

Fábio Nogueira da Silva

**Observadores de Estado Orientados por
Dados: Sintonia, Programação Dinâmica
Aproximada e Aprendizagem por Reforço**

São Luís

2021

Fábio Nogueira da Silva

**Observadores de Estado Orientados por Dados:
Sintonia, Programação Dinâmica Aproximada e
Aprendizagem por Reforço**

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da UFMA como parte dos requisitos necessários para a obtenção do grau de Doutor em Engenharia Elétrica. Área de concentração: Automação e Controle.

Universidade Federal do Maranhão
Centro de Ciências Exatas e Tecnologia
Programa de Pós-Graduação em Engenharia Elétrica

Orientador: Prof. Dr. João Viana da Fonseca Neto

São Luís

2021

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).
Diretoria Integrada de Bibliotecas/UFMA

Silva, Fábio Nogueira da.

Observadores de Estado Orientados por Dados: Sintonia,
Programação Dinâmica Aproximada e Aprendizagem por Reforço
/ Fábio Nogueira da Silva. - 2021.
134 f.

Orientador(a): João Viana da Fonseca Neto.

Tese (Doutorado) - Programa de Pós-graduação em
Engenharia Elétrica/ccet, Universidade Federal do
Maranhão, São Luís, 2021.

1. Aprendizagem por Reforço. 2. Observadores de
Estado. 3. Programação Dinâmica Aproximada. 4.
Realimentação de Saída. 5. Sintonia. I. Fonseca Neto,
João Viana da. II. Título.

Fábio Nogueira da Silva

Observadores de Estado Orientados por Dados: Sintonia, Programação Dinâmica Aproximada e Aprendizagem por Reforço

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da UFMA como parte dos requisitos necessários para a obtenção do grau de Doutor em Engenharia Elétrica. Área de concentração: Automação e Controle.

Trabalho submetido. São Luís, 01 de Junho de 2020:

Prof. Dr. João Viana da Fonseca Neto
Orientador

Prof. Dr. Ginalber Luiz de Oliveira Serra
Examinador Interno

Prof. Dr. Francisco das Chagas de Souza
Examinador Interno

Prof. Dr. Walter Barra Junior
Examinador Externo

Prof. Dr. Antônio Da Silva Silveira
Examinador Externo

São Luís
2021

Aos meus pais Bonifácio José da Silva e Maria do Socorro Nogueira da Silva (in memoriam), a minha irmã, Karine Nogueira da Silva e a minha companheira, Ivanilda Cristina Teixeira da Costa, que sempre torceram, me incentivaram e apoiaram, as minhas escolhas, ao longo de minha vida e ao longo do desenvolvimento desta Tese.

Agradecimentos

Agradeço primeiramente a Deus por permitir com que este trabalho seja concluído.

Agradeço ao Prof. Dr. João Viana da Fonseca Neto pelo incentivo, confiança, amizade, conselhos, orientação, apoio e pelas diversas discussões ao longo de todo o período de construção deste trabalho de pesquisa ao longo desses anos de trabalho e por sempre me estimular a avaliar diferentes pontos de vista, com o propósito de formação não só acadêmica como também intelectual e cultural.

Agradeço também ao Prof. Ph.D Allan Kardec Duailibe Barros Filho, pelo incentivo, apoio e orientação acadêmica ao longo de minha formação e por ter apresentado ao Prof. Ph.D José Carlos Príncipe, que agradeço imensamente pelo acolhimento, suporte e discussões científicas, como meu có-orientador na Universidade da Florida, no período do Doutorado Sanduíche nos Estados Unidos. Também agradeço aos membros do Computational NeuroEngineering Laboratory - CNEL na Universidade da Florida, que me receberam bem e pelos debates e conversas nas reuniões e discussões sobre diversas questões acadêmicas, de projetos e filosóficas.

Agradeço também a todos os participantes do LAC e LABSECI pelas discussões construtivas, com visões diversas sobre vários aspectos, proporcionando um ambiente frutífero de ideias.

Agradeço a todos os funcionários da UFMA em geral e especialmente a coordenação de pós-graduação. Aos professores que tanto ajudaram em minha formação crítica e científica ao longo destes anos nas disciplinas cursadas.

Agradeço aos professores que participaram da minha qualificação e proposta de tese, pelas contribuições e críticas sobre o meu trabalho, pois, com elas foram possíveis melhorar a qualidade do trabalho produzido e proporcionar uma formação científica mais qualificada.

Resumo

Formulações para observadores de estado para sistemas dinâmicos, baseados nos fundamentos de programação dinâmica aproximada (ADP), controle ótimo e aprendizagem por reforço são propostos, desenvolvidos, aplicadas e analisadas nesta Tese. São ainda apresentadas propostas de algoritmos, métricas para avaliação de desempenho, análises de robustez, convergência e solvabilidade. Estudos sobre sensibilidade paramétrica dos algoritmos, com relação aos sinais de ruídos, condições iniciais dos parâmetros e estados iniciais do sistema dinâmico são apresentadas. A fundamentação para os observadores propostos, são baseados em programação dinâmica aproximada, com aproximação da função valorada realizada por um algoritmo de aprendizagem por reforço (RL), usando o erro de diferenças temporais, visando o acoplamento dos observadores para aplicações on-line, podendo também ser implementados off-line. A formulação do observador é fundamentada no problema de controle ótimo discreto, associado ao regulador linear quadrático (DLQR) com realimentação de saída, necessitando somente dos sinais de entrada e saídas mensuradas. Para estimação de estado com estrutura baseada em ADP, é necessário a disponibilidade de duas matrizes, sendo proposta uma formulação que resulta em um sistema de equações algébricas não lineares para recuperação das matrizes. Para a solução desse problema, são aplicados inicialmente uma rede neuronal direta multicamada, mas por possuir uma complexidade computacional elevada ao longo do processo iterativo, tal solução se verificou inviável. Uma alternativa baseada em uma aproximação é proposta, não sendo necessário solucionar o sistema de equações e com isso diminuindo a complexidade computacional. Para avaliar o desempenho dos algoritmos são propostos métricas de erro, visto que os algoritmos possuem diversos parâmetros sintonizáveis. Para facilitar o processo de sintonia e análise são construídas superfícies de erros com variações dos parâmetros, a fim de observar as sensibilidades paramétricas no algoritmo com relação as métricas de erro e avaliar a solvabilidade e convergência, facilitando o processo de sintonia do observador. A aplicação das metodologias propostas apresenta vantagens como a não necessidade de modelagem ou identificação do sistema dinâmico, da incorporação de mudanças de dinâmica pelo uso de abordagens baseadas em aprendizagem por reforço, além do auxílio no processo de sintonia e análise.

Palavras-chave: Observadores de Estado, Programação Dinâmica Aproximada, Aprendizagem por Reforço, Controle Ótimo, Realimentação de Saída, Sistemas Dinâmicos, Sintonia.

Abstract

Formulations for state observers for dynamical systems, based on the fundamentals of approximate dynamic programming (ADP), optimal control and reinforcement learning are proposed, developed, applied and analyzed in this Thesis. Algorithm proposals, metrics for performance evaluation, robustness, convergence and solvability analysis are also presented. Studies on parametric sensitivity of the algorithms, with respect to noise signals, initial conditions of parameters and initial states of the dynamic system are presented. The rationale for the proposed observers is based on approximate dynamic programming, with approximation of the valued function performed by a reinforcement learning algorithm (RL), using the temporal differences errors, aiming at the coupling of observers for online applications, being able to also be implemented offline. The observer's formulation is based on the discrete optimal control problem, associated with the discrete linear quadratic regulator (DLQR) with output feedback, requiring only the measured input and output signals. For state estimation with ADP-based structure, the availability of two matrices is necessary, and a formulation is proposed that results in a system of nonlinear algebraic equations for matrix recovery. To solve this problem, a feedforward multi layer neural network is initially applied, but due to its high computational complexity throughout the iterative process, such a solution was found to be unfeasible. An alternative based on an approximation is proposed, not being necessary to solve the system of equations and thus reducing the computational complexity. To evaluate the performance of the algorithms, error metrics are proposed, since the algorithms have several tunable parameters. To facilitate the tuning and analysis process, error surfaces are constructed with parameter variations, in order to observe the parametric sensitivities in the algorithm in relation to the error metrics and to evaluate the solvability and convergence, facilitating the observer tuning process. The application of the proposed methodologies has advantages such as the lack of modeling or dynamical system identification, the incorporation of dynamic changes through the use of approaches based on reinforcement learning, in addition to helping in the tuning and analysis process.

Keywords: State Observers, Approximate Dynamic Programming, Reinforcement Learning, Optimal Control, Output Feedback, Dynamical Systems, Tuning.

Lista de ilustrações

Figura 1 – Características Gerais para Observadores Lineares e Não Lineares	21
Figura 2 – Morfismo, Isomorfismo e Difeomorfismo entre Espaços de Estados e Observação	22
Figura 3 – Abordagens Não baseadas em Modelos	30
Figura 4 – Função de saída de um Sistema Fracamente Observável, mas não localmente.	40
Figura 5 – Esquema ADP-RL para Reconstrução dos Estados	67
Figura 6 – Gerador de Política Admissível	85
Figura 7 – Heurística Geral para Sintonia	86
Figura 8 – Erro Total para variância do ruído entre 0 e 3, com intervalos de 0.002 . . .	90
Figura 9 – Erro para variação do fator de desconto entre $\gamma = 0.01$ e 1, com intervalos de 0.01.	91
Figura 10 – Estados Estimados x_{1est} e x_{2est} e saída estimada y_{est} , usando aproximação proposta.	92
Figura 11 – Erro Estimação Estados OPFB	93
Figura 12 – Erro Médio Quadrático para 1000 Simulações.	95
Figura 13 – Erro Absoluto Total para 1000 Simulações.	95
Figura 14 – Erro da estimação dos coeficiente estimados pelo <i>RLS</i>	96
Figura 15 – Coeficientes estimados pelo <i>RLS</i>	97
Figura 16 – Superfície de Erro para Variação do Fator de Desconto e Coeficiente xk , que determina os estados iniciais.	98
Figura 17 – Superfície de Erro para Variação do Fator de Desconto e parâmetro xk	99
Figura 18 – Superfície de Erro para Variação do Fator de Desconto e Estados Iniciais entre -1000 e 1000	100
Figura 19 – Superfície de Erro para Variação do Fator de Desconto e $prls_k$	101
Figura 20 – Superfície de Erro para Variação do Fator de Desconto e Variância do Ruído.	102
Figura 21 – Superfície de Erro com Variação do Fator de Desconto entre 0.01 e 0.3 e ruído 0.1 até 4.	103

Figura 22 – Superfície de Erro com Variação do Fator de Desconto entre 0.01 e 0.35 e ruído 0.1 até 3.5.	104
Figura 23 – Superfície de Erro para Variação dos Estados Iniciais x_1 e x_2	105
Figura 24 – Superfície de Erro para Variação dos Estados Iniciais x_1 e x_2 , com variação de 100 para x_1 e 500 para x_2	106
Figura 25 – Superfície de Erro para Variação dos Estados Iniciais x_1 e x_2 , entre 2750 – 2800 para x_1 e 1000 – 1020 para x_2	107
Figura 26 – Coeficiente estimados $erro = 0.1019$	107
Figura 27 – Erros dos Coeficientes Estimados $erro = 0.1019$	108
Figura 28 – Coeficiente estimados $erro = 0.0740$	109
Figura 29 – Erro Coeficientes Estimados $erro = 0.0740$	109
Figura 30 – Superfície de Erro para variações de x_1 e x_2 entre -10000 e 10000.	111
Figura 31 – Superfície de Erro para variações do fator de desconto (fd) e esquecimento (λ), entre 0.01 – 1, com intervalo de 0.01.	113
Figura 32 – Figura rotacionada da superfície de erro para variações do fator de desconto e fator de esquecimento do RLS, entre 0.01 – 1, com intervalo de 0.01 e eixo do Erro Total limitado a 1000.	114
Figura 33 – Superfície de Erro para variações de fd entre 0.01 – 1, com intervalos de 0.01 e $prls_k$ entre 1 – 10000, com intervalos de 500.	115
Figura 34 – Superfície de Erro para variações de fd entre 0.01 – 1, com intervalo de amostragem de 0.01 e $prls_k$ entre 0 – 10000, intervalos de 500 e erro limitado a 20.	116
Figura 35 – Superfície de Erro para variações de fd e $prls_k$ entre 0.01 – 1, com intervalo de 0.01.	117
Figura 36 – Superfície de Erro para variações de fd e $prls_k$ entre 0.01 e 1, em intervalos de 0.01, com erro total limitado à 20.	118
Figura 38 – Coeficientes Estimados e erro total acumulado de 0.0735.	118
Figura 37 – Superfície de Erro para variações de fd entre 0.1900 – 0.1930 e $prls_k$, entre 0.9800 – 0.9850, com intervalo de amostragem de 0.0001.	119
Figura 39 – Erro dos coeficientes estimados, com erro total acumulado de 0.0735.	119
Figura 40 – Trajetória das Funções ao Longo do Processo Iterativo	122
Figura 41 – Erro das funções em função das iterações	122
Figura 42 – Trajetória do Índice de Desempenho	123

Lista de abreviaturas e siglas

ADP	Programação Dinâmica Aproximada
ADHDP	Programação Dinâmica Heurística Dependente de Ação
IFT	Iterative Feedback Tuning
LQR	Regulador Linear Quadrático
LQG	Linear Quadrático Gaussiano
DLQR	Regulador Linear Quadrático Discreto
MFA	Model Free Adaptive
OPFB	Output Feedback
POMDP	Processo de Decisão Markoviano Parcialmente Observável
PI	Iteração de Política
VI	Iteração de Valor
SPSA	Perturbação Simultânea por Aproximação Estocástica
VRFT	Virtual Reference Feedback Tuning
SISO	Single-Input Single-Output
SIMO	Single-Input Multiple-Output
MISO	Multiple-Input Single-Output
MIMO	Multiple-Input Multiple-Output
RLS	Mínimos Quadrados Recursivos

Lista de símbolos

n	Ordem do Sistema Dinâmico
m	Número de Entradas do Sistema
p	Número de Saídas do Sistema
N	Horizonte de Tempo Finito
γ	Fator de Desconto
λ	Fator de Esquecimento do RLS
x	Vetor de Estados
y	Vetor de Saída
u	Ação de Controle
$\hat{x}$	Estado Estimado
$\mathcal{O}$	Espaço de Observação
μ	Política de Controle
$\bar{y}$	Vetor de Saída Passadas
$\bar{u}$	Vetor Ação de Controle Passadas
$\bar{P}$	Matriz Contendo os Coeficientes a Serem Estimados
$\bar{P}_V$	Matriz Contendo os Valores de Referência pela Solução de Schur
$\bar{P}_E$	Matriz Contendo os Coeficientes Estimados pelo RLS

Sumário

1	Introdução	15
1.1	O Problema de Modelagem em Sistemas Dinâmicos e a Observação de Estados	15
1.2	Justificativa	22
1.3	Contextualização do Problema de Estimação de Estado	24
1.4	Objetivos	25
1.5	Organização do Trabalho	26
2	Elementos da Tese	28
2.1	Observadores Baseados em Modelo	28
2.2	Observadores Orientados por Dados	29
2.3	Programação Dinâmica Aproximada e Aprendizagem por Reforço	33
3	Fundamentação Teórica	36
3.1	Observabilidade em Sistemas Dinâmicos Não Lineares	37
3.1.1	Caracterização Sistemas Dinâmicos Não Lineares	37
3.1.2	Observadores	38
3.1.2.1	Condições Geométricas para Observabilidade	39
3.1.2.2	Condições Analíticas para Observabilidade	41
3.2	Observabilidade e Observadores de Estado em Sistemas Lineares	44
3.2.1	Observabilidade Completa em Sistemas Contínuos	45
3.2.2	Observadores de Estado em Sistemas Lineares	46
3.2.3	Observadores de Estado de Ordem Completa	46
3.2.4	O Problema Dual	47
3.2.5	Condições Necessárias e Suficientes para Observação de Estados para Sistemas Lineares	48
3.3	Projeto de Observadores	49
3.3.1	Projeto Observadores Lineares	49
3.3.2	Projeto Observadores Não Lineares	50
3.3.3	Projeto Baseado em Transformação	51
3.4	Observador de Estados Baseados em Programação Dinâmica Aproximada Alimentado por Dados	52
3.4.1	O Problema de Controle Ótimo Alimentado por Dados com Realimentação de Saída Usando Aprendizagem por Reforço	52
3.4.1.1	Política de Controle DLQR e a Equação de Bellman	53

3.4.1.2	Formulação da Função Valor em Termos dos Dados Mensurados	54
3.4.1.3	Erro de Diferença Temporal Baseado em Dados Mensurados	56
3.4.1.4	Escrevendo a Política de Atualização em Termos dos Dados Mensurados	57
4	O Problema de Estimação de Estado Alimentado por Dados	59
4.1	O Problema da Reconstrução dos Estados Baseada em ADP	59
4.2	Aproximação das Matrizes M_u e M_y	66
4.3	Rede Neuronal Sem Saturação para Solução de Sistemas de Equações Não Lineares	68
4.4	Formulação da Rede para Solução do Sistema de Equações	68
4.5	Observador Estado Baseados em Aprendizagem Q	70
4.6	Observador AD-HDP para o LQR Discreto	71
4.6.1	Caracterização da Configuração Função-Q para o LQR Discreto	72
4.7	Observador de Estados e Aprendizagem Q	74
4.7.1	Formulação da Função-Q para Observador de Estado $(\hat{x}, \tilde{x})$	74
4.7.2	Formulação da Função-Q para Observador de Estado $(\hat{x}, \tilde{y})$	77
4.7.3	Formulação da Função-Q para Observador de Estado $(\hat{x}, u, \tilde{y})$	79
5	Metodologia, Métricas e Heurísticas para Avaliação de Convergência do Algoritmo OPFB	81
5.1	Análise de Convergência	81
5.2	Heurística para Geração das Superfícies e para Seleção dos Parâmetros na Sintonia do Algoritmo OPFB	83
5.3	Gerador de Política	84
5.4	Heurística para Seleção dos Parâmetros	85
5.5	Implementação, Ruído e Fator de Desconto do Algoritmo OPFB	86
6	Análises, Simulações Computacionais e Sintonia de Observadores Alimentados por Dados	88
6.1	Sistema Simulado, Estados Reais e Estimados	89
6.1.1	Seleção do Fator de Desconto e Variância do Ruído	89
6.1.2	Estimação de Estados	91
6.2	Heurística para Sintonia, Análise de Sensibilidade e Solvabilidade	94
6.2.1	Análise da Superfície de Erro para Variações dos Parâmetro xk e $fd(\gamma)$	97
6.2.2	Análise da Superfície de Erro para variação dos Parâmetro $prls_k$ e Fator de Desconto (fd)	100
6.2.3	Análise da Superfície Erro para Variação dos Parâmetros $fd(\gamma)$ e Variância do Ruído	101
6.2.4	Análise das Condições Iniciais para os Estados x_1 e x_2	104
6.3	Análise de Solvabilidade	110
6.3.1	Amplitude dos Limites dos Estados	110
6.3.2	Zonas Proibidas, Fator de Desconto e Esquecimento	112

6.3.3	Variação e Análise dos Parâmetros x_k , fd e prls_k	114
6.4	Experimentos Computacionais para a Solução do Sistema de Equações Não Lineares para Estimação de Estados	120
7	Conclusão	124
7.1	Trabalhos Futuros	125
7.1.1	Trabalhos Publicados	125
	 Referências	 126

Capítulo 1

Introdução

Nesta Introdução é abordado inicialmente o problema sobre a construção de modelos baseados em sistemas dinâmicos, necessários em diversos ramos da ciência, tecnologia e indústria. São abordados vários aspectos relacionados a observação de sistemas dinâmicos e os observadores de estado. A discussão abordada neste capítulo é apresentada, com relação a complexidade para construção de modelos, e as dificuldades e limitações associadas ao processo de modelagem, assim como o dispêndio de tempo necessários para formulação e construção de um modelo adequado para uso. São discutidos ainda a importância dos observadores no fornecimento dos estados necessários por diversas metodologias de controle.

1.1 O Problema de Modelagem em Sistemas Dinâmicos e a Observação de Estados

O mundo é descrito por diversos fenômenos, e com o propósito de tentar compreendê-los, os pesquisadores das mais diversas áreas elaboram modelos simplificados e posteriormente aumentam a sua complexidade, visando descrever o sistema em questão para um melhor entendimento da sua dinâmica. Vários são os tipos de modelos desenvolvidos e aplicados pelos pesquisadores, sendo de interesse o desenvolvimento de estruturas que apresentem uma relação funcional entre as variáveis de estado e as saídas do sistema, facilitando assim o processo de análise do comportamento dinâmico de variação dos estados com relação às saídas do sistema.

Outro aspecto de grande interesse está relacionado a possibilidade de descrever o comportamento do sistema a partir de suas variáveis de estados observadas anteriormente e com isso usar essa informação para prever o seu comportamento futuro. As principais formas para tal

estimação parte da existência ou do desenvolvimento de modelos matemáticos, estatísticos ou heurísticos. Tais modelos podem ser elaborados considerando as leis que regem o processo de interesse ou com metodologias baseadas em dados mensurados experimentalmente e aplicando métodos estatísticos.

O processo de elaboração dos modelos matemáticos, exige que se detenha o conhecimento sobre as leis que regem o fenômeno de estudo, assim como outras variáveis que possam ter influência no fenômeno. Obter conhecimento de todas as variáveis que influenciam um determinado sistema é praticamente impossível na maioria das situações. À medida que mais variáveis são incluídas em um modelo, mais preciso o mesmo pode se tornar, mas ao custo de elevação de sua complexidade, visto que é necessário uma maior quantidade de informação, parâmetros ou variáveis, aumentando o custo computacional para uso do modelo.

A importância dos observadores de estados tem fundamentos na teoria de sistemas dinâmicos. As formulações para observação em sistemas dinâmicos com relação linearidade, podem ser: lineares ou não lineares. Aplicações de sistemas dinâmicos são realizadas em várias áreas de conhecimento, como controle, monitoração ou identificação. Essas são as principais motivações para o desenvolvimento de novas estruturas e métodos para aplicações na indústria que justificam a necessidade de pesquisa de métodos e modelos matemáticos para estimação de estados que tem por base a mensuração das saídas do sistema dinâmico observado.

Desta forma, é cada vez mais comum que os cientistas realizem estudos estatísticos com base em dados amostrais extraídos de experimentos com o propósito de construir modelos que descrevam o comportamento do sistema em estudo. Neste aspecto, vários autores desenvolveram trabalhos de modelagem de sistemas dinâmicos lineares, não lineares, discretos, contínuos, determinísticos ou estocásticos [HERMANN; KRENER, 1977; ISIDORI, 1995; CHEN, 1995; ELLIS, 2002]. Estudos comparativos entre metodologias para controle baseadas em modelos e outras baseadas em dados usando, por exemplo: regressão polinomial para controle preditivo são abordados em [LI; YAMAMOTO, 2016]. Outro trabalho com aplicação de controle ótimo adaptativo orientado por dados, faz uso do método de sintonia "*Virtual Reference Feedback Tuning*" (VRFT), que é abordado em [ROMAN et al., 2016]. Como aplicação dessa metodologia, temos uma aplicação em controle de compensação de falhas, com realimentação na saída com controle digital PID para sistemas com dinâmica desconhecida abordada no trabalho [WANG; YANG, 2016a].

Diversos pesquisadores realizaram estudos sobre identificação de sistemas, controle para sistemas não baseados em modelos, assim como a avaliação de metodologias baseadas somente por dados. Diversos trabalhos foram realizados, que não baseados em modelos, relacionados a teoria de controle, identificação de sistemas e estimação de parâmetros, em sistemas lineares, não lineares, discretos ou com linearizações de plantas não lineares, abordados nos trabalhos de [ZHU et al., 2015; XU et al., 2014; XU; JAGANNATHAN, 2014; ZHAO, 2015; HOU ZHONGSHENG; JIN, 2013].

Na construção de modelos por meio do uso de dados existem vantagens e desvantagens. Um ponto positivo está relacionado a possibilidade da construção de modelos para sistemas complexos que possuam muitas variáveis ou se detém pouca informação, dificultando assim, o processo de elaboração de um modelo matemático, assim como o seu ajuste. Uma desvantagem da construção de modelos aproximados por dados, é que muitas metodologias podem necessitar de muita memória ou hardware com elevado poder de processamento computacional, para realizar o levantamento e atualização dos modelos. Existem situações que se demanda de muito tempo para realizar os cálculos a partir dos dados e para alguns sistemas pode se tornar inviável sua aplicação. Outro aspecto relevante são os custos, que em algumas situações podem ser muito elevados, inviabilizando a implementação da metodologia baseada em dados.

O objetivo de se possuir um modelo está relacionado diretamente com o conhecimento do comportamento de variação das variáveis de estado e da saída do sistema, ou seja, da relação do estado com a interação e resposta do sistema dinâmico. Na representação em Espaço de Estados, tem-se que o sistema dinâmico, representado pelo estado que é ponderado pela matriz com a dinâmica do sistema e matriz de controle, usados para estimar o comportamento dos estados, e a saída ponderada por uma matriz de saída com representação dos canais de saída com as ponderações da influência de cada um dos estados. Através das variáveis de estado ponderadas que representem um sistema, pode-se inferir sobre o seu comportamento, desde que se possua um modelo matemático ou os dados experimentais. Com o conhecimento da influência das variáveis de estado do sistema em sua resposta, é possível, em alguns casos, uma estimação aceitável, e com isso a realização da inferência do estado e seu comportamento futuro.

Inúmeras são as dificuldades no processo de elaboração de modelos baseados em dados. Uma das dificuldades está relacionada a aquisição dos dados, a partir dos experimentos reais mensurados por sensores, visto que muitos experimentos podem ser complexos, e nem sempre as informações mensuradas pelos sensores é precisa, podendo conter sinais ruidosos de diversos tipos, tanto brancos quanto coloridos. Muitas vezes também não é possível realizar certos experimentos, por elevado custo, ou mesmo por impossibilidade de implementação. Nesses casos pode-se fazer uso de simulações tanto em hardware quanto por softwares para esses sistemas e com isso realizar a análise dos dados gerados.

Independentemente da escolha para aquisição dos dados, sejam por meio de experimentos ou simulação baseados em modelos, ambas as abordagens possuem características e limitações próprias. Quando se detém *a priori* um modelo matemático de um sistema, pode-se fazer uso do mesmo para a geração de dados por meio de simulações e assim observar as respostas do sistema a diferentes excitações e configurações, assim como realizar a avaliação da robustez dos sistemas de controle ou a estimação de estados frente a variações paramétricas da planta ou com relação a rejeição de ruído. Deste modo é possível testar diferentes cenários para um dado modelo podendo assim avaliar situações mais próximas das situações reais ou para situações desejadas.

Sempre que informações amostrais de um experimento estão disponíveis pode ser possível avaliar a aplicação de modelos estatísticos e técnicas de aprendizado de máquina para a construção de modelos baseados nas amostras, por meio da aplicação da teoria de regressão, por exemplo. A teoria de regressão possui um conjunto de técnicas, que por meio do uso dos dados amostrais realizam o levantamento de modelos, realizando a verificação da possibilidade de ajuste dos dados à curvas parametrizadas por diversos tipos de modelos, a partir da estimação dos parâmetros do modelo que se propõe testar o ajuste de curva.

Em algumas situações somente informações parciais sobre os estados estão disponíveis. No trabalho de [SHOUSONG; QIXIN, 2003], uma aplicação de controle ótimo estocástico orientado somente por dados com longos atrasos é abordado, com a discussão de casos em que se possui informação parcial dos estados e suas implicações e dificuldades. Outra aplicação de controle ótimo orientado por dados para a conservação de energia em construções é apresentado em [NISHIGUCHI et al., 2010].

Outra técnica usada para aproximação de funções são as redes neurais, que são aproximadores universais de funções. As redes neurais, assim como outras técnicas que desenvolvem algoritmos bio-inspirados como algoritmos evolutivos, que possuem também propriedades adaptativas, o que permite o auto ajuste e a adaptabilidade, à medida que os dados estejam disponíveis para o treinamento das arquiteturas de interesse. A não linearidade nas funções de ativação nas redes neurais podem modelar sistemas não lineares.

Deste modo a importância do estudo de abordagens não baseadas em modelos é de fundamental interesse da comunidade científica, visto que o processo de modelagem e identificação dos sistemas é trabalhoso e complexo, resultando em muitas situações um modelo insatisfatório. Para solucionar parte desse problema trabalhos usando de algoritmos adaptativos e técnicas com inspiração biológica, como redes neurais diretas, recorrentes ou base radial, são amplamente aplicadas pela característica de mapeamento universal. A aplicação de redes neurais para identificação de sistemas é abordada no trabalho de [CHU et al., 1990]. Outro trabalho com aplicação para identificação de sistemas voltadas para aplicações em controle, usando rede neuronal com função de ativação variada é abordado no trabalho de [NECHYBA; XU, 1994]. Outras aplicações das redes neurais, como aproximador de mapeamentos, para otimização do processo de adaptação e para aproximar os mapeamentos das funções de transição envolvidas no projeto de controladores/observadores, temos os trabalhos de [WERBOS, 1989], [CHEN et al., 1990] e [SCHRÖDER et al., 2013]. Nestes trabalhos são abordados aplicações em redes neurais e em sistemas híbridos neuro-fuzzy, para identificação de sistemas não lineares. Em outro trabalho, uma rede neuronal recorrente do tipo Hopfield para estimação on-line dos parâmetros da planta observada, a partir da estimação on-line dos parâmetros associados a dinâmica do sistema, discutidos em [ALONSO et al., 2009].

Outros aproximadores universais são os algoritmos baseados em lógica Fuzzy, que podem ser combinados com as redes neurais, a fim de melhorar o desempenho no processo de

estimação e aproximação de funções. Neste sentido, aplicações para estimação de parâmetros usando técnicas de aproximadores de funções para controle adaptativo em sistemas não lineares são discutidos em [SPOONER et al., 2004]. Outra teoria com interesse nas técnicas para estimação com base em dados são os algoritmos de otimização estocástica tratados em [SPALL, 2005], com aplicações em estimação, controle e simulação, com técnicas específicas relacionadas a estimação do gradiente, baseados em dados, usando somente uma função de desempenho. Em ambos os casos, existe ainda a dependência dos dados, que dependendo da quantidade e qualidade da informação adquirida, pode gerar estimativas polarizadas, ou seja, com resultados deslocadas dos verdadeiros valores, causados pela sensibilidade dos modelos às amostras enviesadas ou com tamanho de amostra insuficiente para garantir uma margem de erro aceitável para inferir-se sobre a população estimada.

No contexto de observação de estados e estimação de estados, o problema de observação em sistemas dinâmicos e observadores sejam eles model-free (livre de modelo) e model-dependent (dependente de modelo), são os principais temas abordados nesta Tese. As principais aplicações dos observadores são: monitoração, estimação e controle. As principais ideias e técnicas para identificação de sistemas e estimação de estados para ambas as formulações são aqui discutidas. O desenvolvimento do trabalho apoia-se no horizonte de tempo e estado-da-arte de observadores de estado e seus elementos cronologicamente, pelas citações referenciadas, tanto nos textos clássicos quanto nos mais recentes na literatura.

O interesse no estudo das variáveis de estado que nem sempre são mensuráveis justifica o estudo de metodologias baseadas em dados, com o propósito de realizar a estimação desses estados não diretamente mensuráveis. Uma teoria que faz uso de dados para construir modelos é a teoria de identificação de sistemas, que por meio da aplicação de vários métodos é possível levantar o modelo dinâmico de sistemas lineares ou não lineares, somente por observações passadas. Algumas técnicas envolvidas para identificação utilizam-se de aproximadores polinomiais, estimadores paramétricos, métodos dos mínimos quadrados, métodos iterativos, redes neurais [CHEN et al., 1990].

É de grande uso pela comunidade científica a combinação de técnicas de identificação de sistemas para controle, como a inclusão de observadores de estado para realizar a estimação dos estados em um sistema dinâmico, a fim de minimizar o esforço com relação a ação de controle. A fundamentação teórica para observadores em sistemas não lineares que possuem bases na geometria diferencial, com os conceitos de indistinguibilidade entre estados, condições para observabilidade em sistemas dinâmicos, caracterização do espaços de estados e de observação, assim como síntese e realização de tais observadores são abordados em [GUILDAS, 2007] e [ANGUELOVA, 2004]. Parte desses conceitos são apresentados nesta Tese, como fundamentação.

O estudo de observadores de estados para sistemas em geral é de grande interesse, em especial para o caso em que os estados não podem ser diretamente mensurados, ou por serem

meramente matemáticos. Monitoração e controle com realimentação de estado, são aplicações do uso de observadores de estado em sistemas dinâmicos. O processo de estimação de estados tem como propósito a reconstrução dos estados por meio da observação dos sinais de entrada e saída do sistema [CHEN, 1995], ou mesmo somente pela observação das saídas do sistema dinâmico [TSINIAS, 1989; SOROUGH, 1997; GENC et al., ; KAIPIO; SOMERSALO, 1999].

Uma das aplicações dos observadores de estado é a identificação de sistemas. Abordagens nesse sentido, podemos citar as formulações propostas por *Luenberger* e *Kalman*, que visam aproximar o sistema dinâmico a ser identificado pela inclusão de sistemas auxiliares, os observadores de estados [BAUMEISTER; LEITÃO, 2008], que a partir das informações de entrada e saída do sistema é utilizado um estimador para prever os estados e a saída do sistema, com base em informações a priori da dinâmica do sistema. Vários são os trabalhos clássicos que relatam este tipo de abordagem, com formulações, aplicações e análises dos métodos [ELLIS, 2002; CHEN, 1995; KAILATH, 1980; O'REILLY, 1983].

Outros trabalhos mais que tratam do uso de técnicas baseadas em inteligência artificial, em especial a aplicação de redes neurais como nos trabalhos de [CHU et al., 1990] e [NECHYBA; XU, 1994], por ser um aproximador universal e com isso otimizando ainda mais o processo de adaptação e aproximação intrínseco dos observadores, [SPOONER et al., 2004; ALONSO et al., 2009; WERBOS, 1989; CHEN et al., 1990; SCHRÖDER et al., 2013].

Em sistemas não lineares, temos que algumas condições devem ser verificadas a fim de avaliar a observabilidade de tais sistemas, como as condições geométricas e analíticas. Tanto as condições geométricas quanto as analíticas possuem subcritérios específicos para que haja uma caracterização mais robustas de tais critérios. A Figura 1 apresenta um diagrama com as principais condições geométricas e analíticas para a caracterização dos observadores em sistemas não lineares, abordados em detalhes em [GUILDAS, 2007].

A abordagem geométrica refere-se as definições sobre a estrutura topológica dos espaços de estado e observação, assim como os mapeamentos entre esses espaços, responsáveis pela reconstrução dos estados. Tal reconstrução é feita pelo uso de funções de aproximação baseada na existência de um mapeamento ($\mathcal{H}$) entre as variedades dos espaços de observação e estados, possuindo esses espaços características específicas, tais como, as variedades devem ser diferenciáveis e o mapeamento ($\mathcal{H}$) entre os espaços deve ser um difeomorfismo entre as variedades, ou seja, uma função bijetiva, diferenciável e deve existir mapeamento inverso ($\mathcal{H}^{-1}$), sendo esse mapeamento inverso também diferenciável [SPOONER et al., 2004]. A Figura 2 apresenta um esquema gráfico dos mapeamentos entre o espaço de estados ($\mathcal{X}$) e de observação ($\mathcal{O}$), com a ilustração de Morfismo, Isomorfismo e Difeomorfismo, explicitando as diferenças entre os mapeamentos.

Na Figura 2 o mapeamento $\mathcal{H}^{-1}$, representa o mapeamento do espaço de observação ($\mathcal{O}$), para o espaço de estados ($\mathcal{X}$), que é o mapeamento que reconstrói os estados x_0 a partir das observações $y_e(x_0)$.

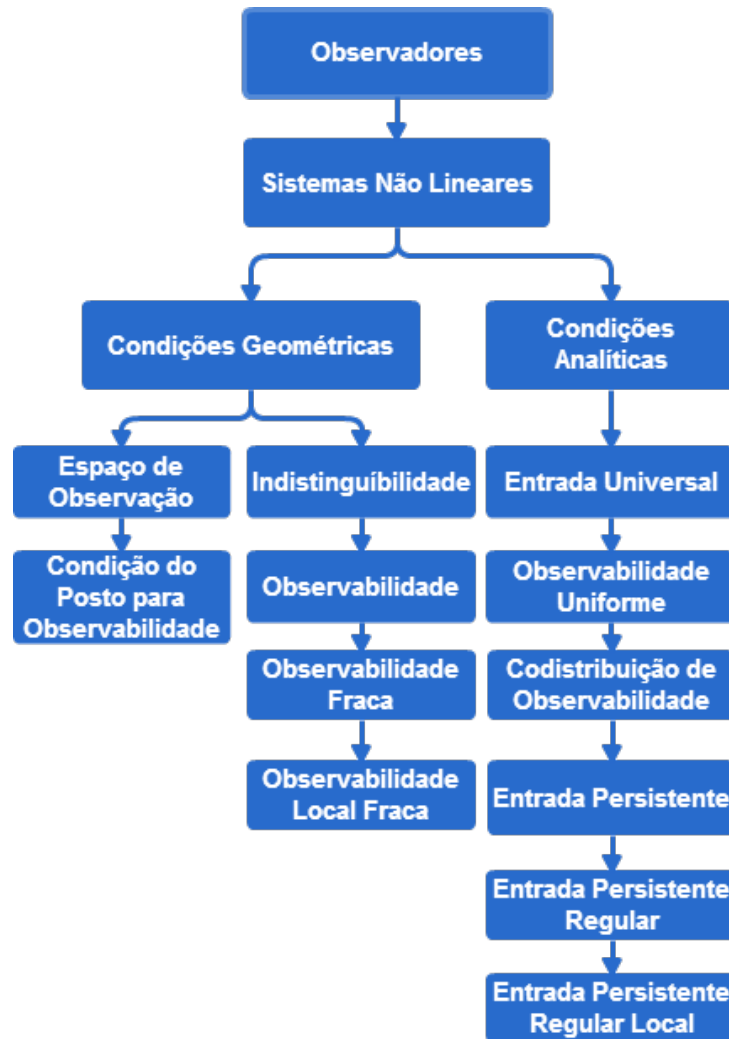


Figura 1 – Características Gerais para Observadores Lineares e Não Lineares

Fonte: Produzido pelos Autores.

É de extrema importância a descrição da formulação do sistema dinâmico tratados nesta Tese, pois a base da formulação depende da presunção sobre a descrição do sistema dinâmico. A fundamentação teórica usada para a descrição e reconstrução dos estados de dado sistema está relacionada com a geometria diferencial, da caracterização dos espaços topológicos, onde se encontram os estados e da existência de funções inversíveis e diferenciáveis [BAUMEISTER; LEITAO, 2005], que permitirão a reconstrução dos estados a partir das observações dos sistemas estudados.

Diversas formulações sobre a caracterização e definição das condições geométricas e analíticas foram desenvolvidas e discutidas por alguns pesquisadores nos trabalhos [MAYNE; BROCKETT, 1973; ISIDORI, 1995; MORGANSEN, 2004; HIRSCHORN; LEWIS, 2002]. Nestes trabalhos é possível encontrar uma vasta descrição sobre conceitos de controlabilidade e observabilidade no ponto de vista da geometria diferencial para sistemas não lineares.

