dos controles em situações de perturbação (Figura 17): percentual de resíduo (PR) e tempo de resposta à perturbação (TR).

Figura 17 – Exemplo dos Indicadores para Avaliação de Robustez



Fonte: Autor (2023)

O PR avalia a porção de sinal de perturbação não rejeitada pelo controlador em regime permanente, sendo calculado como o erro percentual entre setpoint e saída média após acomodação em relação à amplitude da perturbação. Já o TR avalia o intervalo entre o instante de início da perturbação e o tempo para acomodação da saída do sistema. Esses indicadores sugerem a capacidade e viabilidade temporal do controlador em lidar com situações críticas à problemática, como a mudança abrupta da saída do sistema ou alterações significativas não previstas em seu modelo.

3.3 VERIFICAÇÃO DO DIRAC PARA PRESSÃO ARTERIAL

Neste trabalho são diferenciados oito perfis estáticos de pacientes (Quadro 6), variando gradualmente os parâmetros do modelo de pressão arterial, conforme apresentado por Silva et al. (2019).

Quadro 6 – Perfis de Pacientes

Identificador	Perfil do Paciente	K (mmHg/mlh)	τ (s)	τ_i (s)	τ_c (s)	α
Perfil P.1	Insensível	0.25	30	20	30	0.4
Perfil P.2		0.50	34	25	35	0.2
Perfil P.3		0.75	37	30	40	0.4
Perfil P.4	Nominal	1.00	40	40	45	0.4
Perfil P.5		2.00	45	45	55	0.2
Perfil P.6		4.00	55	50	60	0.4
Perfil P.7		6.00	65	55	65	0.2
Perfil P.8	Sensível	8.00	75	60	75	0.4

Fonte: Adaptado de Silva et al. (2019).

Dessa forma, o desempenho do controlador é avaliado em perfis de resposta ao medicamento em oposição à atividade de background, sendo essa com amplitude senoidal de 2 mmHg , ganho estocástico de 10 mmHg e limiar para reflexo de 70 mmHg .

Diferente das métricas de desempenho temporal consideradas no Quadro 4 para comparar os trabalhos da literatura, neste trabalho adota-se critérios temporais próximos ao procedimento de crises hipertensivas indicados por Sousa e Júnior (2014):

- Normalização da pressão arterial, reduzindo entre 20 e 25% do valor inicial de pressão arterial média na primeira hora de tratamento;
- Manter o nível corrigido da pressão arterial entre a segunda e sexta hora de tratamento;
- Manutenção dos níveis corrigidos de pressão arterial durante as próximas 24 horas seguintes.

Nessa perspectiva, são estabelecidas simulações de 6 horas do tratamento e espera-se que a ação do controlador idealmente proporcione níveis normalizados de PA em até duas horas. Dessa maneira, este trabalho foca na primeira etapa de tratamento baseada em infusão de Nitroprussiato de Sódio, sendo as demais etapas possivelmente ministradas via medicamentos orais (FERREIRA; JAZBIK; BRANDÃO, 2009).

Baseado na Tabela 3, o setpoint de controle é definido como 120 mmHg , enquanto o valor de offset base do modelo de pressão arterial é 180 mmHg .

Tabela 3 – Classificação da pressão arterial de acordo com a medição no consultório a partir de 18 anos de idade

Classificação	Pressão Arterial Sistólica (mmHg)
PA ótima	<120
PA normal	120-129
Pré-hipertensão	130-139
Hipertensão Arterial em Estágio 1	140-159
Hipertensão Arterial em Estágio 2	160-179
Hipertensão Arterial em Estágio 3	>179

Fonte: Adaptado de Barroso et al. (2021).

Logo, espera-se que a emergência com risco à vida ocasionada por Hipertensão Arterial em Estágio 3 seja controlada até níveis normalizados de PA, dado um quadro estático. Porém, a redução de pressão arterial deve evitar níveis abaixo de 100 *mmHg*, preservando as funcionalidades cerebrais, renais e sanguíneas do paciente (RODRIGUES, 2002).

Portanto, este trabalho adere ao procedimento gradual de controle da pressão arterial nas primeiras horas conforme previsto nas Diretrizes Brasileiras de Hipertensão Arterial (BARROSO et al., 2021, p. 596). A atenção ao procedimento clínico durante o projeto visa manter o foco na problemática estudada, integrando a proposta de controlador aos estudos da literatura e às boas práticas das ciências médicas.

Nesse contexto, adota-se os seguintes requisitos para determinar o desempenho do algoritmo de controle em relação ao procedimento clínico:

- Sobressinal Máximo deve ser menor que 40 *mmHg* e idealmente menor que 20 *mmHg* em relação ao setpoint final, indicando que a PA não causa estado de choque no paciente ou atinge níveis inferiores à 100 *mmHg* e, também, não retorna aos estágios de hipertensão arterial;
- Tempo de Acomodação Máximo em relação ao setpoint final não deve ultrapassar 21600 segundos (seis horas) e idealmente abaixo de 7200 segundos (duas horas), indicando aprendizagem do algoritmo em tempo hábil para prosseguir o tratamento padrão;
- Tempo de Acomodação Máximo em relação ao setpoint final é idealmente acima de 900 segundos (meia hora), indicando que o tratamento não é demasiadamente agressivo;
- Erro de Estado Estacionário Máximo deve ser acima de 18 *mmHg*, indicando que as oscilações em regime permanente observadas estão acima das esperadas na atividade de background acrescida de 5% do valor de setpoint.

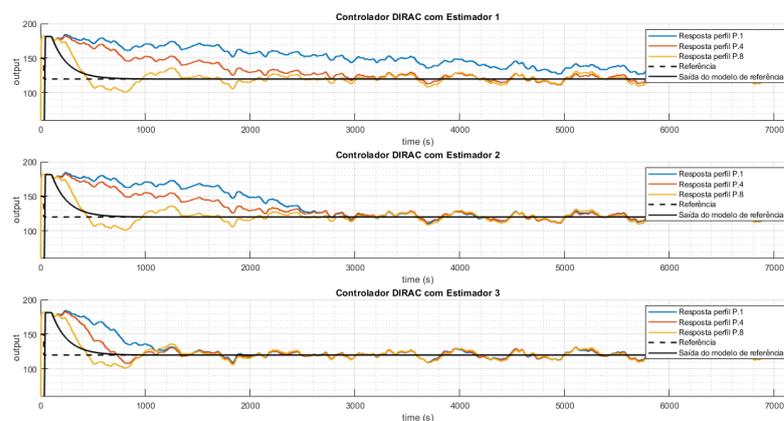
Assim, espera-se que a resposta dinâmica resultante indique retrogressão dos estágios de hipertensão com segurança aos diferentes perfis de paciente.

As restrições de segurança propostas no modelo pelas Equações 89-91 são

relacionadas ao sinal de comando resultante do controlador, portanto, podem ser impostas ao sistema saturando o sinal computado. Já as Equações 92-93 referem-se ao nível de pressão arterial média lido do paciente e são compensadas pela análise dos requisitos em relação ao procedimento das Diretrizes Brasileiras de Hipertensão apontados acima.

Para exemplificar a ação de controle resultando, a Figura 18 ilustra a resposta de quatro perfis de pacientes expostos ao mesmo sinal de atividade de background.

Figura 18 – Comparação entre Quatro Perfis de Paciente em Resposta ao degrau



Fonte: Autor (2023)

Logo, na Figura 18, o controlador PID é adaptado aos diferentes pacientes simulados que partem da situação de Hipertensão Arterial em Estágio 3, isto é, em 180 mmHg de PA até um estado de pressão arterial normalizada em torno de 120 mmHg de forma segura e em temporalmente viável.

Nota-se, também, a distinção entre as respostas dos diferentes controladores DIRAC em cada perfil de paciente, isto é, a tendência de crescimento de SS em pacientes hipersensíveis e crescimento de TA em pacientes insensíveis. Porém, em vista da influência da atividade de background nos indicadores de desempenho, as respostas em simulação são amostradas em 1000 pacientes de cada perfil presente no Quadro 1.

Nesse contexto, os resultados preliminares obtidos pelos controladores adaptativo são avaliados nas Tabelas 4-6 em relação ao SS, TA e ESS máximos.

Tabela 4 – Avaliação Preliminar dos Sobressinais Máximos ($mmHg$)

Identificador de perfil do paciente	Estimador 1	Estimador 2	Estimador 3
Perfil P.1	17.698	14.361	15.509
Perfil P.2	16.298	14.301	15.535
Perfil P.3	15.388	14.730	16.205
Perfil P.4	15.296	14.785	17.038
Perfil P.5	14.324	14.366	23.362
Perfil P.6	14.582	14.579	25.471
Perfil P.7	21.110	21.199	44.457
Perfil P.8	26.774	27.465	53.286

Tabela 5 – Avaliação Preliminar dos Tempos de Acomodação Máximos (s)

Identificador de perfil do paciente	Estimador 1	Estimador 2	Estimador 3
Perfil P.1	9788.309	3801.171	1335.830
Perfil P.2	5358.774	3211.421	1173.404
Perfil P.3	3710.510	2624.563	1044.656
Perfil P.4	2651.476	2476.244	1030.356
Perfil P.5	1126.146	1180.582	2087.763
Perfil P.6	627.058	628.744	1340.795
Perfil P.7	863.591	729.818	2755.002
Perfil P.8	939.161	948.324	3783.728

Tabela 6 – Avaliação Preliminar dos Erros de Estado Estacionário Máximos ($mmHg$)

Identificador de perfil do paciente	Estimador 1	Estimador 2	Estimador 3
Perfil P.1	19.131	13.033	13.830
Perfil P.2	13.627	12.810	13.943
Perfil P.3	13.836	12.557	14.203
Perfil P.4	12.383	12.528	14.449
Perfil P.5	12.740	12.761	14.854
Perfil P.6	13.249	13.261	14.561
Perfil P.7	14.525	14.481	15.006
Perfil P.8	14.999	15.024	16.056

Nota-se magnitudes inviáveis de SS, sendo maiores que $20 mmHg$ para os perfis P.7 e P.8 para todos os controladores DIRAC adotados.

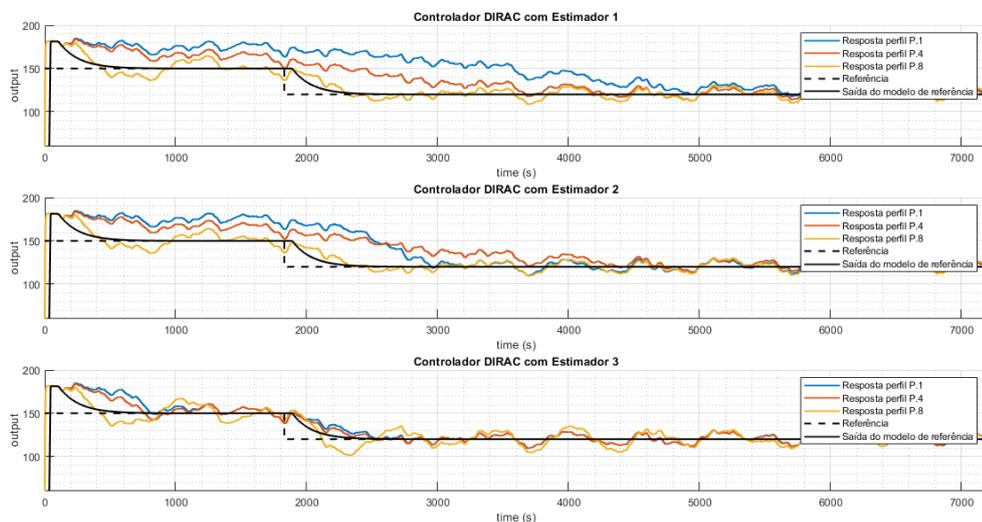
Logo, os controladores indicam serem capazes de controlar a PA em relação à capacidade de perseguir o setpoint em estado estacionário, porém aponta-se falhas

em relação ao SS. Logo, ressalta-se a necessidade de atender diferentes tipos de pacientes torna o projeto de controle árduo, dada as condições de segurança e métricas de desempenho restritivas influenciadas por características estocásticas significativas.

Ações para corrigir os desempenhos podem ser desenvolvidas baseadas nas características de aprendizagem apontadas no capítulo anterior. Nesse intuito, propõe-se alterar o formato da entrada referencial tal que reduza as falhas de SS identificadas, dividindo o comando de referência em um sinal escada composto de dois degraus e excitando o algoritmo em dois momentos de transitório distintos. Assim, espera-se que o controlador atuante na segunda etapa de entrada seja melhor ajustado, resultando em uma resposta dinâmica mais próxima do SS desejado, apesar do possível acréscimo de latência.

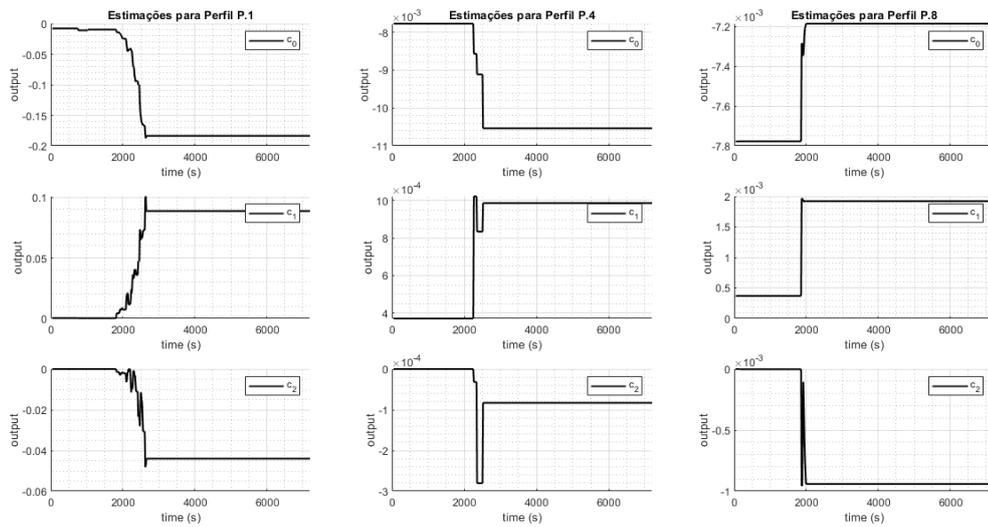
Baseado nessa hipótese, altera-se a entrada de referência para um sinal escada inicialmente definido como 150 mmHg e alterado para 120 mmHg após o intervalo de tempo de acomodação da saída modelo de referência. As Figuras 19-19 ilustram as respostas dinâmicas e respectivos controladores obtidos na simulação de pacientes de sensibilidade mínima, nominal e máxima.