A caracterização e investigação sobre as condições suficientes ou necessárias para a observação em sistemas dinâmicos não lineares, com a necessidade de definição de condições

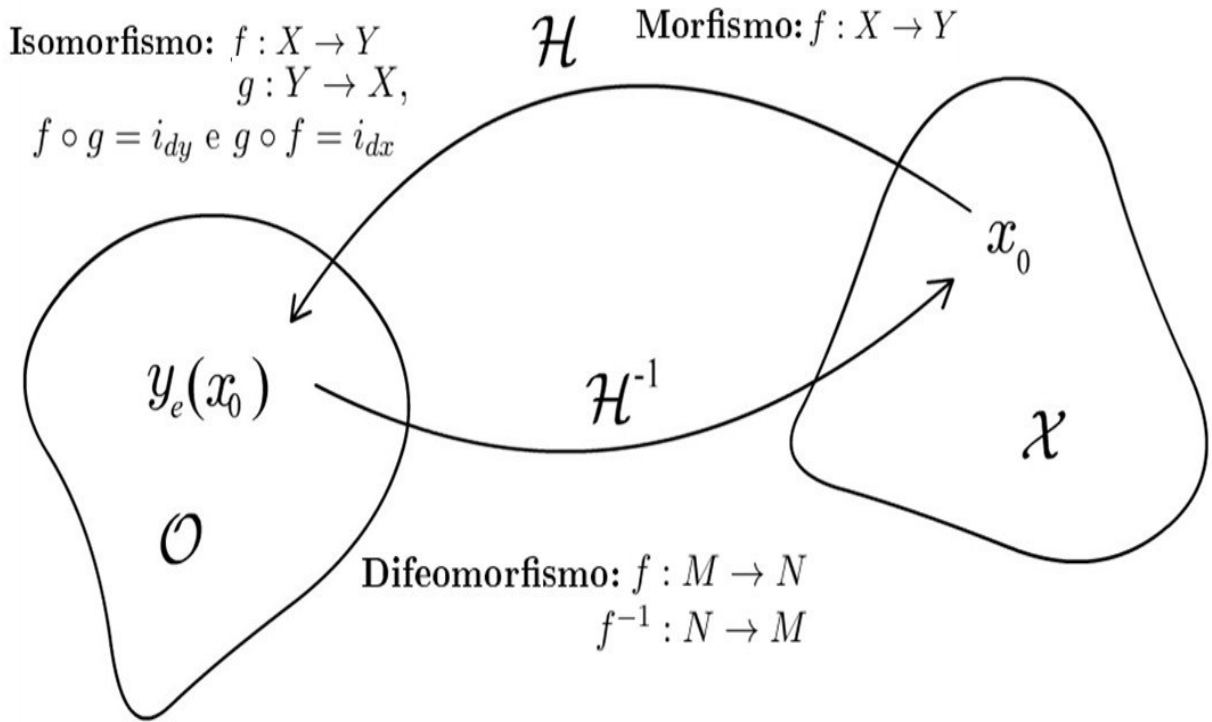


Figura 2 – Morfismo, Isomorfismo e Difeomorfismo entre Espaços de Estados e Observação

Fonte: Produzido pelos Autores.

mais locais, com relação as condições iniciais e de contorno são abordados em [HERMANN; KRENER, 1977].

A seguir são apresentadas as justificativas, as motivações e a contextualização para problema de observação de sistemas dinâmicos, dos objetivos gerais e específicos, seguidos pela descrição da organização da Tese.

1.2 Justificativa

As principais justificativas para o estudo dos observadores de estado, estão associadas à complexidade de se obter informações sobre o comportamento dos estados de um sistema dinâmico, uma vez que um sistema observado pode ser ou não modelado ou predito. A possibilidade de construção de um modelo que represente o sistema adequadamente, mesmo que simplificado, é algo de extrema relevância e importância. A partir de observações passadas, tem-se interesse de entender o comportamento dinâmico do sistema, seja com propósito de monitorar, estimar variáveis de estado ou prever uma resposta sobre o seu comportamento.

Nos mais diversos campos da ciência e indústria, o processo de estimação de variáveis de estado a partir de dados históricos, são de grande importância para um entendimento mais eficiente sobre a dinâmica do sistema estudado. O processo de estimação de estados para sistemas dinâmicos é largamente difundido por vários autores, de diversos campos da ciência, indústrias

e tecnologia, que tem interesse tanto na automatização dos seus processos, quanto para outras ciências que de algum modo interagem com um objeto de estudo.

A pesquisa associada à estimação de estado pode ser classificada com relação a natureza do sistema e com o comportamento das variáveis que a integram, como por exemplo: sistemas dinâmicos discretos ou contínuos, lineares ou não lineares, determinísticos ou estocásticos. De acordo com a natureza do sistema dinâmico, existem diversas abordagens com metodologias consolidadas para a estimação de tais estados. O processo de estimação de estados está relacionado à uma estrutura auxiliar que é acoplado ao sistema dinâmico a ser observado e é chamado de "Observador de Estado".

O desenvolvimento de observadores de estado como parte integrante de tecnologias para estimação de estados são de grande importância para diversas áreas, visto que nem sempre é possível mensurar diretamente os estados pelo uso de sensores. Para se estudar o comportamento dinâmico de fenômenos físicos, químicos, biológicos, energéticos é realizada a construção de modelos matemáticos, com o propósito de simplificar o uso das variáveis envolvidas nos fenômenos e com isso diminuir a complexidade do modelo, facilitando o estudo do comportamento dos fenômenos ao longo do tempo ou mesmo da frequência.

Algumas motivações sobre o processo de observação em sistemas dinâmicos e observadores de estado orientados por dados são apresentados nesta Tese. Alguns dos principais trabalhos e ideias sobre os observadores de estado para sistemas dinâmicos lineares e não lineares são apresentados, assim como as condições necessárias para a observação de estados de um sistema dinâmico. São ainda abordados aplicações e avanços da aplicação de observadores de estados em alguns projetos de observadores, tanto para observadores baseados em modelos quanto de metodologias não baseadas em modelos, tendo como base, técnicas de aprendizado de máquina e bio inspiradas.

Métodos de controle realimentado, controle ótimo e ADP, requerem a disponibilidade completa dos estados internos do sistema dinâmico a ser controlado. Neste trabalho apresenta-se uma abordagem que unifica métodos de aprendizagem por reforço, controle ótimo e técnicas de identificação dos parâmetros de Markov, tendo em vista a implementação de método de ADP usando somente dados mensurados em um horizonte finito, além de também realizar a definição de métricas para avaliação de desempenho, que serão usadas por heurísticas para a sintonia das formulações propostas.

Existe um grande interesse na sintonia de parâmetros em diversas metodologias, independente se a aplicação é para realizar o controle, identificação ou estimação de estado. Sempre que se deseja realizar a aplicação de uma metodologia ou aplicar um algoritmo é necessário realizar o "set up", ou seja, definir os parâmetros iniciais da metodologia a ser aplicada, o que reflete o desempenho da metodologia escolhida ou do algoritmo implementado, pois, estão diretamente ligados às escolhas que foram feitas na inicialização dos algoritmos.

1.3 Contextualização do Problema de Estimação de Estado

O problema de estimação de estados inicialmente é apresentado no contexto de aplicações de controle, atendendo à realimentação de estado e monitoração na detecção de variações significativas no comportamento da planta nas simulações. Para o controle, apresenta-se uma solução associada com métodos de programa dinâmica aproximada para solução da formulação de controle ótimo online model-free (livre de modelo dinâmico).

No caso da estimação de estados para monitoração, a aplicação na detecção de falhas é bastante vasta, podendo ser aplicada em diversos tipos de sistemas, tais como em sistemas industriais, aeroespaciais, indústria automotiva, medicina, em sistemas supervisórios, assim como em sistemas que se realizem o monitoramento e identificação de padrões. Para a execução de tal processo faz-se necessário garantir um melhor rendimento com relação ao custo operacional.

No contexto de desenvolvimento de dispositivos orientados para avaliar o comportamento de sistemas dinâmicos, os observadores de estados baseados em modelos apresentam a solução para problemas de realimentação de estado e monitoração, que funciona muito bem quando se possui um modelo que represente suficientemente bem o sistema modelado.

Novas metodologias apresentam modelos de observadores orientados por dados com maior amplitude e sensibilidade a variações da planta, no sentido de que as estimações são atualizadas a cada instante e com a realização de aprendizagem da dinâmica da planta, perturbação ou ruídos. Métodos para a sintonia e aprendizado dos parâmetros destes modelos, embasados nos paradigmas da programação dinâmica aproximada e aprendizagem por reforço, são apresentados nesta Tese. No arcabouço da teoria são apresentados métodos, métricas, algoritmos, heurísticas e simulações associadas aos observadores de estado.

A seguir apresenta-se um texto que relaciona a importância do desenvolvimento de observadores para aplicações de controle e monitoração. As técnicas de identificação de sistemas tem o propósito de construir um modelo do sistema dinâmico, a fim de descobrir o mapeamento entre as excitações no sistema e suas respostas. Várias metodologias executam o processo de aproximação, que podem ser para estimação paramétrica, visando comparar modelos que possuem estruturas semelhantes e assim classificar os dados observados. A grande maioria dos sistemas em sua natureza dinâmica são propriamente não lineares, possuindo uma componente determinística e outra estocástica, regidas por funções determinísticas e probabilísticas que em ambos os casos são funções desconhecidas.

No processo de construção de um modelo, muitas são as variáveis a serem consideradas, com isso, a tarefa de modelagem pode ser muito complexa e trabalhosa, pois o modelo deve representar adequadamente o sistema que se propõe explicar, sendo, em muitos casos, a modelagem insatisfatória.

O comportamento de diversos fenômenos são não lineares, o que dificulta o processo de construção de um modelo. Para tentar compreender o comportamento dinâmico desses sistemas

pode-se fazer o uso de observadores de estados, com o propósito de identificar, pela aproximação do sistema desconhecido, por um sistema linear ou não linear, determinístico ou estocástico, levantando assim uma estrutura matemática para representar a dinâmica do sistema observado.

O uso de observadores é de interesse para pesquisas baseadas em dados observados dos sistemas dinâmicos, sejam para fins de controle, identificação de sistemas, estimação paramétrica, monitoração ou estimação de estados. Dentre as abordagens de observadores de estados, pode-se identificar as metodologias e nomenclaturas adotadas pelos pesquisadores, associadas a aplicações, somente por meio dos dados observados, sem um modelo formulado previamente.

Quando os pesquisadores em geral referem-se a algum estudo não baseado em modelo, estão referindo-se o estudo do fenômeno somente por meio de observações, não possuindo assim um modelo matemático que represente o comportamento da dinâmica do sistema estudado. Esta escolha deve-se ao fato de que pode ser muito complexo ou dispendioso a obtenção do mesmo, ou por falta de conhecimento das leis que regem o sistema de interesse.

O projeto de observadores de estados aplicados para controle possui características específicas que necessitam da definição sobre quais condições devem ser atendidas e quais são necessárias para se realizar a observação de sistemas dinâmicos, sejam eles lineares ou não lineares. É de extrema importância a definição dos espaços vetoriais associados aos espaços de observação e estados, com a caracterização dos espaços topológicos e métricas para definir quais condições geométricas e analíticas são minimamente necessárias para a realização do processo de observação e possivelmente reconstrução dos estados. Tais condições e detalhes sobre o tema são extensivamente discutidos por [Mayne e Brockett \[1973\]](#), [Guildas \[2007\]](#) e [Brockett \[2014\]](#), .

1.4 Objetivos

Gerais: Desenvolver formulações, métodos e heurísticas, para estimação de estados e sintonia de observadores alimentados por dados, baseados em programação dinâmica aproximada e aprendizagem por reforço.

Específicos:

- Desenvolver heurísticas para sintonia de observadores de estados alimentados por dados, baseados em programação dinâmica aproximada;
- Aplicar técnicas de aprendizagem de máquina, mais especificamente aprendizagem por reforço, para estimação dos estados em horizonte de tempo finito;
- Definir heurísticas para sintonia do processo de aprendizagem dos métodos propostos para estimação de estados, com o propósito de melhoria no desempenho e robustez, com relação ao processo de alimentação de dados, busca de solução e rejeição a dados enviesados;
- Desenvolver gerador de dados de entrada e saída, para as simulações computacionais;

- Formular métricas para avaliação de desempenho e análise de convergência;
- Desenvolver heurísticas para análise de sensibilidade para as formulações propostas;
- Propor métodos alternativos e adaptativos para resolução do sistema de equações não lineares, para estimação de estados;
- Propor e desenvolver métricas e heurísticas para seleção e sintonia dos parâmetros associados às formulações propostas e desenvolvidas, para fins de análise de desempenho e convergência;

1.5 Organização do Trabalho

Esta trabalho é organizado em capítulos e seções que contém o referencial teórico, descrição do problema, formulações propostas, metodologias, análise de resultados, conclusão e referências. Por sua vez, os capítulos são organizados em seções que estão conectados no contexto de observadores de estados orientados por dados e metodologias não baseadas em modelos. Apresenta-se no texto o desenvolvimento de formulações e propostas associadas a métodos e algoritmos para estimação de estados, sintonia de parâmetros e ganhos de observadores, baseados em abordagens de programação dinâmica aproximada, aprendizagem por reforço e algoritmos bio-inspirados.

No Capítulo 2 uma discussão geral do problema de observação de estados é apresentado. A abordagem é discutida sobre observadores baseados em modelos e orientados por dados, sendo esses os principais objetos de estudo abordados nesta Tese. Além de apresentar ainda parte do estado da arte relacionado a estrutura de observadores baseados em modelos, alimentados por dados ou mesmo metodologias baseadas em dados que possuem alguma estrutura interna de aprendizagem ou adaptabilidade baseadas em dados para estimação de estados direta ou indiretamente. É discutido ainda a abordagem relacionada ao uso de técnicas de programação dinâmica aproximada e aprendizagem por reforço.

No Capítulo 3, o Referencial Teórico é apresentado, com os principais tópicos sobre observabilidade em sistemas dinâmicos e observadores de estado, a estrutura baseada em geometria diferencial, observabilidade em sistemas não lineares e programação dinâmica aproximada e aprendizagem por reforço alimentado por dados. Os seguintes tópicos são abordados nas seções deste capítulo:

- Seção 3.1: Nesta Seção discuti-se sobre a Observabilidade em Sistemas Dinâmicos Não Lineares, sob o ponto de vista de Caracterização Sistemas Dinâmicos Não Lineares e sobre Observadores e suas condições Geométricas e Analíticas para Observabilidade.
- Seção 3.2: São abordados os temas sobre Observabilidade e Observadores de Estado em Sistemas Lineares, abordando a Observabilidade Completa de Sistemas Contínuos, Obser-

vadores de Estado em Sistemas Lineares, Observadores de Estado de Ordem Completa, O Problema Dual e as Condições Necessárias e Suficientes para Observação de Estados em Sistemas Lineares.

- Seção 3.3: Aborda o projeto de observadores com base na fundamentação da Seção 3.1, sob o ponto de vista de Projeto Observadores Lineares e Não Lineares, baseados em transformação.
- Seção 3.4: Aborda-se o problema de controle orientado por dados, com a estrutura de controle com realimentação de saída, em uma formulação não baseada em modelo, fundamentada em programação dinâmica aproximada, somente utilizando dados de entrada e saída passados armazenados. São descritos ainda o Problema de Controle Ótimo Alimentado por Dados com Realimentação na Saída, usando aprendizagem por reforço baseado no Regulador Linear Quadrático Discreto (DLQR), adaptando suas estruturas em termos dos dados mensurados, como a Política de Controle do DLQR, a Equação de Bellman, a Formulação da Função Valor, o Erro de Diferença Temporal, Política de Atualização e aspectos sobre sua implementação.

No Capítulo 4, apresenta-se a Descrição e Formulação do Problema de Estimação de Estados Alimentados por Dados. O Problema da Reconstrução dos Estados para Realimentação de Saída, a Rede Neuronal sem Saturação para Solução de Sistemas de Equações Não Lineares, a Formulação da Rede para Solução do Sistema de Equações e formulações propostas para observadores de estados baseados em aprendizagem Q.

No Capítulo 5 apresentam-se as Métricas e Heurísticas para Avaliação de Convergência do Algoritmo OPFB, enfatizando-se a Análise de Convergência, frente a variações paramétricas e amplitude de sinal de ruído, assim como a influência de condições iniciais. São apresentados heurísticas para Geração de Superfícies e Seleção dos Parâmetros na Sintonia do Algoritmo OPFB, além de análises para implementação. Foram ainda avaliados o fator de desconto do algoritmo OPFB e fator de esquecimento do algoritmo dos mínimos quadrados recursivo (RLS) e condições iniciais para a matriz de covariância inicial do RLS. Outras análises ainda foram realizadas neste capítulo, com a discussão sobre a análise dos Experimentos Computacionais, a Sintonia do Algoritmo OPFB, a Amplitude dos Limites dos Estados, as Zonas Proibidas, o Fator de Desconto e Esquecimento e os Fatores Multiplicativos x_k e $prls_k$.

No Capítulo 6, as simulações computacionais das formulações propostas são apresentadas, juntamente com diversas análises dos resultados relacionadas aos aspectos das simulações e análise dos algoritmos. São apresentadas simulações para determinação dos estados, aspectos computacionais, sintonia dos parâmetros, análises de solvabilidade, desempenho, convergência e comportamento dos algoritmos frente presença de ruídos e variações paramétricas.

No Capítulo 7 é apresentado a Conclusão, com as considerações finais, trabalhos futuros e artigos publicados, seguidos do referencial citado no trabalho.

Capítulo 2

Elementos da Tese

Neste Capítulo são apresentadas as referências sobre métodos clássicos de observadores de estados para sistemas lineares, baseados em modelos, assim como referências sobre métodos alimentados por dados que podem possuir ou não observadores de estados explícitos em suas formulações, mas que possuem algum tipo de mecanismo para observação de estados internamente. Também são relatadas diversas referências sobre trabalhos usando programação dinâmica aproximada, que suporta o estudo dos observadores de estado em sistemas dinâmicos e outros trabalhos com metodologias para sintonia e solução de problemas modelados por sistemas dinâmicos. Inicialmente é apresentada a teoria associada à observação em sistemas dinâmicos não lineares, que tem como caso particular o caso para sistemas lineares. A fundamentação teórica apresentada é fundamentada em uma abordagem geométrica, com base em geometria diferencial, espaços topológicos, mapeamentos e no difeomorfismos entre os espaços de observação e estados.

Alguns dos temas abordados nesta Tese são: Observadores de estados e os paradigmas baseados em modelos (Model-Based Observers) e orientados por dados (Model-Free Observers), bem como as estratégias para sintonia dos observadores, associados à programação dinâmica aproximada e aprendizagem por reforço.

2.1 Observadores Baseados em Modelo

Os primeiros trabalhos abordando observadores para sistemas lineares foram apresentados por *Luenberger* no trabalho [LUENBERGER, 1964], seguidos outros trabalhos sobre observadores para sistemas lineares multivariáveis [LUENBERGER, 1966], assim como as estruturas de projetos de observadores com formas canônicas [LUENBERGER, 1967]. Neste último trabalho são apresentadas estruturas que simplificariam o projeto dos observadores para

as formas canônicas em sistemas lineares multivariáveis.

Posteriormente, outros trabalhos foram apresentados por *Luenberger*, como uma introdução ao projeto de observadores de estados de modo mais amplo, com as formulações da época e uma compilação dos modelos, aplicações e exemplos práticos abordados em [LUENBERGER, 1971] e [LUENBERGER, 1979].

Usualmente, na abordagem moderna dos observadores de estados orientados por modelo do tipo *Luenberger*, a notação $\hat{x}$ é usada para o vetor de estado estimado do observador. Em situações práticas usamos o vetor de estado do observador para determinar a lei de controle do sistema realimentado.

Para apresentar a estrutura de um observador do tipo *Luenberger*, considere o sistema dinâmico linear invariante no tempo:

$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx\end{aligned}\quad (2.1)$$

Um observador é um subsistema usado para reconstruir o vetor de estados de uma planta, por meio dos dados mensurados e sinais de entradas passados, considerando o conhecimento do modelo, ou seja, das matrizes A , B e C de um sistema dinâmico, como o sistema (2.1), [CHEN, 1995]. O modelo do observador é basicamente o mesmo da planta, exceto por possuir um termo a mais que inclui o erro da estimação para compensar a acurácia das matrizes A e B , assim como o sinal de erro inicial. O sinal de erro de estimação é dado pela diferença entre a saída medida e a estimada. O erro inicial é dado pela diferença entre o estado inicial e a estimativa de estado inicial. Portanto, pode-se definir o modelo do observador sendo:

$$\begin{aligned}\dot{\hat{x}} &= A\hat{x} + Bu + K_e(y - C\hat{x}) \\ \hat{x} &= (A - K_eC)\hat{x} + Bu + K_ey,\end{aligned}\quad (2.2)$$

onde $\hat{x}$, é o estado estimado e $C\hat{x}$ é a saída estimada. A entrada do observador é a saída y mensurada e u , a entrada de controle. A matriz K_e , que é chamada de matriz de ganho do observador, uma matriz de ponderação para a correção da diferença entre a saída medida e a saída estimada. Este termo corrige continuamente a saída do modelo e melhora o desempenho do observador.

Diversas outras estruturas de observadores foram propostas ao longo dos anos com diferentes tipos de interconexão, além de formas canônicas baseados em modelos como os descritos nos trabalhos de [KAILATH, 1980], [CHEN et al., 1990] e [OGATA, 2010].

2.2 Observadores Orientados por Dados

Nesta seção são apresentados os trabalhos realizados por diversos pesquisadores ao longo das últimas décadas, apresentando formulações que estão ligadas direta ou indiretamente com o

problema de observação de estados. Brevemente discuti-se sobre trabalhos científicos e aplicados, com as referências das principais formulações e métodos não baseados em modelos, alimentados por dados ou baseado em dados, sejam eles para treinamento online ou offline de abordagens associadas a sistemas dinâmicos lineares ou não lineares, discretos ou contínuos, determinísticos ou estocásticos, relacionados a projetos de observadores/controladores.

Embora possamos classificar de diversos modos os observadores orientados por dados, aqui selecionamos uma classificação com relação aos métodos associados a sistemas lineares e não lineares, como é apresentado na Figura 3. Nesta classificação, os métodos não baseados em modelos para sistemas lineares, tem fundamento em técnicas de identificação de subespaço, aplicadas ao projeto de controle Linear Quadratic Gaussian (LQG) e controladores $\mathcal{H}_\infty$. Para sistemas não lineares, são descritos metodologias baseadas em Programação Dinâmica Aproximada (ADP), Perturbação Simultânea por Aproximação Estocástica (SPSA) e a metodologia Model Free Adaptive (MFA). As metodologias apresentadas para sistemas não lineares contidas na Figura 3, fazem uso de técnicas de aprendizagem de máquina, baseadas em inteligência artificial, mais especificamente, redes neurais e lógica fuzzy como aproximadores de funções ou como estimadores de parâmetros associados a identificação de sistemas.

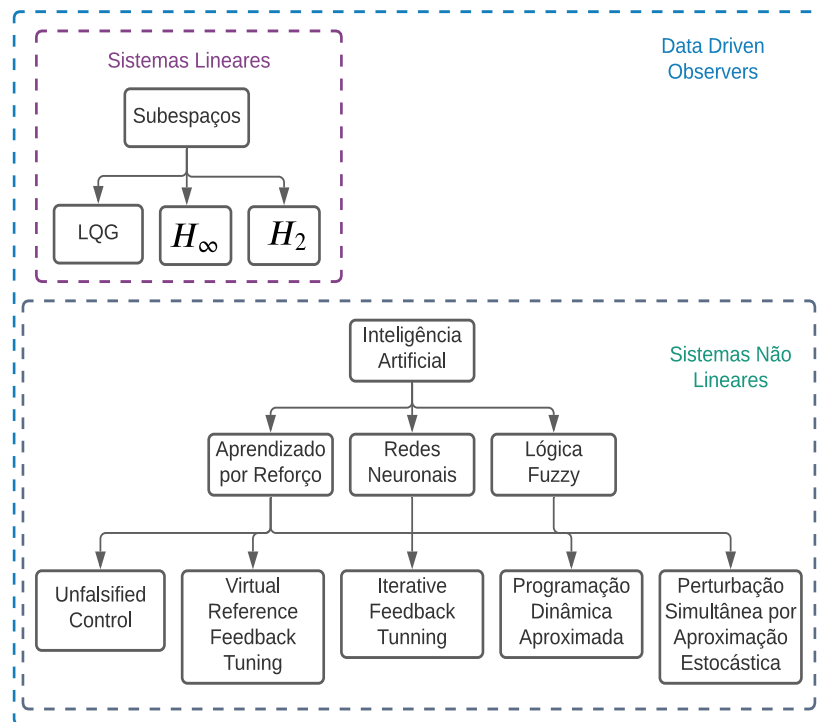


Figura 3 – Abordagens Não baseadas em Modelos

Fonte: Produzido pelos Autores.

Algumas abordagens usadas para sistemas não lineares apresentadas na Figura 3, fazem usos das técnicas de redes neurais, assim como lógica fuzzy, com o propósito de realizar o mapeamento da função de transição de estado desconhecida, por meio da aproximação de

funções a partir dos dados, como um ajuste de curva, ou seja, para identificação da dinâmica das transições de estado e de saída do sistema dinâmico não linear, com a finalidade de incorporar as não linearidades intrínsecas do sistema dinâmico desconhecido. Outro aspecto que justifica a utilização de tais métodos estão relacionados a capacidade de adaptação dessas técnicas pelas suas características adaptativas e de aprendizagem, relacionadas a incorporação de variações na dinâmica da planta, perturbações ou ruídos.

Quando abordamos questões sobre uso de técnicas e metodologias que fazem uso de dados amostrados, para identificação, controle ou estimação, nos referimos ao processo de construção ou aplicação de métodos supondo a existência dos dados por meio de experimentação direta ou fazendo uso de dados gerados por modelos. Neste contexto, consideramos os processos de decisão Markovianos parcialmente observáveis (POMDP), nos quais nem todos os estados são acessíveis por meio de medição direta.

Grande parte do desenvolvimento teórico em sistemas de controle supõem que o sistema seja parcialmente ou completamente observável. Particularmente em relação ao campo de controle ótimo, algumas referências consideram seus sistemas como sendo processos Markovianos parcialmente observáveis, como os trabalhos [SONDIK, 1971] e [FLEMING, 1968], que tratam de problemas de controle ótimo.

Uma revisão de métodos de aproximação para solução de POMDP foi realizada no trabalho de [ABERDEEN, 2003]. No trabalho de [DAVIS; VARAIYA, 1973], são analisadas as condições necessárias para à aplicação de métodos de programação dinâmica para processos de decisão Markoviano parcialmente observáveis.

Trabalhos mais contemporâneos sobre POMDP são tratados por [XU; LIU, 2015], com aplicações em veículos aéreos sob sensoriamento e deslocamentos em presença de incerteza para a navegação com obstáculos, na aplicação de deslocamento para uma localidade específica.

Na abordagem MFA apresentada em [CHENG, 2004], o sistema dinâmico é parcialmente desconhecido e com isso não se possui acesso aos estados para se realizar um controle mais assertivo, assim é desenvolvido uma formulação baseada em uma rede neuronal direta que possui elementos associados internamente para realizar a estimação de estados e com isso realizar o controle. No trabalho é comentado que embora seja possível usar o modelo para controle de sistemas, caixa branca, cinza ou preta, o modelo MFA é mais adequado para sistemas caixa cinza, justificando que a maior parte das plantas industriais são caixa cinza. Também é citando como vantagem da metodologia, não ser necessário um projeto de controlador, ajuste paramétrico manual e nem a necessidade de implantação de qualquer mecanismo de identificação no sistema.

Outra abordagem voltada para o desenvolvimento de controladores não baseados em modelos é apresentada por [DONG; CHEN, 2012], originalmente desenvolvida por [SPALL; CRISTION, 1993], que possui trabalhos sobre a estimação de gradientes de funções não lineares pelo uso de aproximação estocástica. Em [DONG; CHEN, 2012], o observador de estado é

incluído internamente como uma das etapas do processo iterativo de atuação ao sistema dinâmico, por meio do uso de aprendizagem por reforço.

Aplicações com denominações Model-Free, com estado da arte sobre metodologias Model-Free alimentadas por dados são abordados no trabalho de [HOU; XU, 2009]. Trabalhos relacionados a otimização estocástica de sistemas alimentados por dados e por simulações são tratados em [SPALL, 2005]. Projetos de controladores e observadores de estados baseados em dados e projetos de controle ótimo, são abordados em [AANGENENT et al., 2005]. A aplicação de um observador Model-Free aplicado a um quadrotor é apresentado em [YOUNES et al., 2015]. Outra aplicação model-free para controle preditivo baseados em programação dinâmica aproximada é discutido em [DONG; CHEN, 2012]. Uma aplicação de controle em sistemas discretos model-free, é discutido em [SPALL; CRISTION, 1993]. Uma aplicação Model-free para controle preditivo em sistemas de energia solar é tratado em [CHEN et al., 2014].

A ideia original usada por [SPALL, 1992] é baseada na estimação de um controlador sem considerar a existência de um modelo, sendo assim construída pelo uso de um aproximador de função (FA), como uma rede neuronal ou função polinomial. Tal processo envolve a estimação de parâmetros, não sendo pressuposto nenhum conjunto de equações para o sistema, de modo que o gradiente de uma função de custo não estará disponível para uso por algoritmos de otimização padrão. Os trabalhos [SPALL; CRISTION, 1993; SPALL; CRISTION, 1992; DONG; CHEN, 2012] usam algoritmos de aproximação estocástica, que é baseado na aproximação do gradiente por perturbação simultânea, que necessita somente das medidas do sistema e não de um modelo do sistema. O método baseado em SPSA é apresentado como mais eficiente se comparado com algoritmos padrões de aproximação estocástica baseados em aproximadores de gradiente por diferenças finitas [SPALL, 2005].

Em Hou e Wang [2013] é apresentado um survey dos problemas existentes nas abordagens baseadas em modelos e nas metodologias baseados em dados, tanto em projetos de controladores, como em métodos de análise. É feita uma abordagem histórica do controle, aplicações e formulações, assim como aspectos futuros da teoria. Neste trabalho também são apontados as diferenças entre as abordagens baseadas em modelos e metodologias que utilizam somente dados. Outro tópico relevante é discutido com relação a métodos baseados em dados como otimização e modelagem.

As abordagens propostas anteriormente pelos autores citados trabalham com controle preditivo, robusto e adaptativo, não baseados em modelos, sendo que para observadores não baseados em modelo, tem-se poucas referências sobre o assunto. Outro observador de estado não baseados em modelos, podemos citar o "Observador de Thau", juntamente com a abordagem não baseadas em modelos discutidos em [YOUNES et al., 2015]. Neste artigo, um observador não linear Model-Free (NMFO) é proposto para estimar os estados do veículo quadrotor. NMFO é uma estrutura integrada da técnica Model-Free e do observador não linear Thau. O observador Thau irá acomodar as não linearidades no modelo quadrotor e o observador Model-Free irá

compensar a dinâmica não modelada, erros de modelagem e incertezas do sistema. Os resultados dos testes de voo validaram o desempenho do observador proposto para condições livres de falha e de falha do atuador.

A aplicação de métodos de subespaço, usando projeções e sem o uso de modelo, somente com dados, com aplicações de técnicas de controle ótimo do tipo Linear Quadrático Gaussiano (LQG) são abordados em [FAVOREEL et al., 1999a] e [FAVOREEL et al., 1999b], com projeto de controladores robustos do tipo $\mathcal{H}_2$, tratado em [HINNEN et al., 2008] e $\mathcal{H}_\infty$ abordado em [WOODLEY et al., 2001]. Tal abordagem resolve o problema da inclusão de um observador desacoplado, pois calcula diretamente o ganho do controlador, solucionando assim o problema da disponibilidade dos estados no projeto do Regulador Linear Quadrático. No projeto do LQG, que inclui um observador de estado ao sistema para fornecer os estados para que o controlador realize o controle adequadamente, visto que uma série de operações são necessárias para aplicação dessas metodologias, como a solução da equação de *Riccati* para computar o ganho do controlador e do observador. Uma extensão do método proposto pelos autores, com aplicações em controle $\mathcal{H}_\infty$, discutidos em [WOODLEY et al., 2001], outras aplicações podem ainda ser vistas nos trabalhos de Chiera e White [2008] e Chen et al. [2014].

No livro [HOU ZHONGSHENG; JIN, 2013] é apresentado tanto o controle baseado em modelo quanto o controle orientado por dados, algumas de suas vantagens e características. A nomenclatura para se referir a metodologias baseadas em dados é ampla, como por exemplo, orientadas ou alimentadas por dados (Data-Driven), metodologias livre de modelos, sem modelo ou não baseado em modelo (Model-Free). Neste livro é realizada a classificação das metodologias não baseadas em modelos de duas formas, uma com relação às metodologias online, offline ou híbridas e outra classificação com relação ao conhecimento da estrutura do controlador. As principais questões abordadas no livro são: linearização dinâmica, controle adaptativo livre de modelo, controle adaptativo preditivo livre de modelo, controle adaptativo por aprendizagem iterativa para sistemas não lineares discretos tipo SISO, MIMO e análise de estabilidade em aplicações práticas.

Um aspecto de interesse das metodologias sejam elas livres de modelo ou não, é o da sintonia dos seus parâmetros, para melhorar o desempenho e acurácia global dos algoritmos. Uma aplicação para sintonia de controladores via VRFT em controle de sistemas multivariados é discutido no trabalho de Campestrini et al. [2016].

2.3 Programação Dinâmica Aproximada e Aprendizagem por Reforço

Programação Dinâmica aproximada e aprendizagem por reforço estão ligadas por serem abordagens com aplicações não baseadas em modelos relacionadas a ADP e algoritmos de aprendizagem por reforço (RL), que possuem uma família de métodos usados na resolução de

problemas de controle ótimo para um tempo futuro, em especial em sistemas discretos. Diversos trabalhos são associados a esses métodos, desde estudos sobre o problema de estabilidade, assim como aplicações, como nos trabalhos [XU; JAGANNATHAN, 2014], [BALAKRISHNAN et al., 2008] e [JIANG; JIANG, 2013]. Alguns métodos são usados em conjunto com as formulações de ADP, como SPSA. Ambas metodologias têm o propósito de realizar o controle preditivo e rastreamento em sistemas não lineares, com o uso também de redes neuronais como apresentado no trabalho de Dong e Chen [2012].

Metodologias baseadas em programação dinâmica e aprendizagem por reforço também relacionadas ao processo Markoviano parcialmente observável são amplamente estudados por pesquisadores. Aplicações para a determinação e computação de políticas ótimas, com uso de notações compactas é discutido em [BOUTILIER; POOLE, 1996]. Trabalhos relacionados a aprendizagem de sistemas, sem estimação de estados foram realizados e discutidos em [SINGH et al., 1994]. Outra aplicação de métodos para aproximação da função valor em sistemas parcialmente observáveis é abordado no trabalho [HAUSKRECHT, 2000].

As abordagens anteriormente citadas abordam formulações não baseadas em modelos para identificação de sistemas e estimação de estados, apontando diversas dificuldades, tanto para a formulação como também para sintonia, treinamento e implementação das metodologias. Assim justifica-se o trabalho aqui apresentado, relacionado com o desenvolvimento de métodos não baseados em modelos para estimação dos estados, levando em consideração os aspectos relacionados a sintonia, análise de convergência e robustez, não esquecendo os aspectos mais interessantes dessas metodologias que estão relacionadas a aprendizagem da dinâmica que pode ser feita on-line ou por etapas. Nesse sentido devemos enfatizar na adaptabilidade intrínseca da aprendizagem por reforço para aprendizado on-line.

Em [MODARES et al., 2014], são aplicadas técnicas de aprendizagem por reforço com ação integral, tendo conhecimento somente de informação parcial para projeto de controladores on-line. Outras abordagens da aplicação de controle com informação parcial e aprendizagem são discutidas por Safonov e Tsao [1997] e Battistelli et al. [2014]. Nestas referências são avaliadas as possibilidades de se identificar ações de controle, que sejam capazes de atender às especificações de desempenho, antes de serem inseridas no processo de realimentação. Esta técnica é chamada de "Unfalsified Control". Este tipo de metodologia é apresentada como um controlador adaptativo para seleção das ações de controle mais adequadas, com aplicações em tempo real. Outros projetos de controladores on-line, on-line adaptativo e controle de veículos aéreos orientados por dados utilizando a metodologia "Unfalsified Control" e discutidos nos trabalhos [SAEKI et al., 2014], zciteunfalsified2015data, [LIMING et al., 2015] e [SAJJANSHETTY; SAFONOV, 2015].

Aplicações relacionadas ao uso de dados com a aplicação da programação dinâmica aproximada e aprendizagem por reforço tem sido largamente aplicadas no controle de sistemas dinâmicos. Aplicações para a estabilização ótima em sistemas não lineares discretos utilizando a aproximação de valor por iteração são abordadas em [LI et al., 2014]. Aplicações para controle de

sistemas discretos com atrasos, usando aprendizagem por reforço são apresentadas em [ZHANG et al., 2016]. Outro trabalho tratando de problemas de rastreamento de sistemas discretos com múltiplos atrasos é discutido em [LIU et al., 2016]. Aplicação de técnicas de aprendizagem por reforço para controle on-line, com uso de inferência Bayesiana são apresentados em [XIA; ZHAO, 2016]. Aplicação em controle de sistemas, com tolerância a falhas com realimentação de saída em sistema com dinâmica desconhecida, é discutido em [WANG; YANG, 2016b]. Em [LUO et al., 2016], uma aplicação para o problema de rastreamento ótimo via aplicação da técnica "*Q*-learning" somente para a função de avaliação do crítico, que é responsável por avaliar se a política de controle é adequada e se precisa de atualização.

Aplicações utilizando o algoritmo *Q*-learning, para análise de convergência para sistemas discretos determinísticos é discutida em [WEI et al., 2017]. Aplicações para sistemas discretos não lineares com iteração de política dinâmica são abordados nos trabalhos de Wei e Liu [2015b] e Wei e Liu [2015a].

Controladores do tipo PID com o uso de técnicas baseadas em aprendizagem por reforço, como o projeto para controladores PID adaptativos em braços robóticos, são apresentados em [AKBARIMAJD, 2015]. Outro projeto de controlador PID usando a lógica Fuzzy, aplicando o algoritmo *Q*-Learning com função de recompensa manipulada é discutido em [AGHAEI et al., 2015]. Outra aplicação de ADP, baseada no ator-crítico adaptativo, com um projeto de ator-crítico adaptativo, em que o ator tem o papel de escolher qual a ação será aplicada na planta e o crítico de avaliar o quão boa é a política e de ajustá-la sempre que necessária. Em [FAN; YANG, 2016] é abordado o projeto de controle em modo deslizante com ação integral para sistemas não lineares parcialmente desconhecidos, com distúrbios no sinal de entrada. Neste trabalho existem perturbações na ação de controle, são aplicados dois observadores, um baseado em uma rede neuronal para realizar a aproximação da função dos termos desconhecidos não lineares e outro observador para realizar a estimação das perturbações sobre as ações de controle.

Um trabalho com abordagem ADP que usa dados off-line para iterativamente realizar a aproximação do controle ótimo de um sistema dinâmico não linear discreto desconhecido é apresentado por Li et al. [2014]. Outra aplicação com base em dados, temos o projeto de controladores ótimo e robusto orientado por dados para sistemas em cascata incertos, usando iteração de valor, que é abordada em [BIAN; JIANG, 2015].

Algoritmos baseados em ADP e RL tem forte relação com o desempenho computacional para sua implementação. Estudos com relação a metodologias para acelerar o processo de convergência de algoritmos de iteração de valor em sistemas que consideram o POMDP, são discutidos no trabalho de Zhang e Zhang [2001]. Para o caso de problemas de horizonte finito e infinito aplicados em controle ótimo, temos os trabalhos de Smallwood e Sondik [1973] e Sondik [1978], respectivamente. Alguns Surveys com a fundamentação teórica, estado da arte, teoria, modelos, algoritmos e tutorial para POMDP, são abordados nos trabalhos de Monahan [1982], Lovejoy [1991], Zhang [2001], Littman [2009] e III [1991].

Capítulo 3

Fundamentação Teórica

No intuito de apresentar a fundamentação teórica desta Tese de forma sistematizada, são apresentados os principais elementos da Tese para o desenvolvimento de modelos de observadores de estados baseados em modelos e orientados por dados para monitoração e controle, tendo como tema central a formulação do problema de estimação de estados por observação de sistemas dinâmicos discretos alimentados por dados baseados em programação dinâmica aproximada com o projeto de controlador ótimo online, baseado no Regulador Linear Quadrático Discreto (DLQR), com realimentação de saída. Outros aspectos também abordados na Tese, estão relacionados a formulações de métricas usadas pelas heurísticas propostas para sintonia, análise de desempenho e sensibilidade paramétrica, visando avaliar a robustez dos algoritmos apresentados.

A fundamentação teórica é dividida em quatro partes, sendo a primeira parte a Seção Observabilidade em Sistemas Dinâmicos Não Lineares, tratando de uma formulação mais geral baseada em conceitos topológicos para a definição e caracterização sobre observabilidade, bem como as definições matemáticas necessárias para o entendimento das formulações, com alguns exemplos. A segunda parte trata dos conceitos sobre Observabilidade e Observadores de Estado em Sistemas Lineares, abordando os conceitos clássicos para observabilidade em sistemas lineares com a estrutura de observadores de ordem completa e o problema dual, para alocação de polos entre controlador e observador. A terceira parte trata do projeto de observadores e do projeto baseado em transformação, considerando a abordagem apresentada na primeira seção. A quarta parte aborda a metodologias livre de modelo que contem metodologias de aprendizagem por reforço e programação dinâmica aproximada com realimentação por saída, que a partir de uma formulação matemática nos fornecerá a estrutura que contém uma formulação teórica para problema de estimação de estado, não solucionada em [LEWIS; VAMVOUDAKIS, 2011] e que será abordada neste trabalho com vários desenvolvimentos e propostas que solucionam o

problema.

3.1 Observabilidade em Sistemas Dinâmicos Não Lineares

Nesta seção são apresentados os conceitos relacionados a observação em sistemas dinâmicos, juntamente com as condições analíticas e geométricas para observabilidade em sistemas dinâmicos não lineares. Projetos para observadores em sistemas lineares e não lineares são apresentados, assim como o projeto baseado em transformação. Será ainda discutido o caso particular do observador de ordem completa e a dualidade do problema de observação com relação ao problema dual de alocação de polos para sistemas realimentados, assim como as condições necessárias e suficientes para observabilidade em sistemas lineares.

O observador de estado estima as variáveis de estado de um sistema dinâmico, baseados na observação do mesmo, ou seja, pelas saídas mensuradas do sistema e ações de controle passadas [CHEN, 1995]. Anteriormente a esse processo, devemos primeiramente nos certificar que o sistema seja observável. O conceito de observabilidade tratando nesta Tese refere-se a observação de sistemas dinâmicos, sejam eles lineares ou não lineares.

Independentemente se um sistema ser linear ou não linear, ambos possuem suas próprias características, o foco desta seção é avaliar as propriedades necessárias e suficientes para a caracterização do conceito de observabilidade em sistemas dinâmicos, com base nas condições geométricas e analíticas para observabilidade em sistemas dinâmicos definidas nas seções seguintes. As principais formulações e definições apresentadas neste trabalho tem como base os trabalhos de Kailath [1980], Isidori [1995], Chen [1995] e Guildas [2007].

3.1.1 Caracterização Sistemas Dinâmicos Não Lineares

Nesta seção é apresentado a formulação do sistema dinâmico não linear para a definição dos observadores de estado, no sentido mais geral. A caracterização topológica do sistema dinâmico é apresentada com base na geometria diferencial, mas especificamente com relação à definição dos espaços topológicos como sendo variedades diferenciáveis, para então ser possível definir o conceito de indistinguibilidade e assim as definições de observador, observabilidade, espaço de observação e as condições geométricas e analíticas para a observabilidade [GUILDAS, 2007].

Considere o sistema dinâmico não linear genérico:

$$\begin{aligned}\dot{x}(t) &= f(x(t), u(t)), \\ y(t) &= h(x(t)),\end{aligned}\tag{3.1}$$

- $x \in X$, vetor de estados de uma variedade X de dimensão n
- $u \in U$, vetor de entradas externas conhecidas de um subconjunto aberto U de $\mathbb{R}^m$

- $y \in Y$, vetor das saídas mensuradas de um subconjunto aberto Y de $\mathbb{R}^p$
- f e h sendo funções, em geral assumi-se serem de classe C^∞ , com relação aos seus argumentos e as funções de entrada $u(\cdot)$ serem localmente limitadas e mensuráveis em um conjunto $\mathcal{U}$

Mais geralmente, um sistema pode depender do tempo explicitamente, por meio de $f(x(t), u(t), t)$, enquanto y pode depender diretamente de u e mesmo de t , por meio de $h(x(t), u(t), t)$. Esta dependência explícita do tempo é chamada de "variante no tempo" e generaliza o sistema (3.1),

$$\begin{aligned}\dot{x}(t) &= f(x(t), u(t), t) \\ y(t) &= h(x(t), u(t), t).\end{aligned}\tag{3.2}$$

Após a apresentação do sistema dinâmico seguiu-se para as definições das condições geométricas e analíticas do projeto de observadores, com base na equivalência entre sistemas e do projeto por transformação.

3.1.2 Observadores

Uma definição mais geral da estrutura que chamada de observador de estado para sistemas dinâmicos não lineares é apresentada, com a caracterização das propriedades associadas a estrutura mínima que um observador deve possuir para ser caracterizado como observador de estado.

Definição 1 (Observador): Considere o sistema (3.1). Um observador é dado por um sistema auxiliar [GUILDAS, 2007],

$$\begin{aligned}\dot{\hat{X}}(t) &= F(\hat{X}(t), u(t), y(t), t) \\ \hat{x}(t) &= H(\hat{X}(t), u(t), y(t), t)\end{aligned}\tag{3.3}$$

Tal que:

- (i) $\hat{x}(0) = x(0) \Rightarrow \hat{x}(t) = x(t), \forall t \geq 0$;
- (ii) $\|\hat{x}(t) - x(t)\| \rightarrow 0$ quando $t \rightarrow \infty$

Se (ii) é valido para qualquer $x(0)$, $\hat{x}(0)$, o observador é global.

Se (ii) é valido com convergência exponencial, o observador é exponencial.

Se (ii) é valido com taxa de convergência que pode ser sintonizada, o observador é sintonizável.

Apesar desta definição não ser usual, a mesma é bem geral e possui uma representação mais genérica, representando de forma mais abrangente o comportamento de variação do espaço de estados e do estimador em função do espaço de estado e entradas/saídas. Essa definição apresenta a dinâmica das variedades cujos elementos são as funções dos espaços de estados, para

um dado sinal de entrada e saída para um dado instante, dependentes de uma função de transição F . Além do mais, explicita o estimador de estados como função do espaço de estado, do sinal de entrada e saída mensurada em um dado instante.

Em geral $\mathcal{X}_u(t, x_{t_0})$ denotará a solução da equação de estado (3.1), para a aplicação de uma entrada u em $[t_0, t]$, que satisfaça $\mathcal{X}_u(t, x_{t_0}) = x_{t_0}$, sendo u omitido para casos não controlados.

As condições geométricas e analíticas para a observabilidade em sistemas dinâmicos são apresentadas posteriormente, assim com o projeto de observadores com base em transformação.

3.1.2.1 Condições Geométricas para Observabilidade

Nesta seção são apresentados as condições geométricas associadas ao conceito de indistinguibilidade entre estados e a definição do conceito de observabilidade com relação a um estado particular x_0 , assim como a definição do espaço de observação e um teorema que associa observabilidade local a observabilidade fraca [GUILDAS, 2007].

As condições geométricas aqui apresentadas são baseadas na teoria de controle geométrico apresentadas nos trabalhos de Isidori [1995] e Guildas [2007].

Definição 2 (Indistinguibilidade): O par $(x_0, x'_0) \in \mathbb{R}^n \times \mathbb{R}^n$ é indistinguível para o sistema (3.1), se for verificada a seguinte condição [ISIDORI, 1995; GUILDAS, 2007]:

$$\exists u \in \mathcal{U} \forall t \geq 0, h(\mathcal{X}_u(t, x_0)) = h(\mathcal{X}_u(t, x'_0)) \quad (3.4)$$

Um estado x é indistinguível de x_0 , se o par (x, x_0) é indistinguível.

Definição 3 (Observabilidade)[com relação a x_0]: O sistema (3.1) será considerado observável [com relação a x_0], se ele não admitir nenhum par indistinguível (com relação a qualquer estado indistinguível de x_0) [GUILDAS, 2007; ANGUELOVA, 2004; MORGANSEN, 2004].

A condição dada pela Definição 2 é bastante geral (global) e pouco aplicável de forma prática, devido principalmente à sua representação global da observabilidade e o fato de que o observador deve estar principalmente interessado nos estados distinguíveis de seus vizinhos.

Por exemplo, considere o caso do seguinte sistema:

$$\begin{cases} \dot{x} = u, \\ y = \sin(x). \end{cases} \quad (3.5)$$

claramente y não ajuda a distinguir entre x_0 e $x_0 + 2k\pi$, portanto o sistema é não observável. Ao passo que y permite distinguir os estados $]-\frac{\pi}{2}, \frac{\pi}{2}[$.

Isto nos leva a considerar uma versão mais fraca de observabilidade:

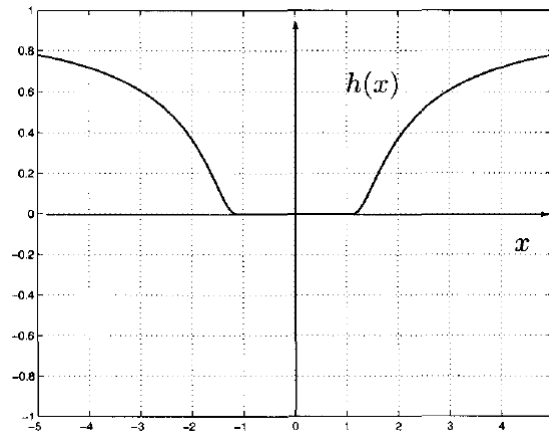


Figura 4 – Função de saída de um Sistema Fracamente Observável, mas não localmente.

Fonte: Figura extraída do livro: [GUILDAS, 2007].

Definição 4 (Observabilidade Fraca)[com relação a x_0]: O Sistema 3.1 é considerado fracamente observável (com relação a x_0), se existir uma vizinhança U para algum x (com relação a x_0), tal que não exista nenhum estado indistinguível de x (com relação a x_0) em U [GUILDAS, 2007; ANGUELOVA, 2004].

Observe que isso não impede os casos em que as trajetórias têm que ir muito longe de U , antes que se possa distinguir entre dois estados de U .

Por exemplo, considere o sistema:

$$\begin{cases} \dot{x} = u, \\ y = h(x). \end{cases} \quad (3.6)$$

com $h(x)$ uma função de classe C^∞ como na Figura 4. Claramente o sistema é fracamente observável, visto que qualquer estado é distinguível de qualquer outro, aplicando alguma entrada u não nula, mas distinguir quaisquer dois estados no intervalo entre $[-1, 1]$ é necessário esperar que y se mova para além de 0.

Portanto, para prevenir esta situação é necessário uma definição mais local de observabilidade, definida a seguir.

Definição 5 (Observabilidade Local Fraca)[com relação a x_0]: O Sistema (3.1) é localmente fracamente observável, com relação a x_0 , se existir uma vizinhança U para qualquer x (com relação a x_0), tal que para qualquer vizinhança V de x (com relação a x_0) contida em U , não exista nenhum estado indistinguível de x em V , considerando os intervalos de tempo para o qual as trajetórias permaneçam em V [ANGUELOVA, 2004; GUILDAS, 2007].

Isso significa que se pode distinguir cada estado de seus vizinhos sem "ir muito longe". Esta noção é mais interessante na prática e também apresenta a vantagem de admitir alguma caracterização de "condição de posto".

Tal condição se baseia na noção de *espaço de observação*, correspondendo aproximadamente ao espaço de todos os estados observáveis.

Definição 6 (Espaço de Observação): O Espaço de Observação ($\mathcal{O}(h)$) para o sistema (3.1), é definido como sendo, o menor espaço vetorial real de funções de classe $\mathcal{C}^\infty$, contendo as componentes de h e fechado com relação a derivada de Lie ao longo de $f_u := f(\cdot, u)$ para qualquer $u \in \mathbb{R}^m$ (ou seja, tal que para qualquer $\varphi \in \mathcal{O}(h)$, $L_{f_u}\varphi \in \mathcal{O}(h)$, sendo a derivada de Lie $L_{f_u}\varphi = \frac{\partial h}{\partial x} f(x, u)$) [ISIDORI, 1995; HERMANN; KRENER, 1977].

Definição 7 (Condição de Posto para Observabilidade)[com relação a x_0]: Um Sistema (3.1) é dito satisfazer a condição de posto para observabilidade (com relação a x_0) se:

$$\forall x, \dim d\mathcal{O}(h)|_x = n, \text{ (com relação a } d\mathcal{O}(h)|_{x_0} = n), \quad (3.7)$$

sendo $d\mathcal{O}(h)|_x$ é o conjunto de $d\varphi(x)$ com $\varphi \in \mathcal{O}(h)$.

Teorema 1: Um sistema (3.1) satisfazendo a condição de posto para observabilidade em x_0 é fracamente localmente observável em x_0 .

Mais geralmente um sistema (3.1) satisfazendo a condição de posto de observabilidade é localmente fracamente observável.

Reciprocamente, um sistema (3.1) localmente fracamente observável, satisfaz as condições de posto para observabilidade em um subconjunto aberto denso de X .

Resumidamente segue dos fatos já mencionados:

- i) A condição do posto de observabilidade em algum x_0 , significa a existência de n elementos do espaço de observação, definindo um difeomorfismo em torno de x_0 ;
- ii) Para qualquer par $(x_0; x'_0)$ indistinguível e qualquer elemento de $\varphi \in \mathcal{O}(h)$, $\varphi(x_0) = \varphi(x'_0)$.

Como exemplo da aplicação, considere o sistema (3.5): para esse sistema fica claro, $d\mathcal{O}(h) = \text{span}\{\cos(x)dx, \sin(x)dx\}$ e portanto $d\mathcal{O}(h)|_{x_0} = 1$ para qualquer x_0 , ou seja, o sistema satisfaz a condição de posto para observabilidade.

Na próxima seção serão apresentadas as condições necessárias para a observabilidade com relação à excitação do sistema, ou seja associadas às entradas do sistema e o quanto essas entradas influenciam na dinâmica de observação do sistema.

3.1.2.2 Condições Analíticas para Observabilidade

Neste seção são apresentadas as condições analíticas para observadores de estado, relacionadas às definições de entrada universal, entrada persistente e entrada regular, assim como

observabilidade uniforme com relação à entrada universal, em um horizonte de tempo finito. É definido ainda o conceito de codistribuição para observabilidade, que avalia o comportamento de variação do espaço de observação do sistema dinâmico segundo [Guildas \[2007\]](#).

Definição 8 (Entrada Universal)[com relação a x_0]: Uma entrada u é universal com relação ao sistema (3.1) se

$$\forall x_0 \neq x'_0, \exists \tau \geq 0, \tau \in [0, t] \mid h(\mathcal{X}_u(\tau, x_0)) \neq h(\mathcal{X}_u(\tau, x'_0)) \quad (3.8)$$

Uma entrada u é entrada singular se ela não for universal.

Definição 9 (Observabilidade Uniforme): Uma sistema é considerado uniformemente localmente observável, se cada entrada for universal dentro do intervalo de tempo finito $[0, t]$.

Definição 10 (Codistribuição para Observabilidade $d\mathcal{O}$): Codistribuição de observabilidade $d\mathcal{O}$ é a matriz formada pelas componentes de $\mathcal{O}$, sendo cada linha dessa matriz é dada por:

$$d\mathcal{O} = \nabla_x O_i \mid O_i \in \mathcal{O} \quad (3.9)$$

onde, $\nabla_x O_i$, representa o vetor gradiente da função O_i .

Se um sistema satisfaz a condição de posto $\forall x \in \mathcal{X}'$, então o mesmo será localmente fracamente observável em x_0 e se $\mathcal{X}' = \mathcal{X}$, e o mesmo atender a condição do posto de observabilidade $\forall x \in \mathcal{X}'$ e $\forall u \in \mathcal{U}'$, o mesmo será *Uniformemente Observável* em $\mathcal{X}' \times \mathcal{U}'$ e se esta condição for satisfeita para $\mathcal{X}' = \mathcal{X}$ e $\mathcal{U}' = \mathcal{U}$, então o sistema será completamente uniformemente observável (*CUO*).

Proposição 1: Uma entrada u é uma entrada universal em $[0, t]$ para o sistema (3.1), se e somente se, $\exists \tau \geq 0, \tau \in [0, t]$,

$$\int_0^t \|h(\mathcal{X}_u(\tau, x_0)) - h(\mathcal{X}_u(\tau, x'_0))\|^2 d\tau > 0, \forall x_0 \neq x'_0 \quad (3.10)$$

Definição 11 (Entradas Persistentes): Uma entrada u é persistente para um sistema (3.1), se $\exists t_0, T : \forall t \geq t_0, \forall x_t \neq x'_t$,

$$\int_t^{t+T} \|h(\mathcal{X}_u(\tau, x_t)) - h(\mathcal{X}_u(\tau, x'_t))\|^2 d\tau > 0 \quad (3.11)$$

Esta condição garante a observabilidade em um intervalo de tempo, mas não garante a mesma

para quando o tempo tende à infinito.

Definição 12 (Entradas Persistentes Regulares): Uma entrada u é regularmente persistente para um sistema (3.1), se

$$\begin{aligned} \exists t_0, T : \forall x_{t-T}, x'_{t-T}, \forall t \geq t_0, \\ \int_{t-T}^t \|h(\mathcal{X}_u(\tau, x_{t-T})) - h(\mathcal{X}_u(\tau, x'_{t-T}))\|^2 d\tau \geq \beta(\|x_{t-T} - x'_{t-T}\|) \end{aligned} \quad (3.12)$$

para algumas funções β de classe $\mathcal{K}$ [GUILDAS, 2007].

Da definição acima, pode-se usar então a definição usual para sistemas de estado afim.

Proposição 2: Para sistemas de estado afim, entradas regularmente persistente u são entradas que [GUILDAS, 2007]:

$$\exists t_0, T, \alpha : \int_{t-T}^t \Phi_u^T(\tau, t-T) C^T C \Phi_u(\tau, t-T) d\tau \geq \alpha I > 0 \quad \forall t \geq t_0, \quad (3.13)$$

com $\Phi_u(\tau, t)$ a matriz de transição classicamente definida por:.

$$\frac{d\Phi_u(\tau, t)}{d\tau} = A(u(\tau))\Phi_u(\tau, t), \quad \Phi_u(t, t) = I. \quad (3.14)$$

Essa é uma consequência direta da aplicação da Definição 12, para o caso de sistemas de estado afim, com $\beta(\|z\|) = \alpha \|z\|^2$

O lado esquerdo da desigualdade na Eq.(3.13) corresponde ao *Gramiano de Observabilidade*, classicamente definido para sistemas variantes no tempo (LTV), para qualquer $t_1 < t_2 \in \mathbb{R}$, como:

$$\Gamma(t_1, t_2) = \int_{t_1}^{t_2} \Phi^T(\tau, t_1) C^T(\tau) C(\tau) \Phi(\tau, t_1) d\tau, \quad (3.15)$$

onde Φ denota a matriz de transição de estados para a parte autônoma do sistema.

Definição 13 (Entradas Persistentes Regulares Locais): Uma entrada u é localmente regularmente persistente para um sistema (3.1), se $\exists T_0, \alpha : \forall x_{t-T}, x'_{t-T}, \forall T \leq T_0, \forall t \geq T$,

$$\int_{t-T}^t \|h(\mathcal{X}_u(\tau, x_{t-T})) - h(\mathcal{X}_u(\tau, x'_{t-T}))\|^2 d\tau \geq \beta(\|x_{t-T} - x'_{t-T}\|, \frac{1}{T}), \quad (3.16)$$

para algumas funções β de classe $\mathcal{KL}$ [GUILDAS, 2007].

Esta propriedade garante observabilidade para pequenos instantes de tempo. Obviamente quando T tende para zero, a observabilidade não pode ser garantida, o que explica a característica de decaimento de β .

Após as definições da estrutura de um observador e das condições geométricas e analíticas para a observabilidade, apresenta-se a seguir o projeto de observadores com base nas definições e com os conceitos de observadores uniformes, não uniformes e o projeto de observadores lineares, não lineares e baseados em transformação.

3.2 Observabilidade e Observadores de Estado em Sistemas Lineares

Nesta seção são abordados os conceitos gerais sobre observabilidade e observadores de estado para sistemas lineares, sempre que possível estimar todos os estados internos do sistema. Algumas definições clássicas são apresentadas para um melhor entendimento inicial.

Considere o sistema não forçado descrito pelas equações:

$$\begin{aligned}\dot{x} &= Ax, \\ y &= Cx,\end{aligned}\tag{3.17}$$

sendo, $x \in \mathbb{R}^n$ o vetor de estados, $y \in \mathbb{R}^m$ o vetor de saída, $A \in \mathbb{R}^{(n \times n)}$ a matriz dinâmica do sistema, $C \in \mathbb{R}^{(m \times n)}$ a matriz de saída do sistema.

O sistema (3.17) é dito ser completamente observável se qualquer estado $x(t_0)$ pode ser determinado a partir das observações $y(t)$ em um tempo finito, $t_0 \leq t \leq t_1$. O sistema é portanto, completamente observável se a cada transição de estado ocorre mudança em cada elemento do vetor de saída. O conceito de observabilidade é útil na resolução de problemas em que é necessário realizar a reconstrução dos estados diretamente a partir dos dados mensurados. Como resultado é necessário, assim, estimar as variáveis que não podem ser medidas diretamente. Esse processo de estimar as variáveis de estado só é possível, se e somente se, o sistema for completamente observável. Nesta seção os sistemas lineares invariantes no tempo são abordados. Portanto, sem perda de generalidade, assumi-se que $t_0 = 0$.

A razão para o sistema (3.17) ser não forçado é a seguinte. Se o sistema é descrito pelas equações

$$\begin{aligned}\dot{x} &= Ax + Bu, \\ y &= Cx + Du,\end{aligned}\tag{3.18}$$

então,

$$x(t) = e^{At}x(0) + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau,\tag{3.19}$$

e

$$y(t) = Ce^{At}x(0) + C \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau + Du.\tag{3.20}$$

Desde que as matrizes A , B , C e D sejam conhecidas e $u(t)$ também seja conhecido, os últimos dois termos do lado direito da equação (3.20) são valores conhecidos. Portanto, podem ser subtraídos dos valores observados de $y(t)$. Sendo assim, para investigar as condições necessárias e suficientes para a observabilidade completa de um sistema é suficiente o sistema descrito pela equação (3.17).

Uma definição clássica para observabilidade é apresentada por [Chen \[1995\]](#) em seu livro sobre sistemas lineares, que é apresentada a seguir.

Definição 14: [Observabilidade]: O sistema (3.18) é dito ser observável, se para qualquer estado inicial desconhecido $x(0)$, existe $t_1 > 0$, tal que o conhecimento da entrada u e saída y em $[0, t_1]$ seja suficiente para determinar unicamente o estado inicial $x(0)$. Caso contrário, o sistema é dito ser não observável.

3.2.1 Observabilidade Completa em Sistemas Contínuos

Considere o sistema descrito em (3.17). O vetor de saída $y(t)$ é

$$y(t) = Ce^{At}x(0). \quad (3.21)$$

Pode-se reescrever e^{At} , após aplicação da formula de interpolação de Sylvester como segue:

$$e^{At} = \sum_{k=0}^{n-1} \alpha^k(t) A^k, \quad (3.22)$$

onde n é o grau do polinômio característico do sistema (3.17).

Portanto tem-se que

$$y(t) = \sum_{k=0}^{n-1} \alpha^k(t) C A^k x(0), \quad (3.23)$$

ou

$$y(t) = \alpha_0(t) C x(0) + \alpha_1(t) C A x(0) + \cdots + \alpha_{n-1}(t) C A^{n-1} x(0). \quad (3.24)$$

Se o sistema é completamente observável, então dada a saída $y(t)$ no intervalo $0 \leq t \leq t_1$, $x(0)$ é unicamente determinado pela equação (3.24), e o posto da matriz $(nm \times n)$, [[OGATA, 2010](#)],

$$\begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}, \quad (3.25)$$

é n .

A partir desta análise, pode-se realizar a definição de observabilidade completa como segue: O sistema (3.17), é completamente observável, se e somente se, a matriz $(n \times nm)$

$$[C \mid AC \mid \cdots \mid (A)^{n-1}C], \quad (3.26)$$

tem posto cheio, ou seja, de ordem n . Esta matriz é chamada de *matriz de observabilidade* [CHEN, 1995], [OGATA, 2010].

3.2.2 Observadores de Estado em Sistemas Lineares

No projeto de controle de sistemas com abordagem de alocação de polos, presume-se que todas as variáveis de estados estejam disponíveis para a realimentação. Na prática nem sempre todos os estados estão disponíveis para a realimentação, sendo assim necessário a estimação dos estados não disponíveis. O processo de estimar os estados não disponíveis é chamado de estimação de estado ou simplesmente de observador.

Se um observador de estado observa todas as variáveis de estado de um sistema, o chama-se de observador de ordem completa, ao passo que se são observadas somente as variáveis que não são mensuradas ou mesmo uma quantidade de variáveis menor que a dimensão do vetor de estados, então o chamamos de observador de ordem reduzida.

3.2.3 Observadores de Estado de Ordem Completa

Nesta seção discuti-se os observadores de estado de ordem completa, na planta descrita pelo sistema (2.1) e o modelo do observador definido pelo sistema (2.2).

Para obtermos a equação do erro do observador, subtraímos as equações das dinâmicas dos estados e estados estimados dados dos sistemas (2.1) e (2.2):

$$\begin{aligned} \dot{x} - \dot{\hat{x}} &= Ax - A\hat{x} - K_e(Cx - C\hat{x}) \\ &= (A - K_eC)(\dot{x} - \dot{\hat{x}}) \end{aligned} \quad (3.27)$$

Seja a diferença entre x e $\hat{x}$ como o vetor de erro e ,

$$e = x - \hat{x} \quad (3.28)$$

Então a equação (3.27) se torna

$$\dot{e} = (A - K_eC)e \quad (3.29)$$

Da equação (3.29), percebe-se que o comportamento dinâmico do vetor de erros é determinado pelos autovalores da matriz $A - K_eC$. Se a matriz $A - K_eC$ é uma matriz estável, o vetor de erros irá convergir para zero, para qualquer vetor de erros iniciais $e(0)$. Isto é, $\hat{x}$

irá convergir para x , independentemente das condições iniciais $x(0)$ e $\hat{x}(0)$. Se os autovalores da matriz $(A - K_e C)$ possuírem comportamento dinâmico do vetor de erros assintoticamente estável e adequadamente rápidos, então qualquer vetor de erros tenderá para zero com velocidade adequada.

Se a planta é completamente observável, então pode ser provado que é possível escolher a matriz K_e , tal que $A - K_e C$ possua autovalores desejados arbitrários. Isto é, a matriz de ganho do observador K_e pode ser determinada de modo a produzir a matriz $A - K_e C$ desejada.

Outras formulações de observadores de ordem reduzida além de outras estruturas de observadores com formas canônicas são discutidos nos trabalhos de Kailath [1980], Chen [1995] e Ogata [2010].

Na seção seguinte apresenta-se o problema de alocação de polos para o controle de um sistema é dual ao projeto de um observador de estados, fazendo com que o a alocação do observador seja realizado de modo análogo ao problema de controle.

3.2.4 O Problema Dual

O problema do projeto de observadores de ordem completa, é o de determinar a matriz de ganho K_e , tal que a dinâmica do erro definida pela equação (3.29) seja assintoticamente estável com velocidade de resposta suficiente, determinado pelos autovalores da matriz $A - K_e C$. Portanto, o projeto de observadores de ordem completa torna-se o de determinar a matriz K_e , tal que os autovalores da matriz $A - K_e C$, sejam alocados como desejados. Assim o problema do projeto do observador se torna equivalente ao da alocação de polos. De fato os dois problemas são matematicamente equivalentes. Esta propriedade é chamada de dualidade.

Considere o sistema definido por

$$\begin{aligned}\dot{x} &= Ax + Bu, \\ y &= Cx.\end{aligned}\tag{3.30}$$

No projeto de observadores de estado de ordem completa, podemos solucionar o problema dual, isto é, o problema da alocação de polos do sistema dual

$$\begin{aligned}\dot{z} &= A^* z + C^* v, \\ s &= B^* z,\end{aligned}\tag{3.31}$$

considerando $v = -Kz$, o "sinal de controle".

Se o sistema dual é completamente controlável, então a matriz de ganho K do sistema realimentado, pode ser determinada tal que $A^* - C^* K$ terá um conjunto de autovalores desejados.

Se $\mu_1, \mu_2, \dots, \mu_n$ são os autovalores desejados da matriz do observador de estado, portanto tomando o mesmos μ_i 's, como autovalores desejados da matriz de ganho do sistema realimentado do sistema dual temos,

$$|sI - (A^* - C^* K)| = (s - \mu_1)(s - \mu_2) \cdots (s - \mu_n).$$

Note que os autovalores de $A^* - C^*K$ e os de $A - K^*C$ são os mesmos, assim temos

$$|sI - (A^* - C^*K)| = |sI - (A - K^*C)|. \quad (3.32)$$

Comparando o polinômio característico $|sI - (A - K^*C)|$ e o polinômio característico $|sI - (A - K_e C)|$ para o sistema do observador (2.2), temos que K_e e K^* estão relacionados, tal que $K_e = K^*$.

Portando usando a matriz K determinada por alocação de polos do sistema dual, a matriz de ganho do observador K_e do sistema original, pode ser determinada usando a relação $K_e = K^*$.

Na seção seguinte apresentaremos as condições necessárias e suficientes par a observação de estados em sistemas lineares invariantes no tempo.

3.2.5 Condições Necessárias e Suficientes para Observação de Estados para Sistemas Lineares

Como já abordado, as condições necessárias e suficientes para a determinação da matriz de ganho do observador K_e para os autovalores desejados de $A - K_e C$ é que o dual do sistema original

$$\dot{z} = A^* z + C v, \quad (3.33)$$

seja completamente controlável. A condição de controlabilidade para o sistema dual é que o posto da matriz

$$[C^* \mid A^* C^* \mid \dots], \quad (3.34)$$

seja de ordem n .

Esta condição de observabilidade completa do sistema original (3.30), significa que a condição necessária e suficiente para a observação do estado do sistema (3.30), é que o sistema seja completamente observável [CHEN, 1995].

Uma vez selecionados os autovalores desejados, o projeto do observador de estado de ordem completa pode ser construído, desde que a planta seja completamente observável. Os autovalores desejados da equação característica devem ser escolhidos, tal que o observador de estado possua resposta pelo menos três ou cinco vezes mais rápida que o sistema de malha fechada [OGATA, 2010]. Como abordado anteriormente, a equação do observador de ordem completa é:

$$\begin{aligned} \hat{x} &= A\hat{x} + Bu + K_e(y - C\hat{x}), \\ &= (A - K_e C)\hat{x} + Bu + K_e y. \end{aligned} \quad (3.35)$$

Assumi-se que as matrizes A , B e C do observador sejam exatamente as mesmas da planta física. Se existirem discrepâncias entre as matrizes do observador e da planta, a dinâmica do observador não será mais governada pela equação (3.29). Isto significa que o erro pode não se

aproximar de zero como esperado. Portanto, deve-se escolher K_e , de modo que o observador seja estável e o erro permaneça aceitavelmente pequeno na presença de pequenos erros de modelagem.

Após apresentados a definição de observador e as condições geométricas e analíticas para a observabilidade em sistemas dinâmicos, o projeto de observadores de estado é apresentado a seguir.

3.3 Projeto de Observadores

Serão apresentados nesta seção uma classificação de observadores com relação ao tipo de excitação do sistema, como sendo uniformes e não uniformes, assim como o projeto de observadores lineares e não lineares, com base nas condições geométricas e analíticas apresentadas anteriormente. Por fim será apresentado o projeto de observador com base na equivalência entre sistemas, para quando não é possível ou é muito complexo o projeto de um observador para um dado sistema de interesse.

Geralmente o problema do projeto de observadores é dependente das entradas do sistema ou das condições analíticas de observabilidade como discutidos em [GUILDAS, 2007].

Para Guildas [2007], o projeto de observadores pode ser classificado, com relação as entradas ou tempo:

- Observadores Uniformes (Observador de *Luenberger* para Sistemas Invariantes no Tempo (LTI))
- Observadores Não Uniformes (Observadores de *Kalman* para Sistemas Variante no Tempo (LTV))

3.3.1 Projeto Observadores Lineares

Para o projeto de observadores lineares consideraremos sistemas do tipo *LTI* para apresentação do observador de Luenberger. Considere o seguinte sistema:

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t).\end{aligned}\tag{3.36}$$

Teorema 2 (Observador de *Luenberger*): Se o sistema (3.36) satisfizer as condições de posto para observabilidade, então existe um observador da forma:

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) - K(C\hat{x}(t) - y(t)),\tag{3.37}$$

com K , tal que $A - KC$ é estável.

Considere o seguinte Sistema Variante no Tempo (LTV):

$$\begin{aligned}\dot{x}(t) &= A(t)x(t) + B(t)u(t), \\ y(t) &= C(t)x(t),\end{aligned}\tag{3.38}$$

com $A(t)$, $C(t)$ uniformemente limitadas.

Teorema 3 (Observador de Kalman): Se o sistema (3.38) é uniformemente completamente observável, então existe um observador da seguinte forma:

$$\dot{\hat{x}}(t) = A(t)\hat{x}(t) + B(t)u(t) - K(t)(C(t)\hat{x}(t) - y(t)),\tag{3.39}$$

com $K(t)$ dado por:

$$\begin{aligned}\dot{M}(t) &= A(t)M(t) + M(t)A^T(t) - M(t)C^T(t)W^{-1}C(t)M(t) + V + \delta M(t), \\ M(0) &= M_0 = M_0^T > 0, W = W^T > 0, \\ K(t) &= M(t)C^T(t)W^{-1},\end{aligned}\tag{3.40}$$

com a condição de ou $\delta > 2 \|A(t)\|$ para todo t , ou $V = V^T > 0$ e as matrizes M e W , matrizes de ponderação associadas as matrizes dinâmicas do sistema A e saídas C , respectivamente, com dimensões equivalentes as dimensões das matrizes A e C , respectivamente.

3.3.2 Projeto Observadores Não Lineares

Com base no Teorema 3, pode-se estender o conceito para sistemas não lineares, por meio de uma aproximação de 1ª ordem das trajetórias estimadas.

Teorema 4 (Filtro de Kalman Extendido [EKF]): Dado um sistema não linear da forma:

$$\begin{aligned}\dot{x}(t) &= f(x(t), u(t)), \\ y(t) &= h(x(t)).\end{aligned}\tag{3.41}$$

O filtro de Kalman estendido correspondente é dado por:

$$\dot{\hat{x}}(t) = f(\hat{x}(t), u(t)) - K(t)(h(\hat{x}(t)) - y(t)),\tag{3.42}$$

sendo $K(t)$ dado pelo observador de Kalman Eq.(3.40) e,

$$A(t) := \frac{\partial f}{\partial x}(\hat{x}(t), u(t)), \quad C(t) := \frac{\partial h}{\partial x}(\hat{x}(t)).\tag{3.43}$$

Este é um candidato ao projeto sistemático de observadores para sistemas não lineares, mas em geral a convergência não é garantida, exceto para certos tipos de estruturas específicas.

3.3.3 Projeto Baseado em Transformação

Os projetos de observadores estão associados a estruturas isoladas ou interconectadas, sendo assim possível transformar estimadores de estado em uma forma conhecida por meio de uma transformação. A mais comum é com relação a mudança de coordenadas dos estados. Essa relação define a equivalência entre sistemas [GUILDAS, 2007].

Definição 15 (Equivalência de Sistemas [com relação a x_0]): Um sistema descrito por:

$$\begin{aligned}\dot{x} &= f(x, u) = f_u(x), x \in \mathbb{R}^n, u \in \mathbb{R}^m, \\ y &= h(x) \in \mathbb{R}^p,\end{aligned}\tag{3.44}$$

é dito ser *equivalente* (com relação a x_0) ao sistema:

$$\begin{aligned}\dot{z} &= F(z, u) = F_u(z), \\ y &= H(z),\end{aligned}\tag{3.45}$$

se existir um *difeomorfismo* $z = \Phi(x)$ definido em $\mathbb{R}^n$ (para alguma vizinhança de x_0) tal que:

$$\forall u \in \mathbb{R}^m, \frac{\partial \Phi}{\partial x} f_u(x)|_{x=\Phi^{-1}(z)} = F_u(z) \quad e \quad h \circ \Phi^{-1} = H\tag{3.46}$$

Os sistemas (3.45) e (3.46) são então chamados de *equivalente* por $z = \Phi(x)$. O interesse pela busca de projeto de observadores com esta propriedade é discutida na proposição a seguir.

Proposição 3: Considere os sistemas (Σ_1) e (Σ_2) , respectivamente definidos por:

$$\begin{aligned}(\Sigma_1) &\begin{cases} \dot{x} = X(x, u), \\ y = h(x), \end{cases} \\ (\Sigma_2) &\begin{cases} \dot{z} = Z(z, u), \\ y = H(z), \end{cases}\end{aligned}\tag{3.47}$$

e equivalentes por $z = \Phi(x)$,

Se:

$$(\mathcal{O}_2) \begin{cases} \dot{\hat{z}} = Z(\hat{z}, u) + k(w, H(\hat{z}) - y), \\ \dot{w} = F(w, u, y), \end{cases}\tag{3.48}$$

é um observador para (Σ_2) ,

Então:

$$(\mathcal{O}_2) \begin{cases} \dot{\hat{x}} = X(\hat{x}, u) + \left(\frac{\partial \Phi}{\partial x}\right)_{\hat{x}}^{-1} k(w, h(\hat{x}) - y), \\ \dot{w} = F(w, u, y), \end{cases}\tag{3.49}$$

é um observador para (Σ_1) .

Portanto se um sistema não possuir uma estrutura apropriada para o projeto de um observador, mas for equivalente a algum outro sistema que o seja, então o problema de observação pode ser resolvido para o sistema original. Informações adicionais com relação a síntese de observadores, interconexão em observadores, observadores de ganho elevado, assim como formas canônicas para sistemas uniformemente observáveis, são abordados em [GUILDAS, 2007].

3.4 Observador de Estados Baseados em Programação Dinâmica Aproximada Alimentado por Dados

Nesta seção apresenta-se o problema de estimação de estados inicialmente como um problema de controle ótimo baseado em programação dinâmica aproximada orientado por dados, em que algoritmos de aprendizagem por reforço são também usados para estimação on-line. O problema de controle ótimo usado é o Regulador Linear Quadrático Discreto (DLQR) com realimentação de saída. É apresentada a abordagem model-free de um problema de programação dinâmica aproximada, juntamente com a formulação da Equação de Bellman, usando somente os dados mensurados e ações de controle passadas. É realizada a formulação do erro de diferença temporal e os algoritmos de avaliação e melhoria de política que solucionam o problema de controle, por meio da determinação da política de controle que estabiliza o sistema, visando a estimação de estados.

3.4.1 O Problema de Controle Ótimo Alimentado por Dados com Realimentação de Saída Usando Aprendizagem por Reforço

Um dos principais problemas em teoria de controle baseados somente em dados é o definir uma lei de controle por meio da observação das ações de controle e saídas passadas associadas. Uma proposta para solucionar este problema é apresentada na metodologia de aprendizagem por reforço para sistemas parcialmente observáveis com realimentação na saída, com proposta de um algoritmo chamado de *OPFB – OutputFeedback* [LEWIS; VAMVOUDAKIS, 2011]. Tal metodologia apresenta uma formulação que torna possível por meio de algoritmos de aprendizagem por reforço (RL), a estimação das políticas de controle que minimizam um índice de desempenho quadrático associado as entradas e saídas passadas ponderadas, com a adição de um ruído branco, este relacionado a propriedade de excitação persistente, necessária para os algoritmos de aprendizagem por reforço. Quando nos referimos a entradas, estamos nos referindo as ações de controle.