Figura 19 – Comparação visual da resposta dinâmica ao sinal escada



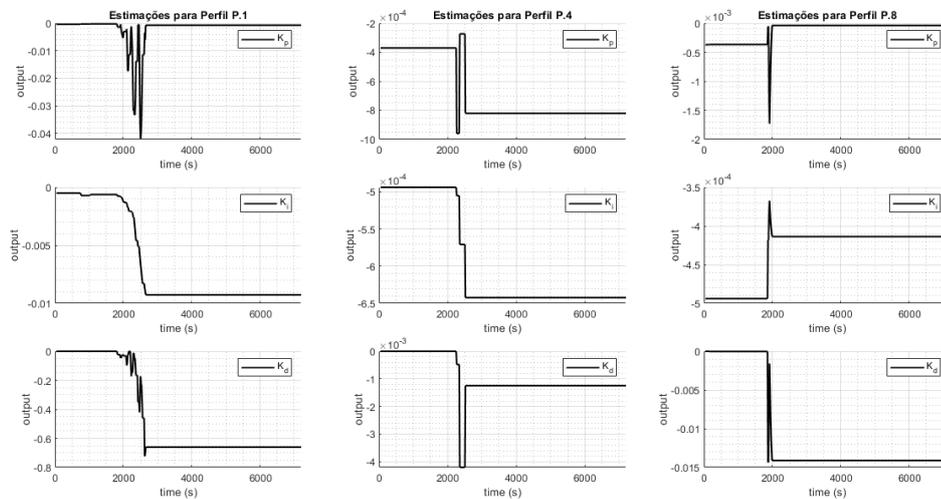
Fonte: Autor (2023)

Figura 20 – Comparação visual da estimação gerada pelo sinal escada



Fonte: Autor (2023)

Figura 21 – Comparação visual da estimação convertida em formato PID gerada pelo sinal escada



Fonte: Autor (2023)

Verifica-se a adaptação determinante do controlador por meio das Figuras 20 e 21, alterando bruscamente os coeficientes estimados próximo ao instante 1800 s. Já as Tabelas 7-9 contêm os resultados ao reproduzir o experimento realizado anteriormente.

Tabela 7 – Avaliação dos Sobressinais Máximos ($mmHg$)

Identificador de perfil do paciente	Estimador 1	Estimador 2	Estimador 3
Perfil P.1	17.009	14.544	16.027
Perfil P.2	15.995	14.415	16.409
Perfil P.3	15.211	14.507	17.051
Perfil P.4	15.036	14.509	17.719
Perfil P.5	14.199	14.277	20.566
Perfil P.6	14.572	14.585	24.170
Perfil P.7	16.841	16.927	36.396
Perfil P.8	18.599	18.618	44.613

Tabela 8 – Avaliação dos Tempos de Acomodação Máximos (s)

Identificador de perfil do paciente	Estimador 1	Estimador 2	Estimador 3
Perfil P.1	9166.631	4392.034	2692.766
Perfil P.2	5599.002	3827.912	2312.345
Perfil P.3	3994.412	3582.555	2229.367
Perfil P.4	3941.641	3483.598	2330.643
Perfil P.5	2542.809	2537.446	3126.284
Perfil P.6	2267.615	2277.608	4054.354
Perfil P.7	2270.737	2269.328	5469.717
Perfil P.8	2287.444	2928.908	6382.494

Tabela 9 – Avaliação dos Erros de Estado Estacionário Máximos ($mmHg$)

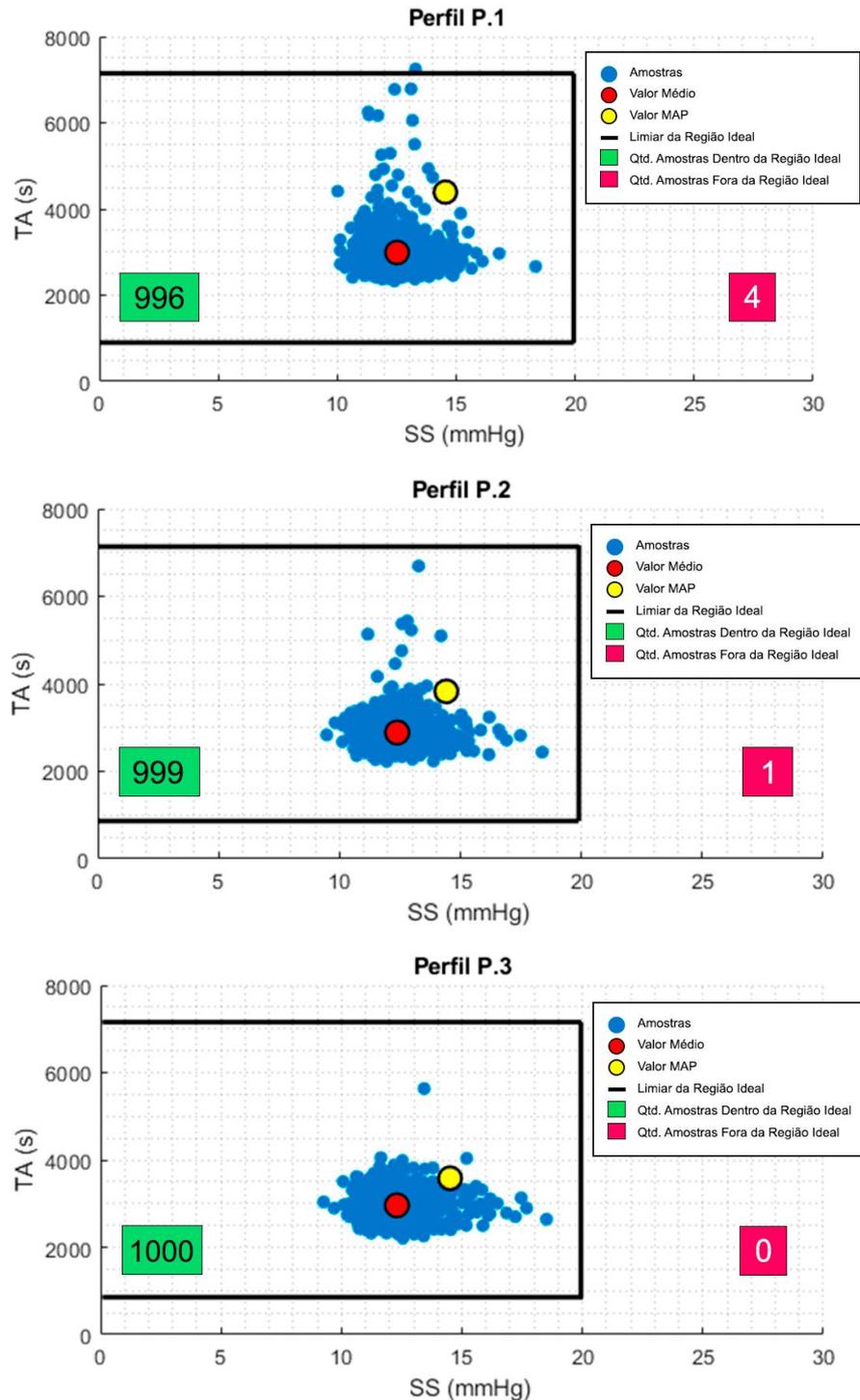
Identificador de perfil do paciente	Estimador 1	Estimador 2	Estimador 3
Perfil P.1	14.732	13.221	14.139
Perfil P.2	12.292	13.003	14.419
Perfil P.3	12.345	12.622	14.630
Perfil P.4	12.390	12.583	14.919
Perfil P.5	12.750	12.772	15.292
Perfil P.6	13.260	13.266	15.538
Perfil P.7	14.435	14.479	15.842
Perfil P.8	14.967	15.013	16.245

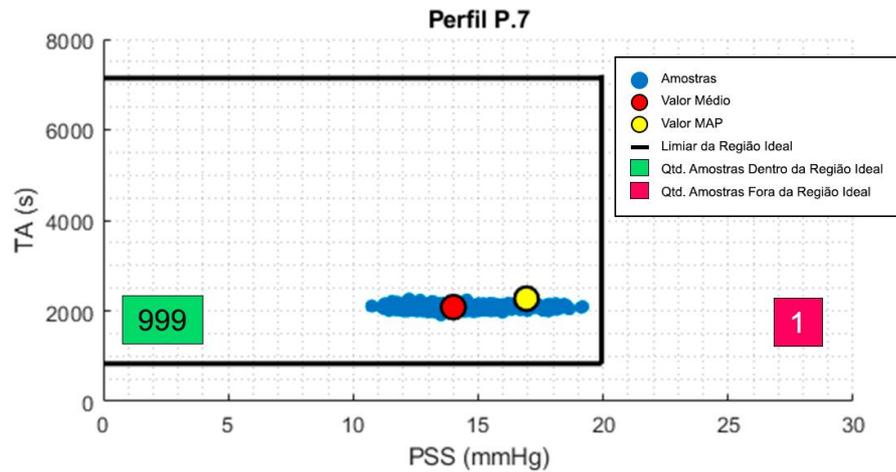
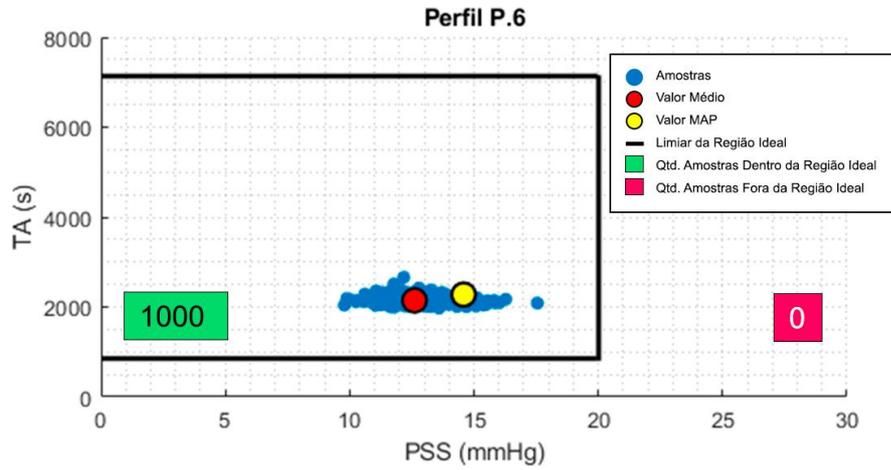
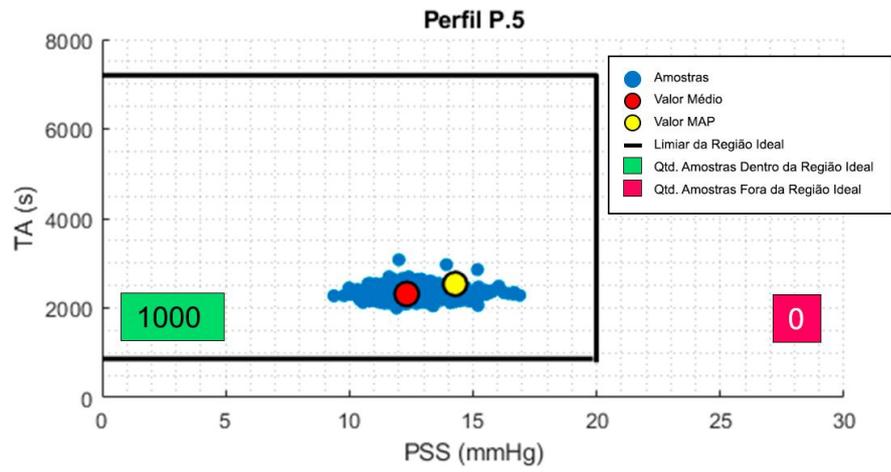
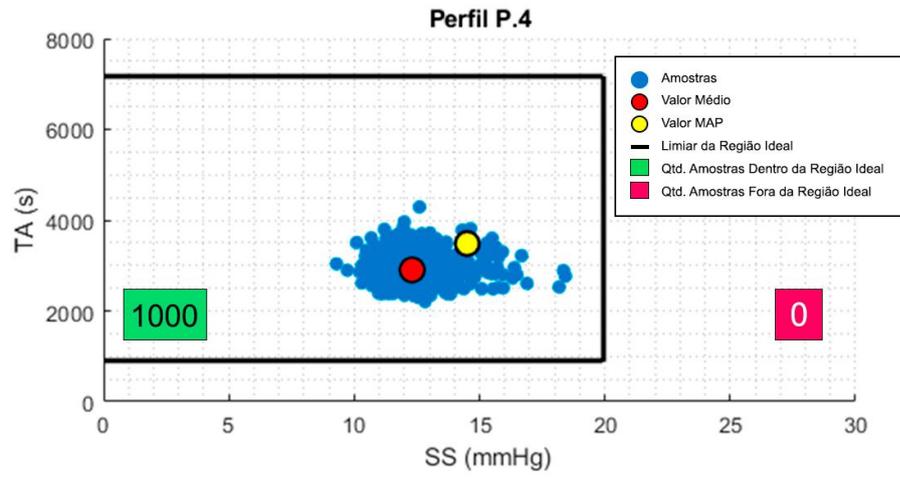
Logo, nota-se melhoria no resultado dos DIRAC com Estimadores 1 e 2 ao adotar o sinal de referência em escada, uma vez que as condições de SS inviáveis anteriores são superadas para todos os perfis de paciente. Já o DIRAC com Estimador 3 indica falhar em relação ao SS do perfil *P.8* apesar de apresentar expressiva melhora em

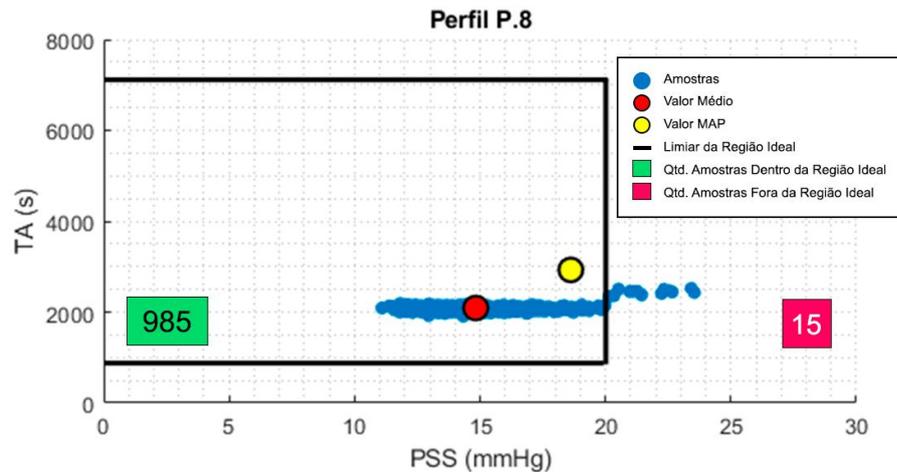
relação ao experimento com referência em degrau.

Finalmente, as Figuras abaixo contêm as 1000 amostras e os indicadores computados de SS e TA para ilustrar como as medições se relacionam aos valores médios e MAP correspondentes do controlador DIRAC com Estimador 2, validando visualmente os resultados encontrados.

Figura 22 – Dispersão de 1000 Amostras em Comparação à Média, MAP e Região Ideal do DIRAC com Estimador 2







Fonte: Autor (2023)

Assim, verifica-se que os valores MAP adotados expressam com sucesso a generalização do comportamento encontrado e compactuam com a confiabilidade esperada. Portanto, o controlador DIRAC integrado ao Estimador 2 demonstra ser capaz de controlar quadros estáticos de hipertensão respeitando os critérios de desempenhos adotados, porém, com maior dificuldade nos perfis P.1 e P.8.

Para averiguar a robustez do algoritmo, estipula-se um acréscimo no valor de offset da atividade de background tal que regresse o quadro de pressão normalizada ao de Hipertensão em Estágio 1 (Tabela 3). Para isso, um sinal degrau de 25 mmHg iniciado após três horas de tratamento é somado ao offset inicial, resultando nas avaliações de PR e TR médios para 1000 amostras de cada perfil de paciente conforme as Tabelas 10 e 11.