O problema de programação dinâmica, a construção do problema de controle ótimo do tipo Regulador Linear Quadrático Discreto (DLQR) e aplicação da aprendizagem por reforço, por meio do algoritmo do erro de diferença temporal, para o caso de realimentação na saída com iteração de política e iteração de valor são apresentados a seguir.

3.4.1.1 Política de Controle DLQR e a Equação de Bellman

Considere o seguinte sistema linear discreto invariante no tempo:

$$\begin{aligned}x_{k+1} &= Ax_k + Bu_k, \\y_k &= Cx_k,\end{aligned}\tag{3.50}$$

sendo $A \in \mathbb{R}^{n \times n}$, $x_k \in \mathbb{R}^n$ o vetor de estado, $B \in \mathbb{R}^{n \times m}$, $u_k \in \mathbb{R}^m$ o vetor de entrada da planta, $C \in \mathbb{R}^{(p \times n)}$ e $y_k \in \mathbb{R}^p$ o vetor de saída. Assuma o par (A, B) controlável e (A, C) observável.

Escolhida uma política de controle estabilizadora $u_k = \mu(x_k)$, define-se a função de custo ou índice de desempenho,

$$V^\mu(x_k) = \sum_{i=k}^{\infty} (y_i^T Q y_i + u_i^T R u_i),\tag{3.51}$$

sendo $Q = Q^T \geq 0$ e $R = R^T > 0$, matrizes de ponderação em relação as saídas e ações de controle, respectivamente e o par $(A, C\sqrt{Q})$ sendo observável [LEWIS; SYRMOS, 1995]. Este índice também chamado de função valor é representado pela equação de Bellman que é dada por:

$$V^\mu(x_k) = y_k^T Q y_k + u_k^T R u_k + V^\mu(x_{k+1}).\tag{3.52}$$

No projeto de controle DLQR [LEWIS; SYRMOS, 1995], deve-se estabelecer uma política de controle $\mu^*(x_k)$ que minimiza a função valor (3.51), com relação a todas as possíveis políticas associadas ao sistema (3.50), isto é, $V^{\mu^*}(x_k) \leq V^\mu(x_k), \forall \mu, x_k$.

De acordo com o princípio da otimalidade de Bellman, a função valor ótima V^* , pode ser determinada usando a equação de Hamilton-Jacobi-Bellman (HJB), isto é,

$$V^*(x_k) = \min_{u_k} (y_k^T Q y_k + u_k^T R u_k + V^*(x_{k+1})),\tag{3.53}$$

e a política de controle ótimo é dada por:

$$\mu^*(x_k) = \arg \min_{u_k} (y_k^T Q y_k + u_k^T R u_k + V^*(x_{k+1})).\tag{3.54}$$

Para o caso do DLQR, qualquer valor é quadrático no estado, tal que o custo associado a qualquer política $u_k = \mu(x_k)$, não necessariamente ótima, seja dada por:

$$V^\mu(x_k) = x_k^T P x_k,\tag{3.55}$$

para alguma matriz $P \geq P^T$, de ordem $n \times n$.

Substituindo a equação (3.55) em (3.53), a equação de Bellman para o DLQR assume a seguinte forma:

$$x_k^T P x_k = y_k^T Q y_k + u_k^T R u_k + x_{k+1}^T P x_{k+1}.\tag{3.56}$$

Até então foram desenvolvidas questões relacionadas ao desenvolvimento teórico da fundamentação sobre programação dinâmica, com relação aos estados do sistema. Nas próximas seções a mudança da dependência dos estados, a formulação em termos dos dados mensurados, os algoritmos de iteração de política (PI) e iteração de valor (VI) usados neste trabalho são apresentados.

3.4.1.2 Formulação da Função Valor em Termos dos Dados Mensurados

Apresenta-se nesta seção a função valor em termos dos dados mensurados de saída e ações de controle passadas. A propriedade de observabilidade significa que existem observações y_k suficientes dentro do horizonte de tempo para que seja possível reconstruir completamente o estado x_k .

A dinâmica de um sistema para um instante k , em um horizonte de tempo descrita como $[k - N, k]$, pode ser representada, pela equação de estado e saída do sistema, por um modelo simplificado, pela definição da equação de estado e saída a seguir:

$$x_k = A^N x_{k-N} + U_N \bar{u}_{[k-1, k-N]}, \quad (3.57)$$

$$\bar{y}_{[k-1, k-N]} = V^N x_{k-N} + T_N \bar{u}_{[k-1, k-N]}, \quad (3.58)$$

sendo U_N a matriz de controlabilidade e V_N a matriz de observabilidade, sendo $V_N \in \mathbb{R}^{pN \times n}$

$$U_N = [B \ AB \ \dots \ A^{N-1}B], \quad (3.59)$$

$$V_N = \begin{bmatrix} CA^{N-1} \\ \vdots \\ CA \\ C \end{bmatrix}, \quad (3.60)$$

e T_N uma matriz Toeplitz dos parâmetros de Markov.

$$T_N = \begin{bmatrix} 0 & CB & CAB & \dots & CA^{N-2}B \\ 0 & 0 & CB & \dots & CA^{N-3}B \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & 0 & CB \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (3.61)$$

Os vetores $\bar{u}_{[k-1, k-N]}$ e $\bar{y}_{[k-1, k-N]}$ são as entradas e saídas passadas atrasadas para o horizonte de tempo $[k - 1, k - N]$, respectivamente, representando os dados mensurados disponíveis:

$$\bar{u}_{[k-1, k-N]} = \begin{bmatrix} u_{k-1} \\ u_{k-2} \\ \vdots \\ u_{k-N} \end{bmatrix} \in \mathbb{R}^{pN}, \quad (3.62)$$

$$\bar{y}_{[k-1, k-N]} = \begin{bmatrix} y_{k-1} \\ y_{k-2} \\ \vdots \\ y_{k-N} \end{bmatrix} \in \mathbb{R}^{mN}. \quad (3.63)$$

Desde que (A, C) seja observável, existe um índice de observabilidade K , tal que $\text{posto}(V_N) < n$ se $N < K$ e $\text{posto}(V_N) = n$ se $N \geq K$. Portanto V_N tem posto cheio n , e existe uma matriz $M \in \mathbb{R}^{(n \times pN)}$ tal que:

$$A^N = MV_N. \quad (3.64)$$

Este procedimento foi utilizado por [Aangenent et al. \[2005\]](#), para determinar o controle por meio da identificação dos parâmetros de Markov.

Desde que V_N tenha posto cheio, o seu inverso à esquerda é dado por:

$$V_N^+ = (V_N^T V_N)^{-1} V_N^T, \quad (3.65)$$

tal que,

$$M = A^N V_N^+ + Z(I - V_N V_N^+) \equiv M_0 + M_1, \quad (3.66)$$

para qualquer matriz Z , com M_0 o operador de norma mínima e $P(\mathbb{R}^\perp(V_N)) = I - V_N V_N^+$, sendo a projeção na imagem perpendicular a V_N [\[LEWIS; VAMVOUDAKIS, 2011\]](#).

A principal contribuição de [Lewis e Vamvoudakis \[2011\]](#) foi apresentar uma formulação para o cálculo da política de controle, sem a necessidade da estimação dos estados de modo explícito. A partir dos dados mensurados de entrada e saída, foi possível aplicar DLQR com realimentação na saída, por meio do uso de algoritmos de erros de diferenças temporais. O Lema a seguir descreve os estados em função das entradas e saídas passadas no horizonte de tempo $[k-1, k-N]$.

Lema 1: Seja o sistema (3.50) observável. Então os estados do sistema são dados unicamente em termos dos dados mensurados,

$$\begin{aligned} x_k &= M_0 \bar{y}_{[k-1, k-N]} + (U_N - M_0 T_N) \bar{u}_{[k-1, k-N]}, \\ x_k &\equiv M_y \bar{y}_{[k-1, k-N]} + M_u \bar{u}_{[k-1, k-N]}, \end{aligned} \quad (3.67)$$

ou

$$x_k = [M_u \ M_y] \begin{bmatrix} \bar{u}_{[k-1, k-N]} \\ \bar{y}_{[k-1, k-N]} \end{bmatrix}, \quad (3.68)$$

sendo $M_u = U_N - M_0 T_N$ e $M_y = M_0$, $M_0 = A^N V_N^+$ e $V_N^+ = (V_N^T V_N)^{-1} V_N^T$, $A^N V_N^+$, o inverso à esquerda da matriz de observabilidade (3.60) e $N > K$, sendo K o índice de observabilidade.

Para se escrever a equação de estado (3.68) em função das entradas e saídas passadas, considere $\bar{z}$ o vetor contendo as ações de controle e saídas passadas no horizonte $[k-1, k-N]$,

$$\bar{z}_{[k-1, k-N]} = \begin{bmatrix} \bar{u}_{[k-1, k-N]} \\ \bar{y}_{[k-1, k-N]} \end{bmatrix} \quad (3.69)$$

Seja $V^\mu(x_k)$ a função valor associada aos estados e usando a equação (3.68),

$$V^\mu(x_k) = x_k^T P x_k = \bar{z}_{[k-1, k-N]}^T \begin{bmatrix} M_u^T \\ M_y^T \end{bmatrix} P [M_u \ M_y] \bar{z}_{[k-1, k-N]}, \quad (3.70)$$

$$V^\mu(x_k) = \bar{z}_{[k-1, k-N]}^T \begin{bmatrix} M_u^T P M_u & M_u^T P M_y \\ M_y^T P M_u & M_y^T P M_y \end{bmatrix} \bar{z}_{[k-1, k-N]} = \bar{z}_{[k-1, k-N]}^T \bar{P} \bar{z}_{[k-1, k-N]}. \quad (3.71)$$

Note que $\bar{u}_{[k-1, k-N]} \in \mathbb{R}^{mN}$, $\bar{y}_{[k-1, k-N]} \in \mathbb{R}^{pN}$, $\bar{z}_{[k-1, k-N]} \in \mathbb{R}^{(m+p)N}$ e $\bar{P} \in \mathbb{R}^{(m+p)N \times (m+p)N}$.

A equação (3.71) expressa a função valor no instante k , como uma forma quadrática dos termos das entradas e saídas passadas, sendo que o núcleo da matriz $\bar{P}$ na equação (3.71) ainda depende das matrizes A , B e C , pelas matrizes M_0 , M_u e M_y .

A seguir apresenta-se como escrever o problema de controle ótimo e sua formulação associada à técnicas de aprendizagem por reforço, para expressar a função valor em função de diferenças temporais.

3.4.1.3 Erro de Diferença Temporal Baseado em Dados Mensurados

Nesta seção apresenta-se um método de aprendizagem por reforço baseado em diferenças temporais para a determinação da função valor em tempo real, somente com os dados mensurados. Métodos RL baseados em diferenças temporais fornecem procedimentos "para frente no tempo", para a aprendizagem on-line da função valor e política de controle ótimas, que podem ser implementadas em tempo real.

Seja a equação do erro de diferença temporal de Bellman para o caso do $DLQR$, com relação aos estados descrita por:

$$e_k = -x_k^T P x_k + y_k^T Q y_k + u_k^T R u_k + x_{k+1}^T P x_{k+1}. \quad (3.72)$$

A estrutura da equação (3.72) pode ser escrita somente com relação às sequências de saídas observadas y_k e sequências de ações de controle u_k , [LEWIS; VAMVOUDAKIS, 2011]. Para aprender a estrutura interna da matriz $\bar{P}$, sem depender das matrizes A , B e C , escreve-se o

erro de diferença temporal da Equação de Bellman para o DLQR da equação (3.72) para dados mensurados:

$$e_k = -\bar{z}_{[k-1,k-N]}^T \bar{P} \bar{z}_{[k-1,k-N]} + y_k^T Q y_k + u_k^T R u_k + \bar{z}_{[k,k-N+1]}^T \bar{P} \bar{z}_{[k,k-N+1]} \quad (3.73)$$

Usando o erro de diferença temporal, a etapa de avaliação de política de qualquer forma de aprendizagem por reforço baseado na equação do erro de diferença temporal de Bellman (3.72), pode ser desempenhada usando somente dados mensurados e não mais os estados, e com isso aprender os coeficientes da matriz $\bar{P}$, sem conhecimento das matrizes A, B e C, como é apresentado na próxima seção.

3.4.1.4 Escrevendo a Política de Atualização em Termos dos Dados Mensurados

O algoritmo de aprendizagem por reforço Q-learning é usado para aprender o valor de uma ação para um dado estado. O algoritmo pode desempenhar Iteração de Valor e Iteração de Política, sem o conhecimento da dinâmica do sistema. Detalhes sobre o algoritmo Q-learning, sua formulação, variações e implementações, são discutidos em [SUTTON; BARTO, 2018] e [BERTSEKAS et al., 2005], que apresentam as formulações, implementações e variações do algoritmo. De maneira análoga, a partir da construção já apresentada, pode-se definir uma política de melhoramento em termos dos dados mensurados. Considere a política de controle $\mu(x_k)$:

$$\mu(x_k) = \min_{u_k} (y_k^T Q y_k + u_k^T R u_k + x_{k+1}^T P x_{k+1}), \quad (3.74)$$

$$\mu(x_k) = \min_{u_k} (y_k^T Q y_k + u_k^T R u_k + \bar{z}_{[k,k-N+1]}^T \bar{P} \bar{z}_{[k,k-N+1]}). \quad (3.75)$$

Particionando $\bar{z}_{[k,k-N+1]}^T \bar{P} \bar{z}_{[k,k-N+1]}$, obtêm-se,

$$\bar{z}_{[k,k-N+1]}^T \bar{P} \bar{z}_{[k,k-N+1]} = \begin{bmatrix} u_k \\ \bar{u}_{[k-1,k-N+1]} \\ \bar{y}_{[k,k-N+1]} \end{bmatrix}^T \begin{bmatrix} p_0 & p_u & p_y \\ p_u & P_{22} & P_{23} \\ p_y & P_{32} & P_{33} \end{bmatrix} \begin{bmatrix} u_k \\ \bar{u}_{[k-1,k-N+1]} \\ \bar{y}_{[k,k-N+1]} \end{bmatrix}, \quad (3.76)$$

sendo os coeficientes p_0, p_u e p_y , usados como parte integrante para definir a ação de controle, que atuará na planta. Tais coeficientes possuem dimensões: $p_0 \in \mathbb{R}^{m \times m}$, $p_u \in \mathbb{R}^{m \times (m(N-1))}$ e $p_y \in \mathbb{R}^{m \times pN}$ e são estimados pelo algoritmo RLS. Detalhes sobre a formulação e implementação do RLS são apresentados no trabalho de Bradtke et al. [1994].

Agora o problema de otimização da equação (3.75), pode ser determinado diferenciando-se (3.76) função de u_k e igualando-se a 0,

$$0 = R u_k + p_0 u_k + p_u \bar{u}_{[k-1,k-N+1]} + p_y \bar{y}_{[k,k-N+1]}, \quad (3.77)$$

ou

$$u_k = -(R + p_0)^{-1} + (p_u \bar{u}_{[k-1,k-N+1]} + p_y \bar{y}_{[k,k-N+1]}). \quad (3.78)$$

Este controlador é um modelo do tipo auto regressivo de média móvel dinâmico, que gera as ações de controle atuais em função das ações e saídas passadas. Assim como no método Q -learning, chamado por [WERBOS, 1992], de "aprendizado dependente de ação", o problema de minimização pode ser desenvolvido em termos do núcleo da matriz $\bar{P}$, sendo assim dispensável conhecimento da dinâmica do sistema.

O algoritmo de Iteração de Política e Iteração de Valor para o controlador apresentado, usando as formulações apresentadas, baseadas no erro de diferença temporal, são apresentados a seguir:

Algoritmo PI - Iteração de Política Usando Realimentação da Saída (OPFB)

Selecione uma política de controle estabilizadora $u_k^0 = \mu^0$ para $j = 0, 1, \dots$, desempenhe até a convergência:

1. **Avaliação de Política:** Calcule $\bar{P}^{j+1}$, tal que,

$$0 = -\bar{z}_{[k-1, k-N]}^T \bar{P}^{j+1} \bar{z}_{[k-1, k-N]} + y_k^T Q y_k + (u_k^j)^T R u_k^j + \bar{z}_{[k, k-N+1]}^T \bar{P}^{j+1} \bar{z}_{[k, k-N+1]}. \quad (3.79)$$

2. **Melhoria de Política:** Particione $\bar{P}$, como em Eq.(3.76) e defina a política de melhoria u_k^{j+1} por:

$$\begin{aligned} u_k^{j+1} &= \mu^{j+1}(x_k) \\ &= -(R + p_0^{j+1})^{-1} (p_u^{j+1})^{-1} \cdot (\bar{u}_{[k-1, k-N+1]}^{j+1} + p_y^{j+1} \bar{y}_{[k, k-N+1]}). \end{aligned} \quad (3.80)$$

Algoritmo VI - Iteração de Valor Usando Realimentação da Saída (OPFB)

Selecione uma política de controle $u_k^0 = \mu^0$ para $j = 0, 1, \dots$, desempenhe até a convergência:

1. **Avaliação de Política:** Calcule $\bar{P}^{j+1}$, tal que,

$$\bar{z}_{[k-1, k-N]}^T \bar{P}^{j+1} \bar{z}_{[k-1, k-N]} = y_k^T Q y_k + (u_k^j)^T R u_k^j + \bar{z}_{[k, k-N+1]}^T \bar{P}^j \bar{z}_{[k, k-N+1]}. \quad (3.81)$$

2. **Melhoria de Política:** Particione $\bar{P}$, como em Eq.(3.76) e defina a política de melhoria u_k^{j+1} por:

$$\begin{aligned} u_k^{j+1} &= \mu^{j+1}(x_k) \\ &= -(R + p_0^{j+1})^{-1} (p_u^{j+1})^{-1} \cdot (\bar{u}_{[k-1, k-N+1]}^{j+1} + p_y^{j+1} \bar{y}_{[k, k-N+1]}). \end{aligned} \quad (3.82)$$

Capítulo 4

O Problema de Estimação de Estado Alimentado por Dados

Neste capítulo são apresentadas contribuições, com formulações propostas para estimação de estados não baseadas em modelos, utilizando somente dados de entrada e saída de um sistema simulado. Algumas das formulações são fundamentadas no problema de programação dinâmica heurística aproximada dependente de ação (ADHDP) e algoritmos de aprendizagem por reforço baseados em Q-learning. Também são propostas duas formulações para a solução do problema de estimação de estado proposto neste capítulo associado à formulação apresentada na Seção [3.4.1](#).

4.1 O Problema da Reconstrução dos Estados Baseada em ADP

A formulação do problema de reconstrução dos estados abordada nesta seção têm como base o Lema 1 apresentado em [[LEWIS; VAMVOUDAKIS, 2011](#)] na Seção [3.4.1.2](#), em que os estados do sistema dinâmico linear discreto podem ser expressos em função dos dados de entrada/saída mensurados em um horizonte de tempo. Embora o lema tenha sido apresentado no trabalho citado, não foi apresentado um método para a estimação de estados explicitamente. Como contribuição, apresenta-se nesta Tese uma formulação com a estrutura algébrica para o cálculo de matrizes de blocos da matriz $\bar{P}$ na equação (3.71), mais especificamente das matrizes M_u e M_y da equação (4.1), necessárias para a estimação dos estados.

O Lema novamente apresentado para facilitar a apresentação do problema de recuperação

das matrizes M_u e M_y , necessários para a estimação dos estados. O problema de recuperação de matrizes é equivalente ao de solucionar um sistema de equações não lineares, o desenvolvimento deste sistema é apresentado a seguir.

Lema 1: Seja o sistema (3.50) observável. Então os estados do sistema são dados unicamente em termos dos dados mensurados,

$$\begin{aligned} x_k &= M_0 \bar{y}_{[k-1, k-N]} + (U_N - M_0 T_N) \bar{u}_{[k-1, k-N]}, \\ &\equiv M_y \bar{y}_{[k-1, k-N]} + M_u \bar{u}_{[k-1, k-N]}, \end{aligned} \quad (4.1)$$

ou

$$x_k = [M_u \ M_y] \begin{bmatrix} \bar{u}_{[k-1, k-N]} \\ \bar{y}_{[k-1, k-N]} \end{bmatrix}, \quad (4.2)$$

sendo,

- $M_u = U_N - M_0 T_N$ e $M_y = M_0$;
- $M_0 = A^N V_N^+$, $V_N^+ = (V_N^T V_N)^{-1} V_N^T$;
- V_N^+ , Inversa à Esquerda da matriz V_N ;
- $N > K$, e K o Índice de Observabilidade.

Para a determinação dos estados pelo uso da equação (4.1), é necessária à disponibilidade das matrizes M_u e M_y , visto que os demais dados são entradas/saídas no horizonte de tempo.

Após a formulação do problema na Seção 3.4, baseada em programação dinâmica aproximada para o DLQR, é realizada a construção da função valor em função dos dados mensurados e a definição da matriz de bloco $\bar{P}$ na equação (3.71), que é estimada pelo RLS para o fornecimento dos coeficientes, necessários para calcular o sinal de controle u_k .

A função valor $V^\mu(x_k)$ em função dos dados mensurados, nos apresenta a matriz $\bar{P}$, com as matrizes de blocos a serem estimadas:

$$\begin{aligned} V^\mu(x_k) &= \bar{z}_{[k-1, k-N]}^T \begin{bmatrix} M_u^T P M_u & M_u^T P M_y \\ M_y^T P M_u & M_y^T P M_y \end{bmatrix} \bar{z}_{[k-1, k-N]}, \\ &= \bar{z}_{[k-1, k-N]}^T \bar{P} \bar{z}_{[k-1, k-N]}. \end{aligned} \quad (4.3)$$

Note que:

- $\bar{u}_{[k-1, k-N]} \in \mathbb{R}^{(mN)}$ e $\bar{y}_{[k-1, k-N]} \in \mathbb{R}^{(pN)}$;
- $\bar{z}_{[k-1, k-N]} \in \mathbb{R}^{[(m+p)N]}$;
- $\bar{P} \in \mathbb{R}^{[(m+p)N \times (m+p)N]}$.

Para realizar a reconstrução efetiva dos estados é necessário computar as matrizes M_u e M_y , a partir dos blocos da matriz $\bar{P}$, sendo eles: $M_u^T P M_u$, $M_u^T P M_y$, $M_y^T P M_u$ e $M_y^T P M_y$.

Esse problema é semelhante ao de solucionar o problema: $X^T A X = B$, que é um problema de congruência. Quando não se tem conhecimento algum sobre as matrizes X e A , não existe solução para esse problema [TERÁN, 2010]. Quando se possuem informações sobre simetria ou positividade das matrizes, o problema torna-se menos complexo.

Para facilitar o desenvolvimento, considere a partição das matrizes de blocos de $\bar{P}$:

$$\bar{P} = \begin{bmatrix} M_u^T P M_u & M_u^T P M_y \\ M_y^T P M_u & M_y^T P M_y \end{bmatrix} = \begin{bmatrix} \bar{P}_{uu} & \bar{P}_{uy} \\ \bar{P}_{yu} & \bar{P}_{yy} \end{bmatrix}, \quad (4.4)$$

assim, tem-se o seguinte sistema:

$$\begin{cases} M_u^T P M_u = \bar{P}_{uu} \\ M_u^T P M_y = \bar{P}_{uy} \\ M_y^T P M_u = \bar{P}_{yu} \\ M_y^T P M_y = \bar{P}_{yy} \end{cases} \quad (4.5)$$

Para um sistema de ordem 2, tem-se a matriz $\bar{P} \in \mathbb{R}^{4 \times 4}$ e $(P, M_u, M_y) \in \mathbb{R}^{2 \times 2}$ e as matrizes,

$$M_u = \begin{bmatrix} m_{u11} & m_{u12} \\ m_{u21} & m_{u22} \end{bmatrix}, M_y = \begin{bmatrix} m_{y11} & m_{y12} \\ m_{y21} & m_{y22} \end{bmatrix}, P = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix}, \quad (4.6)$$

$$\bar{P}_{uu} = \begin{bmatrix} \bar{p}_{11}^{uu} & \bar{p}_{12}^{uu} \\ \bar{p}_{21}^{uu} & \bar{p}_{22}^{uu} \end{bmatrix}, \bar{P}_{uy} = \begin{bmatrix} \bar{p}_{11}^{uy} & \bar{p}_{12}^{uy} \\ \bar{p}_{21}^{uy} & \bar{p}_{22}^{uy} \end{bmatrix}, \quad (4.7)$$

$$\bar{P}_{yu} = \begin{bmatrix} \bar{p}_{11}^{yu} & \bar{p}_{12}^{yu} \\ \bar{p}_{21}^{yu} & \bar{p}_{22}^{yu} \end{bmatrix}, \bar{P}_{yy} = \begin{bmatrix} \bar{p}_{11}^{yy} & \bar{p}_{12}^{yy} \\ \bar{p}_{21}^{yy} & \bar{p}_{22}^{yy} \end{bmatrix},$$

substituindo-se as equações (4.6) e (4.7) em (4.5), obtêm-se o sistema:

$$\begin{cases} M_u^T P M_u = \begin{bmatrix} m_{u11} & m_{u21} \\ m_{u12} & m_{u22} \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} \begin{bmatrix} m_{u11} & m_{u12} \\ m_{u21} & m_{u22} \end{bmatrix} = \begin{bmatrix} \bar{p}_{11}^{uu} & \bar{p}_{12}^{uu} \\ \bar{p}_{21}^{uu} & \bar{p}_{22}^{uu} \end{bmatrix}, \\ M_u^T P M_y = \begin{bmatrix} m_{u11} & m_{u21} \\ m_{u12} & m_{u22} \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} \begin{bmatrix} m_{y11} & m_{y12} \\ m_{y21} & m_{y22} \end{bmatrix} = \begin{bmatrix} \bar{p}_{11}^{uy} & \bar{p}_{12}^{uy} \\ \bar{p}_{21}^{uy} & \bar{p}_{22}^{uy} \end{bmatrix}, \\ M_y^T P M_u = \begin{bmatrix} m_{y11} & m_{y21} \\ m_{y12} & m_{y22} \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} \begin{bmatrix} m_{u11} & m_{u12} \\ m_{u21} & m_{u22} \end{bmatrix} = \begin{bmatrix} \bar{p}_{11}^{yu} & \bar{p}_{12}^{yu} \\ \bar{p}_{21}^{yu} & \bar{p}_{22}^{yu} \end{bmatrix}, \\ M_y^T P M_y = \begin{bmatrix} m_{y11} & m_{y21} \\ m_{y12} & m_{y22} \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} \begin{bmatrix} m_{y11} & m_{y12} \\ m_{y21} & m_{y22} \end{bmatrix} = \begin{bmatrix} \bar{p}_{11}^{yy} & \bar{p}_{12}^{yy} \\ \bar{p}_{21}^{yy} & \bar{p}_{22}^{yy} \end{bmatrix}, \end{cases} \quad (4.8)$$

Desenvolvendo-se o bloco do termo $M_u^T P M_u$ e efetuando-se as operações de multiplicações das matrizes da primeira equação do sistema (4.8), obtêm-se o sistema:

$$M_u^T P M_u \Rightarrow \begin{cases} m_{u11}^2 p_{11} + 2m_{u11}m_{u21}p_{12} + m_{u21}^2 p_{22} = \bar{p}_{11}^{uu} \\ m_{u11}m_{u12}p_{11} + m_{u12}^2 p_{21} + m_{u11}m_{u22}p_{12} + m_{u21}m_{u22}p_{22} = \bar{p}_{12}^{uu} \\ m_{u11}m_{u12}p_{11} + m_{u12}^2 p_{12} + m_{u11}m_{u22}p_{21} + m_{u21}m_{u22}p_{22} = \bar{p}_{21}^{uu} \\ m_{u12}^2 p_{11} + 2m_{u22}m_{u12}p_{21} + m_{u22}^2 p_{22} = \bar{p}_{22}^{uu} \end{cases} \quad (4.9)$$

Como a matriz $\bar{P}$ é simétrica, tem-se que $M_u^T P M_y = M_y^T P M_u \implies \bar{P}_{uy} = \bar{P}_{yu}$. Como matrizes M_u , M_y e P são simétricas, então o sistema (4.9) possuirá somente 3 equações,

$$M_u^T P M_u \Rightarrow \begin{cases} m_{u11}^2 p_{11} + 2m_{u11}m_{u21}p_{12} + m_{u21}^2 p_{22} = \bar{p}_{11}^{uu} \\ m_{u11}m_{u12}p_{11} + m_{u12}^2 p_{12} + m_{u11}m_{u22}p_{12} + m_{u21}m_{u22}p_{22} = \bar{p}_{12}^{uu} \\ m_{u12}^2 p_{11} + 2m_{u22}m_{u12}p_{21} + m_{u22}^2 p_{22} = \bar{p}_{22}^{uu} \end{cases} \quad (4.10)$$

Para os termos simétricos $a_{ji} = a_{ij}$, usaremos a notação a_{ij} somente. Reescrevendo as equações (4.10) e colocando os coeficiente p_{ij} em evidencia, obtêm-se o sistema:

$$M_u^T P M_u \Rightarrow \begin{cases} m_{u11}^2 p_{11} + 2m_{u11}m_{u12}p_{12} + m_{u12}^2 p_{22} = \bar{p}_{11}^{uu} \\ m_{u11}m_{u12}p_{11} + (m_{u12}^2 + m_{u11}m_{u22})p_{12} + m_{u12}m_{u22}p_{22} = \bar{p}_{12}^{uu} \\ m_{u12}^2 p_{11} + 2m_{u12}m_{u22}p_{12} + m_{u22}^2 p_{22} = \bar{p}_{22}^{uu} \end{cases} \quad (4.11)$$

Em notação matricial, o sistema (4.11) é representado como:

$$\begin{bmatrix} m_{u11}^2 & 2m_{u11}m_{u12} & m_{u12}^2 \\ m_{u11}m_{u12} & (m_{u12}^2 + m_{u11}m_{u22}) & m_{u12}m_{u22} \\ m_{u12}^2 & 2m_{u12}m_{u22} & m_{u22}^2 \end{bmatrix} \begin{bmatrix} p_{11} \\ p_{12} \\ p_{22} \end{bmatrix} = \begin{bmatrix} \bar{p}_{11}^{uu} \\ \bar{p}_{12}^{uu} \\ \bar{p}_{22}^{uu} \end{bmatrix} \quad (4.12)$$

Do mesmo modo que foi desenvolvido o termo da matriz do bloco $M_u^T P M_u$, desenvolve-se analogamente para os termos, $M_u^T P M_y$, $M_y^T P M_u$ e $M_y^T P M_y$, com $M_u^T P M_y = M_y^T P M_u$,

pois, a matriz $\bar{P}$ é simétrica. Assim, tem-se que $M_u^T P M_y$, resulta no seguinte sistema:

$$M_u^T P M_y \Rightarrow \begin{cases} m_{u11}m_{y11}p_{11} + m_{u21}m_{y11}p_{21} + m_{u11}m_{y21}p_{12} + m_{u21}m_{y21}p_{22} = \bar{p}_{11}^{uy} \\ m_{u11}m_{y12}p_{11} + m_{u21}m_{y12}p_{21} + m_{u11}m_{y22}p_{12} + m_{u21}m_{y22}p_{22} = \bar{p}_{12}^{uy} \\ m_{u12}m_{y11}p_{11} + m_{u22}m_{y11}p_{21} + m_{u12}m_{y21}p_{12} + m_{u22}m_{y21}p_{22} = \bar{p}_{21}^{uy} \\ m_{u12}m_{y12}p_{11} + m_{u22}m_{y12}p_{21} + m_{u12}m_{y22}p_{12} + m_{u22}m_{y22}p_{22} = \bar{p}_{22}^{uy} \end{cases} \quad (4.13)$$

Como a matriz P é simétrica, tem-se que $p_{12} = p_{21}$ e evidenciando-se os termos em função de p_{ij} , o sistema (4.13), assumirá a forma:

$$M_u^T P M_y \Rightarrow \begin{cases} m_{u11}m_{y11}p_{11} + (m_{u21}m_{y11} + m_{u11}m_{y21})p_{12} + m_{u21}m_{y21}p_{22} = \bar{p}_{11}^{uy} \\ m_{u11}m_{y12}p_{11} + (m_{u21}m_{y12} + m_{u11}m_{y22})p_{12} + m_{u21}m_{y22}p_{22} = \bar{p}_{12}^{uy} \\ m_{u12}m_{y11}p_{11} + (m_{u22}m_{y11} + m_{u12}m_{y21})p_{12} + m_{u22}m_{y21}p_{22} = \bar{p}_{21}^{uy} \\ m_{u12}m_{y12}p_{11} + (m_{u22}m_{y12} + m_{u12}m_{y22})p_{12} + m_{u22}m_{y22}p_{22} = \bar{p}_{22}^{uy} \end{cases} \quad (4.14)$$

A forma matricial do sistema (4.14) é representada a seguir,

$$\begin{bmatrix} m_{u11}m_{y11} & (m_{u21}m_{y11} + m_{u11}m_{y21}) & m_{u21}m_{y21} \\ m_{u11}m_{y12} & (m_{u21}m_{y12} + m_{u11}m_{y22}) & m_{u21}m_{y22} \\ m_{u12}m_{y11} & (m_{u22}m_{y11} + m_{u12}m_{y21}) & m_{u22}m_{y21} \\ m_{u12}m_{y12} & (m_{u22}m_{y12} + m_{u12}m_{y22}) & m_{u22}m_{y22} \end{bmatrix} \begin{bmatrix} p_{11} \\ p_{12} \\ p_{22} \end{bmatrix} = \begin{bmatrix} \bar{p}_{11}^{uy} \\ \bar{p}_{12}^{uy} \\ \bar{p}_{21}^{uy} \\ \bar{p}_{22}^{uy} \end{bmatrix} \quad (4.15)$$

Como $M_u^T P M_y = M_y^T P M_u$, obtêm-se o mesmo sistema de equações para $M_y^T P M_u$ e por fim somente o sistema composto pelo termo $M_y^T P M_y$, apresentados a seguir,

$$M_y^T P M_y \Rightarrow \begin{cases} m_{y11}^2 p_{11} + 2m_{y11}m_{y21}p_{12} + m_{y21}^2 p_{22} = \bar{p}_{11}^{yy} \\ m_{y11}m_{y12}p_{11} + m_{y12}^2 p_{21} + m_{y11}m_{y22}p_{12} + m_{y21}m_{y22}p_{22} = \bar{p}_{12}^{yy} \\ m_{y11}m_{y12}p_{11} + m_{y12}^2 p_{12} + m_{y11}m_{y22}p_{21} + m_{y21}m_{y22}p_{22} = \bar{p}_{21}^{yy} \\ m_{y12}^2 p_{11} + 2m_{y22}m_{y12}p_{21} + m_{y22}^2 p_{22} = \bar{p}_{22}^{yy} \end{cases} \quad (4.16)$$

Como a segunda e terceira equações do sistema (4.16) são iguais pela simetria, obtêm-se

somente três equações,

$$M_y^T P M_y \Rightarrow \begin{cases} m_{y11}^2 p_{11} + 2m_{y11}m_{y21}p_{12} + m_{y21}^2 p_{22} = \bar{p}_{11}^{yy} \\ m_{y11}m_{y12}p_{11} + m_{y12}^2 p_{12} + m_{y11}m_{y22}p_{12} + m_{y21}m_{y22}p_{22} = \bar{p}_{12}^{yy} \\ m_{y12}^2 p_{11} + 2m_{y22}m_{y12}p_{21} + m_{y22}^2 p_{22} = \bar{p}_{22}^{yy} \end{cases} \quad (4.17)$$

Evidenciando-se os termos em função de p_{ij} , obtêm-se o sistema,

$$M_y^T P M_y \Rightarrow \begin{cases} m_{y11}^2 p_{11} + 2m_{y11}m_{y12}p_{12} + m_{y12}^2 p_{22} = \bar{p}_{11}^{yy} \\ m_{y11}m_{y12}p_{11} + (m_{y12}^2 + m_{y11}m_{y22})p_{12} + m_{y12}m_{y22}p_{22} = \bar{p}_{12}^{yy} \\ m_{y12}^2 p_{11} + 2m_{y12}m_{y22}p_{12} + m_{y22}^2 p_{22} = \bar{p}_{22}^{yy} \end{cases} \quad (4.18)$$

Na forma matricial em notação mais compacta obtêm-se o sistema,

$$\begin{bmatrix} m_{y11}^2 & 2m_{y11}m_{y12} & m_{y12}^2 \\ m_{y11}m_{y12} & (m_{y12}^2 + m_{y11}m_{y22}) & m_{y12}m_{y22} \\ m_{y12}^2 & 2m_{y12}m_{y22} & m_{y22}^2 \end{bmatrix} \begin{bmatrix} p_{11} \\ p_{12} \\ p_{22} \end{bmatrix} = \begin{bmatrix} \bar{p}_{11}^{yy} \\ \bar{p}_{12}^{yy} \\ \bar{p}_{22}^{yy} \end{bmatrix} \quad (4.19)$$

Assim o sistema completo com todas as equações assume a forma,

$$\left\{ \begin{array}{l} m_{u11}^2 p_{11} + 2m_{u11}m_{u12}p_{12} + m_{u12}^2 p_{22} = \bar{p}_{11}^{uu} \\ m_{u11}m_{u12}p_{11} + (m_{u12}^2 + m_{u11}m_{u22})p_{12} + m_{u12}m_{u22}p_{22} = \bar{p}_{12}^{uu} \\ m_{u12}^2 p_{11} + 2m_{u12}m_{u22}p_{12} + m_{u22}^2 p_{22} = \bar{p}_{22}^{uu} \\ m_{u11}m_{y11}p_{11} + (m_{u21}m_{y11} + m_{u11}m_{y21})p_{12} + m_{u21}m_{y21}p_{22} = \bar{p}_{11}^{uy} \\ m_{u11}m_{y12}p_{11} + (m_{u21}m_{y12} + m_{u11}m_{y22})p_{12} + m_{u21}m_{y22}p_{22} = \bar{p}_{12}^{uy} \\ m_{u12}m_{y11}p_{11} + (m_{u22}m_{y11} + m_{u12}m_{y21})p_{12} + m_{u22}m_{y21}p_{22} = \bar{p}_{21}^{uy} \\ m_{u12}m_{y12}p_{11} + (m_{u22}m_{y12} + m_{u12}m_{y22})p_{12} + m_{u22}m_{y22}p_{22} = \bar{p}_{22}^{uy} \\ m_{y11}^2 p_{11} + 2m_{y11}m_{y12}p_{12} + m_{y12}^2 p_{22} = \bar{p}_{11}^{yy} \\ m_{y11}m_{y12}p_{11} + (m_{y12}^2 + m_{y11}m_{y22})p_{12} + m_{y12}m_{y22}p_{22} = \bar{p}_{12}^{yy} \\ m_{y12}^2 p_{11} + 2m_{y12}m_{y22}p_{12} + m_{y22}^2 p_{22} = \bar{p}_{22}^{yy} \end{array} \right. \quad (4.20)$$

O sistema (4.20), possui dez equações, com as seis incógnitas, $m_{u11}, m_{u12}, m_{u22}, m_{y11}, m_{y12}, m_{y22}$. A forma matricial do sistema (4.20), assume a forma,

$$\begin{bmatrix} m_{u11}^2 & 2m_{u11}m_{u12} & m_{u12}^2 \\ m_{u11}m_{u12} & (m_{u12}^2 + m_{u11}m_{u22}) & m_{u12}m_{u22} \\ m_{u12}^2 & 2m_{u12}m_{u22} & m_{u22}^2 \\ m_{u11}m_{y11} & (m_{u21}m_{y11} + m_{u11}m_{y21}) & m_{u21}m_{y21} \\ m_{u11}m_{y12} & (m_{u21}m_{y12} + m_{u11}m_{y22}) & m_{u21}m_{y22} \\ m_{u12}m_{y11} & (m_{u22}m_{y11} + m_{u12}m_{y21}) & m_{u22}m_{y21} \\ m_{u12}m_{y12} & (m_{u22}m_{y12} + m_{u12}m_{y22}) & m_{u22}m_{y22} \\ m_{y11}^2 & 2m_{y11}m_{y12} & m_{y12}^2 \\ m_{y11}m_{y12} & (m_{y12}^2 + m_{y11}m_{y22}) & m_{y12}m_{y22} \\ m_{y12}^2 & 2m_{y12}m_{y22} & m_{y22}^2 \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} & p_{22} \end{bmatrix} = \begin{bmatrix} \bar{p}_{11}^{uu} \\ \bar{p}_{12}^{uu} \\ \bar{p}_{22}^{uu} \\ \bar{p}_{11}^{uy} \\ \bar{p}_{12}^{uy} \\ \bar{p}_{21}^{uy} \\ \bar{p}_{22}^{uy} \\ \bar{p}_{11}^{yy} \\ \bar{p}_{12}^{yy} \\ \bar{p}_{22}^{yy} \end{bmatrix} \quad (4.21)$$

Para a solução do sistema (4.20), que é não linear, pelos termos quadráticos e multiplicativos, pode-se aplicar técnicas de resolução de sistemas de equações não lineares. Um algoritmo adaptativo com passo de aprendizagem variável é usado para a busca da solução do sistema de equações não lineares.

O problema anteriormente citado é equivalente ao problema de resolução do sistema de equações ou mesmo como um problema de decomposição de matrizes. Uma decomposição que pode ser investigada seria a *Decomposição Real de Schur*, descrita pelo seguinte teorema:

Teorema [Decomposição Real de Schur]: Se $W \in \mathbb{R}^{n \times n}$, então existe uma matriz $G \in \mathbb{R}^{n \times n}$, ortogonal, tal que

$$G^T W G = \begin{bmatrix} R_{11} & R_{12} & \cdots & R_{1m} \\ 0 & R_{22} & \cdots & R_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R_{mm} \end{bmatrix} \quad (4.22)$$

sendo, cada R_{ii} uma matriz (1×1) ou uma matriz (2×2) , contendo os autovalores complexos conjugados.

Para a resolução do problema $M_u^T P M_u$, teria-se a alternativa de selecionar a matriz P e então verificar se a propriedade sobre ortogonalidade pode ser verificada, sendo assim possível a busca de uma solução para o problema em particular para valores de P pré-fixados. A matriz P , na formulação do *OPFB*, está relacionado ao problema do *DLQR* relacionado à alocação de polos ótima, associado ao índice de desempenho que minimiza os desvios dos estados, com menor esforço de controle.

Alguns resultados com relação a simetria de matrizes como a *Lei da Inércia* são necessários, para entendimento de algumas transformações, que podem levar à solução do problema

$X^T A X$. A inércia de uma matriz simétrica A , é uma tripa de inteiros não negativos (a, b, c) , sendo a , b e c , o número de autovalores negativos, nulos e positivos respectivamente.

Teorema [Lei de Inércia de Sylverter]: Se $A \in \mathbb{R}^{(n \times n)}$ é simétrico e $X \in \mathbb{R}^{(n \times n)}$ é não singular, então A e $X^T A X$ tem a mesma inércia.

A partir do resultado da inércia pode-se tentar verificar uma decomposição que mantenha a inércia da matriz.

A transformação da forma $A \rightarrow X^T A X$, sendo X não singular é chamado de transformação de congruência, se existe X . Portanto, uma transformação de congruência de uma matriz simétrica preserva a inércia.

Sabe-se que da teoria de congruências [KRESSNER; LIU, 2013] que se, $\tilde{X} = S^T X S$

$$\tilde{X} S^{-1} A S^{-T} X = S^T X S \quad (4.23)$$

para qualquer $S \in \mathbb{C}^{(n \times n)}$ não singular, sendo S^T a inversa de S e S^{-T} a inversa da transposta de S . Portanto pode-se substituir as transformações A e B pelas transformações dos seus contra-gradientes simultâneos:

$$A \mapsto S^{-1} A S^{-T}, B \mapsto S^T B S \quad (4.24)$$

Na seção seguinte apresenta-se uma alternativa para a solução do sistema não linear apresentado por meio da proposta de uma aproximação.

4.2 Aproximação das Matrizes M_u e M_y

Para a estimação dos estados usando a equação (4.1) é necessária a disponibilidade das matrizes M_u e M_y . A equação Eq.(3.71) contém as matrizes de bloco $M_u^T P M_u$ e $M_y^T P M_y$, que possuem dimensões equivalentes as das matrizes M_u e M_y , respectivamente, assim como também as propriedades de simetria e positividade.

Existe um resultado apresentado por Greville [1974], que para uma determinada matriz A , a solução da equação matricial $X A X = X$ existe, se e somente se, puder ser expressável na forma $X = (E A F)^\dagger$, onde E e F são matrizes Hermitianas e Idempotentes e $\dagger$ indica a inversa de Moore-Penrose.

A matriz $\bar{P}$ na equação (3.71), possui internamente as matrizes de blocos $M_u^T P M_u$ e $M_y^T P M_y$, que é iterativamente estimada pelos mínimos quadrados recursivo (RLS) e com isso seus valores numéricos estão disponíveis por meio da estimativa de seus coeficientes a cada iteração do algoritmo. Detalhes sobre a implementação do RLS são apresentados e discutidos em [BRADTKE et al., 1994].

Como as matrizes M_u e M_y são simétricas, apenas o bloco triangular superior é realmente considerado, tanto que apenas ele é usado para o cálculo da nova ação de controle, conforme descrito pelas equações (3.77) e (3.78).

Usando os blocos $M_u^T P M_u$ e $M_y^T P M_y$, a estimativa dos estados assume a seguinte forma:

$$x_k \approx M_y^T P M_y \bar{y}_{[k-1, k-N]} + M_u^T P M_u \bar{u}_{[k-1, k-N]}, \quad (4.25)$$

onde esses blocos são estimados a partir da matriz $\bar{P}$ pelo RLS a cada iteração. Em uma das simulações desenvolvidas neste trabalho, foi necessário incluir um fator de correção $[1 \ 1/\kappa]^T$ ao sistema testado, com o objetivo de penalizar o estado x_2 para realizar uma sintonia e correção de desvios.

Assim, a formula usada para estimar os estados foi,

$$x_k \approx M_y^T P M_y \bar{y}_{[k-1, k-N]} [1 \ 1/\kappa]^T + M_u^T P M_u \bar{u}_{[k-1, k-N]}. \quad (4.26)$$

Um esquema que descreve o processo geral sobre a observação do sistema dinâmico é apresentado na Figura 5, com a formulação baseada em programação dinâmica aproximada, usando algoritmo de aprendizagem por reforço por meio do iteração de política ou iteração de valor e as aproximações das matrizes M_u e M_y .

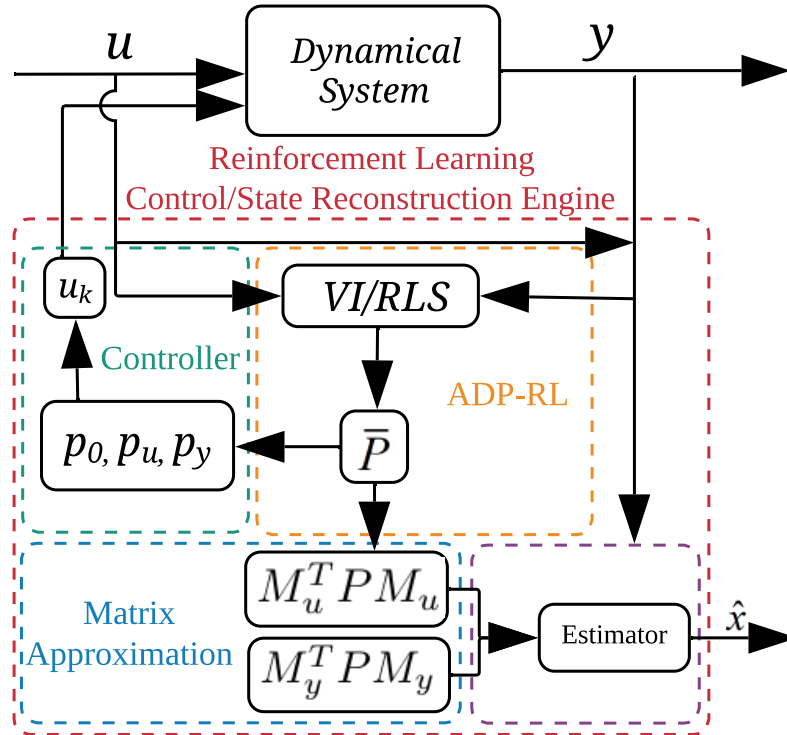


Figura 5 – Esquema ADP-RL para Reconstrução dos Estados

Fonte: Produzido pelos Autores.

Na seção a seguir é apresentada outra alternativa para a solução do sistema de equações não lineares, pelo uso de uma rede neuronal.

4.3 Rede Neuronal Sem Saturação para Solução de Sistemas de Equações Não Lineares

Nesta seção é apresentado a estrutura da rede neuronal, para solução do sistema de equações não lineares (4.20). Os pesos a serem calculados pela rede são necessariamente as raízes das equações que compõem o sistema. A abordagem apresentada a seguir tem como base as metodologias apresentadas por Li e Zeng [2008], Margaris e Adamopoulos [2007] e Margaris e Goulianas [2012]. No trabalho apresentado por Li e Zeng [2008], foi abordada uma rede para a solução de sistemas de equações não lineares em que o número de equação e incógnitas são iguais. Neste trabalho abordaremos o caso em que o número de equações e incógnitas são diferentes, mais especificamente um sistema de décima ordem com seis incógnitas, dado pela equação (4.20).

Trabalhos relacionados à solução de sistemas de equações algébricas, que representam as equações do sistema na camada de saída da rede usando redes neurais diretas, são discutidos nos trabalhos de Margaris e Adamopoulos [2007] e Margaris e Goulianas [2012]. Um trabalho aplicando redes neurais recorrentes do tipo Hopfield é abordado em [GOULIANAS et al., 2013]. Neste trabalho são apresentadas estruturas de redes cujas saídas representam as equações que compõem o sistema e com isso aplicam o treinamento de retro propagação do erro (backpropagation) em uma rede direta, usando a função identidade como função de ativação, que também é usada neste trabalho.

4.4 Formulação da Rede para Solução do Sistema de Equações

Considere o sistema de equações não lineares descrito a seguir:

$$\begin{cases} f_1(x_1, x_2, \dots, x_s) = 0 \\ f_2(x_1, x_2, \dots, x_s) = 0 \\ \vdots \\ \vdots \\ f_q(x_1, x_2, \dots, x_s) = 0 \end{cases}, \quad (4.27)$$

A solução do sistema de equações não lineares (4.27), dada uma condição inicial $X^0 = [x_1^0, x_2^0, \dots, x_s^0]^T$, implica que para qualquer i , ($i = 1, 2, \dots, q$), $f_i(x_1, x_2, \dots, x_s) = 0$, [LI; ZENG, 2008]. O ponto $(x_1, x_2, \dots, x_s)$, representam as raízes para o sistema, sendo na rede apresentada a seguir o conjunto de pesos a serem adaptados ao longo do processo de aprendizagem no treinamento da rede.

Para o processo de treinamento da rede, considere a função de erro descrita pela diferença entre o valor que a função $f_i(x_1, x_2, \dots, x_s)$ assume a cada iteração do processo de treinamento

com o valor desejado, que iguala cada uma das funções do sistema a 0, ou seja, a função do erro:

$$e(k) = 0 - f_i(x_1, x_2, \dots, x_s) = -f_i(x_1, x_2, \dots, x_s), \quad (4.28)$$

sendo, $e(i)$ o erro da equação f_i no ponto $(x_1, x_2, \dots, x_s)$.

Considere o índice de desempenho J a seguir:

$$J = \frac{1}{2} \sum_{i=1}^q e^2(i) \quad (4.29)$$

sendo, J o índice de desempenho da ponderação dos erros associados a cada uma das f_i funções que compõem o sistema, para um dado conjunto de pontos de $X(h) = [x_1(h), x_2(h), \dots, x_s(h)]^T$, sendo $X(h)$, o vetor contendo os pesos e h o número iteração para o treinamento.

Para a realização do treinamento da rede deve-se determinar o conjunto de pesos ótimos que representam as raízes do sistema, de modo a minimizar o índice de desempenho J , que sucessivamente também minimizam as equações que compõem o sistema. Deste modo diferencia-se a função de desempenho J com relação aos pesos x_j , de modo a identificar-se uma regra recursiva usando a regra do gradiente descendente a cada uma das funções f_i , ($i = 1, 2, \dots, q$), com relação ao conjunto de pesos ao longo do treinamento:

Seja Δx_j^i a variação dos pesos x_j^i na m -ésima iteração, sendo j o índice referente ao peso x_j e i referente ao índice associado as q equações que compõem o sistema em questão. Tem-se que a variação dos pesos pode ser expressa como:

$$\Delta x_j^i = -\eta(i) \frac{\partial J}{\partial x_j^i} = \eta(i) e(i) \frac{\partial f_i(x_1, x_2, \dots, x_s)}{\partial x_j^i}, \quad (4.30)$$

sendo $\eta(i)$ a taxa de aprendizagem variável associado a cada uma das funções f_i , podendo assim realizar um processo de aprendizagem ao longo de treinamento, de modo variável e assim acelerar a convergência e melhorar a exatidão.

Como $\Delta x_j^i = x_j^i - x_j^{i+1}$, isolando a variável x_j^{i+1} e substituindo na equação (4.30), obtemos a seguinte regra recursiva para atualização dos pesos da rede ao longo do processo de treinamento.

$$x_j^{i+1} = x_j^i + \eta(i) e(i) \frac{\partial f_i(x_1, x_2, \dots, x_s)}{\partial x_j^i}, \quad (j = 1, 2, \dots, s) \quad (4.31)$$

Seja $\mathbf{X}^i = [x_1^i, x_2^i, \dots, x_s^i]^T$, e o vetor $\mathbf{F}_X(i)$, sendo o vetor que contém as derivadas parciais das funções f_i , ($i = 1, 2, \dots, q$), com relação às variáveis x_j , ($j = 1, 2, \dots, s$), no ponto definido pelo vetor X ao longo do treinamento, como sendo,

$$\mathbf{F}_X(i) = \left[\frac{\partial f_i(\mathbf{X}^i)}{\partial x_1^i}, \frac{\partial f_i(\mathbf{X}^i)}{\partial x_2^i}, \dots, \frac{\partial f_i(\mathbf{X}^i)}{\partial x_s^i} \right]^T, \quad (4.32)$$

obtem-se a seguinte regra iterativa para a atualização dos pesos, ao longo do processo iterativo,

$$\mathbf{X}^{i+1} = \mathbf{X}^i + \eta(i) e(i) \mathbf{F}_X(i), \quad (4.33)$$

sendo $\eta(i)$ a taxa de aprendizagem adaptativa, com passo variável e $0 < \eta(i) < 1$.

Para assegurar a convergência absoluta do algoritmo o seguinte teorema foi usado por [Li e Zeng \[2008\]](#), para a seleção da taxa de aprendizagem apropriadamente.

Teorema 2: Suponha $f_i \in C[a_i, b_i]$, somente quando

$$0 < \eta(i) < \frac{2}{\left\| \frac{\partial f_i(\mathbf{X}^i)}{\partial \mathbf{X}^i} \right\|^2} \quad (4.34)$$

o algoritmo é convergente, sendo o operador $\|\cdot\|^2$, a normal Euclidiana e $\frac{\partial f_i(\mathbf{X}^i)}{\partial \mathbf{X}^i}$, o vetor contendo as derivadas parciais da função f_i com relação as variáveis x_j^i na iteração i , representadas em notação compacta pelo vetor $\mathbf{X}^i$.

Para a seleção ótima do parâmetro de aprendizagem adaptativo n_{opt} , em [Li e Zeng \[2008\]](#) é recomendado uso de valores para o numerador para calcular n_{opt} , entre 1.0 e 1.6. Na equação a seguir o numerador é apresentado o intervalo entre 1 e 1.6.

$$\eta_{opt}(i) = \frac{(1.0 - 1.6)}{\left\| \frac{\partial f_i(\mathbf{X}^i)}{\partial \mathbf{X}^i} \right\|^2} \quad (4.35)$$

4.5 Observador Estado Baseados em Aprendizagem Q

Nesta seção são apresentadas, contribuições com formulações para observadores de estados e algoritmos baseadas no paradigma de aprendizagem por reforço, embasadas em aprendizagem-Q. Nas subseções são apresentadas as formulações e descrições das funções Q, para estimação de estado em sistemas dinâmicos discretos. As formulações apresentadas, visam avaliar o comportamento dinâmico dos observadores de estado e do estimador com relação às saídas estimadas.

As abordagens apresentadas tem como motivação, o problema de controle ótimo formulado a partir de abordagens em programação dinâmica (DP) e programação dinâmica aproximada dependente de ação (AD-HDP), apresentados nos trabalhos de [Lewis e Syrmos \[1995\]](#), [Lewis et al. \[2012\]](#), [Athans e Falb \[2013\]](#) e [Bertsekas et al. \[2005\]](#). Outro trabalho associado ao problema de controle ótimo e programação dinâmica aproximada, que aborda, desde as formulações, variações, aspectos de implementação e análise é o trabalho de [Powell \[2007\]](#). Essas metodologias e algoritmos propõem estratégias para solucionar o problema de controle ótimo por meio da minimização de um índice de desempenho, associado ao esforço de controle e variações dos estados.

O algoritmo Action Dependent Heuristic Dynamic Programming (AD-HDP) abordado por [Al-Tamimi et al. \[2007\]](#), apresenta uma formulação que não depende de modelo dinâmico do sistema para solucionar o problema de controle ótimo, com o conhecimento dos estados e entradas passadas, calcula o ganho do controlador, [[LEE et al., 2009](#)].

Na abordagem a seguir apresenta-se uma formulação proposta, para a estimação de estado em sistemas dinâmicos discretos, independente de modelo dinâmico do sistema.

4.6 Observador AD-HDP para o LQR Discreto

O observador de estados orientado por dados que é proposto nessa seção é baseado no problema de programação dinâmica heurística dependente de ação, a formulação AD-HDP [SUTTON; BARTO, 2018]. A principal contribuição na formulação proposta é a utilização indireta do estado estimado por uma aproximação linear, em função das saídas mensuradas e levando em consideração a dualidade do filtro de Kalman.

Sabe-se que no algoritmo da programação dinâmica Heurística - HDP, necessita-se do conhecimento das matrizes A e B . Inicialmente tem-se interesse de estudar o comportamento de variação dos estados com relação as ações de controle e com isso tem-se que diferenciar em relação ao controle u_k como mostra a Eq.(4.36).

$$\frac{\partial x_{k+1}}{\partial u_k} = g(x_k). \quad (4.36)$$

Entretanto é exigido do sistema a matriz de entrada $g(x_k)$. Para se evitar a necessidade de qualquer conhecimento do sistema, deve-se fornecer uma alternativa para se obter derivadas parciais em relação à entrada de controle. Neste seção, o princípio da otimalidade de *Bellman* é formulado usando o conceito de Função-Q (*Q-function*), ao invés da Função Valor padrão. A equação de *Bellman* nos permite calcular o valor usando qualquer política de controle admissível. Entretanto, fazendo uso da aproximação $x_k = C^{-1}y$, escolhendo uma matriz C inversível, ou mesmo fazendo uso de alguma forma de pseudo inversa de Penrose, pode-se definir a Função-Q associada com a política $u_k = h(C^{-1}y_k)$ da seguinte forma:

$$Q_h(C^{-1}y_k, u_k) = r(C^{-1}y_k, u_k) + \gamma V_h(C^{-1}y_{k+1}), \quad (4.37)$$

sendo, C a matriz de transição de estados associado a saída do sistema dinâmico e considerando que a mesma seja inversível. Embora a matriz C possa não ser quadrada, é necessário manipular a matriz de modo a permitir que a mesma se torne quadrada, ou caso não seja considerada, pode-se calcular a pseudo inversa de Moore-Penrose.

Percebe-se que a Função-Q, fica em função da saída ponderada pela matriz $C^{-1}y_k$ e do controle u_k no tempo k . A Função-Q ótima é dada por:

$$Q^*(C^{-1}y_k, u_k) = r(C^{-1}y_k, u_k) + \gamma V^*(C^{-1}y_{k+1}). \quad (4.38)$$

Em termos de Q^* , pode-se escrever a equação de otimalidade de *Bellman* da seguinte forma:

$$V^*(C^{-1}y_k) = \min_{u_k} (Q^*(C^{-1}y_k, u_k)), \quad (4.39)$$

e o controle ótimo da forma:

$$h^*(C^{-1}y_k) = \arg \min_{u_k} (Q^*(C^{-1}y_k, u_k)), \quad (4.40)$$

O valor mínimo então pode ser obtido diferenciando e igualando a zero, como segue,

$$\frac{\partial Q^*(C^{-1}y_k, u_k)}{\partial u_k} = 0. \quad (4.41)$$

Ao contrário da Eq.(4.36), assumindo-se que se conheça a Função-Q para todo $(C^{-1}y_k, u_k)$, não necessita-se do conhecimento da dinâmica do sistema. No algoritmo HDP, deve-se aprender e armazenar o valor ótimo para todos os estados $x_k = C^{-1}y_k$ possíveis. Em contraste, na aprendizagem pela Função-Q, deve-se armazenar o valor ótimo da Função-Q, para todos os valores de $(C^{-1}y_k, u_k)$ possíveis, isto é, para toda ação de controle realizada a cada estado possível. Assim, tem-se muito mais informação a ser processada.

Para determinar-se a equação de ponto fixo para a Função-Q, nota-se que:

$$Q_h(C^{-1}y_k, h(x_k)) = V_h(C^{-1}y_k). \quad (4.42)$$

Então, a "equação de *Bellman*" para Q será,

$$Q_h(C^{-1}y_k, u_k) = r(C^{-1}y_k, u_k) + \gamma Q_h(C^{-1}y_{k+1}, h(C^{-1}y_{k+1})). \quad (4.43)$$

O valor Q ótimo:

$$Q^*(C^{-1}y_k, u_k) = r(C^{-1}y_k, u_k) + \gamma Q^*(C^{-1}y_{k+1}, h^*(C^{-1}y_{k+1})) \quad (4.44)$$

A Eq.(4.43) é a equação de ponto fixo ou "equação de *Bellman*" para a Função-Q. A partir dela pode-se aplicar qualquer técnica de aprendizagem por reforço para se obter a solução, incluindo PI e VI.

4.6.1 Caracterização da Configuração Função-Q para o LQR Discreto

A partir da formulação já proposta usando Função-Q, agora leva-se a uma aplicação para o caso do DLQR. Partindo-se da Eq.(4.37) tem-se:

$$\begin{aligned} Q_K(C^{-1}y, u_k) &= [C^{-1}y_k]^T Q [C^{-1}y_k] + u_k^T R u_k + \gamma \left[(C^{-1}y_{k+1})^T P C^{-1}y_{k+1} \right] = \\ &= \begin{bmatrix} y_k^T (C^{-1})^T \\ u_k^T \end{bmatrix}^T H \begin{bmatrix} C^{-1}y_k \\ u_k \end{bmatrix} = \begin{bmatrix} y_k^T \\ u_k^T \end{bmatrix}^T (C^{-1})^T H C^{-1} \begin{bmatrix} y_k \\ u_k \end{bmatrix}. \end{aligned} \quad (4.45)$$

Sendo P a solução da equação de *Lyapunov* para um determinado ganho de controle K e uma matriz H associada a solução de *Lyapunov*. Desenvolvendo-se a Eq.(4.45) tem-se:

$$\begin{aligned} \begin{bmatrix} y_k^T (C^{-1})^T \\ u_k^T \end{bmatrix}^T H \begin{bmatrix} C^{-1} y_k \\ u_k \end{bmatrix} &= y_k^T (C^{-1})^T Q C^{-1} y_k + u_k^T R u_k + \gamma (y_{k+1}^T (C^{-1})^T Q C^{-1} y_{k+1}) = \\ &= \begin{bmatrix} y_k^T \\ u_k^T \end{bmatrix}^T \begin{bmatrix} (C^{-1})^T Q C^{-1} & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} y_k \\ u_k \end{bmatrix} + \gamma \left(\begin{bmatrix} y_k^T \\ u_k^T \end{bmatrix}^T \begin{bmatrix} (C^{-1})^T A^T \\ B^T \end{bmatrix} P \begin{bmatrix} A C^{-1} \\ B \end{bmatrix}^T \begin{bmatrix} y_k \\ u_k \end{bmatrix} \right). \end{aligned} \quad (4.46)$$

Desenvolvendo-se a Eq.(4.46) para o lado direito de (4.45) tem-se que:

$$Q_K(C^{-1}y_k, u_k) = \begin{bmatrix} y_k^T \\ u_k^T \end{bmatrix}^T \begin{bmatrix} (C^{-1})^T Q C^{-1} + \gamma (C^{-1})^T A^T P A C^{-1} & \gamma (C^{-1})^T A^T P B \\ \gamma B^T P A C^{-1} & R + \gamma B^T P B \end{bmatrix} \begin{bmatrix} y_k \\ u_k \end{bmatrix}. \quad (4.47)$$

A Eq.(4.47) é a Função-Q, quadrática em (y_k, u_k) para o LQR discreto. A matriz H então é dada por:

$$H = \begin{bmatrix} (C^{-1})^T Q C^{-1} + \gamma (C^{-1})^T A^T P A C^{-1} & \gamma (C^{-1})^T A^T P B \\ \gamma B^T P A C^{-1} & R + \gamma B^T P B \end{bmatrix}. \quad (4.48)$$

Aplicando a dualidade para o filtro de Kalman, em que, $A = A^T$ e $B = C^T$, assim tem-se H reescrito como:

$$H = \begin{bmatrix} (C^{-1})^T Q C^{-1} + \gamma (C^{-1})^T A P A^T C^{-1} & \gamma (C^{-1})^T A P C^T \\ \gamma C^T P A C^{-1} & R + \gamma C^T P C^T \end{bmatrix}. \quad (4.49)$$

Agora tem-se a função H usando somente as matrizes A e C , e usando a "equação de *Bellman*", Eq.(4.43) para o caso da Função-Q, através da Eq.(4.44), tem-se para o caso do LQR discreto:

$$\begin{aligned} \begin{bmatrix} y_k^T \\ u_k^T \end{bmatrix}^T H \begin{bmatrix} y_k \\ u_k \end{bmatrix} &= r(C^{-1}y_k, u_k) + \gamma Q^*(C^{-1}y_{k+1}, u_{k+1}), \\ &= y_k^T Q y_k + u_k^T R u_k + \gamma \begin{bmatrix} y_{k+1}^T \\ u_{k+1}^T \end{bmatrix}^T H \begin{bmatrix} y_{k+1} \\ u_{k+1} \end{bmatrix}. \end{aligned} \quad (4.50)$$

Atribuindo-se $h(C^{-1}y_k) = u_k = -KC^{-1}y_k$, tem-se:

$$\begin{aligned}
&= \begin{bmatrix} y_k^T \\ u_k^T \end{bmatrix}^T \begin{bmatrix} (C^{-1})^T QC^{-1} & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} y_k \\ u_k \end{bmatrix} \\
&+ \gamma \begin{bmatrix} (AC^{-1}y_k + Bu_k)^T \\ -(K(AC^{-1}y_k + Bu_k))^T \end{bmatrix}^T H \begin{bmatrix} AC^{-1}y_k + Bu_k \\ -K(AC^{-1}y_k + Bu_k) \end{bmatrix}, \\
&= \begin{bmatrix} y_k^T \\ u_k^T \end{bmatrix}^T \begin{bmatrix} (C^{-1})^T QC^{-1} & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} y_k \\ u_k \end{bmatrix} + \\
&+ \gamma \begin{bmatrix} (AC^{-1}y_k)^T + (Bu_k)^T \\ -((AC^{-1}y_k)^T + (Bu_k)^T)K^T \end{bmatrix}^T H \begin{bmatrix} AC^{-1}y_k + Bu_k \\ -KAC^{-1}y_k - KBu_k \end{bmatrix}, \\
&= \begin{bmatrix} y_k^T \\ u_k^T \end{bmatrix}^T \begin{bmatrix} (C^{-1})^T QC^{-1} & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} y_k \\ u_k \end{bmatrix} + \\
&+ \gamma \begin{bmatrix} y_k^T (C^{-1})^T A^T + u_k^T B^T \\ -y_k^T (C^{-1})^T A^T K^T - u_k^T B^T K^T \end{bmatrix}^T H \begin{bmatrix} AC^{-1}y_k + Bu_k \\ -KAC^{-1}y_k - KBu_k \end{bmatrix}, \\
&= \begin{bmatrix} y_k^T \\ u_k^T \end{bmatrix}^T \begin{bmatrix} (C^{-1})^T QC^{-1} & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} y_k \\ u_k \end{bmatrix} + \\
&+ \gamma \begin{bmatrix} y_k^T \\ u_k^T \end{bmatrix}^T \begin{bmatrix} (C^{-1})^T A^T & B^T \\ -(C^{-1})^T A^T K^T & -B^T K^T \end{bmatrix}^T H \begin{bmatrix} AC^{-1} & B \\ -KAC^{-1} & -KB \end{bmatrix} \begin{bmatrix} y_k \\ u_k \end{bmatrix}.
\end{aligned} \tag{4.51}$$

4.7 Observador de Estados e Aprendizagem Q

Nesta seção são apresentadas contribuições desta Tese, com três abordagens model free para estimação de estados seguindo a linha de desenvolvimento do algoritmo AD-HDP com características específicas para cada uma das formulações apresentadas. Em cada uma das formulações apresentadas é modificado a dependência das informações e dados necessários para aplicação das mesmas, consequentemente gerando diferentes tipos de observadores.

4.7.1 Formulação da Função-Q para Observador de Estado $(\hat{x}, \tilde{x})$

Nesta seção é apresentado a formulação para o problema de observação de estados a partir das saídas mensuradas e dos erros entre as saídas estimadas e mensuradas. Nesta seção é apresentado a formulação da função Q , evidenciando a dependência dos estados estimados e erros dos estados estimados do observador.

Considere o sistema dinâmico discreto:

$$\begin{aligned}
x_{k+1} &= Ax_k + Bu_k + Gw_k, \\
y_k &= Cx_k,
\end{aligned} \tag{4.52}$$

sendo A a matriz dinâmica do sistema, C a matriz associadas as saídas do sistema e G a matriz associada as excitações sobre o sistema e ponderado por um ruído Gaussiano w_k .

Considere o sistema dinâmico discreto do observador de estados:

$$\begin{aligned}\hat{x}_{k+1} &= A\hat{x}_k + L(y_k - \hat{y}_k), \\ \hat{y}_k &= C\hat{x}_k,\end{aligned}\tag{4.53}$$

sendo $\hat{x}_k$ o estado estimado e $\hat{y}_k$ a saída estimada, L o ganho do observador e y_k a saída mensurada do sistema (4.52), na iteração k .

A metodologia descrita nesta seção tem por objetivo apresentar a formulação da função valor da saída estimada $V(\hat{y}_k)$, por meio da construção de uma aproximação pela função- $Q(\hat{y}, \tilde{y})$. A função $r(\hat{y}_k, \tilde{y}_k)$, representa o custo de transição das saídas estimadas e dos erros entre as saídas mensuradas e estimadas, ou seja, $\tilde{y}_k = y_k - \hat{y}_k$. Seja $V(\hat{y}_k)$ a função valor aproximada pela função Q definida por:

$$Q(\hat{y}_k, \tilde{y}_k) = r(\hat{y}_k, \tilde{y}_k) + \gamma Q(\hat{y}_{k+1}, \tilde{y}_{k+1}),\tag{4.54}$$

$$Q(\hat{y}_k, \tilde{y}_k) = \hat{y}_k^T Q \hat{y}_k + \tilde{y}_k^T R \tilde{y}_k + \gamma(\hat{y}_{k+1}^T P \hat{y}_{k+1}),\tag{4.55}$$

sendo, γ o fator de desconto utilizado para descontar os custos futuros ao longo do tempo.

Assim pode-se definir a equação de ponto fixo da função valor $V(\hat{y}_k)$ na Função-Q, considerando que a saída mensurada $y_k = Cx_k$, definindo $\tilde{x}_k = x_k - \hat{x}_k$. Considerando o observador de estado com equação de estados estimados e saída no sistema (4.53), tem-se o seguinte desenvolvimento para função valor $V(\hat{y}_k)$ e função Q associada.

$$\begin{aligned}Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{y}_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} \hat{y}_k \\ \tilde{y}_k \end{bmatrix} + \gamma(\hat{y}_{k+1}^T P \hat{y}_{k+1}) \\ Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} (C\hat{x}_k)^T \\ (y_k - \hat{y}_k)^T \end{bmatrix}^T \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} C\hat{x}_k \\ y_k - \hat{y}_k \end{bmatrix} + \gamma(\hat{y}_{k+1}^T P \hat{y}_{k+1}) \\ Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} (C\hat{x}_k)^T \\ (Cx_k - C\hat{x}_k)^T \end{bmatrix}^T \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} C\hat{x}_k \\ (Cx_k - C\hat{x}_k) \end{bmatrix} + \gamma[(C\hat{x}_{k+1})^T PC\hat{x}_{k+1}] \\ Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T C^T \\ (C(x_k - \hat{x}_k))^T \end{bmatrix}^T \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} C\hat{x}_k \\ C\tilde{x}_k \end{bmatrix} + \gamma[(C(A\hat{x}_k + L\tilde{y}_k))^T PC(A\hat{x}_k + L\tilde{y}_k)] \\ Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T C^T \\ (C\tilde{x}_k)^T \end{bmatrix}^T \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} C\hat{x}_k \\ C\tilde{x}_k \end{bmatrix} + \gamma[(A\hat{x}_k + L\tilde{y}_k)^T C^T PC(A\hat{x}_k + L\tilde{y}_k)] \\ Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T C^T \\ \tilde{x}_k^T C^T \end{bmatrix}^T \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} C\hat{x}_k \\ C\tilde{x}_k \end{bmatrix} + \gamma[((A\hat{x}_k)^T C^T + (L\tilde{y}_k)^T C^T)P(CA\hat{x}_k + CL\tilde{y}_k)] \\ Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T \\ \tilde{x}_k^T \end{bmatrix}^T \begin{bmatrix} C^T QC & 0 \\ 0 & C^T RC \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{x}_k \end{bmatrix} + \gamma[(\hat{x}_k^T A^T C^T + \tilde{y}_k^T L^T C^T)P(CA\hat{x}_k + CLC\tilde{x}_k)] \\ Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T \\ \tilde{x}_k^T \end{bmatrix}^T \begin{bmatrix} C^T QC & 0 \\ 0 & C^T RC \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{x}_k \end{bmatrix} + \\ &\gamma[(\hat{x}_k^T A^T C^T + (C\tilde{x}_k)^T L^T C^T)P(CA\hat{x}_k + CLC\tilde{x}_k)]\end{aligned}$$

$$\begin{aligned}
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T \\ \tilde{x}_k^T \end{bmatrix}^T \begin{bmatrix} C^T Q C & 0 \\ 0 & C^T R C \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{x}_k \end{bmatrix} + \\
&\gamma \left[(\hat{x}_k^T A^T C^T + \tilde{x}_k^T C^T L^T C^T) P (C A \hat{x}_k + C L C \tilde{x}_k) \right] \\
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T \\ \tilde{x}_k^T \end{bmatrix}^T \begin{bmatrix} C^T Q C & 0 \\ 0 & C^T R C \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{x}_k \end{bmatrix} + \gamma \left[\begin{bmatrix} \hat{x}_k^T \\ \tilde{x}_k^T \end{bmatrix}^T \begin{bmatrix} A^T C^T \\ C^T L^T C^T \end{bmatrix} P \begin{bmatrix} C A \\ C L C \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{x}_k \end{bmatrix} \right] \\
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T \\ \tilde{x}_k^T \end{bmatrix}^T \begin{bmatrix} C^T Q C & 0 \\ 0 & C^T R C \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{x}_k \end{bmatrix} + \\
&\gamma \left[\begin{bmatrix} \hat{x}_k^T \\ \tilde{x}_k^T \end{bmatrix}^T \begin{bmatrix} A^T C^T P C A & A^T C^T P C L C \\ C^T L^T C^T P C A & C^T L^T C^T P C L C \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{x}_k \end{bmatrix} \right]
\end{aligned}$$

$$\begin{aligned}
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T \\ \tilde{x}_k^T \end{bmatrix}^T \begin{bmatrix} C^T Q C & 0 \\ 0 & C^T R C \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{x}_k \end{bmatrix} + \\
&\begin{bmatrix} \hat{x}_k^T \\ \tilde{x}_k^T \end{bmatrix}^T \begin{bmatrix} \gamma A^T C^T P C A & \gamma A^T C^T P C L C \\ \gamma C^T L^T C^T P C A & \gamma C^T L^T C^T P C L C \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{x}_k \end{bmatrix} \\
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T \\ \tilde{x}_k^T \end{bmatrix}^T \begin{bmatrix} C^T Q C + \gamma A^T C^T P C A & \gamma A^T C^T P C L C \\ \gamma C^T L^T C^T P C A & C^T R C + \gamma C^T L^T C^T P C L C \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{x}_k \end{bmatrix} \quad (4.56)
\end{aligned}$$

Definindo $H_{\hat{x}, \tilde{x}}$, como o core da matriz na Eq.(4.56), temos:

$$H_{\hat{x}, \tilde{x}} = \begin{bmatrix} C^T Q C + \gamma A^T C^T P C A & \gamma A^T C^T P C L C \\ \gamma C^T L^T C^T P C A & C^T R C + \gamma C^T L^T C^T P C L C \end{bmatrix}. \quad (4.57)$$

Na equação (4.56), tem-se uma estrutura que depende dos estados estimados e do erro de estimação somente. Essa matriz pode ser estimada por métodos iterativos de estimação paramétrica.

4.7.2 Formulação da Função-Q para Observador de Estado ($\hat{x}$, $\tilde{y}$)

Nesta seção apresenta-se a formulação para o problema de observação de estado a partir das saídas mensuradas e dos erros entre as saídas estimadas e mensuradas. A formulação da função Q é apresentada, evidenciando a dependência dos estados estimados e dos erros entre as saídas estimadas e mensuradas.

Considere o sistema dinâmico discreto:

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + Gw_k, \\ y_k &= Cx_k, \end{aligned} \quad (4.58)$$

sendo, A a matriz dinâmica do sistema, C a matriz associadas as saídas do sistema e G a matriz associada as excitações o sistema ponderados por um ruído Gaussiano w_k .

Considere o sistema dinâmico discreto do observador de estado:

$$\begin{aligned} \hat{x}_{k+1} &= A\hat{x}_k + L(y_k - \hat{y}_k), \\ \hat{y}_k &= C\hat{x}_k, \end{aligned} \quad (4.59)$$

sendo, $\hat{x}_k$ o estado estimado e $\hat{y}_k$ a saída estimada, L o ganho do observador e y_k a saída mensurada do sistema (4.58), na iteração k .

A metodologia descrita nesta seção tem por objetivo apresentar a formulação da função valor da saída estimada $V(\hat{y}_k)$, aproximada por uma função- $Q(\hat{y}_k, \tilde{y}_k)$. A função $r(\hat{y}_k, \tilde{y}_k)$, representa o custo de transição das saídas estimadas e dos erros entre as saídas mensuradas e estimadas.

Nesta abordagem tem-se interesse de identificar o comportamento da custo acumulado das saídas estimadas para um dado erro estimado $\tilde{y}_k = y_k - \hat{y}_k$. Seja a função Valor $V(\hat{y}_k)$, aproximada pela função- Q a seguir:

$$Q(\hat{y}_k, \tilde{y}_k) = r(\hat{y}_k, \tilde{y}_k) + \gamma Q(\hat{y}_{k+1}, \tilde{y}_{k+1}), \quad (4.60)$$

$$Q(\hat{y}_k, \tilde{y}_k) = \hat{y}_k^T Q \hat{y}_k + \tilde{y}_k^T R \tilde{y}_k + \gamma(\hat{y}_{k+1}^T P \hat{y}_{k+1}), \quad (4.61)$$

sendo γ , o fator de desconto utilizado para descontar os custos futuros ao longo do tempo.