Tabela 10 – Avaliação dos Percentuais Residuais da Perturbação (%)

Identificador de perfil do paciente	Estimador 1	Estimador 2	Estimador 3
Perfil P.1	88.723	88.408	88.408
Perfil P.2	4.416	2.886	3.351
Perfil P.3	3.721	3.104	3.045
Perfil P.4	3.586	3.450	2.960
Perfil P.5	3.283	3.219	3.040
Perfil P.6	3.050	3.038	3.015
Perfil P.7	2.908	2.904	3.163
Perfil P.8	2.894	2.893	3.187

Tabela 11 – Avaliação dos Tempos de Resposta à Perturbação (s)

Identificador de perfil do paciente	Estimador 1	Estimador 2	Estimador 3
Perfil P.1	10795.975	10795.975	10795.975
Perfil P.2	1597.883	229.083	2155.166
Perfil P.3	1165.306	418.121	1503.287
Perfil P.4	1048.238	568.831	1137.755
Perfil P.5	480.120	461.829	1073.904
Perfil P.6	310.810	307.128	962.569
Perfil P.7	251.689	255.567	997.136
Perfil P.8	265.232	269.842	967.577

O controlador DIRAC mostra-se eficaz em reverter a perturbação imposta, possuindo resíduo médio entre cerca de 2.9 e 4.4% com exceção do perfil P.1. O controlador também é eficaz em atuar em tempo hábil, reduzindo a pressão no intervalo de cerca de 229 até 2155 segundos, porém com exceção do perfil P.1. O comportamento diferenciado do perfil P.1 é justificado pela presença de saturadores no sinal de comando, impossibilitando que a taxa elevada de infusão necessária para tratar pacientes insensíveis seja infundida.

A ação de saturadores é averiguada pela amplitude média em regime permanente do sinal de controle (Tabela 12).

Tabela 12 – Avaliação do nível de infusão em Resposta à Perturbação (mL/h)

Identificador de perfil do paciente	Estimador 1	Estimador 2	Estimador 3
Perfil P.1	179.778	180	180
Perfil P.2	140.347	140.772	140.214
Perfil P.3	80.343	80.406	80.449
Perfil P.4	70.327	70.308	70.387
Perfil P.5	30.155	30.156	30.192
Perfil P.6	17.596	17.596	17.597
Perfil P.7	10.056	10.056	10.058
Perfil P.8	8.799	8.798	8.802

Na Tabela 12 os valores são decrescentes entre os perfis listados conforme esses apresentam-se mais sensíveis ao medicamento, exigindo amplitudes gradativamente menores para resultar em uma mesma redução de PA. Além disso, a amplitude do sinal coincide com o valor de saturação de $180 mL/h$ previsto na Equação 89, confirmando a justificativa.

Portanto, o valor residual e latência resposta percebidos para o perfil P.1 na presença da perturbação não indicam falhas do algoritmo, mas demonstram uma limitação geral do tratamento de hipertensos com Nitroprussiato de Sódio. Finalmente, o controlador DIRAC desenvolvido neste trabalho indica ser robusto, capaz de reduzir a pressão arterial de pacientes sujeitos a emergências hipertensivas de forma eficaz e operar conforme os procedimentos clínicos previstos nas diretrizes brasileiras de hipertensão.

4 CONCLUSÕES

Este trabalho teve como objetivo desenvolver controladores explicita e diretamente adaptados para tratamento intensivo de pacientes hipertensos, contextualizando, também, temas como algoritmos de estimação, modelagem do problema e implicações clínicas. Adota-se neste trabalho um controlador PID DIRAC iniciado no formato PI, sendo ajustado previamente pelo método Lambda e otimizado recursivamente pelo algoritmo RLS em relação ao comportamento de um modelo de referência de primeira ordem. A metodologia utilizada é baseada em simulações numéricas, desenvolvendo os conceitos essenciais dos algoritmos e analisando os resultados dos controladores propostos.

A automatização de processos por meio de malhas fechadas tem mostrado potencial ao longo dos anos e é vantajosa ao corrigir a atuação a fim de compensar efeitos de perturbações, variações e imprecisões. O controlador PID é amplamente utilizado em malhas fechadas e encontra-se difundido em diversas aplicações dado seu apelo intuitivo, sendo encontrado em cerca de 95 % das malhas fechadas industriais e majoritariamente no formato PI.

Nessa perspectiva, o tratamento automatizado de pacientes hipertensos é aplicado em milhares de pacientes e alguns desses demonstram agravantes no tratamento como hipersensibilidade ou insensibilidade ao medicamento. Levando em conta essas divergências entre grupos de pacientes e variações estocásticas, controladores de ganhos fixos podem causar oscilações danosas ao indivíduo.

Para aperfeiçoar o desempenho dos sistemas controle estuda-se métodos adaptativos que aprendam sobre o sistema em que atuam e então ajustem o controlador recursivamente. Dessa forma, permite-se que o algoritmo de controle se ajuste à demanda de melhoria, otimizando a ação de controle mesmo em processos com variações desconhecidas ou imperfeições de modelagem.

O controlador PID desenvolvido no trabalho é dito com adaptação direta, ou seja, não se baseia na estimação da planta propriamente mas diretamente dos coeficientes do controlador. A estimação é estruturada através da filtragem dos sinais de entrada e saída do processo e computada tal que solucione o problema de otimização formulado, ajustando o controlador diretamente com o algoritmo de identificação Recursive Least Squares.

As possíveis aplicações de algoritmos DIRAC são numerosas, porém no processo de pesquisa bibliográfica deste trabalho constatou-se baixa exploração na comunidade acadêmica desses controladores em aplicações de controle de pressão arterial. Em especial, o estudo dos métodos DIRAC apresentado por Keyser (2000)

e Keyser e De (1989) é particularmente oportuno, pois, apesar de presente em outra aplicação biomédica, não foram localizadas investigações na literatura sobre esses controladores no tratamento de hipertensos.

As variações do controlador DIRAC apresentadas neste trabalho são diferenciadas pela forma de esquecimento do RLS, sendo estipuladas três formas do fatores de esquecimento: unitário (ausência de esquecimento), breve horizonte de tempo e variante no tempo. Após introduzir o RLS, discutiu-se problemas relacionados a ponderação de dados e as alternativas do algoritmo capazes de evitar-los. Assim, atualizações condicionais e diferentes fatores de esquecimento constantes foram apresentados de tal forma que resultem em hipóteses distintas de projeto quando integradas ao controlador DIRAC.

Ainda na perspectiva das limitações do algoritmo identificador, a influência da excitação do sinal na estimação de parâmetros foi investigada, levando em consideração sua formulação e guiando os resultados esperados. O estudo teórico dos sinais resultou em possíveis melhorias dos experimentos de controle, uma vez que a primitiva de rejeitar perturbações e o sinal originado pela realimentação sugerem presença de baixos níveis de excitação.

Outros temas pertinentes ao projeto de controle adaptativo foram discutidos, como período de amostragem, efeitos de perturbação, experimentos de robustez, emprego de conhecimento prévio e métodos de escolha da estimativa inicial. Assim, após introduzir tópicos essenciais para compreender a fundamentação de algoritmos adaptativos e os resultados esperados, esses temas foram integrados no projeto do controlador DIRAC.

Também foi introduzido o contexto da hipertensão arterial para fins didáticos e compreensão da problemática abordada, definindo suas características de diagnóstico, tratamento e impactos social e econômico. Nessa perspectiva, distinguiu-se casos de urgência e emergência hipertensas, sendo diferenciadas entre sintomas, gravidade e procedimento terapêutico. Além disso, reforçou-se a complexidade do sistema considerando a quantidade e interação dos agentes químicos envolvidos.

O modelo matemático que descreve o comportamento estacionário da pressão arterial e a resposta ao medicamento responsável por diminuí-la foram apresentados. O modelo é submetido a critérios de segurança que são introduzidos na forma de saturadores do medicamento, evitando gerar sinais de controle com magnitudes danosas ao paciente.

Na perspectiva de controle, estipula-se a atualização condicional tal que desligue a adaptação ao detectar acomodação da pressão arterial para evitar a influência prolongada do sinal em regime permanente na aprendizagem do algoritmo. Além disso, são descritas três hipóteses em relação ao método de definição do fator de esquecimento, originando três controladores DIRAC estudados neste trabalho e,

portanto, diferenciados entre si pelos estimadores que utilizam.

Para verificar o funcionamento do algoritmo de controle, foram diferenciados oito perfis estáticos de pacientes em relação aos parâmetros de resposta ao medicamento. Já o objetivo de controle foi baseado no procedimento presente na Diretriz Brasileira de Hipertensão, sugerindo reduzir a PA com risco à vida até o nível normalizado como parte de um tratamento por etapas de emergências hipertensivas, mantendo o compromisso com o tratamento clínico esperado em campo.

Logo, nas simulações, o paciente parte da Hipertensão em Estágio 3, isto é, 180 mmHg de PA até o setpoint de 120 mmHg em até duas horas e evitando situações de risco ao paciente. Para isso, o tratamento adaptativo foi avaliado por aferições probabilísticas de indicadores de desempenho encontrados na literatura: Sobressinal, Tempo de Acomodação e Erro de Estado Estacionário. As métricas adotadas buscam evitar situações como atingir estado de choque, volta à estágios de hipertensão, danos cerebrais causados por reduções abruptas e atraso em prosseguir o tratamento devido à latência para conversão ao setpoint.

Além da capacidade de controlar quadros estáticos de hipertensão, foi averiguada a robustez do controlador a perturbações abruptas, utilizando um sinal de perturbação em degrau após a acomodação inicial em pressão normalizada tal que o paciente regredisse ao Estágio de Hipertensão 1. Para avaliar numericamente o desempenho em rejeitar a perturbação foram propostos os indicadores de Residual e Tempo de Resposta à Perturbação. A partir desses indicadores, avaliou-se também a capacidade percentual e a latência para perseguir novamente o sinal de referência após certa amplitude de perturbação.

Finalmente, o controlador DIRAC resultante mostrou ser capaz de reduzir a pressão arterial de pacientes em emergência hipertensiva ao utilizar diferentes conceitos de controle adaptativo durante o projeto, sendo eficaz em operar conforme o procedimento clínico adotado com determinada confiabilidade.

No entanto, indica-se continuar a investigação da robustez desse método de controle em trabalhos futuros, considerando diferentes variações e perturbações. Na mesma perspectiva, sugere-se buscar modelos de medicamentos que substituam o Nitroprussiato de Sódio, dada suas limitações por toxicidade. Além disso, estudos podem ser realizados para ajustar ou alternar o modelo de referência e algoritmo estimador, tal que aumentem a confiabilidade dos resultados.

REFERÊNCIAS

- ALBERTOS, P.; SALGADO, M.; OLIVARES, M. Are delays in digital control implementation always bad. In: **Proc. Asian Control Conference**. Shanghai, China: [s.n.], 2000. v. 1, p. 634–639.
- ALEXANDER, T. S. **Adaptive signal processing: theory and applications**. New York: Springer Science & Business Media, 2012.
- ARNSPARGER, J. M. et al. Adaptive control of blood pressure. **IEEE Transactions on Biomedical Engineering**, n. 3, p. 168–176, 1983.
- ÅSTRÖM, K.; HAGGLUND, T. Pid controllers: theory, design and tuning. **Instrument Society of America**, 1995.
- ÅSTRÖM, K. J.; HÄGGLUND, T. **Advanced PID control**. [S.l.]: ISA-The Instrumentation, Systems, and Automation Society Research Triangle Park, 2006. v. 461.
- ÅSTRÖM, K. J.; NEUMANN, L.; GUTMAN, P.-O. A comparison between robust and adaptive control of uncertain systems. **IFAC Proceedings Volumes**, Elsevier, v. 20, n. 2, p. 43–48, 1987.
- ÅSTRÖM, K. J.; WITTENMARK, B. **Adaptive control**. Mineola, New York: Courier Corporation, 2013.
- BAILEY, J. M.; HADDAD, W. M. Drug dosing control in clinical pharmacology. **IEEE Control Systems Magazine**, v. 25, n. 2, p. 35–51, 2005.
- BÁNYÁSZ, C.; KEVICZKY, L. Direct methods for self-tuning pid regulators. **IFAC Proceedings Volumes**, Elsevier, v. 15, n. 4, p. 1395–1400, 1982.
- BARROSO, W. K. S. et al. Diretrizes brasileiras de hipertensão arterial–2020. **Arquivos Brasileiros de Cardiologia**, SciELO Brasil, v. 116, p. 516–658, 2021.
- BASHA, A. A.; VIVEKANANDAN, S.; PARTHASARATHY, P. Evolution of blood pressure control identification in lieu of post-surgery diabetic patients: a review. **Health information science and systems**, Springer, v. 6, n. 1, p. 1–10, 2018.
- BOBÁL, V. et al. **Digital Self-tuning Controllers: Algorithms, Implementation and Applications**. Springer London, 2006. (Advanced Textbooks in Control and Signal Processing). ISBN 9781846280412. Disponível em: <https://books.google.com.br/books?id=SNi1n0QnD4QC>.
- CARDOSO, M. E. Emergências hipertensivas. **Revista Hospital Universitário Pedro Ernesto**, v. 12, n. 3, 2013.
- CHANDLER, D. L. **Explained: Sigma**. MIT News Office, 2012. Disponível em: <https://news.mit.edu/2012/explained-sigma-0209#:~:text=One%20standard%20deviation%2C%20or%20one,sigmas%20would%20include%2099.7%20percent>.
- DORF, R. C.; BISHOP, R. H. **Modern Control Systems**. 12. ed. Upper Saddle River, Nova Jersey: Prentice Hall; 12 edition, 2010. 1104 p. ISBN 0136024580.