Desenvolvendo a Eq.(4.61) e fazendo as devidas substituições com base nas equações já apresentadas, segue o desenvolvimento:

$$\begin{aligned}
Q(\hat{y}_k, \tilde{y}_k) &= \hat{y}_k^T Q \hat{y}_k + \tilde{y}_k^T R \tilde{y}_k + \gamma(\hat{y}_{k+1}^T P \hat{y}_{k+1}) \\
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{y}_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} \hat{y}_k \\ \tilde{y}_k \end{bmatrix} + \gamma(\hat{y}_{k+1}^T P \hat{y}_{k+1}) \\
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} (C\hat{x}_k)^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} \hat{y}_k \\ \tilde{y}_k \end{bmatrix} + \gamma[(C\hat{x}_{k+1})^T P C \hat{x}_{k+1}] \\
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T C^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} C\hat{x}_k \\ \tilde{y}_k \end{bmatrix} + \gamma[(C(A\hat{x}_k + L\tilde{y}_k))^T P C (A\hat{x}_k + L\tilde{y}_k)] \\
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} C^T Q C & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{y}_k \end{bmatrix} + \gamma[(A\hat{x}_k + L\tilde{y}_k)^T C^T P C (A\hat{x}_k + L\tilde{y}_k)] \\
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} C^T Q C & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{y}_k \end{bmatrix} + \gamma[((A\hat{x}_k)^T C^T + (L\tilde{y}_k)^T C^T) P (C A \hat{x}_k + C L \tilde{y}_k)] \\
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} C^T Q C & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{y}_k \end{bmatrix} + \gamma[(\hat{x}_k^T A^T C^T + \tilde{y}_k^T L^T C^T) P (C A \hat{x}_k + C L \tilde{y}_k)] \\
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} C^T Q C & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{y}_k \end{bmatrix} + \gamma \left[\begin{bmatrix} \hat{x}_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} A^T C^T \\ L^T C^T \end{bmatrix} P \begin{bmatrix} C A \\ C L \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{y}_k \end{bmatrix} \right] \\
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} C^T Q C & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{y}_k \end{bmatrix} + \gamma \left[\begin{bmatrix} \hat{x}_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} A^T C^T \\ L^T C^T \end{bmatrix} P \begin{bmatrix} C A \\ C L \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{y}_k \end{bmatrix} \right] \\
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} C^T Q C & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{y}_k \end{bmatrix} + \gamma \left[\begin{bmatrix} \hat{x}_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} A^T C^T P C A & A^T C^T P C L \\ L^T C^T P C A & L^T C^T P C L \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{y}_k \end{bmatrix} \right] \\
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} C^T Q C & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{y}_k \end{bmatrix} + \begin{bmatrix} \hat{x}_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} \gamma A^T C^T P C A & \gamma A^T C^T P C L \\ \gamma L^T C^T P C A & \gamma L^T C^T P C L \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{y}_k \end{bmatrix} \\
Q(\hat{y}_k, \tilde{y}_k) &= \begin{bmatrix} \hat{x}_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} C^T Q C + \gamma A^T C^T P C A & \gamma A^T C^T P C L \\ \gamma L^T C^T P C A & R + \gamma L^T C^T P C L \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \tilde{y}_k \end{bmatrix} \quad (4.62)
\end{aligned}$$

Obtem-se a matriz $H_{\hat{x}, \tilde{y}}$, sendo o core da matriz da Eq.(4.62),

$$H_{\hat{x}, \tilde{y}} = \begin{bmatrix} C^T Q C + \gamma A^T C^T P C A & \gamma A^T C^T P C L \\ \gamma L^T C^T P C A & R + \gamma L^T C^T P C L \end{bmatrix}. \quad (4.63)$$

A partir da Eq.(4.62), é possível calcular a derivada de primeira ordem, para achar o ponto extremo da equação de ponto fixo da função $Q(\hat{y}_k, \tilde{y}_k)$ em relação ao erro da estimação, de modo a determinar o ganho L do observador.

4.7.3 Formulação da Função- Q para Observador de Estado $(\hat{x}, u, \tilde{y})$

Nesta seção a formulação da função Q é apresentada, que avalia os estados estimados, em função dos erros estimados $\tilde{y}_k = y_k - \hat{y}_k$, sendo saída mensurada y_k e das saídas estimadas $\hat{y}_k$, para o sistema dinâmico discreto para uma dada ação de controle u_k .

Considere o sistema dinâmico discreto descrito pelas equações:

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + Gw_k, \\ y_k &= Cx_k + Dv_k, \end{aligned} \quad (4.64)$$

sendo, A a matriz dinâmica do sistema, C a matriz associadas as saídas do sistema e G a matriz associada as excitações sobre o sistema ponderados por um ruído Gaussiano w_k e D matriz de alimentação direta ponderada por ruído Gaussiano v_k .

Considere o sistema dinâmico discreto para o observador de estados:

$$\begin{aligned} \hat{x}_{k+1} &= A\hat{x}_k + Bu_k + L(y_k - \hat{y}_k), \\ \hat{y} &= C\hat{x}_k, \end{aligned} \quad (4.65)$$

sendo, $\hat{x}$ o estado estimado, $\hat{y}$ a saída estimada, L o ganho do observador e y_k a saída mensurada do sistema (4.64), na iteração k .

A metodologia descrita nesta seção tem por objetivo apresentar a formulação da função valor da saída estimada $V(\hat{y}_k)$, aproximada pela função $Q(\hat{y}, u_k, \tilde{y})$ desenvolvida a seguir. A função $r(\hat{y}_k, u_k, \tilde{y}_k)$, representa o custo de transição das saídas estimadas e erros entre as saídas mensuradas e estimadas, $\tilde{y}_k = y_k - \hat{y}_k$, para uma dada política de controle u_k . Seja a função Valor $V(\hat{y}_k)$, aproximada pela função- Q a seguir:

$$Q(\hat{x}_k, u_k, \tilde{y}_k) = r(\hat{x}_k, u_k, \tilde{y}_k) + \gamma Q(\hat{x}_{k+1}), \quad (4.66)$$

$$Q(\hat{x}_k, u_k, \tilde{y}_k) = \hat{x}_k^T Q \hat{x}_k + u_k^T R u_k + \tilde{y}_k^T S \tilde{y}_k + \gamma(\hat{x}_{k+1}^T P \hat{x}_{k+1}), \quad (4.67)$$

sendo, γ o fator de desconto utilizado para descontar os custos futuros ao longo do tempo. Assim pode-se então definir a equação de ponto fixo da função valor $V(\hat{y}_k)$ na função- Q , $Q(\hat{x}_k, u_k, \tilde{y}_k)$. A seguir o desenvolvimento para função valor pela função $Q(\hat{x}_k, u_k, \tilde{y}_k)$ é realizado.

$$Q(\hat{x}_k, u_k, \tilde{y}_k) = r(\hat{x}_k, u_k, \tilde{y}_k) + \gamma V(\hat{x}_{k+1})$$

$$Q(\hat{x}_k, u_k, \tilde{y}_k) = \hat{x}_k^T Q \hat{x}_k + u_k^T R u_k + \tilde{y}_k^T S \tilde{y}_k + \gamma(\hat{x}_{k+1}^T P \hat{x}_{k+1})$$

$$Q(\hat{x}_k, u_k, \tilde{y}_k) = \begin{bmatrix} \hat{x}_k^T \\ u_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} Q & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & S \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ u_k \\ \tilde{y}_k \end{bmatrix} + \gamma \left[(A\hat{x}_k + Bu_k + L\tilde{y}_k)^T P (A\hat{x}_k + Bu_k + L\tilde{y}_k) \right]$$

$$Q(\hat{x}_k, u_k, \tilde{y}_k) = \begin{bmatrix} \hat{x}_k^T \\ u_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} Q & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & S \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ u_k \\ \tilde{y}_k \end{bmatrix} + \gamma \left[(\hat{x}_k^T A^T + u_k^T B^T + \tilde{y}_k^T L^T) P (A\hat{x}_k + Bu_k + L\tilde{y}_k) \right]$$

$$Q(\hat{x}_k, u_k, \tilde{y}_k) = \begin{bmatrix} \hat{x}_k^T \\ u_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} Q & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & S \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ u_k \\ \tilde{y}_k \end{bmatrix} + \gamma \left[\begin{bmatrix} \hat{x}_k^T \\ u_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} A^T \\ B^T \\ L^T \end{bmatrix} P \begin{bmatrix} A \\ B \\ L \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ u_k \\ \tilde{y}_k \end{bmatrix} \right]$$

$$Q(\hat{x}_k, u_k, \tilde{y}_k) = \begin{bmatrix} \hat{x}_k^T \\ u_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} Q & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & S \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ u_k \\ \tilde{y}_k \end{bmatrix} + \gamma \left[\begin{bmatrix} \hat{x}_k^T \\ u_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} A^T P A & A^T P B & A^T P L \\ B^T P A & B^T P B & B^T P L \\ L^T P A & L^T P B & L^T P L \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ u_k \\ \tilde{y}_k \end{bmatrix} \right]$$

$$Q(\hat{x}_k, u_k, \tilde{y}_k) = \begin{bmatrix} \hat{x}_k^T \\ u_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} Q & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & S \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ u_k \\ \tilde{y}_k \end{bmatrix} +$$

$$\begin{bmatrix} \hat{x}_k^T \\ u_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} \gamma A^T P A & \gamma A^T P B & \gamma A^T P L \\ \gamma B^T P A & \gamma B^T P B & \gamma B^T P L \\ \gamma L^T P A & \gamma L^T P B & \gamma L^T P L \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ u_k \\ \tilde{y}_k \end{bmatrix}$$

$$Q(\hat{x}_k, u_k, \tilde{y}_k) = \begin{bmatrix} \hat{x}_k^T \\ u_k^T \\ \tilde{y}_k^T \end{bmatrix}^T \begin{bmatrix} \gamma A^T P A + Q & \gamma A^T P B & \gamma A^T P L \\ \gamma B^T P A & \gamma B^T P B + R & \gamma B^T P L \\ \gamma L^T P A & \gamma L^T P B & \gamma L^T P L + S \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ u_k \\ \tilde{y}_k \end{bmatrix}. \quad (4.68)$$

sendo a matriz $H_{(\hat{x}_k, u_k, \tilde{y}_k)}$, o core da matriz da equação (4.68),

$$H_{(\hat{x}_k, u_k, \tilde{y}_k)} = \begin{bmatrix} \gamma A^T P A + Q & \gamma A^T P B & \gamma A^T P L \\ \gamma B^T P A & \gamma B^T P B + R & \gamma B^T P L \\ \gamma L^T P A & \gamma L^T P B & \gamma L^T P L + S \end{bmatrix}. \quad (4.69)$$

A partir da equação (4.68), pode-se diferenciar em função de u_k caso se deseje calcular a ação de controle que minimize o custo associado as transições dos estados estimados. Pode-se ainda diferenciar em função $Q(\hat{x}_k, u_k, \tilde{y}_k)$, com relação ao erro de estimação com o intuito de se calcular o ganho L do observador, associado às ponderações dos erros entre às saídas estimadas e mensuradas.

Capítulo 5

Metodologia, Métricas e Heurísticas para Avaliação de Convergência do Algoritmo OPFB

Neste Capítulo, são abordados as estratégias usadas para realizar a avaliação da convergência do algoritmo OPFB, por meio da descrição das métricas usadas para avaliar o comportamento de convergência do algoritmo ao longo do processo iterativo, assim como heurísticas para a sintonia dos algoritmos usados para a geração dos dados, para a obtenção de dados mais confiáveis, por parte do modelo de geração dos dados. Para o processo de sintonia dos parâmetros dos algoritmos é proposto uma heurística para a sintonia, baseadas na geração de superfícies de erros, com o propósito de gerar uma visualização da influência dos parâmetros no desempenho global do algoritmo, facilitando assim o processo de sintonia paramétrica, minimizando o trabalho do projetista.

Inicialmente são definidos os critérios usados para avaliar a convergência do algoritmo, assim como as métricas usadas em função de uma norma. Posteriormente é proposto uma heurística para a seleção dos parâmetros por meio da criação de superfícies de erros.

5.1 Análise de Convergência

Na metodologia do algoritmo Output Feedback (OPFB), apresentada por [Lewis e Vamvoudakis \[2011\]](#), tem-se por base a construção de uma matriz $\bar{P}$, que irá fornecer a partir dos seus coeficientes, parâmetros que serão usados para a definição da próxima ação de controle, tais parâmetros são: p_0 , p_u , e $p_y = [p_{y1} \ p_{y2}]$. Esses parâmetros são de extrema importância,

pois, ao longo do processo de aprendizagem pela aplicação dos algoritmos PI ou VI, pois, estão associados ao controlador, atuando no sistema a fim de estabilizá-lo. A estimação da matriz $\bar{P}$, pode ser feita por qualquer algoritmo de estimação de parâmetros tais como: LMS, RLS, Batch LS, Filtro de Kalman, Redes Neurais.

Para comparar o processo de convergência foi gerada uma matriz $\bar{P}$ com base no modelo usado para gerar os dados e assim ter uma matriz $\bar{P}$ de referencia para avaliar se o algoritmo conseguia aprender a matriz e com isso convergir para os valores corretos, podendo assim ser computado um erro entre os coeficientes das matrizes estimadas e pela verdadeira gerada pelo modelo. A seguinte função de acumula o erro total dos coeficientes e é definida por:

$$E_n^i = \sum_{j=1}^h |a_j - b_j| \quad (5.1)$$

sendo,

- E_n^i , erro total composto da soma dos erros dos h parâmetros, estimados na simulação i , após n iterações;
- $h = [(m+p)N][(m+p)(N+1)]/2$, referente a quantidade mínima de termos independentes referentes ao núcleo da matriz $\bar{P}$. Sendo m o número de entradas e p número de saídas e N o horizonte de tempo;
- a_j , valor da estimação do parâmetro j após n iterações;
- b_j , valor desejado do parâmetro j após n iterações;

Outra métrica definida é a variância total do erro de estimação de cada um dos parâmetros estimados ao longo do processo iterativo, sendo a variância total do erro de cada simulação, definida por:

$$S_i^2(E_n^i) = \sum_{j=1}^h \sum_{k=1}^n \frac{|e_{j,k} - \bar{e}_j|^2}{n}, k = 1, 2, \dots, n. \quad (5.2)$$

sendo,

- k , o número de iterações;
- $e_{j,k}$, é o erro do parâmetro j na iteração k ;
- $\bar{e}_j$, a média do erro da estimação do parâmetro j após n iterações;
- $S_i^2(E_n^i)$, sendo a soma das variâncias dos erros dos parâmetros estimados na simulação i após n iterações do algoritmo.

O objetivo da métrica de variância total, está relacionado ao fato de se possuir uma métrica para comprar simulações cujos erros absolutos foram semelhantes, sendo que para esses erros seria melhor escolher o conjunto de parâmetros que gerou uma menor dispersão, consequentemente convergindo mais rapidamente. Essa métrica de variância total só é possível de se aplicar para comparar simulações, caso os valores médios dos erros sejam equiparados, caso contrário não faz sentido comprar as variâncias totais. Para os casos em que as médias sejam relativamente diferentes pode-se utilizar a variância normalizada, com por exemplo o coeficiente de variação, que é dado pela razão entre o desvio padrão dos erros e sua média e com isso chegar em uma medida de dispersão comparável entre as medidas.

Com essas duas métricas para avaliar o comportamento de convergência do algoritmo e a qualidade da solução, pode-se então definir algumas heurísticas com o propósito de guiar o processo de busca da solução na presença de ruído. É possível também avaliar as amplitudes e testar tipos de ruídos diferentes no processo de geração dos dados com a utilização de geradores de números aleatórios.

Percebe-se que em plantas reais o desconhecimento da função que gera o ruído implica em dificuldades a serem superadas pelos métodos de controle, uma vez que a incerteza está presente em muitas situações. O ideal é que no processo de simulação obtenha-se o controle independente de características das perturbações, assim é bastante usual realizar o estudo de simulações de *Monte Carlo*, para avaliar o comportamento dos algoritmos a diferentes sequências de ruídos.

5.2 Heurística para Geração das Superfícies e para Seleção dos Parâmetros na Sintonia do Algoritmo OPFB

Nesta seção apresenta-se a heurística usada para realizar a seleção dos parâmetros e suas etapas, contendo os detalhes para a geração dos dados por parte de um modelo, assim como para as superfícies de erros, visando realizar a sintonia do algoritmo com relação as condições iniciais dos parâmetros associados a um determinado algoritmo.

Foram usadas as seguintes etapas para se observar o comportamento do algoritmo frente as variações dos parâmetros iniciais, em busca de melhores soluções e para o entendimento de sua influência, com relação ao desempenho do algoritmo testado e sua convergência.

O processo de aplicação da heurística inicia-se com a geração de uma superfície do erro absoluto conforme Eq.(5.1), com o propósito de identificar a região onde se encontra a melhor solução para uma dada configuração e a partir desta, ou das regiões com melhores soluções, realiza-se a construção de outras superfícies, visando diminuir o espaço de busca. Repete-se esta operação até que se encontre uma configuração que gere o menor erro absoluto total Eq.(5.1). O uso da variância Eq.(5.2), é feito somente quando existe necessidade de escolher parâmetros cujas simulações retornaram erros equiparados.

A seguir etapas com a heurística a serem desenvolvidas ao longo do processo de escolha dos parâmetros são apresentadas.

- Defini-se a configuração dos parâmetros iniciais, com limites de variação com maior amplitude possível de acordo com a experiência do especialista, com relação aos parâmetros das condições iniciais;
- Realizam-se diversas simulações e seleciona-se a semente geradora de números aleatórios que gerou a melhor solução, menor erro;
- Gera-se uma figura com o erro médio quadrático para observar a influência do ruído ao longo das simulações;
- Gera-se a superfície do erro absoluto, com as variações dos parâmetros de interesse, para a determinação da região onde se encontra os valores das condições iniciais dos parâmetros que retornam o menor erro;
- Uma vez selecionados os parâmetros iniciais realiza-se a construção de outra superfície, agora em torno da melhor solução encontrada inicialmente, só que com intervalo menor de variação dos parâmetros a serem configurados;
- Por fim avalia-se se o erro é satisfatório, caso contrário verifica-se se o número de iterações foi suficiente para a convergência do algoritmo, caso contrário realiza-se o aumento do número de iterações e simula-se novamente.

Após a supervisão da superfície pode-se então identificar qual foi o melhor fator de desconto e condições iniciais dos parâmetros que produzem o menor erro absoluto total.

5.3 Gerador de Política

Para a inicialização de um dos algoritmos, no caso o de iteração de política PI, é necessário que a política inicial seja admissível e estabilizadora, o mesmo não é necessário para o algoritmo de iteração de valor VI. Para o caso do algoritmo PI é realizado o seguinte procedimento.

Algumas etapas são realizadas para o cálculo e geração da matriz P , de modo que a partir da mesma seja possível realizar a estabilização do sistema. A Figura 6 apresenta essa heurística. As etapas podem ser definidas como: a inicialização das variáveis Q , R , P e n , sendo que as variáveis podem ser aleatórias inicialmente, considerando somente que as matrizes Q e R , sejam simétricas e semi-definida positiva e positiva definida, respectivamente. A matriz P pode ser aleatória, pois a mesma que sofrerá modificações posteriormente.

Após a geração da matriz P aleatória, é realizado um procedimento para modificação da matriz em uma matriz simétrica e posteriormente ao resultado outra operação para tornar a

matriz definida positiva, visto que é condição necessária que a matriz P seja simétrica e definida positiva. Posteriormente é realizada a avaliação se a política gerada pela matriz P produz um ganho que estabiliza o sistema em malha fechada.

Para a avaliação de estabilidade em malha fechada, é realizado o cálculo do ganho do regulador linear quadrático discreto (DLQR) e posteriormente calculados os autovalores do sistema realimentado, verificando-se se a parte real se encontra dentro do círculo unitário, tanto para a matriz P gerada, quanto para o sistema realimentado $A-BK$. Se for verificado que a matriz P gerada satisfaz as condições de simetria, positividade e estabilização, então tem-se uma matriz P estabilizadora, caso contrário gera-se outra matriz P aleatória e retorna-se para a etapa de simetria, como é mostrado na Figura 6.

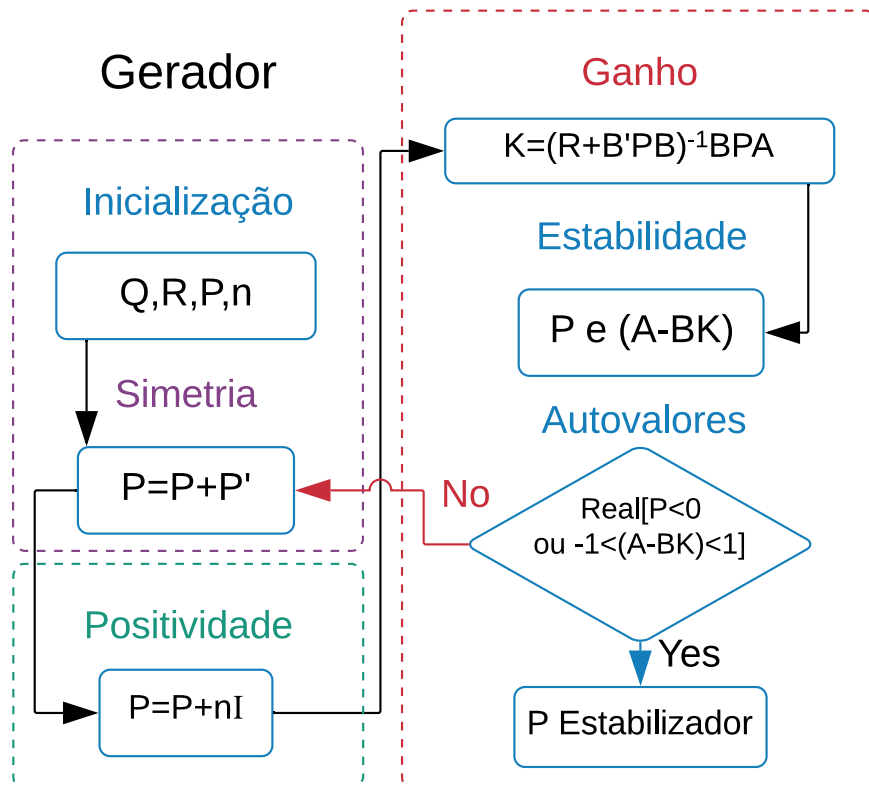


Figura 6 – Gerador de Política Admissível

Fonte: Produzido pelos Autores.

5.4 Heurística para Seleção dos Parâmetros

A heurística geral usada para a seleção dos vários parâmetros usado no algoritmo é apresentada na Figura 7. A estrutura geral do algoritmo é dividida em algumas etapas, sendo elas a inicialização, geração dados/método alimentado por dados, sintonia/variação paramétrica e superfícies/seleção.

Na etapa de inicialização, as variáveis são inicializadas, sendo calculado a matriz P estabilizadora, fornecida pelo gerador definido anteriormente na Figura 6. Posteriormente é realizada a geração dos dados na etapa que irá fornecer os dados para o método não baseado em modelo, nesse caso o OPFB. Nesta etapa que ocorre o processo de aprendizagem e determinação da ação de controle futura, por meio da aplicação da algoritmo de aprendizagem por reforço e uso do RLS , como partes integrantes do $OPFB$, para a estimação da matriz $\bar{P}$ na determinação da ação de controle.

Após a etapa de aplicação do método alimentado por dados obtêm-se ao final o sinal de erro que será usado pelas etapas de geração das superfícies e na seleção dos parâmetros, para seguir para etapa de sintonia e variação dos parâmetros. A etapa da gerações das superfícies de erro e seleção dos parâmetros que melhor sintonizam a metodologia é repetido até que um erro predefinido seja atingido ou até que um número de iterações se alcançado.

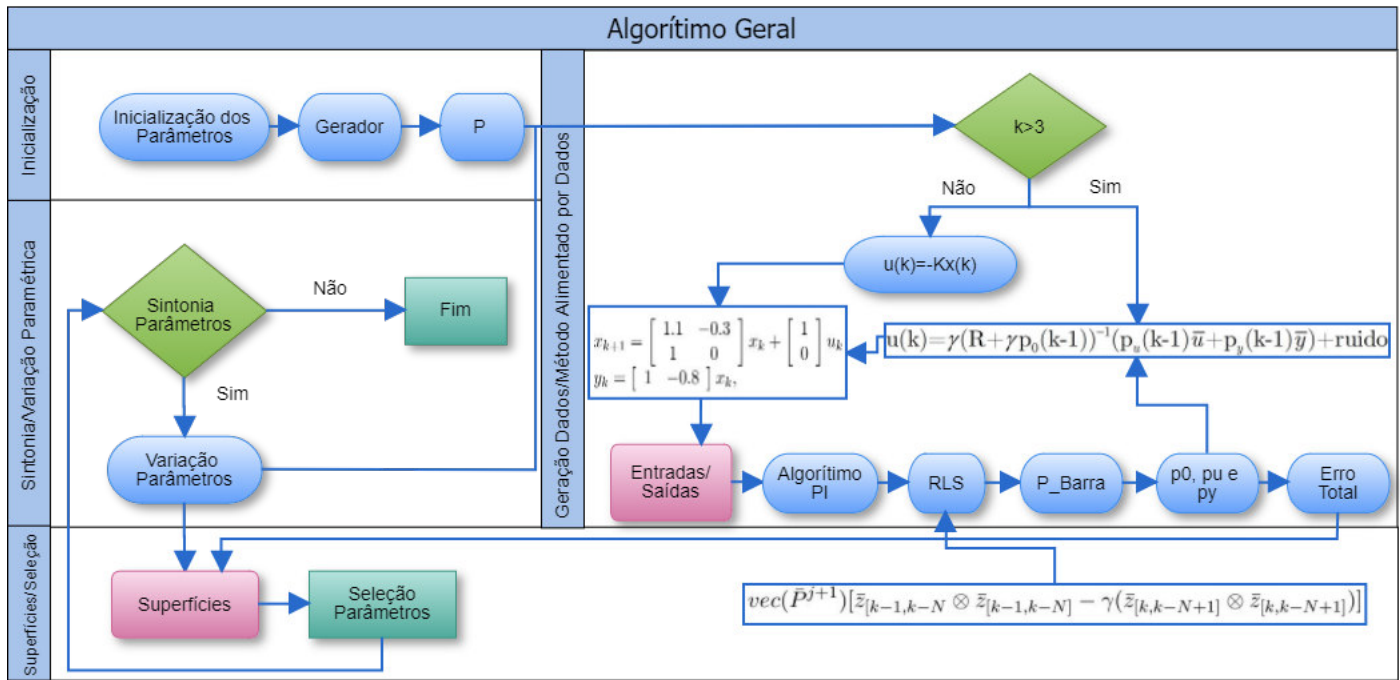


Figura 7 – Heurística Geral para Sintonia

Fonte: Produzido pelos Autores.

5.5 Implementação, Ruído e Fator de Desconto do Algoritmo OPFB

Nesta seção são apresentados os aspectos relacionados a influência do fator de desconto, com o propósito de atenuar os efeitos indesejados de ruídos, que possam convergir para soluções polarizadas, [AL-TAMIMI et al., 2008].

O algoritmo PI apresentado pode ser resolvido online por técnicas padrões como LS ou RLS, por exemplo. Em [BRADTKE et al., 1994] e [SILVA; NETO, 2020], foi usado o RLS para Q -learning para resolver a Eq.(3.79), usando a seguinte forma.

$$vec(\bar{P}^{j+1})[\bar{z}_{[k-1,k-N]} \otimes \bar{z}_{[k-1,k-N]} - \bar{z}_{[k,k-N+1]} \otimes \bar{z}_{[k,k-N+1]}] = y_k^T Q y + (u_k^j)^T R u_k^j, \quad (5.3)$$

sendo $\otimes$, o produto de Kronecker e vec , sendo o operador coluna [BREWER, 1978]. Os termos com índices quadráticos no produto de Kronecker são somados. Informações e detalhes da implementação são discutidos em [BREWER, 1978].

Para o caso do algoritmo VI temos a seguinte forma.

$$vec(\bar{P}^{j+1})[\bar{z}_{[k-1,k-N]} \otimes \bar{z}_{[k-1,k-N]}] = y_k^T Q y + (u_k^j)^T R u_k^j + \bar{z}_{[k,k-N+1]}^T \bar{P}^j \bar{z}_{[k,k-N+1]}. \quad (5.4)$$

Para resolver Eq.(5.3), é necessário que o vetor $[\bar{z}_{[k-1,k-N]} \otimes \bar{z}_{[k-1,k-N]} - \bar{z}_{[k,k-N+1]} \otimes \bar{z}_{[k,k-N+1]}]$ seja linearmente independente ao longo do intervalo de tempo. Esta propriedade conhecida como excitação persistente, que por padrão, injeta um ruído branco à ação de controle, ou seja, $\hat{u}_k = u_k + d_k$, sendo u_k a ação de controle calculada pelo algoritmo PI ou VI e d_k sendo o ruído branco. Do mesmo modo para solucionar a Eq.(5.4), o termo $\bar{z}_{[k-1,k-N]} \otimes \bar{z}_{[k-1,k-N]}$, necessita ser linearmente independente.

É sabido que a inclusão de ruído pode acarretar polarização na saída e com isso erro na identificação da dinâmica do sistema. Em [LEWIS; VAMVOUDAKIS, 2011] é discutido o efeito de polarização para a equação de Bellman, além de avaliar o efeito da inclusão de um fator de desconto para amenizar o efeito indesejado do ruído adicionado.

Com a inclusão do fator de desconto $\gamma < 1$, tem-se a versão modificada do algoritmo PI,

$$vec(\bar{P}^{j+1})[\bar{z}_{[k-1,k-N]} \otimes \bar{z}_{[k-1,k-N]} - \gamma(\bar{z}_{[k,k-N+1]} \otimes \bar{z}_{[k,k-N+1]})] = y_k^T Q y + (u_k^j)^T R u_k^j, \quad (5.5)$$

e para o algoritmo VI,

$$vec(\bar{P}^{j+1})[\bar{z}_{[k-1,k-N]} \otimes \bar{z}_{[k-1,k-N]}] = y_k^T Q y + (u_k^j)^T R u_k^j + \gamma(\bar{z}_{[k,k-N+1]}^T \bar{P}^j \bar{z}_{[k,k-N+1]}). \quad (5.6)$$

Para fins de implementação deste método leva-se em consideração a complexidade computacional relacionadas a ordem do sistema dinâmico e da matriz $\bar{P}$, pois, está será estimada pelo RLS e com isso temos mais um laço de operações para serem computados a cada iteração do algoritmo OPFB. A ordem da matriz $\bar{P}$ possui dimensão $\bar{P} \in \mathbb{R}^{(m+p)N \times (m+p)N}$, ou seja, depende do horizonte finito escolhido N , da quantidade de entradas m e saídas p . A medida que a ordem do sistema se eleva, os valores de N aumentam e para sistemas do tipo MISO (multiple-input, single-output), SIMO (single-input, multiple-output) ou MIMO (multiple-input/output), a complexidade computacional se elevará, com o aumento do número de entradas (m) e/ou número de saídas (p). Um resultado direto é que o número de coeficientes da matriz $\bar{P}$ cresce não linearmente com relação aos parâmetros N , m e p que definem a dimensão de $\bar{P}$. Deve-se também levar em consideração o número de operações necessárias para que o RLS efetue a estimação dos parâmetros de $\bar{P}$.

Capítulo 6

Análises, Simulações Computacionais e Sintonia de Observadores Alimentados por Dados

São apresentados neste capítulo as simulações realizadas, demonstrando o comportamento da metodologia proposta para estimação de estados, baseada em programação dinâmica aproximada e aprendizagem por reforço com realimentação de saída. São realizadas análises sobre a sensibilidade dos parâmetros, com relação a convergência e robustez ao longo do processo de sintonia dos parâmetros, pela aplicação das heurísticas, desde a geração dos dados como das superfícies de erros para seleção dos parâmetros.

Como já mencionado, o algoritmo de Iteração de Política (*PI*) necessita de uma política inicial estabilizadora, para a garantia de convergência do controlador. Para isso foi desenvolvido um simulador que pudesse gerar uma matriz P aleatória, simétrica e definida positiva, para então se calcular o ganho do sistema de malha fechada e assim avaliar se os autovalores do sistema de malha fechada estariam no interior do círculo unitário.

Nesta seção abordaremos os aspectos relacionados a implementação do algoritmos *OPFB*, com Iteração de Política e Iteração de Valor. Os algoritmos *PI* e *VI* foram usados em duas plantas de dois sistemas dinâmicos, um estável e um instável, também usados no trabalho de [Lewis e Vamvoudakis \[2011\]](#). A escolha do sistema instável está relacionado a avaliação de que a metodologia proposta e aplicada, além de conseguir estabilizar também consegue estimar adequadamente os estados. Neste trabalho são apresentados as simulações e experimentos associados a Iteração de Política para aplicação da heurística para sintonia dos

parâmetros. Em outros experimentos para estimação de estados é usado o algoritmo de Iteração de Valor (VI).

A apresentação dos experimentos computacionais é dividida em duas partes, uma associada a implementação do algoritmo PI com sua sintonia e a outra relacionada a resolução do sistema de equações não lineares, que como resultado fornece os coeficientes das matrizes M_u e M_y , necessários para a estimação dos estados.

Foram realizados experimentos computacionais e simulações para identificar a sensibilidade paramétrica dos algoritmos propostos, com relação às condições iniciais, variação do ruído branco utilizado, assim como o fator de desconto usado para atenuar a influência do ruído.

6.1 Sistema Simulado, Estados Reais e Estimados

Para termos como parâmetro para avaliar as formulações propostas foram gerados dados do comportamento de um sistema dinâmico, juntamente com os estados por um modelo sintético, para serem comparados com os estados estimados pela equação (4.26). Primeiramente, fixa-se os parâmetros iniciais e realizam-se as simulações, somente nos parâmetros que se deseja avaliar a influência com relação ao erro absoluto total, Eq.(5.1). Os parâmetros fixados foram: as matrizes de ponderação Q e R iguais a matriz identidade, com dimensões adequadas e estados iniciais iguais a 1. Para a estimação dos coeficientes de $\bar{P}$, foi utilizado o estimador dos mínimos quadrados recursivo (RLS), com fator de esquecimento igual a 1 e matriz de covariância inicial igual a matriz identidade, mais detalhes desta implementação são apresentadas em [BRADTKE et al., 1994].

Para a geração das trajetórias contendo a evolução dos estados, foi utilizado o seguinte sistema dinâmico discreto instável (6.1), ou seja, os autovalores da matriz A estão fora do círculo unitário. O propósito desta escolha está relacionado com o mesmo sistema usado por Lewis e Vamvoudakis [2011], pois, o objetivo é ter um sistema já testado por outros trabalhos, de modo a demonstrar que a metodologia consegue estabilizar também o sistema testado.

$$\begin{aligned} x_{k+1} &= \begin{bmatrix} 1.8 & -0.7700 \\ 1 & 0 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_k, \\ y_k &= \begin{bmatrix} 1 & -0.5 \end{bmatrix} x_k. \end{aligned} \quad (6.1)$$

6.1.1 Seleção do Fator de Desconto e Variância do Ruído

Para avaliar a sensibilidade do algoritmo com relação ao fator de desconto (γ) e variância do ruído Gaussiano, adicionado a política de controle u , foram realizadas simulações, variando cada um dos parâmetros e observando o comportamento do algoritmo com relação à convergência e métrica de performance, avaliando-se o erro absoluto total, Eq.(5.1).

Simulações foram realizadas para se avaliar o erro absoluto total e a influência que o ruído impacta no algoritmo baseado em ADP apresentado na Seção 3.4. Os ruídos usados nas simulações possuíam média fixa em 0 e variância com amplitude entre 0 e 3, com intervalos de variância de 0.002. Os resultados de cada uma das simulações contendo as variações das variâncias do sinal Gaussiano, com relação ao erro absoluto total é apresentado na Figura 8.

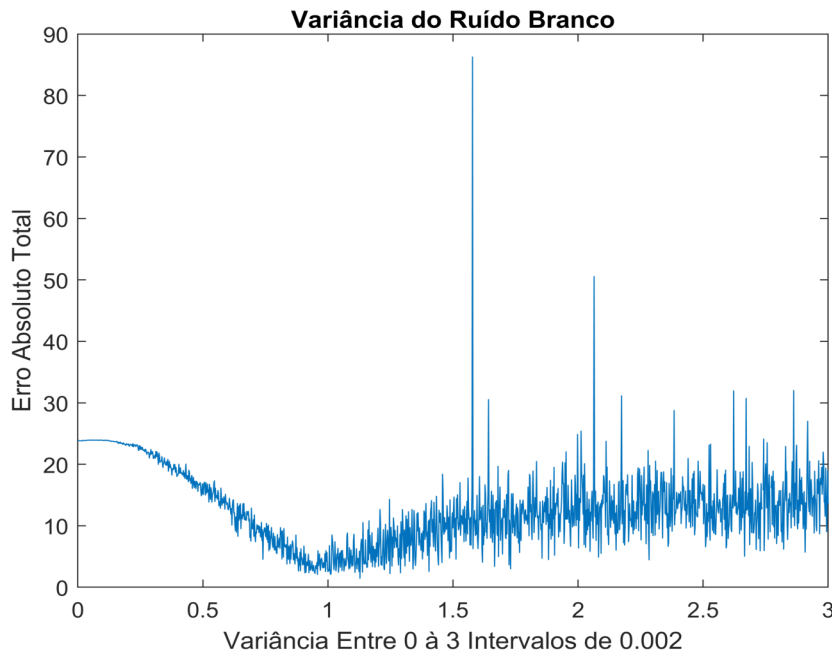


Figura 8 – Erro Total para variância do ruído entre 0 e 3, com intervalos de 0.002

Fonte: Produzido pelos Autores.

A variância que produz menor erro total está centrada em 1. Para a seleção do fator de desconto, são realizadas simulações com o ruído Gaussiano e demais parâmetros fixos.

Outro parâmetro que foi avaliado e que está relacionado a mitigação dos efeitos deletérios do ruído ao desempenho do sistema dinâmico é o fator de desconto, usado pelos algoritmos de avaliação de política ou iteração de valor apresentados na Seção 3.4. Para avaliar a sensibilidade do algoritmo com relação ao fator de desconto, o algoritmo foi simulado com valores entre 0.01 e 1 com intervalos de 0.01 para o fator de desconto. A Figura 9 apresenta o erro absoluto total Eq.(5.1), que é definido como sendo a soma dos erros entre os coeficientes das matrizes $\bar{P}$ verdadeira Eq.(3.76) e a estimada pelo RLS.

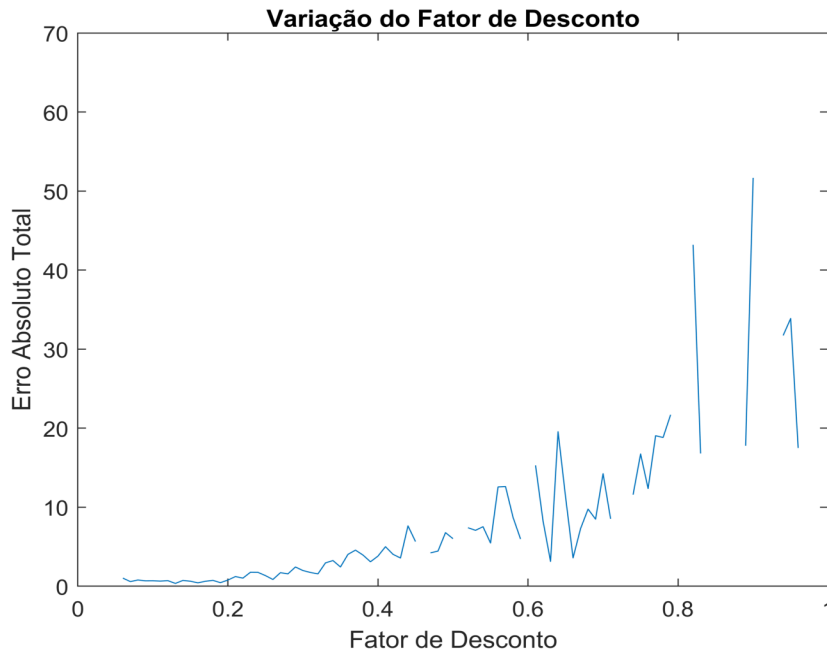


Figura 9 – Erro para variação do fator de desconto entre $\gamma = 0.01$ e 1, com intervalos de 0.01.

Fonte: Produzido pelos Autores.

O erro tende a ser menor à medida que o fator de desconto assume menores valores. Percebe-se que, à medida que o fator de desconto cresce, o erro também aumenta, sendo que em alguns casos o algoritmo não retorna solução, ou seja, diverge.

Vários aspectos podem resultar em divergência, como: o aumento do número de condicionamento ao longo do processo iterativo, o que pode indicar que existem problemas de arredondamentos. Outra possível causa estaria relacionado a divergência do RLS ou mesmo problemas com relação as condições iniciais.

6.1.2 Estimação de Estados

Inicialmente são geradas as trajetórias dos estados x_1 e x_2 , simulados pelo sistema (6.2), para serem comparados com os estados estimados x_{1est} e x_{2est} , produzidos pela equação (4.26). As saídas y , simuladas pelo modelo do sistema (6.2), e as saídas estimadas y_{est} , apresentadas na Figura 10.

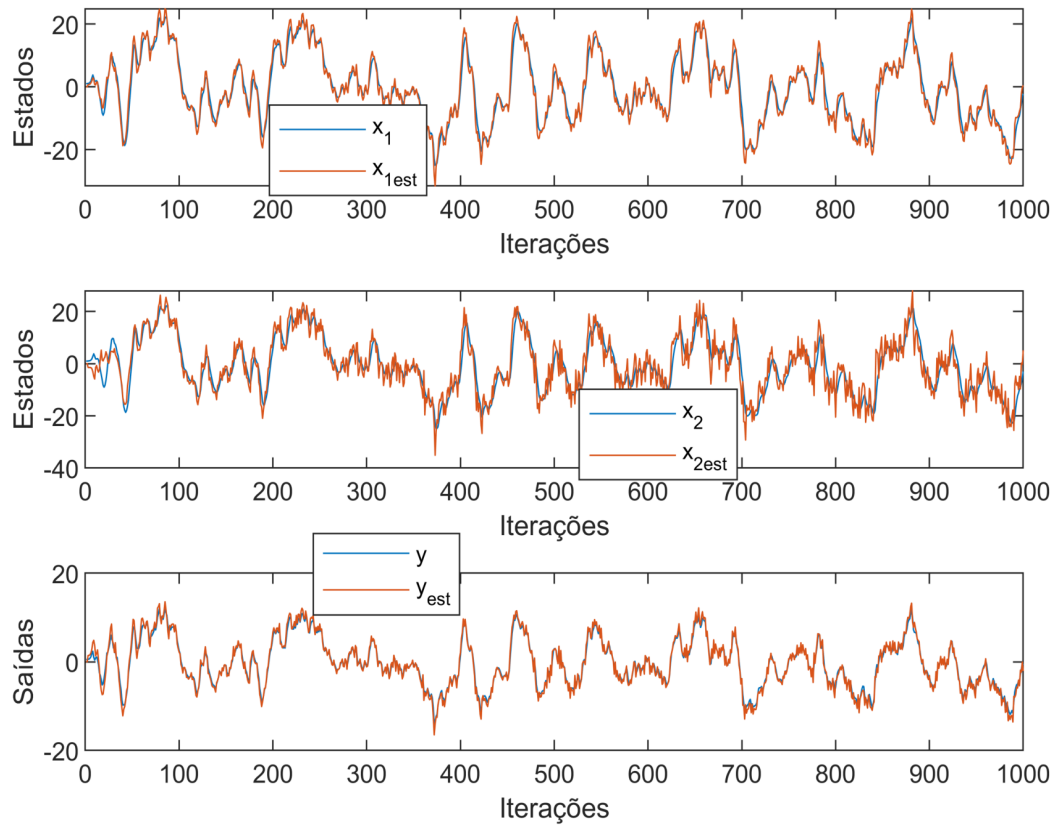


Figura 10 – Estados Estimados x_{1est} e x_{2est} e saída estimada y_{est} , usando aproximação proposta.

Fonte: Produzido pelos Autores.

O estado estimado x_{1est} consegue rastrear o estado simulado de modo mais preciso, enquanto que o estado x_{2est} é mais oscilatório em torno do estado simulado.

A fim de melhor observar o comportamento dos estados estimados e da saída estimada, plotou-se o erro de estimação entre os estados simulados e estimados, sendo o eixo das iterações representados em escala logarítmica na Figura 11. O erro da estimação do estado x_1 é menor que o erro de estimação do estado x_2 , embora esses erros se cancelem, se observados o erro de estimação da saída, que oscila com valores médios abaixo de 2, como é possível verificar-se na Figura 11. Uma vez que o método proposto não é baseado em modelo, não faz muito sentido comparar com filtros tradicionais como o filtro de Kalman ou outros observadores baseados em modelos. Uma possível forma de realizar comparações, poderia ser por meio da complexidade computacional dos algoritmos sob certas condições, visto que, o pré ou pós processamento ou filtragem e mesmo determinado setup de cada método, pode implicar em grandes variações nas quantidades de operações e por isso se torna mais complexo tal comparação e muitas das vezes inviável, visto que é desejado uma comparação sem vieses.

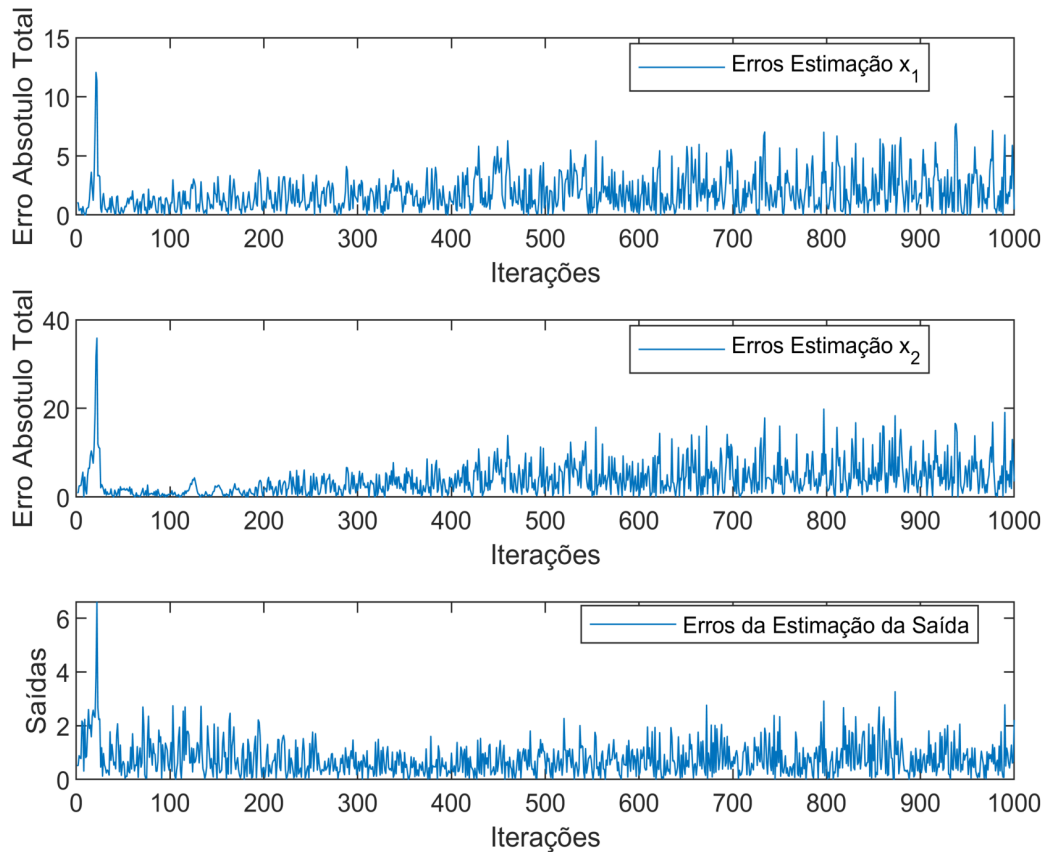


Figura 11 – Erro Estimação Estados OPFB

Fonte: Produzido pelos Autores.

Para realizar uma melhor sintonia é necessário investigar o comportamento do estimador dos coeficientes da matriz $\bar{P}$, nesse caso o RLS padrão, pois, é possível e conhecido que o RLS é sensível a escolha do fator de esquecimento que nesse caso foi utilizado valor igual a 1. As condições iniciais dos estados, também podem influenciar o desempenho do RLS.

A aproximação utilizada na formulação alimentada por dados é vantajosa, uma vez que não é necessário a realização da modelagem matemática do sistema dinâmico para a estimação de estados. Outra vantagem está relacionada ao fato de não ser necessário solucionar o sistema de equações, necessários para a o fornecimento das matrizes M_u e M_y .

Outra vantagem está associada a questão de que os dados mensurados contém informações do fenômeno real com variáveis que nem sempre são inseridas no processo de modelagem, uma vez que não existe conhecimento das mesmas ou mesmo por não serem observáveis.

Como desvantagem, pode-se ter informação ruidosa por perturbações e ruídos de sensores, que podem dificultar a sintonia dos métodos alimentados puramente por dados. Tais problemas podem ser atenuados pela aplicação de filtros, tendo em mente o custo computacional desta escolha. Outro problema está relacionado ao problema de sintonia, pois, a medida que ocorrem variações na dinâmica do sistema, uma nova sintonia faz-se necessária, justificando a formulação

de métodos e ferramentas de análise, como as heurísticas aqui desenvolvidas, para sintonia dos parâmetros, visando a melhoria do desempenho do algoritmo.

6.2 Heurística para Sintonia, Análise de Sensibilidade e Solvabilidade

A heurística para a sintonia dos parâmetros, com as configurações e os experimentos simulados com algoritmo *PI* é apresentado nesta seção, com o propósito de analisar a sensibilidade paramétrica, robustez a condições iniciais, convergência e desempenho do algoritmo, aplicando as heurísticas propostas para a sintonia do algoritmo apresentada nas seções anteriores.

Para as simulações presentes nesta seção foi usado o seguinte sistema dinâmico discreto como gerador dos dados de entrada e saída.

$$\begin{aligned} x_{k+1} &= \begin{bmatrix} 1.1 & -0.3 \\ 1 & 0 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_k, \\ y_k &= \begin{bmatrix} 1 & -0.8 \end{bmatrix} x_k. \end{aligned} \quad (6.2)$$

Inicialmente avalia-se o resultado de 1000 simulações e seleciona-se o menor erro absoluto total para os 10 coeficientes estimados com o RLS, para fator de esquecimento $\lambda = 1$ fator de desconto $\gamma = 0,2$, ruído branco com variância de 1.2 e média 0 e coeficiente x^k que multiplica da matriz identidade associado aos estados iniciais x_1, x_2 . O coeficiente x^k inicial foi $x^k = 200$, assim os estados iniciais foram $x_1 = 200$ e $x_2 = 200$. Cada simulação possuía 1000 iterações e aplicação do erro de diferença temporal e somente uma iteração para o RLS. É necessário uma quantidade mínima de dados para aplicação do algoritmo OPFB. Um mínimo de três iterações foram necessários para geração dos vetores de entrada e saída. Após a quarta iteração é realizada a etapa de estimação dos coeficientes.

Na Figura 12, o erro médio quadrático para 1000 simulações é apresentado. Sendo para cada simulação calculado a variância do erro absoluto total em 1000 iterações de cada simulação. Esse número de 1000 iterações foi escolhido, pois nas simulações testes foram suficientes para ocorrer convergência dos algoritmos.

Ao avaliar-se as Figuras 12 e 13, observa-se que existe um pico, com valores dos ruídos aplicados que resultaram em um erro fora do padrão, comparados com os demais. Tal variação pode ter ocorrido por valores de números aleatórios muito afastados da média, usados pelo gerador de números aleatórios, usada para gerar o sinal de ruído.

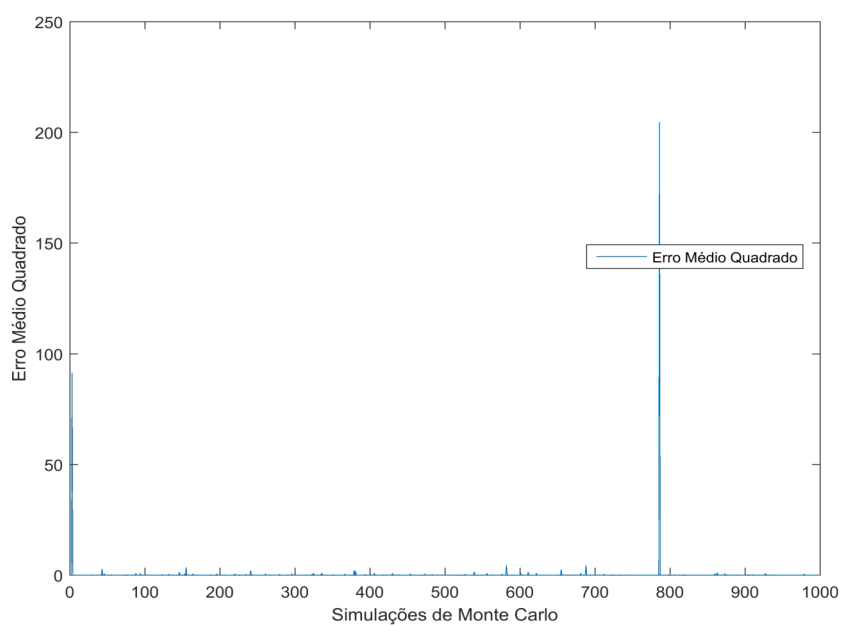


Figura 12 – Erro Médio Quadrático para 1000 Simulações.

Fonte: Produzido pelos Autores.

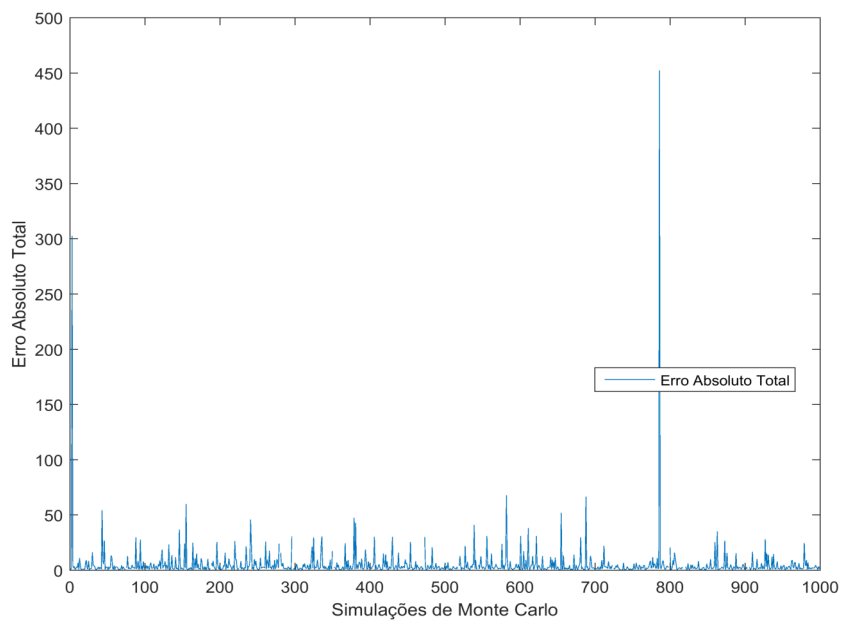


Figura 13 – Erro Absoluto Total para 1000 Simulações.

Fonte: Produzido pelos Autores.

Ao final das 1000 simulações, o menor erro absoluto total atingido foi de 0.4963. As

matrizes com a solução verdadeira e a estimada obtidas foram:

$$\bar{P}_V = \begin{bmatrix} 1.0150 & -0.8440 & 1.1455 & -0.3165 \\ -0.8440 & 0.7918 & -1.0341 & 0.2969 \\ 1.1455 & -1.0341 & 1.3667 & -0.3878 \\ -0.3165 & 0.2969 & -0.3878 & 0.1113 \end{bmatrix} \quad (6.3)$$

$$\bar{P}_E = \begin{bmatrix} 1.0836 & -0.8324 & 1.1263 & -0.3067 \\ -0.8324 & 0.8195 & -0.9302 & 0.2391 \\ 1.1263 & -0.9302 & 1.4073 & -0.3764 \\ -0.3067 & 0.2391 & -0.3764 & 0.1450 \end{bmatrix} \quad (6.4)$$

O erro absoluto total para cada simulação é apresentado na Figura 13, sendo o erro absoluto total, a soma dos valores absolutos dos erros entre os coeficientes da matriz $\bar{P}_V$ na Eq.(6.3) e $\bar{P}_E$ na Eq.(6.4). Na Figura 14, apresenta-se o comportamento de evolução dos erro dos coeficiente, para a simulação com o menor erro dentre as 1000 simulações. O comportamento de convergência dos erros possuem convergências diferentes por conta do processo de estimação de cada um dos coeficientes, que estão sendo estimados pelo RLS. Os valores iniciais com picos mais elevados estão relacionados as condições iniciais do setup, usado para inicialização das variáveis. Detalhes e análises sobre essas questões são realizadas nas seções seguintes.

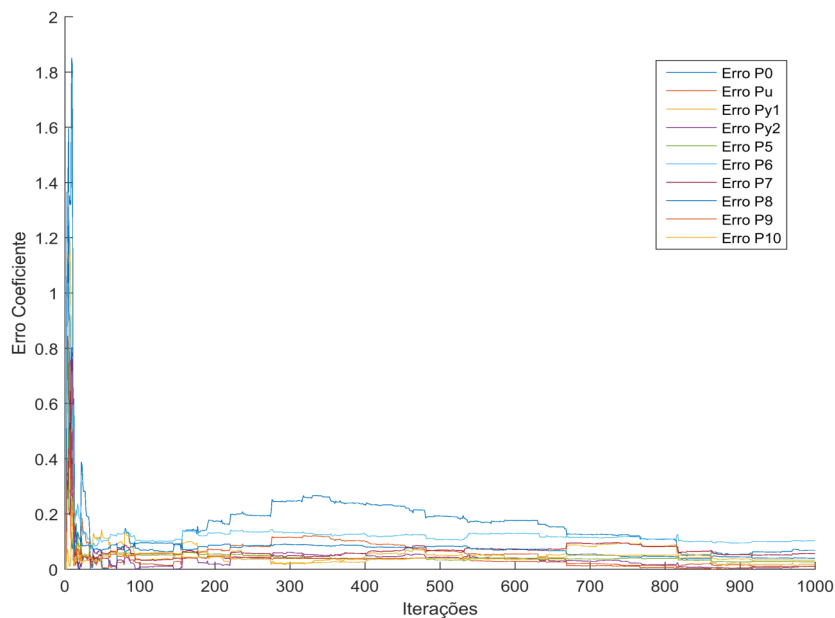


Figura 14 – Erro da estimação dos coeficiente estimados pelo *RLS*.

Fonte: Produzido pelos Autores.

A simulação que gerou o melhor resultado, com a evolução dos coeficientes, ao longo de 1000 iterações é mostrado na Figura 15.

Um objetivo é apresentar a melhoria na sintonia dos parâmetros ao longo da aplicação da heurística proposta, de modo que seja possível identificar a melhoria na convergência dos coeficientes das matriz $\bar{P}_E$, ao longo do processo de sintonia.

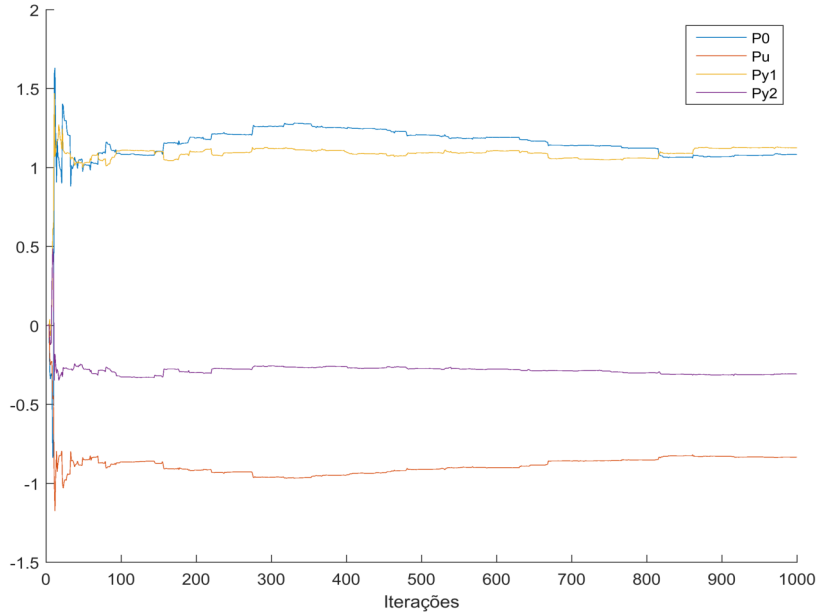


Figura 15 – Coeficientes estimados pelo *RLS*.

Fonte: Produzido pelos Autores.

6.2.1 Análise da Superfície de Erro para Variações dos Parâmetro x_k e $fd(\gamma)$

São discutidos nesta seção, a análise da superfície de erro apresentada na Figura 16. A superfície apresenta o erro absoluto total, com relação a variação do fator de desconto (fd) e da condição inicial do fator multiplicativo x_k , usado para ponderar a condição inicial dos estados utilizados para a geração dos dados de saída.

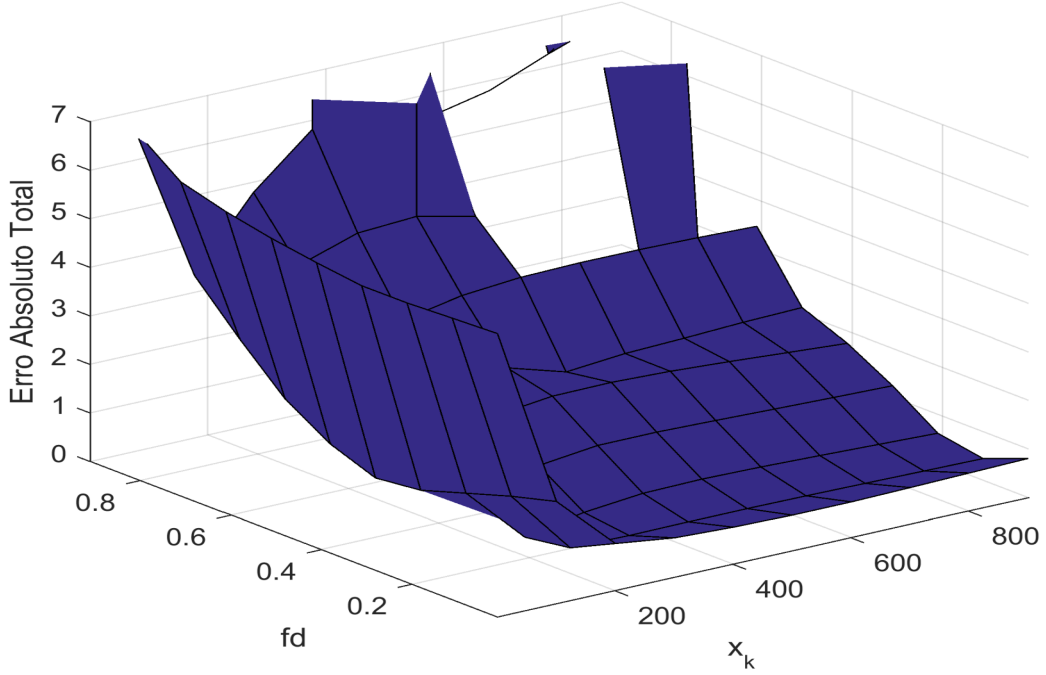


Figura 16 – Superfície de Erro para Variação do Fator de Desconto e Coeficiente x_k , que determina os estados iniciais.

Fonte: Produzido pelos Autores.

A melhor solução na superfície Figura 16 foi 0.3889, para o fator de desconto 0.21 e $x_k = 201$. Foi realizada a variação do fator de desconto em um intervalo de 0.01 até 1 em intervalos de 0.1 e o fator x_k , de 1 até 900, com intervalos de 100 e ruído branco com média 0 e variância de 0.4. As matrizes $\bar{P}_V$ e $\bar{P}_E$ são apresentadas a seguir. Pode-se observar que os coeficientes já estão mais próximos dos verdadeiros valores, se comparado com a matriz inicial.

$$\bar{P}_V = \begin{bmatrix} 1.0150 & -0.8440 & 1.1455 & -0.3165 \\ -0.8440 & 0.7918 & -1.0341 & 0.2969 \\ 1.1455 & -1.0341 & 1.3667 & -0.3878 \\ -0.3165 & 0.2969 & -0.3878 & 0.1113 \end{bmatrix} \quad (6.5)$$

$$\bar{P}_E = \begin{bmatrix} 1.0500 & -0.8559 & 1.1552 & -0.3333 \\ -0.8559 & 0.9419 & -0.9630 & 0.2695 \\ 1.1552 & -0.9630 & 1.4050 & -0.4001 \\ -0.3333 & 0.2695 & -0.4001 & 0.1276 \end{bmatrix} \quad (6.6)$$

Escolhendo agora um intervalo em torno da melhor solução encontrada na superfície da Figura 16 e variando o fator de desconto no intervalo de 0.1 até 1, com intervalo de 0.5 e variando o intervalo de x_k entre 1 e 300, com intervalos de tamanho 10, temos a Figura 17, com uma superfície mais bem definida, podendo identificar melhor quais valores do fator x_k e do fator de desconto, produzem melhores resultados, por meio de uma sintonia mais fina.

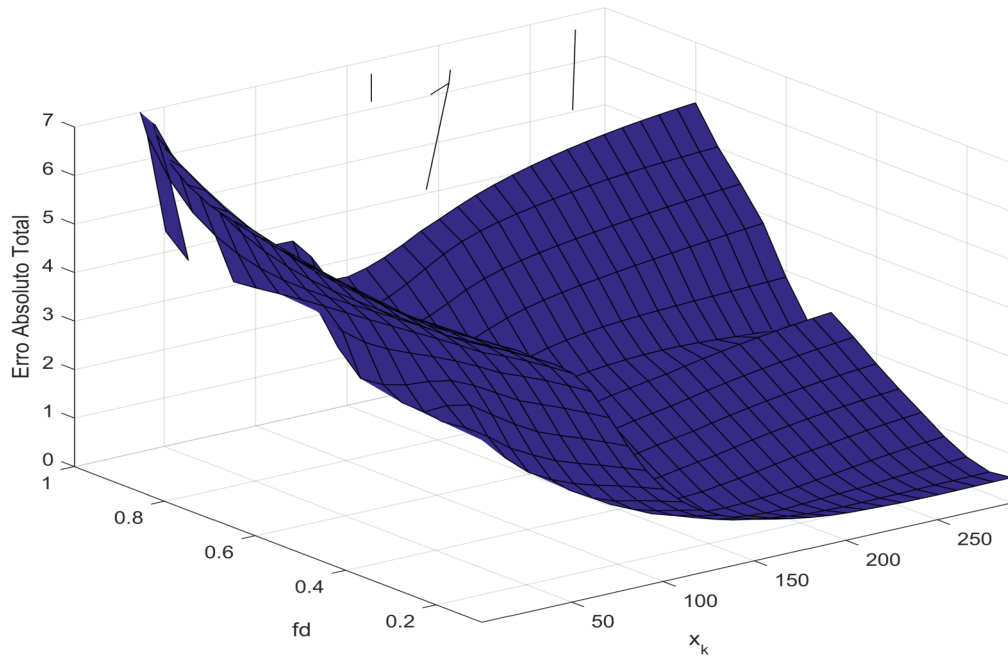


Figura 17 – Superfície de Erro para Variação do Fator de Desconto e parâmetro x_k .

Fonte: Produzido pelos Autores.

A melhor solução encontrada na superfície da Figura 17, foi de 0.3568, para o fator de desconto de 0.20 e fator $x_k = 191$. As matrizes $\bar{P}_V$ e $\bar{P}_E$ obtidas foram:

$$\bar{P}_V = \begin{bmatrix} 1.0150 & -0.8440 & 1.1455 & -0.3165 \\ -0.8440 & 0.7918 & -1.0341 & 0.2969 \\ 1.1455 & -1.0341 & 1.3667 & -0.3878 \\ -0.3165 & 0.2969 & -0.3878 & 0.1113 \end{bmatrix} \quad (6.7)$$

$$\bar{P}_E = \begin{bmatrix} 1.0655 & -0.8444 & 1.1469 & -0.3250 \\ -0.8444 & 0.9152 & -0.9438 & 0.2524 \\ 1.1469 & -0.9438 & 1.3880 & -0.3912 \\ -0.3250 & 0.2524 & -0.39127 & 0.1244 \end{bmatrix} \quad (6.8)$$

Foi realizado uma análise para variações das condições iniciais para o parâmetro x_k para valores entre -1000 e 1000 , com intervalo de tamanho 100, com o objetivo de avaliar a influencia dos estados iniciais para valores negativos e positivos, tal situação é mostrada na Figura 18. Existe um comportamento relativamente de simetria com relação aos estados iniciais nulos. Para estados iniciais iguais, não foram observados variações significativas com relação ao impacto no erro absoluto total.

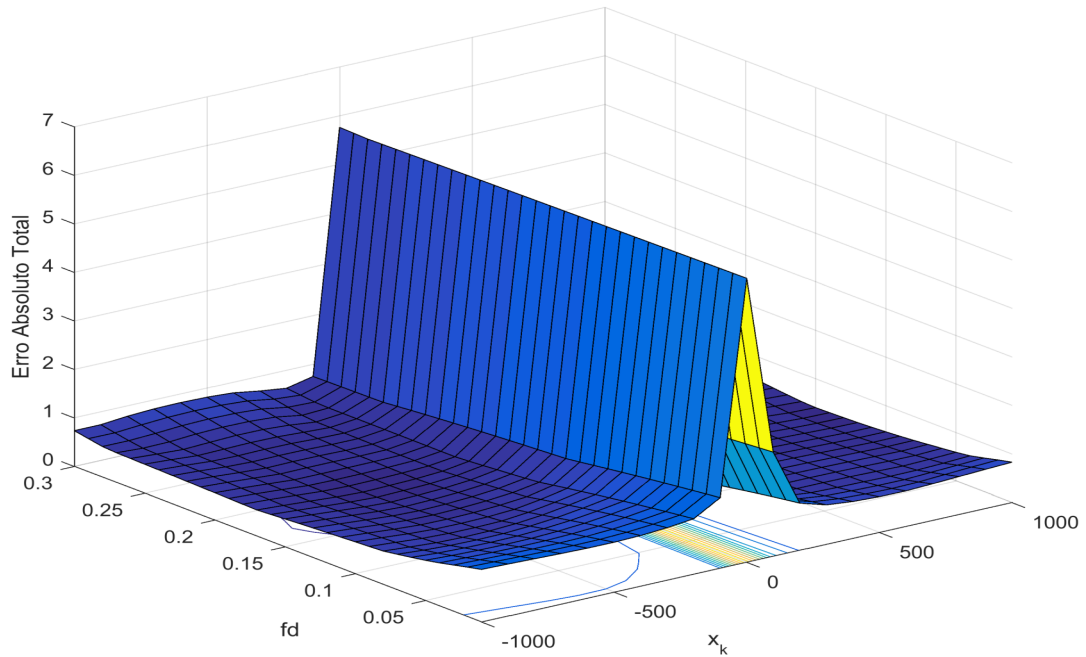


Figura 18 – Superfície de Erro para Variação do Fator de Desconto e Estados Iniciais entre -1000 e 1000 .

Fonte: Produzido pelos Autores.

6.2.2 Análise da Superfície de Erro para variação dos Parâmetro prls_k e Fator de Desconto (fd)

Outro parâmetro analisado foi prls_k , que é um fator de ponderação da matriz de covariância inicial do *RLS*, usado para a estimação dos coeficiente da matriz $\bar{P}$. A superfície de erro para a variação do fator de desconto, com relação a variação do coeficiente prls_k da matriz de covariância inicial, usada para inicializar o *RLS* é apresentado na Figura 19.

Observa-se que o comportamento do erro absoluto total cresce, a medida que o fator de desconto cresce e com isso pode-se assim verificar, que os resultados que trazem um menor erro absoluto total, ocorrem quando o fator de desconto é menor, mas precisamente, o melhor resultado é alcançado quando o fator de desconto é 0.21. Já o parâmetro prls_k , não apresenta tendência clara de influência no erro absoluto total, aparentando ser descorrelacionado com o erro, somente a faixa entre 300 e próximo a 600 que apresentou duas regiões de divergência, não retornando soluções, além de dois picos ao redor da região com fator de desconto entre 0.2 e 0.4. O menor erro encontrado na superfície foi de 0.3637, para fator de desconto igual 0.21 e fator $\text{prls}_k = 1$.

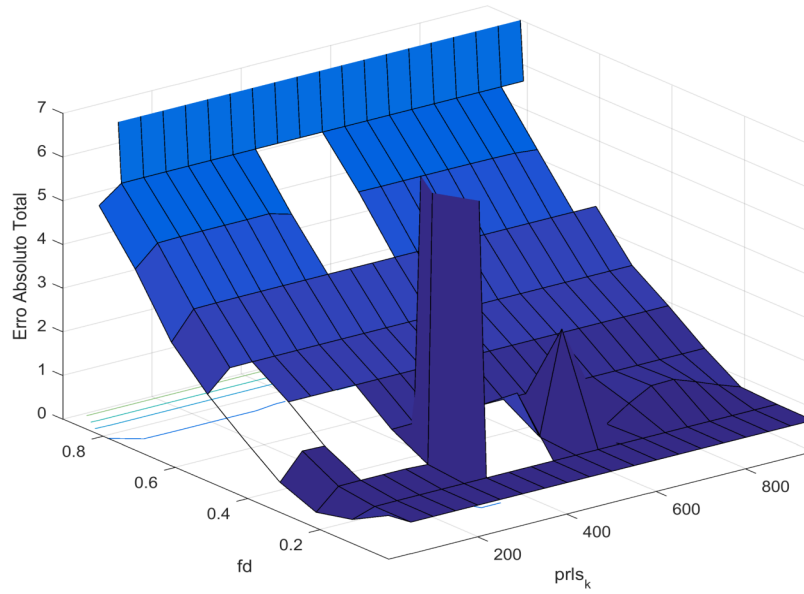


Figura 19 – Superfície de Erro para Variação do Fator de Desconto e prls_k .

Fonte: Produzido pelos Autores.

As matrizes $\bar{P}_V$ e $\bar{P}_E$ obtidas foram:

$$\bar{P}_V = \begin{bmatrix} 1.0150 & -0.8440 & 1.1455 & -0.3165 \\ -0.8440 & 0.7918 & -1.0341 & 0.2969 \\ 1.1455 & -1.0341 & 1.3667 & -0.3878 \\ -0.3165 & 0.2969 & -0.3878 & 0.1113 \end{bmatrix} \quad (6.9)$$

$$\bar{P}_E = \begin{bmatrix} 1.0539 & -0.8476 & 1.1471 & -0.3275 \\ -0.8476 & 0.9275 & -0.9549 & 0.2576 \\ 1.1471 & -0.9549 & 1.3991 & -0.3953 \\ -0.3275 & 0.2576 & -0.3953 & 0.125844 \end{bmatrix} \quad (6.10)$$

6.2.3 Análise da Superfície Erro para Variação dos Parâmetros $\text{fd}(\gamma)$ e Variância do Ruído

Nesta seção é discutido sobre o quão sensível é o algoritmo OPFB, com relação ao erro absoluto total, frente a variações do fator de desconto (γ) e ruídos, com diferentes amplitudes de variâncias.

Avaliou-se o comportamento das superfícies de erros para um ruído branco com média 0 e variância com amplitude entre 0.01 e 10, com intervalos de 0.5 e fator de desconto de 0.01 à 1, com intervalo de 0.01. A Figura 20, mostra o comportamento do erro absoluto total para as variações do ruído e fator de desconto mencionados.

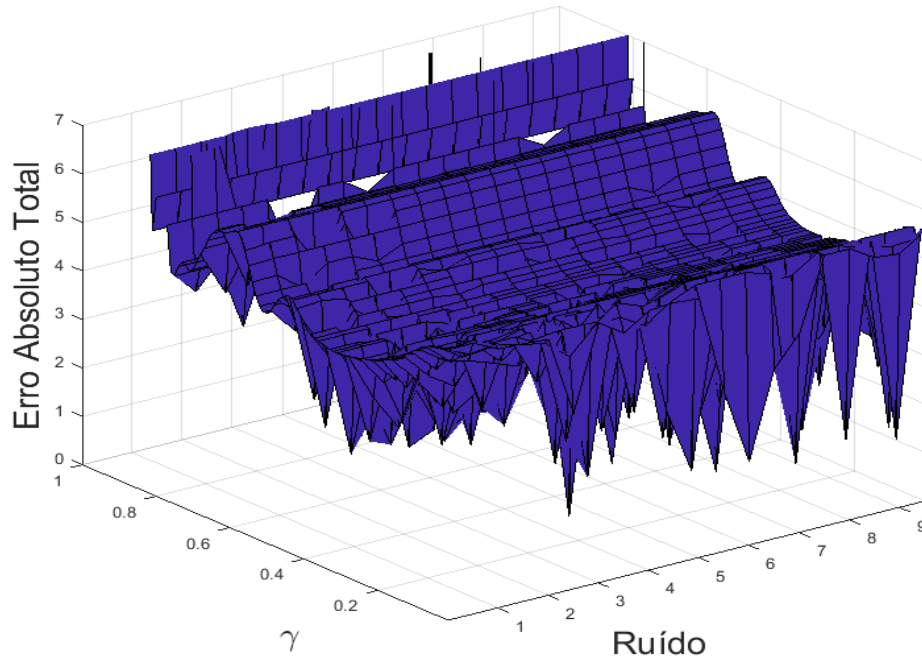


Figura 20 – Superfície de Erro para Variação do Fator de Desconto e Variância do Ruído.

Fonte: Produzido pelos Autores.

Observa-se na Figura 20, que a medida que o fator de desconto aumenta o erro absoluto total também se eleva, indicando que para esta situação é mais indicado selecionar valores menores para o fator de desconto. O menor erro na superfície foi atingido para o fator de desconto igual a $fd = \gamma = 0.09$. Para cada ponto da superfície, foram realizada 5 simulações e escolhida da simulação que produziu o menor erro absoluto total, para este experimento o menor erro obtido foi de 1.1909, esse erro foi atingido para variância de 3.51.

Com relação da influência do ruído, ocorreu uma pequena elevação no erro total a medida que a variância do ruído foi aumentada. O propósito dessas simulações está relacionado ao sinal de ruído gerar uma nova sequência aleatória a cada simulação, sendo gerada por uma distribuição pseudo aleatória, a fim de existir reprodutibilidade do experimento aleatório.

Na Figura 21, o intervalo de variância do ruído foi de 0.1 até 4, com intervalos de tamanho 0.1. O número de simulações foi de 25 para cada ponto da superfície, sendo escolhido o melhor dentre as simulações, com relação ao menor erro absoluto total. Foi definido o intervalo de variação do fator de desconto de 0.01 até 0.3 com intervalos de 0.01, pois menores valores para o fator de desconto retornaram menores erros absolutos, na superfície da Figura 20.