- ESFANDIARI, K.; ABDOLLAHI, F.; TALEBI, H. A. **Neural Network-Based Adaptive Control of Uncertain Nonlinear Systems**. Basel, Suíça: Springer, 2022.
- FEITOSA-FILHO, G. S. et al. Emergências hipertensivas. **Revista Brasileira de Terapia Intensiva**, SciELO Brasil, v. 20, p. 305–312, 2008.
- FERREIRA, R. F.; JAZBIK, C.; BRANDÃO, A. Emergências hipertensivas. **Revista Hospital Universitário Pedro Ernesto (TÍTULO NÃO-CORRENTE)**, v. 8, n. 2, 2009.
- FORTESCUE, T.; KERSHENBAUM, L. S.; YDSTIE, B. E. Implementation of self-tuning regulators with variable forgetting factors. **Automatica**, Elsevier, v. 17, n. 6, p. 831–835, 1981.
- GAO, Y.; ER, M. J. An intelligent adaptive control scheme for postsurgical blood pressure regulation. **IEEE transactions on neural networks**, v. 16, n. 2, p. 475–483, 2005.
- GOODWIN, G. C.; SIN, K. S. **Adaptive filtering prediction and control**. [S.l.]: Courier Corporation, 2014.
- GUIDI, F. H. et al. **Open and closed loop model identification and validation**. 2009. Tese (Doutorado) — University of Pretoria, 2009.
- HE, W.; KAUFMAN, H.; ROY, R. Multiple model adaptive control procedure for blood pressure control. **IEEE transactions on biomedical engineering**, n. 1, p. 10–19, 1986.
- HOSTETTER, G. H. Recursive estimation. In: **Handbook of digital signal processing**. [S.l.]: Elsevier, 1987. p. 899–940.
- HUNT, K. J. A survey of recursive identification algorithms. **Transactions of the Institute of Measurement and Control**, Sage Publications Sage CA: Thousand Oaks, CA, v. 8, n. 5, p. 273–278, 1986.
- IOANNOU, P.; FIDAN, B. **Adaptive control tutorial**. 3600 Market Street, 6th Floor Philadelphia, PA 19104 USA: SIAM, 2006.
- IOANNOU, P. A.; SUN, J. **Robust adaptive control**. North Chelmsford, Massachusetts: Courier Corporation, 2012.
- IONESCU, C.-M.; DE KEYSER, R. A robust adaptive auto-tuner for artificially stimulated muscles. **Control Engineering and Applied Informatics**, v. 8, n. 3, p. 15–22, 2007.
- ISAKA, S.; SEBALD, A. V. Control strategies for arterial blood pressure regulation. **IEEE Transactions on Biomedical Engineering**, v. 40, n. 4, p. 353–363, 1993.
- ISERMANN, R. Practical aspects of process identification. In: **System Identification**. [S.l.]: Elsevier, 1981. p. 575–587.
- ISERMANN, R. **Digital Control Systems, volume II. Stochastic Control, Multivariable Control, Adaptive Control, Applications**. [S.l.]: Springer, 1991.
- JURY, E. **Theory and Application of the z-Transform Method**. Huntington, New York: Wiley New York., 1964. v. 3.

- KEYSER, R.; DE, M. Dirac: A finite impulse response direct adaptive controller. In: **Adaptive Control Strategies for Industrial Use**. [S.l.]: Springer, 1989. p. 65–88.
- DE KEYSER, R. Dirac: A direct adaptive controller. **IFAC Proceedings Volumes**, Elsevier, v. 33, n. 4, p. 173–178, 2000.
- DE KEYSER, R.; IONESCU, C. Adaptive control of a pressure-controlled artificial ventilator: A simulator-based evaluation using real copd patient data. **Computer Methods and Programs in Biomedicine**, Elsevier, v. 104, n. 3, p. e178–e188, 2011.
- KULHAVÝ, R.; ZARROP, M. B. On a general concept of forgetting. **International Journal of Control**, Taylor & Francis, v. 58, n. 4, p. 905–924, 1993.
- LANDAU, I. D. et al. **Adaptive control: algorithms, analysis and applications**. Gray's Inn Road, London: Springer Science & Business Media, 2011.
- LJUNG, L. **System Identification: Theory for the User**. Second. Upper Saddle River, NJ: Prentice Hall, 1999.
- LJUNG, L. Recursive identification algorithms. **Circuits, Systems and Signal Processing**, Springer, v. 21, p. 57–68, 2002.
- LJUNG, L.; GUNNARSSON, S. Adaptation and tracking in system identification—a survey. **Automatica**, Elsevier, v. 26, n. 1, p. 7–21, 1990.
- LOPES, C. D. O. **Emergências hipertensivas**. 2019. Tese (Doutorado) — Universidade da Beira Interior, 2019.
- MALIK, O.; HOPE, G.; CHENG, S. Some issues on the practical use of recursive least squares identification in self-tuning control. **International Journal of Control**, Taylor & Francis, v. 53, n. 5, p. 1021–1033, 1991.
- MARTIN, J. F.; SCHNEIDER, A. M.; SMITH, N. T. Multiple-model adaptive control of blood pressure using sodium nitroprusside. **IEEE transactions on biomedical engineering**, n. 8, p. 603–611, 1987.
- MATLAB. **version 7.10.0 (R2010a)**. Natick, Massachusetts: The MathWorks Inc., 2020.
- MILLARD, R. et al. On using a self-tuning controller for blood pressure regulation during surgery in man. **Computers in biology and medicine**, Elsevier, v. 17, n. 1, p. 1–18, 1987.
- NGUYEN, N. T. Model-reference adaptive control. In: **Model-Reference Adaptive Control**. New York, 10036, United States: Springer, 2018. p. 83–123.
- NISE, N. S.; SILVA, F. da. Engenharia de sistemas de controle, vol. 3. **Rio de Janeiro: Livros Técnicos e Científicos**, 2004.
- NISHIMURA, T.; SAITO, M. A 24-hour prediction model of blood pressure employing endocrine system and autonomic nervous system. In: **Computing and Monitoring in Anesthesia and Intensive Care**. [S.l.]: Springer, 1992. p. 84–92.
- OGATA, K. Engenharia de controle moderno, 3a. edição. **Editora. LTC. Rio de Janeiro**, 1998.

- OPEX, L. T. **Types of Distributions Used in Six Sigma**. OpEx Learning Inc, 2013. Disponível em: <https://opexlearning.com/resources/types-of-distributions-in-six-sigma/12481/>.
- PAJUNEN, G. A.; STEINMETZ, M.; SHANKAR, R. Model reference adaptive control with constraints for postoperative blood pressure management. **IEEE Transactions on Biomedical Engineering**, v. 37, n. 7, p. 679–687, 1990.
- PHILLIPS, C. L.; NAGLE, H. T. **Digital control system analysis and design**. 910 Sylvan Ave, Englewood Cliffs, NJ: Prentice Hall Englewood Cliffs, NJ, 1995.
- RAHIMI, Z.; MORADI, M.; NASRI, H. A systematic review of the role of renin angiotensin aldosterone system genes in diabetes mellitus, diabetic retinopathy and diabetic neuropathy. **Journal of research in medical sciences: the official journal of Isfahan University of Medical Sciences**, Wolters Kluwer–Medknow Publications, v. 19, n. 11, p. 1090, 2014.
- RODRIGUES, C. I. S. Tratamento das emergências hipertensivas. **Rev Bras Hipertens**, v. 9, n. 4, p. 353–8, 2002.
- SHARMA, R. et al. Intelligent automated drug administration and therapy: Future of healthcare. **Drug Delivery and Translational Research**, Springer, v. 11, n. 5, p. 1878–1902, 2021.
- SIDDIQUI, A. et al. Endocrine stress responses and risk of type 2 diabetes mellitus. **Stress**, Taylor & Francis, v. 18, n. 5, p. 498–506, 2015.
- SILVA, S. J. da et al. Simple adaptive pi controller development and evaluation for mean arterial pressure regulation. **Research on Biomedical Engineering**, Springer, v. 35, n. 2, p. 157–165, 2019.
- SLATE, J.; SHEPPARD, L. Automatic control of blood pressure by drug infusion. **IEE Proceedings A (Physical Science, Measurement and Instrumentation, Management and Education, Reviews)**, IET, v. 129, n. 9, p. 639–645, 1982.
- SLATE, J. et al. A model for design of a blood pressure controller for hypertensive patients. **IFAC proceedings volumes**, Elsevier, v. 12, n. 8, p. 867–874, 1979.
- SOUSA, M. G.; JÚNIOR, O. P. Emergências hipertensivas: epidemiologia, definição e classificação. **Revista Brasileira de Hipertensão**, v. 21, n. 3, p. 134–139, 2014.
- RAO SRIPADA, N.; GRANT FISHER, D. Improved least squares identification. **International Journal of Control**, Taylor & Francis, v. 46, n. 6, p. 1889–1913, 1987.
- STEINMETZ, M. J. **Model reference adaptive control of blood pressure**. Boca Raton, Florida, Estados Unidos da América: Florida Atlantic University, 1987.
- TAO, G. **Adaptive control design and analysis**. [S.l.]: John Wiley & Sons, 2003. v. 37.
- VRANČIĆ, D. et al. Model-free vrft-based tuning method for pid controllers. **Mathematics**, MDPI, v. 11, n. 3, p. 715, 2023.
- WESTENSKOW, D. R. Closed-loop control of blood pressure, ventilation, and

anesthesia delivery. **International journal of clinical monitoring and computing**, Springer, v. 4, n. 2, p. 69–74, 1987.

WITTENMARK, B.; ÅSTRÖM, K. J. Practical issues in the implementation of self-tuning control. **Automatica**, Elsevier, v. 20, n. 5, p. 595–605, 1984.

YUAN, T. et al. New insights into oxidative stress and inflammation during diabetes mellitus-accelerated atherosclerosis. **Redox biology**, Elsevier, v. 20, p. 247–260, 2019.

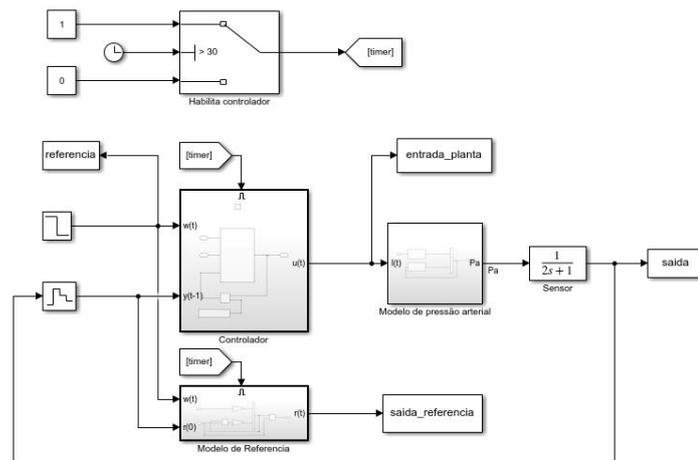
ZHU, K.; ZHENG, H.; LAVANYA, J. An adaptive pi controller for regulation of pressure of hypertension patients. In: **IEEE International Conference on Automation Science and Engineering, 2005**. Edmonton, AB, Canada: Anais, 2005. p. 67–72.

ZHU, K.; ZHENG, H.; LAVANYA, J. Adaptive pi regulation of blood pressure of hypertension patients. In: **2005 IEEE Engineering in Medicine and Biology 27th Annual Conference**. Shanghai, China: Anais, 2006. p. 5536–5539.

APÊNDICE A - DIAGRAMAS DO AMBIENTE DE SIMULAÇÃO

As Figuras 23-35 apresentam o ambiente de simulação utilizado nesse trabalho, sendo composto por diagramas Simulink. O diagrama da Figura 23 é o principal diagrama, condensando as interações da malha de controle.

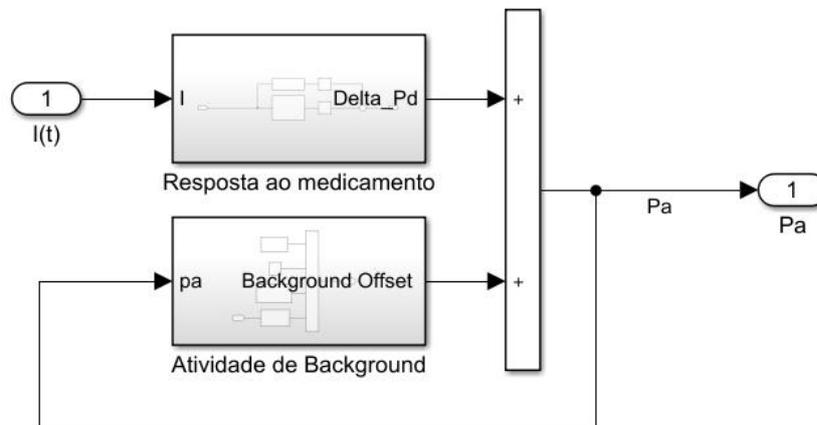
Figura 23 – Diagrama Principal do Ambiente de Simulação



Fonte: Autor (2023).

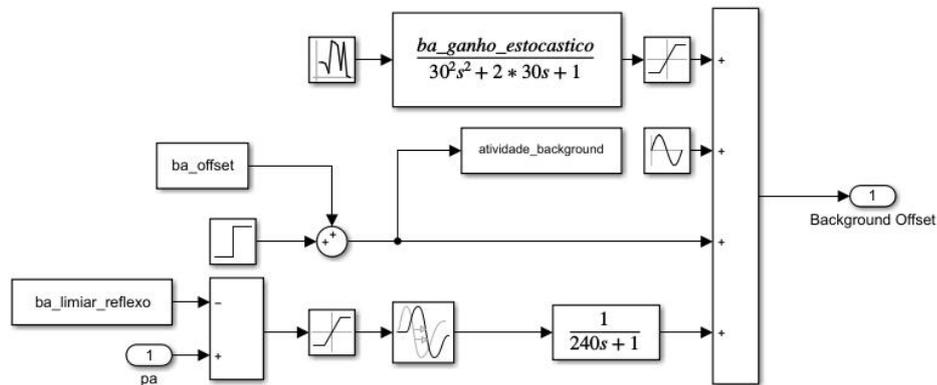
Os diagramas das Figuras 24-26 definem o modelo de pressão arterial, sendo compostos pela atividade de background e resposta ao medicamento.

Figura 24 – Diagrama do Modelo de Pressão Arterial



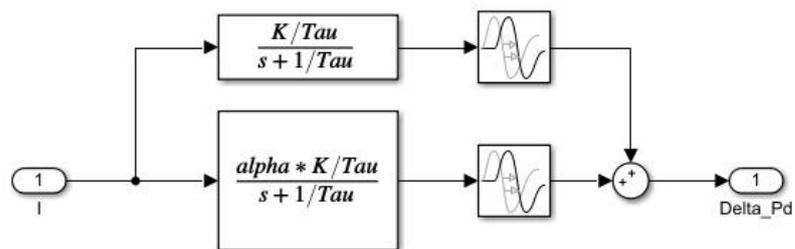
Fonte: Autor (2023).

Figura 25 – Diagrama da Atividade de Background do Modelo de Pressão Arterial



Fonte: Autor (2023).

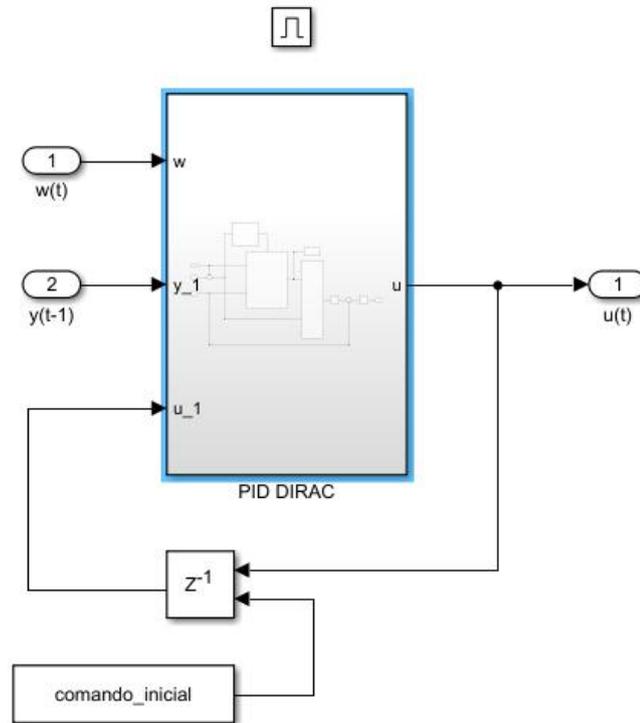
Figura 26 – Diagrama da Resposta ao Medicamento do Modelo de Pressão Arterial



Fonte: Autor (2023).

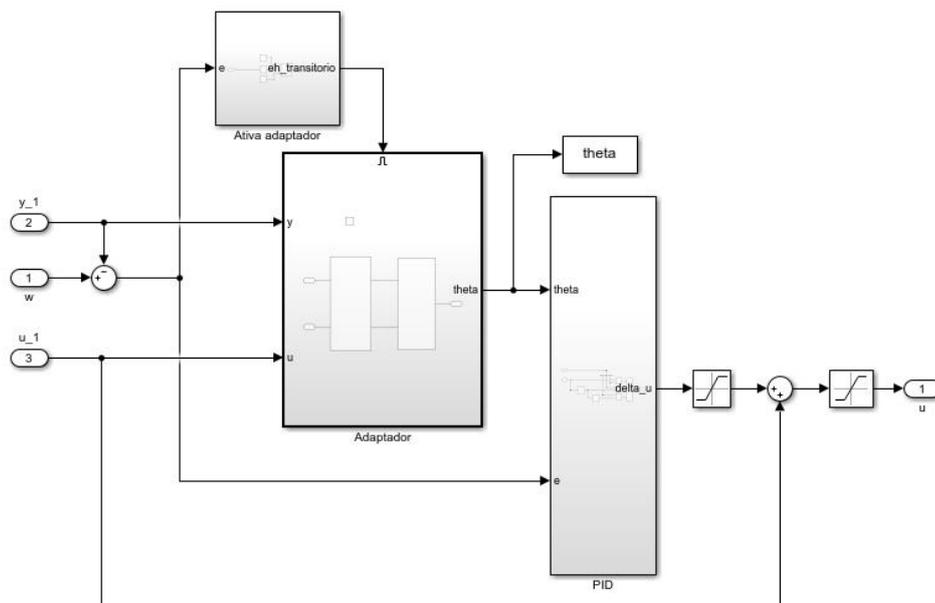
Os diagramas das Figuras 27-35 ilustram as estruturas do controlador, composto pela lógica de ativação, filtragem dos sinais, algoritmo estimador e computação e saturação do sinal de controle.

Figura 27 – Diagrama Principal do Controlador



Fonte: Autor (2023).

Figura 28 – Diagrama Expandido do PID Dirac

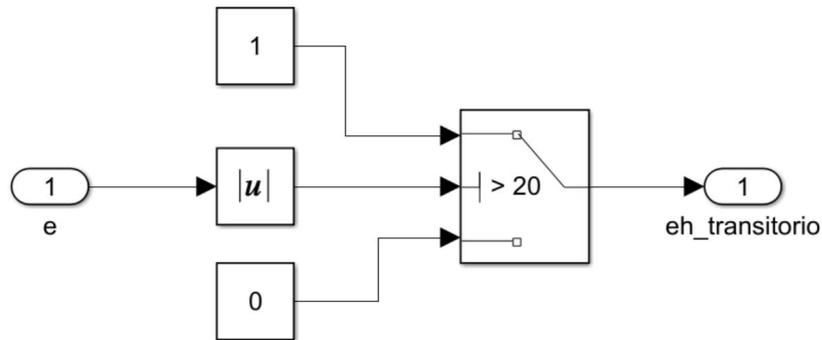


Fonte: Autor (2023).

O dispositivo lógico da Figura 29 identifica o momento de acomodação segundo o sinal

de erro, tomando nível lógico alto caso o sinal esteja fora da região considerada como de acomodação e, portanto, desligando a adaptação.

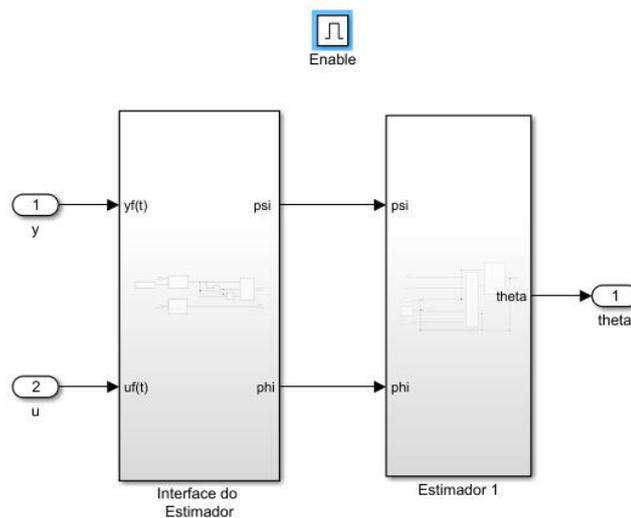
Figura 29 – Diagrama do Mecanismos de Identificação de Acomodação e Ativação do Controlador



Fonte: Autor (2023).

A Figura 30 apresenta a distinção entre a formatação das entradas e processamento do algoritmo estimador, sendo o primeiro dado pelo algoritmo de controle DIRAC e o segundo pela formulação do algoritmo RLS.

Figura 30 – Diagrama Principal do Adaptador

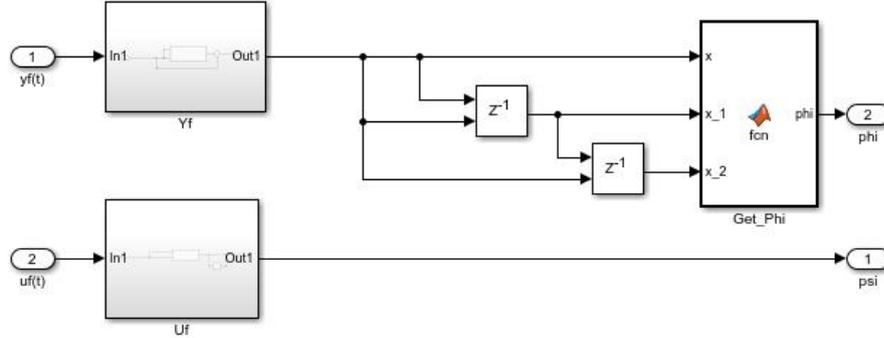


Fonte: Autor (2023).

O bloco de interface do estimador é responsável pela filtragem dos sinais, tal que

resultem nos sinais formatados para uso do estimador.

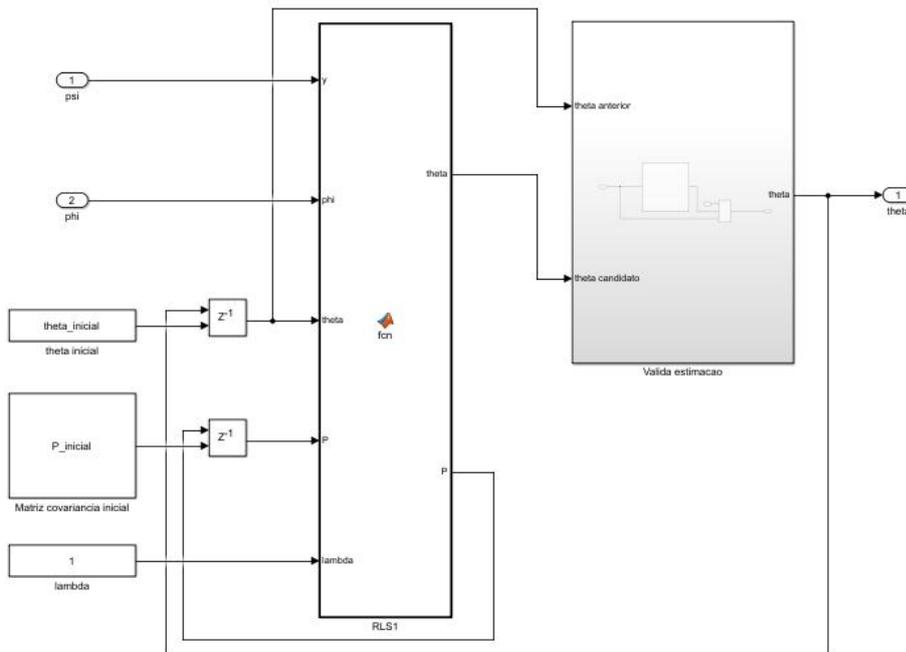
Figura 31 – Diagrama da Filtragem de Sinais para Estimação



Fonte: Autor (2023).

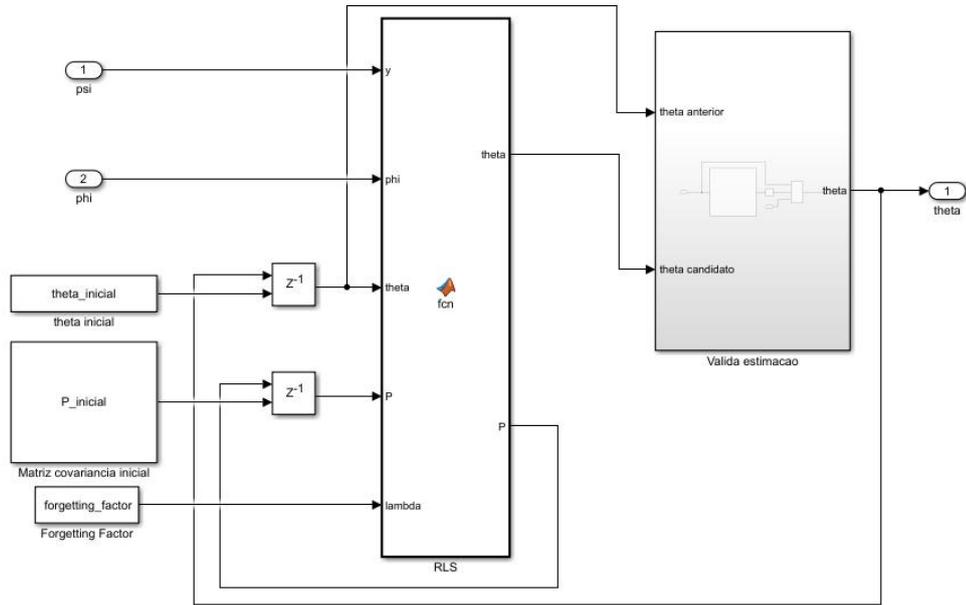
Já as variações de controlador avaliadas no trabalho são diferenciadas alternando o bloco do *Estimador 1* no diagrama da Figura 30 pelos relativos aos demais Estimadores.

Figura 32 – Diagrama do Estimador 1



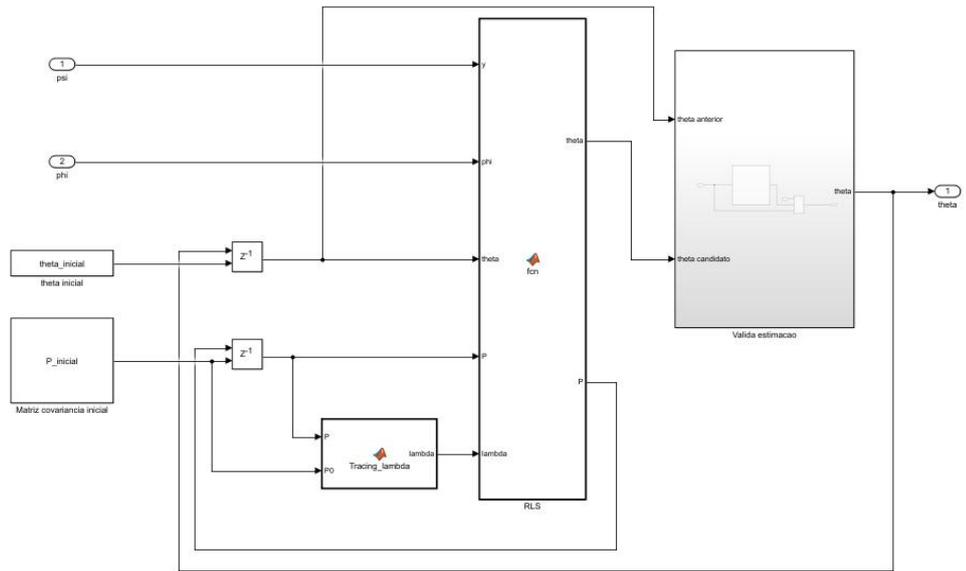
Fonte: Autor (2023).

Figura 33 – Diagrama do Estimador 2



Fonte: Autor (2023).

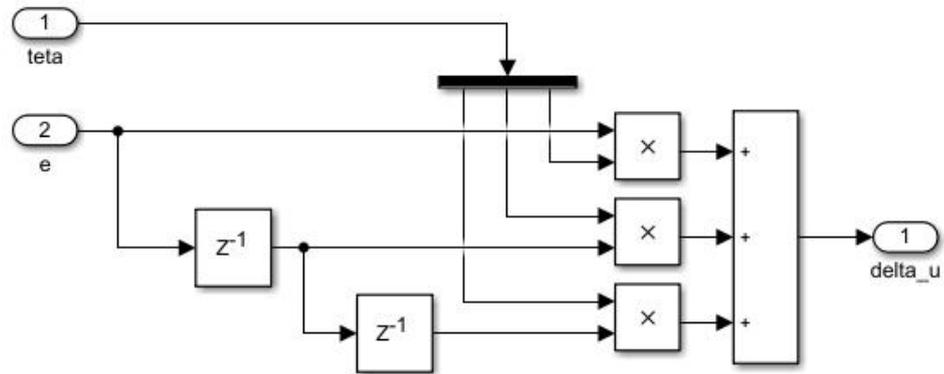
Figura 34 – Diagrama do Estimador 3



Fonte: Autor (2023).

A Figura 35 computa o sinal incremental de controle dado pelos sinais de erro ponderados pelos coeficientes PID estimados.

Figura 35 – Diagrama da Operação Recursiva do PID com Coeficientes Estimados



Fonte: Autor (2023).

Assim, esses diagramas resumem de forma modularizada a dinâmica da planta, algoritmo adaptativo e geração do sinal de controle.

APÊNDICE B - SCRIPTS DO AMBIENTE DE SIMULAÇÃO

Já os códigos abaixo são usados para definir os parâmetros de ambiente e parâmetros do algoritmo além de capturar e processar seus dados das varias simulações. Para isso, os código abaixo são usados respectivamente para a análise sem perturbação, análise de robustez e geração de gráfico de exemplo:

```

1
2 %----- CONFIGURACOES
3
4 ArquivoResultados = 'Analise_sem_perturbacao.xlsx';
5
6 % Configuracoes gerais da simulacao
7 simuTime = 6*60*60; % Tempo de simulacão
8 T = 15; % Tempo de amostragem do controlador
9 Instante_habilita_controlador = 30; % Instante apos transitorio do modelo
10 Amplitude_perturbacao = 0; % Valor somado ao offset
11 Instante_habilita_perturbacao = 3*60*60;
12 setpoint_inicial = 150; % Referencia inicial
13 setpoint_final = 120; % Referencia final 140 ~ 100
14 Criterio_Acomodacao = 20; % Criterio para definir momento de acomodacao
15 n_amostras = 1000; % Configuracoes de afericao estatistica
16
17
18 % Configuracoes do controlador DIRAC
19 Constante_tempo_referencia = 50*3; % Constante de tempo da malha de referencia
20 Tempo_de_acomodacao = Constante_tempo_referencia*4; %Tempo de acomodacao do modelo de
    referencia
21 Tempo_Segundo_Step = 30*60; % Referencia de controle
22 ar = exp(-T/Constante_tempo_referencia); % Parametro de velocidade da malha de referencia
23 d = 5; % Parametro de atraso da malha de referencia , superestimado em 1(multiplo de h nominal)
    ;
24 saida_inicial_maxima = 0; % Estimacao de saida maxima inicial da planta
25 comando_inicial = 0; % Comando inicial do controlador
26
27 % Configuracoes do estimador
28 P_inicial = [0.01 0 0; 0 0.001 0; 0 0 1]; % Matriz Covariancia inicial
29 T_forgetting = Tempo_de_acomodacao; % Horizonte de tempo de memoria (Estimador 1 2)
30 forgetting_factor = exp(-T/T_forgetting); % Fator de esquecimento (Estimador 1 2)
31
32 % Lambda Tuning para
33 K_lambdaTuning = 9; % K = 0.25 ~ 9
34 L_lambdaTuning = 60; % L = 20 ~60
35 Tau_lambdaTuning = 75; % Tau = 30 ~ 75
36 Tcl_lambdaTuning = Constante_tempo_referencia;
37
38 Kp_inicial = -0.01*Tau_lambdaTuning/(K_lambdaTuning*(Tau_lambdaTuning+Tcl_lambdaTuning));
39 Ti_inicial = 0.01*Tau_lambdaTuning;
40 Td_inicial = 0;
41
42 % Calculo de configuracoes dos algoritmos de controle
43 c0_inicial = Kp_inicial*(1+T/Ti_inicial+Td_inicial/T);
44 c1_inicial = -Kp_inicial*(1+2*(Td_inicial/T));
45 c2_inicial = Kp_inicial*(Td_inicial/T);
46 theta_inicial = [c0_inicial; c1_inicial; c2_inicial];

```

```

47
48 % Configuracoes do modelo
49 Parametros_Paciente = { ...
50     'ID',... % Identificador do paciente
51     'K',... % Ganho de pressao em resposta ao medicamento
52     'Tau',... % Constante de tempo
53     'Tau_i',... % Constante de atraso para transporte do medicamento
54     'Tau_c',... % Constante de atraso por recirculaÃ§Ã£o
55     'alpha',... % Fracao de recirculacao
56     'ba_frequencia_seno',... % Frequencia da senoide de background
57     'ba_offset',... % PressÃ£o arterial de offset
58     'ba_amplitude_seno',... % Amplitude da senoidal de background
59     'ba_ganho_estocastico',... % Ganho da atividade estocastica
60     'ba_limiar_reflexo' % Pressao limiar para atividade de reflexo
61 };
62 Pacientes = Get_Pacientes(Parametros_Paciente);
63
64 PSS_amostras_1 = zeros(length(Pacientes),n_amostras);
65 PSS_amostras_2 = zeros(length(Pacientes),n_amostras);
66 PSS_amostras_3 = zeros(length(Pacientes),n_amostras);
67
68 TA_amostras_1 = zeros(length(Pacientes),n_amostras);
69 TA_amostras_2 = zeros(length(Pacientes),n_amostras);
70 TA_amostras_3 = zeros(length(Pacientes),n_amostras);
71
72 ESS_amostras_1 = zeros(length(Pacientes),n_amostras);
73 ESS_amostras_2 = zeros(length(Pacientes),n_amostras);
74 ESS_amostras_3 = zeros(length(Pacientes),n_amostras);
75
76 Resultados_Por_Perfil = zeros(length(Pacientes),9);
77
78 PSS_maximo = zeros(length(Pacientes),3); %1-3
79 TA_maximo = zeros(length(Pacientes),3); %5-6
80 ESS_maximo = zeros(length(Pacientes),3); %7-9
81
82 n_failures_1 = zeros(length(Pacientes),1);
83 n_failures_2 = zeros(length(Pacientes),1);
84 n_failures_3 = zeros(length(Pacientes),1);
85
86 % Graficos para cada paciente
87 for i=1:length(Pacientes)
88
89     %----- DEFINE PACIENTE
90
91     K = Pacientes{i}('K'); % Ganho de pressao em resposta ao medicamento
92     Tau = Pacientes{i}('Tau'); % Constante de tempo
93     Tau_i = Pacientes{i}('Tau_i'); % Constante de atraso para transporte do medicamento
94     Tau_c = Pacientes{i}('Tau_c'); % Constante de atraso por recirculaÃ§Ã£o
95     alpha = Pacientes{i}('alpha'); % Fracao de recirculacao
96     ba_frequencia_seno = Pacientes{i}('ba_frequencia_seno'); % Frequencia da senoide de
97         background
98     ba_offset = Pacientes{i}('ba_offset'); % PressÃ£o arterial de offset
99     ba_amplitude_seno = Pacientes{i}('ba_amplitude_seno'); % Amplitude da senoidal de
100         background
101     ba_ganho_estocastico = Pacientes{i}('ba_ganho_estocastico'); % Ganho da atividade
102         estocastica
103     ba_limiar_reflexo = Pacientes{i}('ba_limiar_reflexo'); % Pressao limiar para atividade de
104         reflexo

```

```

102   for j=1:n_amostras
103
104       %----- SIMULA PACIENTE E COLETA SINAIS
105       semente_ruido = j;
106
107       simu_Controlador_1 = sim('Verificacao_Controlo_Pressao_1','StopTime', sprintf('%.6f',
108           simuTime));
109       saida_planta_1 = simu_Controlador_1.saida;
110       entrada_planta_1 = simu_Controlador_1.entrada_planta;
111       saida_referencia_1 = simu_Controlador_1.saida_referencia;
112       theta_1 = simu_Controlador_1.theta;
113       perturbacao_1 = simu_Controlador_1.perturbacao;
114       referencia_1 = simu_Controlador_1.referencia;
115
116       simu_Controlador_2 = sim('Verificacao_Controlo_Pressao_2','StopTime', sprintf('%.6f',
117           simuTime));
118       saida_planta_2 = simu_Controlador_2.saida;
119       entrada_planta_2 = simu_Controlador_2.entrada_planta;
120       saida_referencia_2 = simu_Controlador_2.saida_referencia;
121       theta_2 = simu_Controlador_2.theta;
122       perturbacao_2 = simu_Controlador_2.perturbacao;
123       referencia_2 = simu_Controlador_2.referencia;
124
125       simu_Controlador_3 = sim('Verificacao_Controlo_Pressao_3','StopTime', sprintf('%.6f',
126           simuTime));
127       saida_planta_3 = simu_Controlador_3.saida;
128       entrada_planta_3 = simu_Controlador_3.entrada_planta;
129       saida_referencia_3 = simu_Controlador_3.saida_referencia;
130       theta_3 = simu_Controlador_3.theta;
131       perturbacao_3 = simu_Controlador_3.perturbacao;
132       referencia_3 = simu_Controlador_3.referencia;
133
134       %----- AVALIA DESEMPENHO
135       [PSS_1,TA_1,ESS_1] = Metricas(saida_planta_1.time,saida_planta_1.data,setpoint_final,
136           Criterio_Acomodacao, Instante_habilita_controlador);
137       [PSS_2,TA_2,ESS_2] = Metricas(saida_planta_2.time,saida_planta_2.data,setpoint_final,
138           Criterio_Acomodacao, Instante_habilita_controlador);
139       [PSS_3,TA_3,ESS_3] = Metricas(saida_planta_3.time,saida_planta_3.data,setpoint_final,
140           Criterio_Acomodacao, Instante_habilita_controlador);
141
142       %----- REGISTRA DESEMPENHO DA AMOSTRA
143       PSS_amostras_1(i,j) = PSS_1;
144       TA_amostras_1(i,j) = TA_1;
145       ESS_amostras_1(i,j) = ESS_1;
146
147       PSS_amostras_2(i,j) = PSS_2;
148       TA_amostras_2(i,j) = TA_2;
149       ESS_amostras_2(i,j) = ESS_2;
150
151       PSS_amostras_3(i,j) = PSS_3;
152       TA_amostras_3(i,j) = TA_3;
153       ESS_amostras_3(i,j) = ESS_3;
154
155       if (PSS_1 > 20 || TA_1 > 7200 || TA_1 < 900)
156           n_failures_1(i,1) = n_failures_1(i,1)+1;
157       end
158
159       if (PSS_2 > 20 || TA_2 > 7200 || TA_2 < 900)
160           n_failures_2(i,1) = n_failures_2(i,1)+1;

```

```

155     end
156
157     if (PSS_3 > 20 || TA_3 > 7200 || TA_3 < 900)
158         n_failures_3(i,1) = n_failures_3(i,1)+1;
159     end
160 end
161
162 %----- AVALIA DESEMPENHO DOS CONTROLADORES PARA O PERFIL DE PACIENTE
163
164 PSS_maximo(i,1) = mean(PSS_amostras_1(i,:)) + 2.*std(PSS_amostras_1(i,:));
165 PSS_maximo(i,2) = mean(PSS_amostras_2(i,:)) + 2.*std(PSS_amostras_2(i,:));
166 PSS_maximo(i,3) = mean(PSS_amostras_3(i,:)) + 2.*std(PSS_amostras_3(i,:));
167
168 TA_maximo(i,1) = mean(TA_amostras_1(i,:)) + 2.*std(TA_amostras_1(i,:));
169 TA_maximo(i,2) = mean(TA_amostras_2(i,:)) + 2.*std(TA_amostras_2(i,:));
170 TA_maximo(i,3) = mean(TA_amostras_3(i,:)) + 2.*std(TA_amostras_3(i,:));
171
172 ESS_maximo(i,1) = mean(ESS_amostras_1(i,:)) + 2.*std(ESS_amostras_1(i,:));
173 ESS_maximo(i,2) = mean(ESS_amostras_2(i,:)) + 2.*std(ESS_amostras_2(i,:));
174 ESS_maximo(i,3) = mean(ESS_amostras_3(i,:)) + 2.*std(ESS_amostras_3(i,:));
175
176 Resultados_Por_Perfil(i,1) = PSS_maximo(i,1);
177 Resultados_Por_Perfil(i,2) = PSS_maximo(i,2);
178 Resultados_Por_Perfil(i,3) = PSS_maximo(i,3);
179 Resultados_Por_Perfil(i,4) = TA_maximo(i,1);
180 Resultados_Por_Perfil(i,5) = TA_maximo(i,2);
181 Resultados_Por_Perfil(i,6) = TA_maximo(i,3);
182 Resultados_Por_Perfil(i,7) = ESS_maximo(i,1);
183 Resultados_Por_Perfil(i,8) = ESS_maximo(i,2);
184 Resultados_Por_Perfil(i,9) = ESS_maximo(i,3);
185 end
186
187 fprintf('%2.3f %2.3f %2.3f | %6.3f %6.3f %6.3f | %2.3f %2.3f %2.3f |\n', Resultados_Por_Perfil
188 .')
189 writetable(...
190     array2table(Resultados_Por_Perfil,...
191         'VariableNames',{'PSS_1','PSS_2','PSS_3','TA_1','TA_2','TA_3','ESS_1','ESS_2','ESS_3'}),...
192     ArquivoResultados);
193
194 % Registra ultimo instante fora da faixa percentual definida pelo criterio
195 function ta = Get_TempoAcomodacao(criteria,output, setpoint_final, tempo,
196     instante_habilitacao)
197     sz = length(output);
198     ta = tempo(end);
199     for i = 1:sz
200         if(abs(setpoint_final - output(i)) > criteria && tempo(i) >= instante_habilitacao)
201             ta = tempo(i);
202         end
203     end
204 end
205 % Registra maior pico abaixo do setpoint_final
206 function us = Get_Undershoot(output, setpoint_final, tempo, instante_habilitacao)
207     sz = length(output);
208     us = 0;
209     cruzou = 0;
210     for i = 2:sz

```

```

212     if(setpoint_final > output(i) && setpoint_final <= output(i-1))
213         cruzou = 1;
214     end
215
216     if(cruzou==1 && abs(setpoint_final - output(i)) > us && tempo(i) >= instante_habilitacao
217         )
218         us = abs(setpoint_final - output(i));
219     end
220 end
221
222 % Registra o erro entre referencia e a saída média nos ultimos 10% do tempo
223 function ess = Get_ESS(output , setpoint_final)
224     sz = length(output);
225
226     start = round(sz*0.9);
227     final_output = output(start:end);
228     ess = max(abs(final_output - setpoint_final.* ones(length(final_output),1)));
229 end
230
231 function [us,ta,ess] = Metricas(tempo,saida , setpoint_final , Criterio_Acomodacao ,
232     Instante_habilita_controlador)
233     ess = Get_ESS(saida , setpoint_final);
234     ta = Get_TempoAcomodacao(Criterio_Acomodacao ,saida , setpoint_final , tempo ,
235         Instante_habilita_controlador);
236     us = Get_Undershoot(saida , setpoint_final , tempo , Instante_habilita_controlador);
237 end
238
239 function pacientes = Get_Pacientes(Parametros_Paciente)
240     % Pacientes sem acao de perturbacao
241
242     Paciente_1 = containers.Map(Parametros_Paciente,[1 -0.25 30 20 30 0.4 1.25663706 180 2 10
243         70]);
244     Paciente_2 = containers.Map(Parametros_Paciente,[2 -0.5 34 25 35 0.2 1.25663706 180 2 10
245         70]);
246     Paciente_3 = containers.Map(Parametros_Paciente,[3 -0.75 37 30 40 0.4 1.25663706 180 2 10
247         70]);
248     Paciente_4 = containers.Map(Parametros_Paciente,[4 -1 40 40 45 0.2 1.25663706 180 2 10
249         70]);
250     Paciente_5 = containers.Map(Parametros_Paciente,[5 -2 45 45 55 0.4 1.25663706 180 2 10
251         70]);
252     Paciente_6 = containers.Map(Parametros_Paciente,[6 -4 55 50 60 0.2 1.25663706 180 2 10
253         70]);
254     Paciente_7 = containers.Map(Parametros_Paciente,[7 -6 65 55 65 0.4 1.25663706 180 2 10
255         70]);
256     Paciente_8 = containers.Map(Parametros_Paciente,[8 -8 75 60 75 0.2 1.25663706 180 2 10
257         70]);
258     pacientes = {Paciente_1 , Paciente_2 , Paciente_3 , Paciente_4 , Paciente_5 , Paciente_6 , Paciente_7 ,
259         Paciente_8};
260 end

```

```

1
2 %----- CONFIGURACOES
3
4 ArquivoResultados = 'Analise_Robustez.xlsx';
5
6 % Configuracoes gerais da simulacao
7 simuTime = 6*60*60; % Tempo de simulacao
8 T = 15; % Tempo de amostragem do controlador
9 Instante_habilita_controlador = 30; % Instante apos transitorio do modelo
10 Amplitude_perturbacao = 25; % Valor somado ao offset
11 Instante_habilita_perturbacao = 3*60*60;
12 setpoint_inicial = 150; % Referencia inicial
13 setpoint_final = 120; % Referencia final 140 ~ 100
14 Criterio_Acomodacao = 15; % Criterio para definir momento de acomodacao
15 n_amostras = 1000; % Configuracoes de afericao estatistica
16 Instante_habilita_analise = 20000;
17
18 % Configuracoes do controlador DIRAC
19 Constante_tempo_referencia = 50*3; % Constante de tempo da malha de referencia (Tau_max*3)
20 Tempo_de_acomodacao = Constante_tempo_referencia*4; %Tempo de acomodacao do modelo de
    referencia
21 Tempo_Segundo_Step = 30*60; % Referencia de controle
22 ar = exp(-T/Constante_tempo_referencia); % Parametro de velocidade da malha de referencia
23 d = 5; % Parametro de atraso da malha de referencia , superestimado em 1(multiplo de h nominal)
    ;
24 saida_inicial_maxima = 0; % Estimacao de saida maxima inicial da planta
25 comando_inicial = 0; % Comando inicial do controlador
26
27 % Configuracoes do estimador
28 P_inicial = [0.01 0 0; 0 0.001 0 ; 0 0 1]; % Matriz Covariancia inicial
29 T_forgetting = Tempo_de_acomodacao; % Horizonte de tempo de memoria (Estimador 1 2)
30 forgetting_factor = exp(-T/T_forgetting); % Fator de esquecimento (Estimador 1 2)
31
32 % Lambda Tuning para
33 K_lambdaTuning = 9; % K = 0.25 ~ 9
34 L_lambdaTuning = 60; % L = 20 ~60
35 Tau_lambdaTuning = 75; % Tau = 30 ~ 75
36 Tcl_lambdaTuning = Constante_tempo_referencia;
37
38 Kp_inicial = -Tau_lambdaTuning*0.001/(K_lambdaTuning*(Tau_lambdaTuning+Tcl_lambdaTuning));
39 Ti_inicial = Tau_lambdaTuning*0.001;
40 Td_inicial = 0;
41
42 % Calculo de configuracoes dos algoritmos de controle
43 c0_inicial = Kp_inicial*(1+T/Ti_inicial+Td_inicial/T);
44 c1_inicial = -Kp_inicial*(1+2*(Td_inicial/T));
45 c2_inicial = Kp_inicial*(Td_inicial/T);
46 theta_inicial = [c0_inicial; c1_inicial; c2_inicial];
47
48 % Configuracoes do modelo
49 Parametros_Paciente = {...
50     'ID',... % Identificador do paciente
51     'K',... % Ganho de pressao em resposta ao medicamento
52     'Tau',... % Constante de tempo
53     'Tau_i',... % Constante de atraso para transporte do medicamento
54     'Tau_c',... % Constante de atraso por recirculacao
55     'alpha',... % Fracao de recirculacao
56     'ba_frequencia_seno',... % Frequencia da senoide de background

```

```

57     'ba_offset',... % Pressão arterial de offset
58     'ba_amplitude_seno',... % Amplitude da senoidal de background
59     'ba_ganho_estocastico',... % Ganho da atividade estocastica
60     'ba_limiar_reflexo' % Pressao limiar para atividade de reflexo
61 };
62 Pacientes = Get_Pacientes(Parametros_Paciente);
63
64 Estd_amostras_1 = zeros(length(Pacientes),n_amostras);
65 Estd_amostras_2 = zeros(length(Pacientes),n_amostras);
66 Estd_amostras_3 = zeros(length(Pacientes),n_amostras);
67
68 TR_amostras_1 = zeros(length(Pacientes),n_amostras);
69 TR_amostras_2 = zeros(length(Pacientes),n_amostras);
70 TR_amostras_3 = zeros(length(Pacientes),n_amostras);
71
72 SM_amostras_1 = zeros(length(Pacientes),n_amostras);
73 SM_amostras_2 = zeros(length(Pacientes),n_amostras);
74 SM_amostras_3 = zeros(length(Pacientes),n_amostras);
75
76 Resultados_Por_Perfil = zeros(length(Pacientes),9);
77
78 % Graficos para cada paciente
79 for i=1:length(Pacientes)
80
81     %----- DEFINE PACIENTE
82
83     K = Pacientes{i}('K'); % Ganho de pressao em resposta ao medicamento
84     Tau = Pacientes{i}('Tau'); % Constante de tempo
85     Tau_i = Pacientes{i}('Tau_i');5; % Constante de atraso para transporte do medicamento
86     Tau_c = Pacientes{i}('Tau_c'); % Constante de atraso por recirculaçã
87     alpha = Pacientes{i}('alpha'); % Fracao de recirculacao
88     ba_frequencia_seno = Pacientes{i}('ba_frequencia_seno'); % Frequencia da senoide de
89         background
90     ba_offset = Pacientes{i}('ba_offset'); % Pressão arterial de offset
91     ba_amplitude_seno = Pacientes{i}('ba_amplitude_seno'); % Amplitude da senoidal de
92         background
93     ba_ganho_estocastico = Pacientes{i}('ba_ganho_estocastico'); % Ganho da atividade
94         estocastica
95     ba_limiar_reflexo = Pacientes{i}('ba_limiar_reflexo'); % Pressao limiar para atividade de
96         reflexo
97
98     for j=1:n_amostras
99
100        %----- SIMULA PACIENTE E COLETA SINAIS
101        semente_ruido = j;
102
103        simu_Controlador_1 = sim('Verificacao_Controlo_Pressao_1','StopTime', sprintf('%6f',
104            simuTime));
105        saida_planta_1 = simu_Controlador_1.saida;
106        entrada_planta_1 = simu_Controlador_1.entrada_planta;
107        saida_referencia_1 = simu_Controlador_1.saida_referencia;
108        theta_1 = simu_Controlador_1.theta;
109        referencia_1 = simu_Controlador_1.referencia;
110        atividade_background_1 = simu_Controlador_1.atividade_background;
111
112        simu_Controlador_2 = sim('Verificacao_Controlo_Pressao_2','StopTime', sprintf('%6f',
113            simuTime));
114        saida_planta_2 = simu_Controlador_2.saida;
115        entrada_planta_2 = simu_Controlador_2.entrada_planta;

```

```

110   saida_referencia_2 = simu_Controlador_2.saida_referencia;
111   theta_2 = simu_Controlador_2.theta;
112   referencia_2 = simu_Controlador_2.referencia;
113   atividade_background_2 = simu_Controlador_2.atividade_background;
114
115   simu_Controlador_3 = sim('Verificacao_Controlo_Pressao_3', 'StopTime', sprintf('%0.6f',
      simuTime));
116   saida_planta_3 = simu_Controlador_3.saida;
117   entrada_planta_3 = simu_Controlador_3.entrada_planta;
118   saida_referencia_3 = simu_Controlador_3.saida_referencia;
119   theta_3 = simu_Controlador_3.theta;
120   referencia_3 = simu_Controlador_3.referencia;
121   atividade_background_3 = simu_Controlador_3.atividade_background;
122
123   %----- AVALIA DESEMPENHO
124
125   [Estd_1, TR_1, SM_1] = Metricas(saida_planta_1.time, ...
126                               saida_planta_1.data, ...
127                               entrada_planta_1.data, ...
128                               setpoint_final, ...
129                               Criterio_Acomodacao, ...
130                               Amplitude_perturbacao, ...
131                               Instante_habilita_perturbacao, ...
132                               Instante_habilita_analise);
133
134   [Estd_2, TR_2, SM_2] = Metricas(saida_planta_2.time, ...
135                               saida_planta_2.data, ...
136                               entrada_planta_2.data, ...
137                               setpoint_final, ...
138                               Criterio_Acomodacao, ...
139                               Amplitude_perturbacao, ...
140                               Instante_habilita_perturbacao, ...
141                               Instante_habilita_analise);
142   [Estd_3, TR_3, SM_3] = Metricas(saida_planta_3.time, ...
143                               saida_planta_3.data, ...
144                               entrada_planta_3.data, ...
145                               setpoint_final, ...
146                               Criterio_Acomodacao, ...
147                               Amplitude_perturbacao, ...
148                               Instante_habilita_perturbacao, ...
149                               Instante_habilita_analise);
150
151   %----- REGISTRA DESEMPENHO DA AMOSTRA
152   Estd_amostras_1(i, j) = Estd_1;
153   Estd_amostras_2(i, j) = Estd_2;
154   Estd_amostras_3(i, j) = Estd_3;
155
156   TR_amostras_1(i, j) = TR_1;
157   TR_amostras_2(i, j) = TR_2;
158   TR_amostras_3(i, j) = TR_3;
159
160   SM_amostras_1(i, j) = SM_1;
161   SM_amostras_2(i, j) = SM_2;
162   SM_amostras_3(i, j) = SM_3;
163
164   end
165
166   %----- AVALIA DESEMPENHO DOS CONTROLADORES PARA O PERFIL DE PACIENTE
167   Resultados_Por_Perfil(i, 1) = mean(Estd_amostras_1(i, :));

```

```

168     Resultados_Por_Perfil(i,2) = mean(Estd_amostras_2(i,:));
169     Resultados_Por_Perfil(i,3) = mean(Estd_amostras_3(i,:));
170     Resultados_Por_Perfil(i,4) = mean(TR_amostras_1(i,:));
171     Resultados_Por_Perfil(i,5) = mean(TR_amostras_2(i,:));
172     Resultados_Por_Perfil(i,6) = mean(TR_amostras_3(i,:));
173     Resultados_Por_Perfil(i,7) = mean(SM_amostras_1(i,:));
174     Resultados_Por_Perfil(i,8) = mean(SM_amostras_2(i,:));
175     Resultados_Por_Perfil(i,9) = mean(SM_amostras_3(i,:));
176 end
177
178 fprintf('%2.3f %2.3f %2.3f | %6.3f %6.3f %6.3f | %3.3f %3.3f %3.3f\n', Resultados_Por_Perfil
    .')
179
180 figure(1);
181 plot(saida_planta_1.time, saida_planta_1.data);
182 figure(2);
183 plot(theta_1.time, theta_1.data);
184 figure(3);
185 plot(atividade_background_1.time, atividade_background_1.data);
186
187
188
189 function [Residual_medio, Tempo_Resposta, Entrada_Media_Estacionaria] = Metricas(tempo, ...
190     saida, ...
191     entrada_planta, ...
192     setpoint, ...
193     Criterio_Acomodacao, ...
194     amplitude_perturbacao, ...
195     Instante_habilita_perturbacao, ...
196     Instante_habilita_analise)
197
198     i_start = find(tempo>Instante_habilita_analise,1,'first');
199     referencia = setpoint.* ones(length(saida(i_start:end)),1);
200     saida_media = mean(saida(i_start:end));
201     Residual = abs(saida_media - referencia)*100/(amplitude_perturbacao);
202     Residual_medio = mean(Residual);
203     Tempo_Resposta = Get_TempoAcomodacao(Criterio_Acomodacao,saida, setpoint, tempo,
        Instante_habilita_perturbacao) - Instante_habilita_perturbacao;
204
205
206     j_start = round((length(entrada_planta)-i_start)*0.9)+i_start;
207     Entrada_Media_Estacionaria = mean(entrada_planta(j_start:end));
208 end
209
210 function ta = Get_TempoAcomodacao(criteria,output, setpoint, tempo, instante_habilitacao)
211     sz = length(output);
212     ta = tempo(end);
213     for i = 1:sz
214         if(abs((setpoint - output(i))/setpoint)*100 > criteria && tempo(i) >=
            instante_habilitacao)
215             ta = tempo(i);
216         end
217     end
218 end
219
220 function pacientes = Get_Pacientes(Parametros_Paciente)
221     % Pacientes sem acao de perturbacao
222
223     Paciente_1 = containers.Map(Parametros_Paciente,[1 -0.25 30 20 30 0.4 1.25663706 180 2 10

```

```
    70]);
224 Paciente_2 = containers.Map(Parametros_Paciente,[2 -0.5 34 25 35 0.2 1.25663706 180 2 10
    70]);
225 Paciente_3 = containers.Map(Parametros_Paciente,[3 -0.75 37 30 40 0.4 1.25663706 180 2 10
    70]);
226 Paciente_4 = containers.Map(Parametros_Paciente,[4 -1 40 40 45 0.2 1.25663706 180 2 10
    70]);
227 Paciente_5 = containers.Map(Parametros_Paciente,[5 -2 45 45 55 0.4 1.25663706 180 2 10
    70]);
228 Paciente_6 = containers.Map(Parametros_Paciente,[6 -4 55 50 60 0.2 1.25663706 180 2 10
    70]);
229 Paciente_7 = containers.Map(Parametros_Paciente,[7 -6 65 55 65 0.4 1.25663706 180 2 10
    70]);
230 Paciente_8 = containers.Map(Parametros_Paciente,[8 -8 75 60 75 0.2 1.25663706 180 2 10
    70]);
231 pacientes = {Paciente_1, Paciente_2, Paciente_3, Paciente_4, Paciente_5, Paciente_6, Paciente_7
    , Paciente_8};
232 end
```

```

1 % Configuracoes gerais da simulacao
2 simuTime = 2*60*60; % Tempo de simulacao
3 T = 15; % Tempo de amostragem do controlador
4 Instante_habilita_controlador = 30; % Instante apos transitorio do modelo
5 Amplitude_perturbacao = 0; % Valor somado ao offset
6 Instante_habilita_perturbacao = 3*60*60;
7 setpoint_inicial = 150; % Referencia inicial
8 setpoint_final = 120; % Referencia final 140 ~ 100
9 Criterio_Acomodacao = 20; % Criterio para definir momento de acomodacao
10
11 % Configuracoes do controlador DIRAC
12 Constante_tempo_referencia = 50*3; % Constante de tempo da malha de referencia (Tau_max*3)
13 Tempo_de_acomodacao = Constante_tempo_referencia*4; %Tempo de acomodacao do modelo de
    referencia
14 Tempo_Segundo_Step = 30*60; % Referencia de controle
15 ar = exp(-T/Constante_tempo_referencia); % Parametro de velocidade da malha de referencia
16 d = 5; % Parametro de atraso da malha de referencia , superestimado em 1(multiplo de h nominal)
    ;
17 saida_inicial_maxima = 0; % Estimacao de saida maxima inicial da planta
18 comando_inicial = 0; % Comando inicial do controlador
19
20 % Configuracoes do estimador
21 P_inicial = [0.01 0 0; 0 0.001 0 ; 0 0 1]; % Matriz Covariancia inicial
22 T_forgetting = Tempo_de_acomodacao; % Horizonte de tempo de memoria (Estimador 1 2)
23 forgetting_factor = exp(-T/T_forgetting); % Fator de esquecimento (Estimador 1 2)
24
25 % Lambda Tuning para
26 K_lambdaTuning = 9; % K = 0.25 ~ 9
27 L_lambdaTuning = 60; % L = 20 ~60
28 Tau_lambdaTuning = 75; % Tau = 30 ~ 75
29 Tcl_lambdaTuning = Constante_tempo_referencia;
30
31 Kp_inicial = -0.01*Tau_lambdaTuning/(K_lambdaTuning*(Tau_lambdaTuning+Tcl_lambdaTuning));
32 Ti_inicial = 0.01*Tau_lambdaTuning;
33 Td_inicial = 0;
34
35 % Calculo de configuracoes dos algoritmos de controle
36 c0_inicial = Kp_inicial*(1+T/Ti_inicial+Td_inicial/T);
37 c1_inicial = -Kp_inicial*(1+2*(Td_inicial/T));
38 c2_inicial = Kp_inicial*(Td_inicial/T);
39 theta_inicial = [c0_inicial; c1_inicial; c2_inicial];
40
41 %----- DEFINE PACIENTE
42 % Configuracoes do modelo
43 Parametros_Paciente = {...
44     'ID',... % Identificador do paciente
45     'K',... % Ganho de pressao em resposta ao medicamento
46     'Tau',... % Constante de tempo
47     'Tau_i',... % Constante de atraso para transporte do medicamento
48     'Tau_c',... % Constante de atraso por recirculacao
49     'alpha',... % Fracao de recirculacao
50     'ba_frequencia_seno',... % Frequencia da senoide de background
51     'ba_offset',... % Pressao arterial de offset
52     'ba_amplitude_seno',... % Amplitude da senoidal de background
53     'ba_ganho_estocastico',... % Ganho da atividade estocastica
54     'ba_limiar_reflexo' % Pressao limiar para atividade de reflexo
55 };
56 Pacientes = Get_Pacientes(Parametros_Paciente);

```

```

57
58 semente_ruido = 294;
59
60 % DIRAC c/ ESTIMADOR 2:
61
62 win_1 = figure(1); % Saida
63 win_1(1) = subplot(3, 1, 1); %Estimador 1
64 win_1(2) = subplot(3, 1, 2); %Estimador 2
65 win_1(3) = subplot(3, 1, 3); %Estimador 3
66 set(win_1, 'Nextplot', 'add');
67
68 win_2 = figure(2); % Theta
69 win_2(1) = subplot(3, 3, 1); % Kp - Paciente 1
70 win_2(2) = subplot(3, 3, 2); % Ti - Paciente 1
71 win_2(3) = subplot(3, 3, 3); % Td - Paciente 1
72 win_2(4) = subplot(3, 3, 4); % Kp - Paciente 2
73 win_2(5) = subplot(3, 3, 5); % Ti - Paciente 2
74 win_2(6) = subplot(3, 3, 6); % Td - Paciente 2
75 win_2(7) = subplot(3, 3, 7); % Kp - Paciente 3
76 win_2(8) = subplot(3, 3, 8); % Ti - Paciente 3
77 win_2(9) = subplot(3, 3, 9); % Td - Paciente 3
78 set(win_2, 'Nextplot', 'add');
79
80 win_3 = figure(3); % Theta
81 win_3(1) = subplot(3, 3, 1); % c0 - Paciente 1
82 win_3(2) = subplot(3, 3, 2); % c1 - Paciente 1
83 win_3(3) = subplot(3, 3, 3); % c2 - Paciente 1
84 win_3(4) = subplot(3, 3, 4); % c0 - Paciente 2
85 win_3(5) = subplot(3, 3, 5); % c1 - Paciente 2
86 win_3(6) = subplot(3, 3, 6); % c2 - Paciente 2
87 win_3(7) = subplot(3, 3, 7); % c0 - Paciente 3
88 win_3(8) = subplot(3, 3, 8); % c1 - Paciente 3
89 win_3(9) = subplot(3, 3, 9); % c2 - Paciente 3
90 set(win_3, 'Nextplot', 'add');
91
92 j =1;
93 for i =1:length(Pacientes)
94
95     Pacientes = Get_Pacientes(Parametros_Paciente);
96     K = Pacientes{i}('K'); % Ganho de pressao em resposta ao medicamento
97     Tau = Pacientes{i}('Tau'); % Constante de tempo
98     Tau_i = Pacientes{i}('Tau_i');5; % Constante de atraso para transporte do medicamento
99     Tau_c = Pacientes{i}('Tau_c'); % Constante de atraso por recirculaÃ§Ã£o
100     alpha = Pacientes{i}('alpha'); % Fracao de recirculacao
101     ba_frequencia_seno = Pacientes{i}('ba_frequencia_seno'); % Frequencia da senoide de
        background
102     ba_offset = Pacientes{i}('ba_offset'); % PressÃ£o arterial de offset
103     ba_amplitude_seno = Pacientes{i}('ba_amplitude_seno'); % Amplitude da senoidal de
        background
104     ba_ganho_estocastico = Pacientes{i}('ba_ganho_estocastico'); % Ganho da atividade
        estocastica
105     ba_limiar_reflexo = Pacientes{i}('ba_limiar_reflexo'); % Pressao limiar para atividade de
        reflexo
106
107     simu_Controlador_1 = sim('Verificacao_Controlo_Pressao_1', 'StopTime', sprintf('%0.6f',
        simuTime));
108     saida_planta_1 = simu_Controlador_1.saida;
109     entrada_planta_1 = simu_Controlador_1.entrada_planta;
110     saida_referencia_1 = simu_Controlador_1.saida_referencia;

```

```

111 theta_1 = simu_Controlador_1.theta;
112 referencia_1 = simu_Controlador_1.referencia;
113
114 simu_Controlador_2 = sim('Verificacao Controle Pressao_2','StopTime', sprintf('%0.6f',
    simuTime));
115 saida_planta_2 = simu_Controlador_2.saida;
116 entrada_planta_2 = simu_Controlador_2.entrada_planta;
117 saida_referencia_2 = simu_Controlador_2.saida_referencia;
118 theta_2 = simu_Controlador_2.theta;
119 referencia_2 = simu_Controlador_2.referencia;
120
121 simu_Controlador_3 = sim('Verificacao Controle Pressao_3','StopTime', sprintf('%0.6f',
    simuTime));
122 saida_planta_3 = simu_Controlador_3.saida;
123 entrada_planta_3 = simu_Controlador_3.entrada_planta;
124 saida_referencia_3 = simu_Controlador_3.saida_referencia;
125 theta_3 = simu_Controlador_3.theta;
126 referencia_3 = simu_Controlador_3.referencia;
127
128
129 plot(win_1(1),saida_planta_1.time, saida_planta_1.data, 'LineWidth',1.25);
130 plot(win_1(2),saida_planta_2.time, saida_planta_2.data, 'LineWidth',1.25);
131 plot(win_1(3),saida_planta_3.time, saida_planta_3.data, 'LineWidth',1.25);
132
133 c0 = theta_2.data(:,1);
134 c1 = theta_2.data(:,2);
135 c2 = theta_2.data(:,3);
136 Kp_2 = -(c1+2*c2);
137 Ki_2 = (c0+c1+c2)./T;
138 Kd_2 = T*c2;
139
140 plot(win_2(i),theta_2.time, Kp_2.', 'k', 'LineWidth',1.25);
141 plot(win_3(i),theta_2.time, c0.', 'k', 'LineWidth',1.25);
142 j = j+1;
143 plot(win_2(i+3),theta_2.time, Ki_2.', 'k', 'LineWidth',1.25);
144 plot(win_3(i+3),theta_2.time, c1.', 'k', 'LineWidth',1.25);
145 j = j+1;
146 plot(win_2(i+6),theta_2.time, Kd_2.', 'k', 'LineWidth',1.25);
147 plot(win_3(i+6),theta_2.time, c2.', 'k', 'LineWidth',1.25);
148 j = j+1;
149 end
150
151 ymax = 200;
152
153 plot(win_1(1),referencia_1.time, referencia_1.data, 'k--', 'LineWidth',1.5);
154 plot(win_1(1),saida_referencia_1.time, saida_referencia_1.data, 'k', 'LineWidth',1.25);
155 legend(win_1(1),{...
156     "Resposta perfil P.1",...
157     "Resposta perfil P.4",...
158     "Resposta perfil P.8",...
159     "Referência", "Saída do modelo de referência"});
160 title(win_1(1),"Controlador DIRAC com Estimador 1");
161 xlabel(win_1(1), 'time (s)');
162 ylabel(win_1(1), 'output');
163 grid(win_1(1), 'on');
164 grid(win_1(1), 'minor');
165 ylim(win_1(1),[60 ymax]);
166 xlim(win_1(1),[0 simuTime]);
167

```

```

168 plot(win_1(2),referencia_2.time, referencia_2.data, 'k--', 'LineWidth',1.5);
169 plot(win_1(2),saida_referencia_2.time, saida_referencia_2.data, 'k', 'LineWidth',1.25);
170 legend(win_1(2),{...
171     "Resposta perfil P.1",...
172     "Resposta perfil P.4",...
173     "Resposta perfil P.8",...
174     "Referência", "Saída do modelo de referência"});
175 title(win_1(2),"Controlador DIRAC com Estimador 2");
176 xlabel(win_1(2), 'time (s)');
177 ylabel(win_1(2), 'output');
178 grid(win_1(2), 'on');
179 grid(win_1(2), 'minor');
180 ylim(win_1(2),[60 ymax]);
181 xlim(win_1(2),[0 simuTime]);
182
183 plot(win_1(3),referencia_3.time, referencia_3.data, 'k--', 'LineWidth',1.5);
184 plot(win_1(3),saida_referencia_3.time, saida_referencia_3.data, 'k', 'LineWidth',1.25);
185 legend(win_1(3),{...
186     "Resposta perfil P.1",...
187     "Resposta perfil P.4",...
188     "Resposta perfil P.8",...
189     "Referência", "Saída do modelo de referência"});
190 title(win_1(3),"Controlador DIRAC com Estimador 3");
191 xlabel(win_1(3), 'time (s)');
192 ylabel(win_1(3), 'output');
193 grid(win_1(3), 'on');
194 grid(win_1(3), 'minor');
195 ylim(win_1(3),[60 ymax]);
196 xlim(win_1(3),[0 simuTime]);
197
198
199 legend(win_2(1), "K_{p}");
200 title(win_2(1), "Estimações para Perfil P.1");
201 xlabel(win_2(1), 'time (s)');
202 ylabel(win_2(1), 'output');
203 grid(win_2(1), 'on');
204 grid(win_2(1), 'minor');
205 xlim(win_2(1),[0 simuTime]);
206
207 legend(win_2(2), "K_{p}");
208 title(win_2(2), "Estimações para Perfil P.4");
209 xlabel(win_2(2), 'time (s)');
210 ylabel(win_2(2), 'output');
211 grid(win_2(2), 'on');
212 grid(win_2(2), 'minor');
213 xlim(win_2(2),[0 simuTime]);
214
215 legend(win_2(3), "K_{p}");
216 title(win_2(3), "Estimações para Perfil P.8");
217 xlabel(win_2(3), 'time (s)');
218 ylabel(win_2(3), 'output');
219 grid(win_2(3), 'on');
220 grid(win_2(3), 'minor');
221 xlim(win_2(3),[0 simuTime]);
222
223 legend(win_2(4), "K_{i}");
224 xlabel(win_2(4), 'time (s)');
225 ylabel(win_2(4), 'output');
226 grid(win_2(4), 'on');

```

```

227 grid(win_2(4), 'minor');
228 xlim(win_2(4), [0 simuTime]);
229
230 legend(win_2(5), "K_{i}");
231 xlabel(win_2(5), 'time (s)');
232 ylabel(win_2(5), 'output');
233 grid(win_2(5), 'on');
234 grid(win_2(5), 'minor');
235 xlim(win_2(5), [0 simuTime]);
236
237 legend(win_2(6), "K_{i}");
238 xlabel(win_2(6), 'time (s)');
239 ylabel(win_2(6), 'output');
240 grid(win_2(6), 'on');
241 grid(win_2(6), 'minor');
242 xlim(win_2(6), [0 simuTime]);
243
244 legend(win_2(7), "K_{d}");
245 xlabel(win_2(7), 'time (s)');
246 ylabel(win_2(7), 'output');
247 grid(win_2(7), 'on');
248 grid(win_2(7), 'minor');
249 xlim(win_2(7), [0 simuTime]);
250
251 legend(win_2(8), "K_{d}");
252 xlabel(win_2(8), 'time (s)');
253 ylabel(win_2(8), 'output');
254 grid(win_2(8), 'on');
255 grid(win_2(8), 'minor');
256 xlim(win_2(8), [0 simuTime]);
257
258 legend(win_2(9), "K_{d}");
259 xlabel(win_2(9), 'time (s)');
260 ylabel(win_2(9), 'output');
261 grid(win_2(9), 'on');
262 grid(win_2(9), 'minor');
263 xlim(win_2(9), [0 simuTime]);
264
265
266 %=====
267 legend(win_3(1), "c_{0}");
268 title(win_3(1), "Estimações para Perfil P.1");
269 xlabel(win_3(1), 'time (s)');
270 ylabel(win_3(1), 'output');
271 grid(win_3(1), 'on');
272 grid(win_3(1), 'minor');
273 xlim(win_3(1), [0 simuTime]);
274
275 legend(win_3(2), "c_{0}");
276 title(win_3(2), "Estimações para Perfil P.4");
277 xlabel(win_3(2), 'time (s)');
278 ylabel(win_3(2), 'output');
279 grid(win_3(2), 'on');
280 grid(win_3(2), 'minor');
281 xlim(win_3(2), [0 simuTime]);
282
283 legend(win_3(3), "c_{0}");
284 title(win_3(3), "Estimações para Perfil P.8");
285 xlabel(win_3(3), 'time (s)');

```

```

286 ylabel(win_3(3), 'output');
287 grid(win_3(3), 'on');
288 grid(win_3(3), 'minor');
289 xlim(win_3(3), [0 simuTime]);
290
291 legend(win_3(4), "c_{1}");
292 xlabel(win_3(4), 'time (s)');
293 ylabel(win_3(4), 'output');
294 grid(win_3(4), 'on');
295 grid(win_3(4), 'minor');
296 xlim(win_3(4), [0 simuTime]);
297
298 legend(win_3(5), "c_{1}");
299 xlabel(win_3(5), 'time (s)');
300 ylabel(win_3(5), 'output');
301 grid(win_3(5), 'on');
302 grid(win_3(5), 'minor');
303 xlim(win_3(5), [0 simuTime]);
304
305 legend(win_3(6), "c_{1}");
306 xlabel(win_3(6), 'time (s)');
307 ylabel(win_3(6), 'output');
308 grid(win_3(6), 'on');
309 grid(win_3(6), 'minor');
310 xlim(win_3(6), [0 simuTime]);
311
312 legend(win_3(7), "c_{2}");
313 xlabel(win_3(7), 'time (s)');
314 ylabel(win_3(7), 'output');
315 grid(win_3(7), 'on');
316 grid(win_3(7), 'minor');
317 xlim(win_3(7), [0 simuTime]);
318
319 legend(win_3(8), "c_{2}");
320 xlabel(win_3(8), 'time (s)');
321 ylabel(win_3(8), 'output');
322 grid(win_3(8), 'on');
323 grid(win_3(8), 'minor');
324 xlim(win_3(8), [0 simuTime]);
325
326 legend(win_3(9), "c_{2}");
327 xlabel(win_3(9), 'time (s)');
328 ylabel(win_3(9), 'output');
329 grid(win_3(9), 'on');
330 grid(win_3(9), 'minor');
331 xlim(win_3(9), [0 simuTime]);
332
333 function pacientes = Get_Pacientes(Parametros_Paciente)
334     % Pacientes sem acao de perturbacao
335     Paciente_1 = containers.Map(Parametros_Paciente, [1 -0.25 30 20 30 0.4 1.25663706 180 2 10
336         70]);
337     Paciente_4 = containers.Map(Parametros_Paciente, [4 -1 40 40 45 0.2 1.25663706 180 2 10
338         70]);
339     Paciente_8 = containers.Map(Parametros_Paciente, [8 -8 75 60 75 0.2 1.25663706 180 2 10
340         70]);
341     pacientes = {Paciente_1, Paciente_4, Paciente_8};
342 end

```