Como para cada simulação variou-se a variância do ruído, assim o erro total não seguirá um padrão suave, gerando assim erros totais muito diferentes. A Figura 21, possui erro absoluto total limitado superiormente e menor que 5. Para avaliar-se melhor essa superfície, deve-se observar a colocação e curvas de nível representadas no plano ($fd \times ruído$), em que os valores

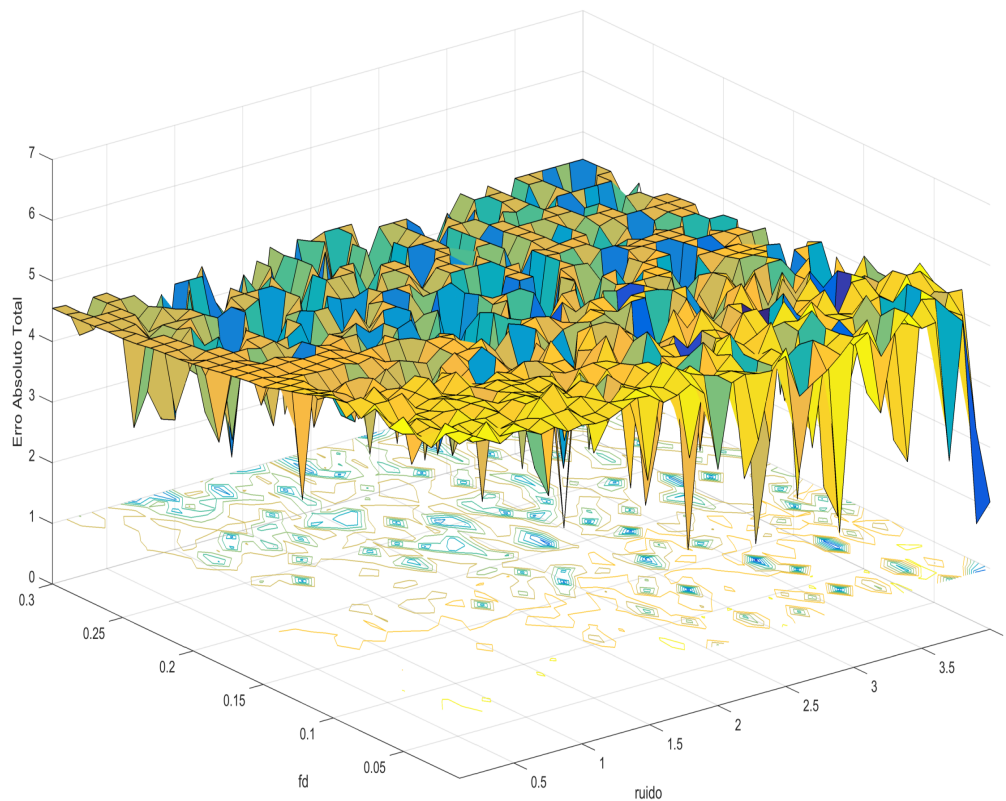


Figura 21 – Superfície de Erro com Variação do Fator de Desconto entre 0.01 e 0.3 e ruído 0.1 até 4.

Fonte: Produzido pelos Autores.

da superfície de erro, são representados pela cor mais escura, no caso esverdeada e azulada, indicando os vales com os menores valores da superfície de erro.

O objetivo da superfície gerada na Figura 21, é de se visualizar a influencia direta do aumento da variância do ruído com relação ao erro absoluto total. Pode-se ainda calcular o coeficiente de correlação de *Pearson* para identificar se existe correlação linear entre o ruído e o erro absoluto total e assim plotar uma outra superfície contendo as correlações das sequências, embora claramente não exista tal correlação linear, por simples inspeção visual.

Restringindo o intervalo de variação do fator de desconto entre 0.01 e 0.35 e ruído com variância entre 0.1 e 3.5, com intervalos de 0.05, observa-se na Figura 22, que assim como na Figura 21, os menores erros se concentram no intervalo de até 0.1 para o fator de desconto e para a variância do ruído em torno de 3. O melhor resultado se obtêm quando a variância do ruído é 2.95, nesta superfície o menor erro foi de 1.1019.

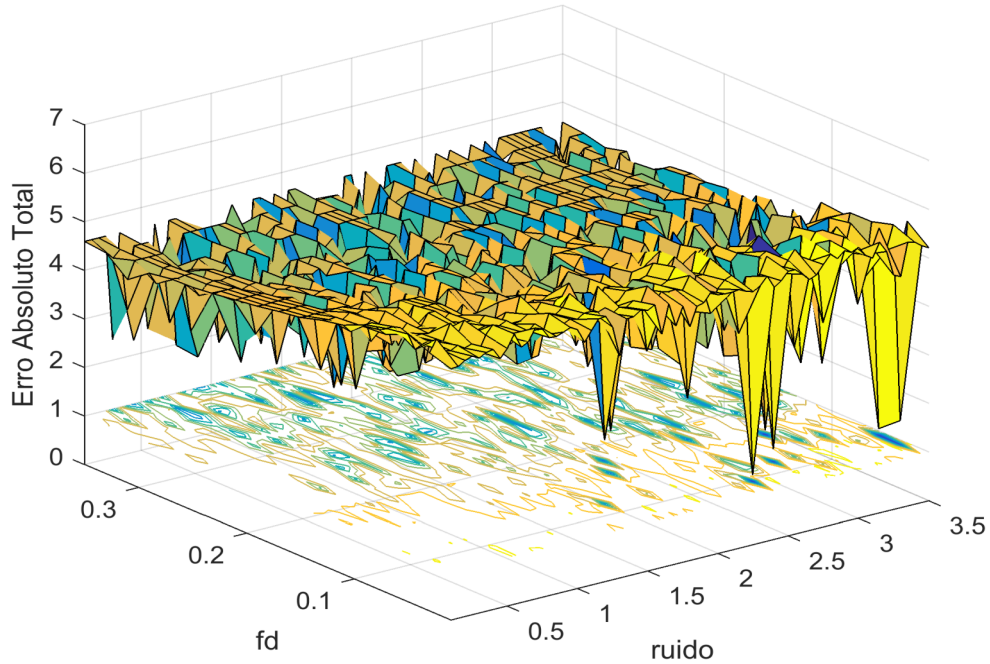


Figura 22 – Superfície de Erro com Variação do Fator de Desconto entre 0.01 e 0.35 e ruído 0.1 até 3.5.

Fonte: Produzido pelos Autores.

O resultado atingido pode ser verificado por inspeção visual na própria superfície na Figura 22, nos pontos $fd = 0.09$ e $ruído = 2.95$. Um dos objetivos do uso destas superfícies é o de facilitar para o projetista a avaliação do comportamento de influência dos parâmetros, com relação a métrica de erro como um instrumento de análise visual.

6.2.4 Análise das Condições Iniciais para os Estados x_1 e x_2 .

Uma vez identificada a influência do ruído e fator de desconto, com relação ao erro absoluto total nas seções anteriores, podemos agora também observar o comportamento dos estados iniciais que são utilizados para a geração dos dados e assim verificar se existe alguma influência significativa para no comportamento de aprendizagem do algoritmo.

Para o caso aqui discutido, a planta em questão possui-se dois estados x_1 e x_2 . Inicialmente observa-se o erro absoluto total, com relação aos estados iniciais x_1 , variando entre 0 e 5000, com intervalo de tamanho 200 e x_2 , variando com intervalo entre -2000 e 2000 e intervalo de tamanho 500.

Na superfície de erro absoluto total da Figura 23, o menor erro foi atingido para $x_1 = 1600$ e $x_2 = 1000$, com erro absoluto total de 0.3205. Para cada par de (x_1, x_2) foram realizadas simulações com ruído fixo previamente selecionado, a fim de que o ruído não influenciasse na

análise, facilitando a identificação de quais seriam os valores para estados iniciais que resultariam no menor erro absoluto total.

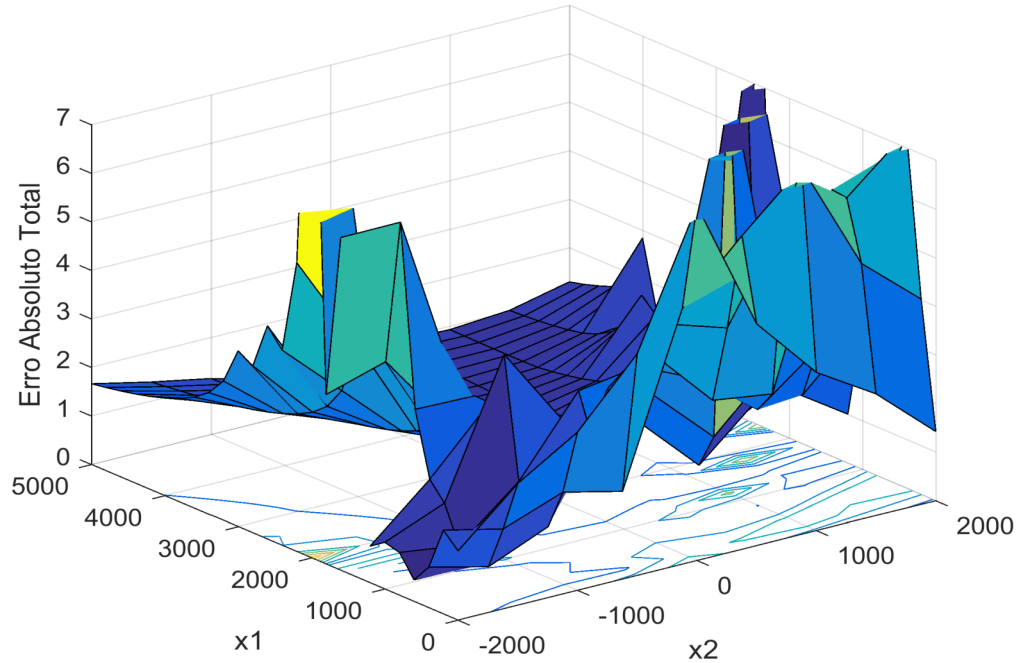


Figura 23 – Superfície de Erro para Variação dos Estados Iniciais x_1 e x_2

Fonte: Produzido pelos Autores.

Selecionando agora um intervalo de variação menor para x_1 com tamanho 100, obtêm-se uma superfície com mais detalhes, facilitando a análise da influência do comportamento de variação dos estados com relação ao erro total. A Figura 24 mostra essa situação, com o menor um erro absoluto total atingido pela superfície, com erro de 0.2644, para $x_1 = 2700$ e $x_2 = 1500$.

Restringindo o intervalo de variação de x_1 entre 2750 à 2800 e x_2 entre 1000 e 1020, com variação de 1 unidade somente, obtêm-se o resultado apresentado na Figura 25, com erro absoluto total de 0.1019, para fator de desconto 0.20.

Por inspeção visual na Figura 25, observa-se que a superfície não possui irregularidades em torno do melhores valores, assim, observa-se que não existem mais variações para x_1 e x_2 que resultem em melhor resultado, para o ruído então considerado. Os valores dos estados que retornaram o menor erro absoluto total foram: $x_1 = 2771$ e $x_2 = 1012$.

Os valores dos coeficientes encontrados nas matrizes $\bar{P}_V$ e $\bar{P}_E$, já possuem pequenos desvios, se comparados.

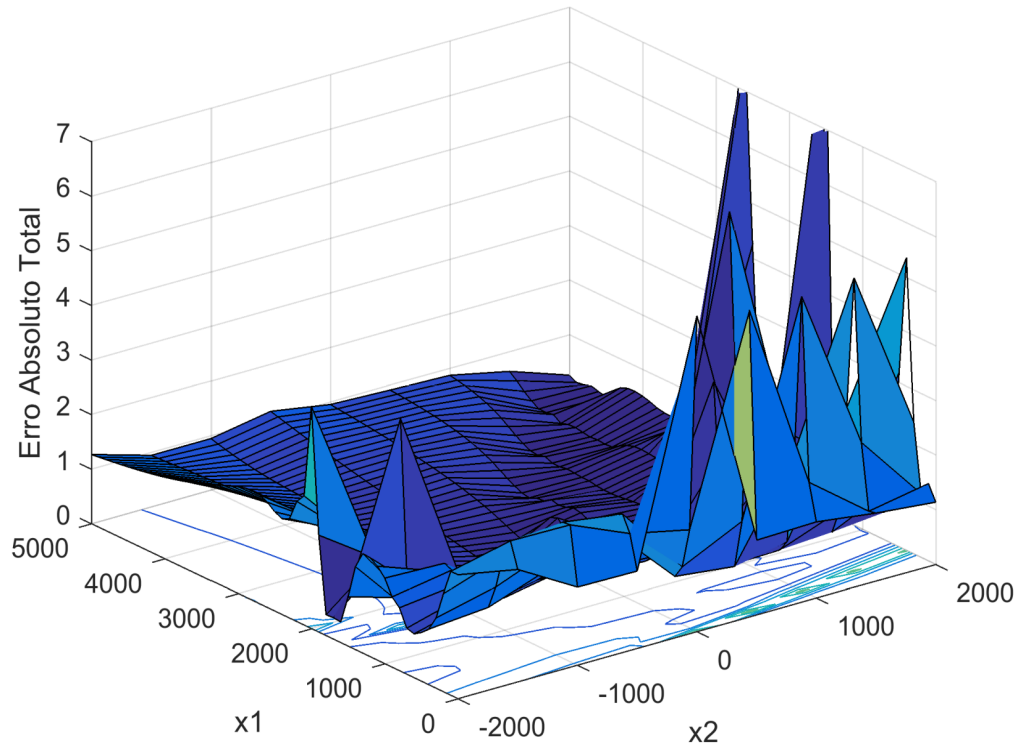


Figura 24 – Superfície de Erro para Variação dos Estados Iniciais x_1 e x_2 , com variação de 100 para x_1 e 500 para x_2 .

Fonte: Produzido pelos Autores.

$$\bar{P}_V = \begin{bmatrix} 1.0150 & -0.8440 & 1.1455 & -0.3165 \\ -0.8440 & 0.7918 & -1.0341 & 0.2969 \\ 1.1455 & -1.0341 & 1.3667 & -0.3878 \\ -0.3165 & 0.2969 & -0.3878 & 0.1113 \end{bmatrix} \quad (6.11)$$

$$\bar{P}_E = \begin{bmatrix} 1.0150 & -0.8429 & 1.1308 & -0.3192 \\ -0.8429 & 0.8049 & -1.0630 & 0.2969 \\ 1.1308 & -1.0630 & 1.3928 & -0.4002 \\ -0.3192 & 0.2969 & -0.4002 & 0.1141 \end{bmatrix} \quad (6.12)$$

Agora pode-se observar o comportamento de convergência dos coeficientes para o erro absoluto total de 0.1019 na Figura 26 e da convergência dos erros dos coeficiente na Figura 27.

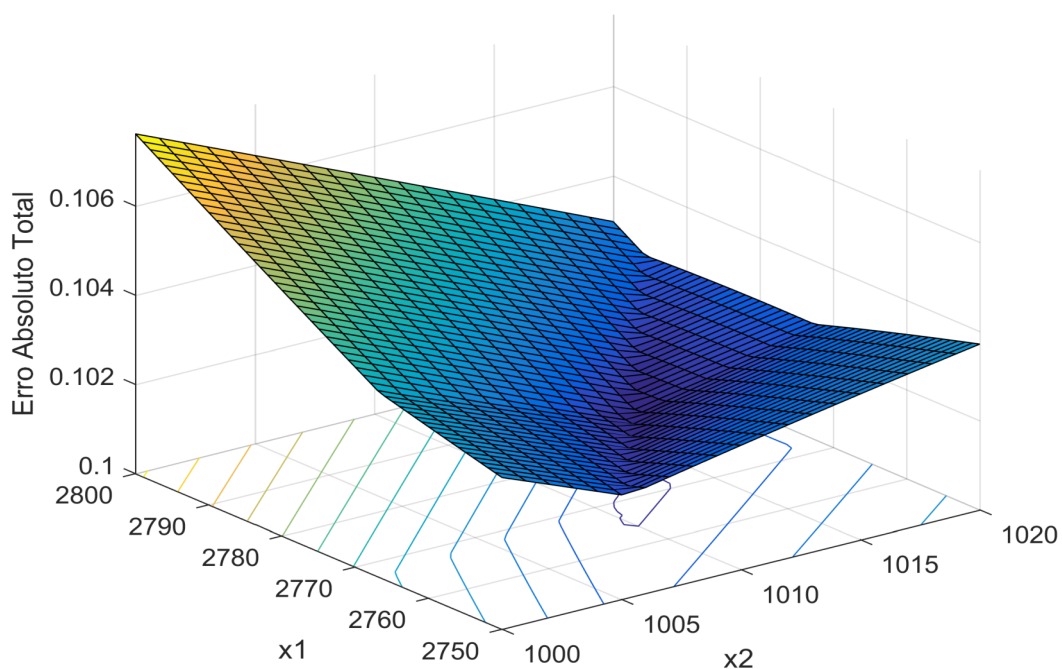


Figura 25 – Superfície de Erro para Variação dos Estados Iniciais x_1 e x_2 , entre 2750 – 2800 para x_1 e 1000 – 1020 para x_2 .

Fonte: Produzido pelos Autores.

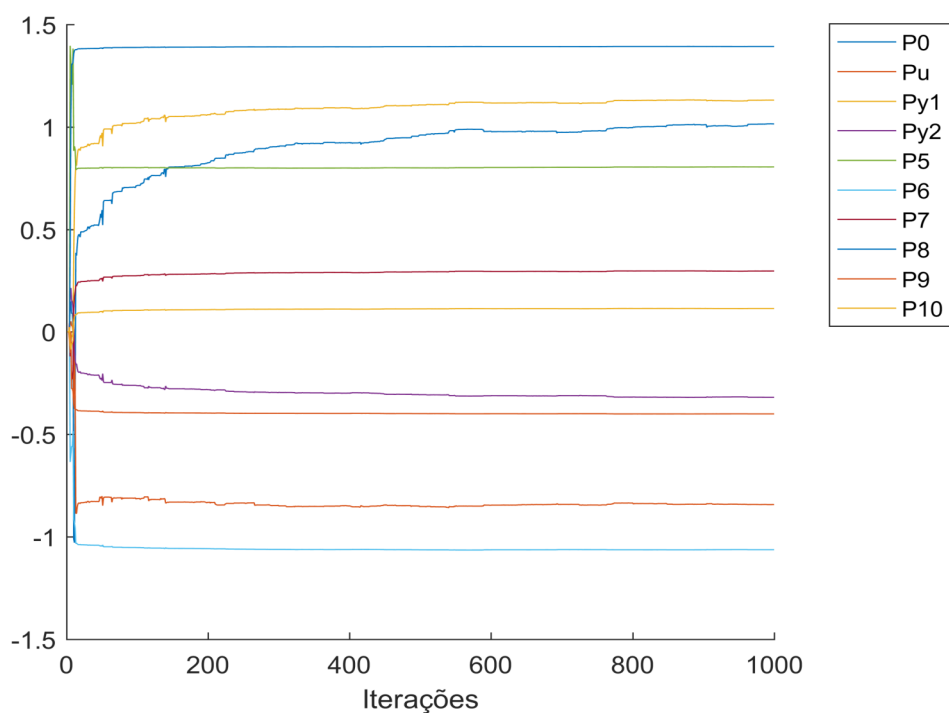


Figura 26 – Coeficiente estimados $erro = 0.1019$.

Fonte: Produzido pelos Autores.

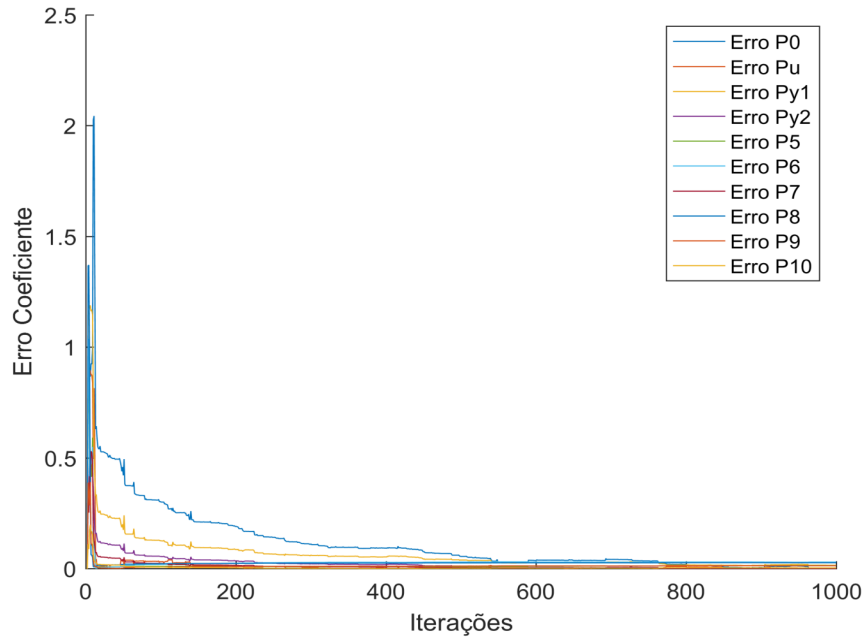


Figura 27 – Erros dos Coeficientes Estimados $erro = 0.1019$.

Fonte: Produzido pelos Autores.

Percebe-se que pela análise dos resultados apresentados, que o algoritmo é sensível aos dados iniciais dos estados, sendo assim de grande importância simulações para avaliação da resposta do algoritmo para diversas condições de iniciais, com o propósito de avaliar tal sensibilidade.

Por fim foi observado que ao mudarmos os estados iniciais, pode-se ainda ter pequenas variações do fator de desconto que poderiam modificar o erro absoluto total. Assim, foram realizadas mais simulações, variando os estados iniciais nas proximidades da melhor solução encontrada e variando o fator de desconto entre 0.19 e 0.21 com intervalos de 0.0001, obtendo-se um erro absoluto total de 0.0740, para $x_1 = 2902$ e $x_2 = 977.8$, com fator de desconto 0.1916.

O resultando das matrizes finais para $\bar{P}_V$ e $\bar{P}_E$ foram:

$$\bar{P}_V = \begin{bmatrix} 1.0150 & -0.8440 & 1.1455 & -0.3165 \\ -0.8440 & 0.7918 & -1.0341 & 0.2969 \\ 1.1455 & -1.0341 & 1.3667 & -0.3878 \\ -0.3165 & 0.2969 & -0.3878 & 0.1113 \end{bmatrix} \quad (6.13)$$

$$\bar{P}_E = \begin{bmatrix} 1.0155 & -0.8440 & 1.1355 & -0.3200 \\ -0.8440 & 0.7883 & -1.0608 & 0.2969 \\ 1.1355 & -1.0608 & 1.3845 & -0.3973 \\ -0.3200 & 0.2969 & -0.3973 & 0.1138 \end{bmatrix} \quad (6.14)$$

O comportamento de convergência dos coeficiente e dos erros dos coeficientes para a simulação que atingiu o menor erro absoluto total de 0.074 é apresentado nas Figuras 28 e Figura

29, mostrando que a convergência ocorreu para a maioria dos coeficientes rapidamente, com exceção dos coeficiente p_8 e p_{10} , que necessitaram de algumas iterações a mais para convergir.

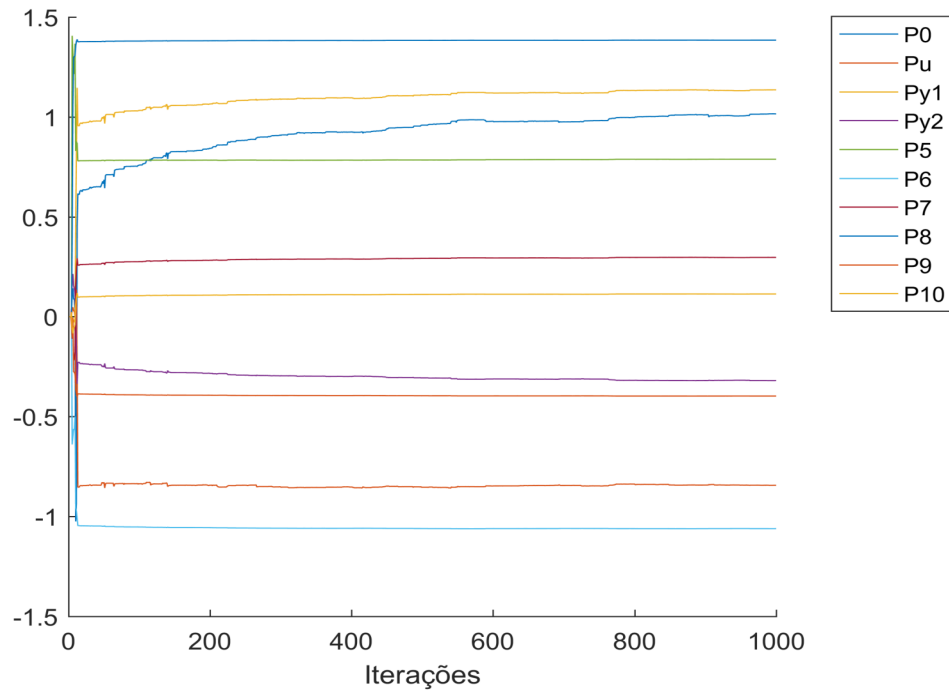


Figura 28 – Coeficiente estimados $erro = 0.0740$.

Fonte: Produzido pelos Autores.

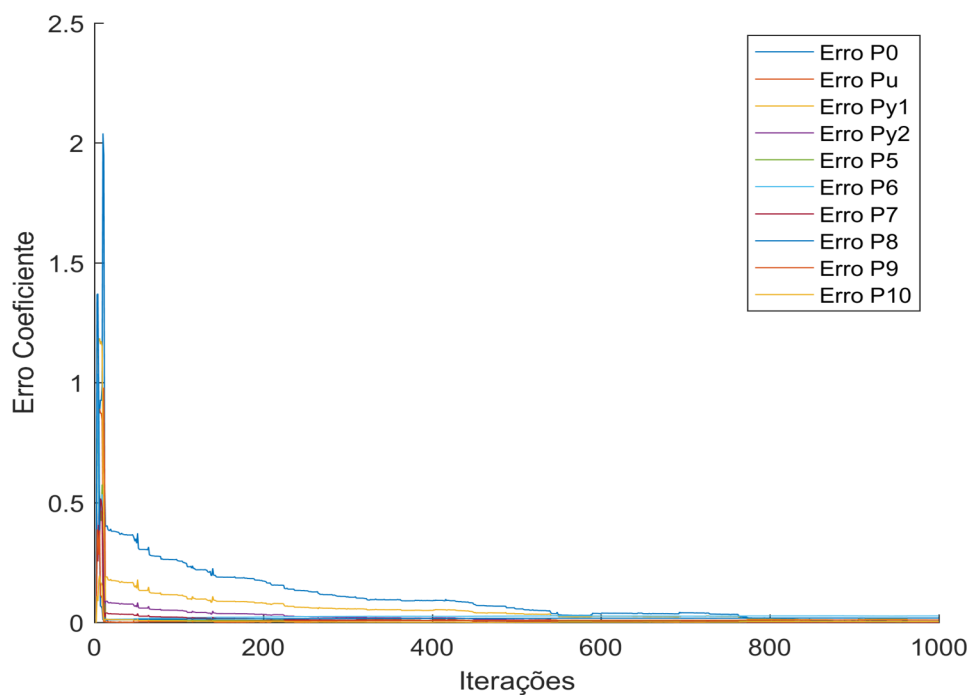


Figura 29 – Erro Coeficientes Estimados $erro = 0.0740$.

Fonte: Produzido pelos Autores.

6.3 Análise de Solvabilidade

A análise de solvabilidade apresentada neste trabalho está relacionada a capacidade do algoritmo como um todo poder retornar solução, para uma dada configuração, com relação a condições iniciais ou mesmo para valores dos parâmetros em uma faixa de valores numéricos superiores aos anteriormente analisados ou em outra faixa, visto que algumas situações ao longo da aplicação da heurística, para seleção dos parâmetros, não retornaram solução, ocorrendo divergência do algoritmo. Alguns experimentos e simulações serão apresentados a fim de realizar-se a análise de solvabilidade.

Algumas questões estão relacionadas a avaliação dos parâmetros que influenciam no erro absoluto total. Abordam-se nesta seção, aspectos relacionados aos limites dos estados, a influência do fator de esquecimento do RLS, da matriz de covariância inicial do RLS, da heurística de busca para a melhor solução, levando em consideração a vizinhança no processo de busca e as regiões que não retornam soluções. Aspectos relacionados a divergência e definição de uma superfície padrão para referência e determinação de padrões, assim como limitantes para aplicações práticas também são discutidos e analisados.

6.3.1 Amplitude dos Limites dos Estados

Em termos práticos é desejável estudar se os limites para os estados iniciais influenciam no processo de geração dos dados. Temos como propósito avaliar os limites em que a metodologia avaliada é capaz de retornar uma solução, uma vez que existem métodos sensíveis a arredondamentos numéricos, que estão relacionadas ao número de condição, propagando assim esses erros de arredondamento e com isso gerando instabilidade numérica, levando a divergência ou mesmo produzindo resultados polarizados. O estudo do condicionamento numérico é de extrema importância, pois alguns algoritmos podem divergir por problemas de singularidade no cálculo de inversão de matrizes, assim como no processo de divisão para números muito pequenos.

Visando observar o comportamento do erro absoluto total, em relação aos estados iniciais, foi gerado uma superfície de erro, com variação dos parâmetros x_1 e x_2 entre -10000 e 10000 , com gerador de números aleatórios com distribuição *Gaussiana* de média 0 e variância 0.4. Cada ponto na Figura 30, foi gerado após 100 simulações e escolhido o resultado que produziu o menor erro absoluto total. O fator de esquecimento usado nas simulações para o RLS foi de 1.

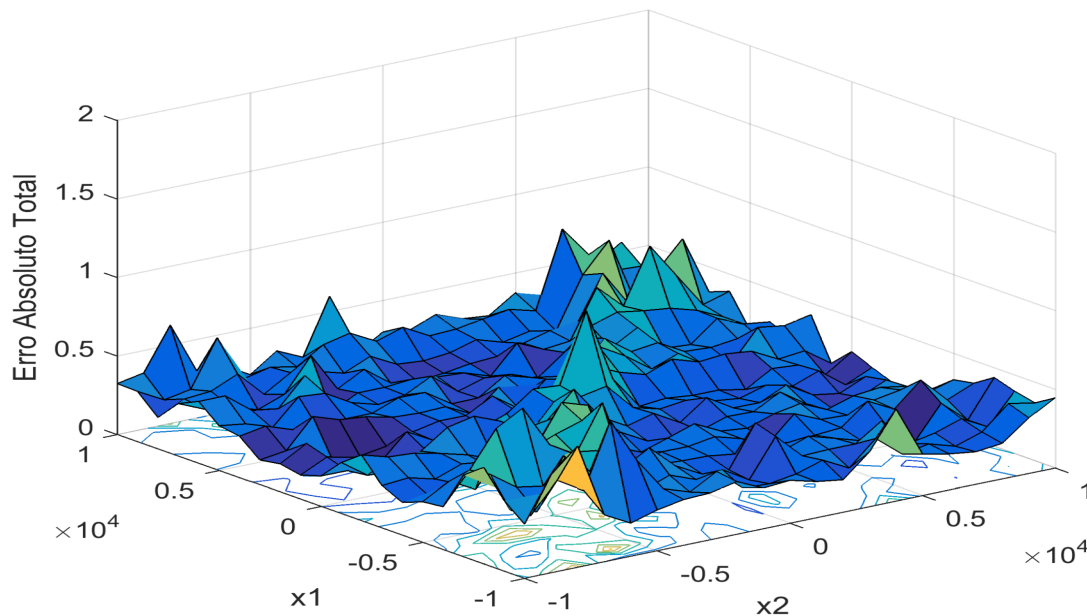


Figura 30 – Superfície de Erro para variações de x_1 e x_2 entre -10000 e 10000.

Fonte: Produzido pelos Autores.

Aparentemente por inspeção visual da Figura 30, para valores de $x_1 = x_2$, obtêm-se maiores picos e valores mais elevados para os erros absolutos, concordante com análise já feita anteriormente e apresentado na Figura 18. Os melhores resultados foram atingidos para valores de $x_1 \neq x_2$, com menores erros. Observa-se na Figura 30, que existem muitos mínimos locais e isso é causado pela variância do ruído, acarretando em desempenhos diferentes a cada simulação, obtendo-se assim uma superfície irregular.

Pode-se observar a existência de diversos mínimos locais na superfície da Figura 30 e com isso identificar o problema de minimização do erro absoluto total, como um problema de otimização combinatória, em que se deseja determinar, quais as condições iniciais dos parâmetros minimizam o erro absoluto total. O desenvolvendo de heurísticas para realizar esse processo de busca de mínimos locais, por meio da elaboração de métricas para a classificação dos elementos no espaço de busca em que os parâmetros se encontram, baseados na correlação entre as variáveis e o erro, assim como outros critérios de interesse do projetista.

A heurística para a seleção em linhas gerais podem ser definida pela seguinte sequência. Após a geração da superfície, avalia-se o comportamento do erro na vizinhança da melhor solução, diminuindo a amplitude de variação dos estados, assim como o intervalo de variação dos estados, com o propósito de refinar a busca e com isso melhorar a solução, visando selecionar mais precisamente quais parâmetros minimizam o erro.

O procedimento que representa tal heurística pode ser matematicamente descrito do seguinte modo:

Seja e^* o menor erro absoluto total na superfície na Figura 30, gerado pelos estados (x_1^*, x_2^*) e $V_\epsilon^{e_i}$ a vizinhança do erro e_i , gerado pelos estados (x_i, x_j) com raio $\epsilon > 0$, tal que $|(x_i, x_j) - (x_1^*, x_2^*)| < \epsilon$.

Selecionam-se os estados pertencentes a vizinhança de $V_\epsilon^{e_i}$ e realiza-se as simulações para cada ponto escolhido, calcula-se o erro absoluto total e observa-se o comportamento do erro, de modo a selecionar os estados que geram o menor erro absoluto total. Diminui-se o intervalo de variação dos estados e simula-se novamente, selecionando os estados que produziram os menores erros totais. Executa-se esse algoritmo por um número de iterações predefinida ou até atingir um erro pre definido. Essa situação pode ser definida como um problema de otimização a seguir de modo simplificado, considerando somente os estados e para um dado fator de desconto, podendo adicionar a inclusão de mais parâmetros, que podem ser mais sensíveis a função de erro ou de interesse do projetista.

O problema de otimização pode ser lido como: quais valores de x_1, x_2 e γ , que minimizam a função de erro absoluto total, matematicamente representado a seguir,

$$\min_{x_1, x_2, \gamma} E(x_1, x_2, \gamma). \quad (6.15)$$

Após se determinar os estados de (x_1, x_2) que geram um menor erro, é realizado o ajuste do fator de desconto, visto que foi observado por meio dos experimentos já realizados nas seções anteriores, pois, para cada conjunto de estados é necessário ajustar os parâmetros, como o fator de desconto, amplitude do ruído, dentre outros parâmetros, como foi tratado nas seções anteriores pela análise das superfícies de erros.

Após a seleção do novo fator de desconto é realizado uma verificação em torno da fronteira da melhor solução, variando os estados novamente e com isso finalizando o processo de ajuste, tanto dos estados quanto do fator de desconto, uma vez que a amplitude vai diminuindo a cada novo ajuste.

Em resumo existem dois passos de ajustes para os estados iniciais e fator de desconto. A heurística funciona nos ajustes dos mesmos a cada iteração, ou seja, para cada novo par de estados, se testa um novo fator de desconto e posteriormente se testa os estados também em sua fronteira e assim sucessivamente até que a variação, tanto dos estados, quanto do fator de desconto sejam suficientemente pequenas, atendendo as especificações do projetista.

6.3.2 Zonas Proibidas, Fator de Desconto e Esquecimento

Nesta seção avalia-se a influência do fator de desconto (fd) e fator de esquecimento (λ) do RLS, em regiões que iremos chamar de zonas proibidas, pelo fato da formulação aplicada não retornar solução, ou seja, divergir por algum motivo como já explicado no início da seção.

As análises aqui apresentadas, iniciam-se a partir da melhor solução obtida com a aplicação da metodologia já apresentada para a seleção dos parâmetros e condições iniciais. Considere a melhor solução já obtida com os estados escolhidos $x_1 = 2902$ e $x_2 = 977$. Para avaliar a influencia do fator de desconto e esquecimento do RLS, foram escolhidos fator de desconto e esquecimento variando entre 0.01 e 1, com intervalos de 0.01. O resultado dessas simulações é apresentado na superfície do erro absoluto total na Figura 31. Observa-se que existe uma grande região que não retorna solução, ou seja, existem diversos valores que tanto o fator de desconto, quanto o fator de esquecimento, não podem ser escolhidos por não retornarem soluções, existindo uma fronteira que mostra claramente que os erros foram muito elevados. Essas regiões de fronteiras são indicativos que separam as zonas proibidas, das zonas de convergência do algoritmo.

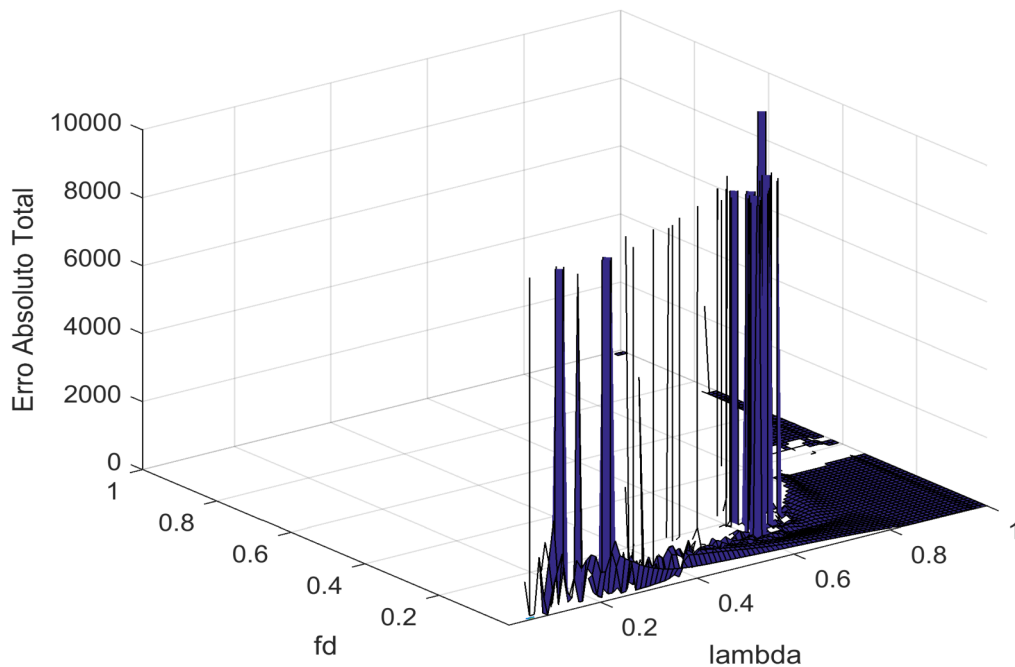


Figura 31 – Superfície de Erro para variações do fator de desconto (fd) e esquecimento ($lambda$), entre 0.01 – 1, com intervalo de 0.01.

Fonte: Produzido pelos Autores.

Os melhores valores do fator de esquecimento e fator de desconto, que retornaram a melhor solução foram, $\lambda(lambda) = 1$ e $fd = 0.19$. Observa-se que a medida que o algoritmo se aproxima das regiões proibidas, os erros totais retornam valores muito elevados.

Na Figura 31, o erro total é mostrado somente até o limite do eixo, contendo o erro de 10000, sendo que foram obtidos erros maiores, mas se perderia a visualização do gráfico e por isso foi limitado. O fato de termos valores de erros muito elevados, piora a visualização do comportamento da superfície e para isso é necessário limitar ainda mais o eixo para uma melhor visualização do erro. A Figura 31 apresenta a situação em que o limite para erro é limitado a

1000. A Figura 31 foi rotacionada para melhor visualização em torno do menor erro, como é mostrado na Figura 32.

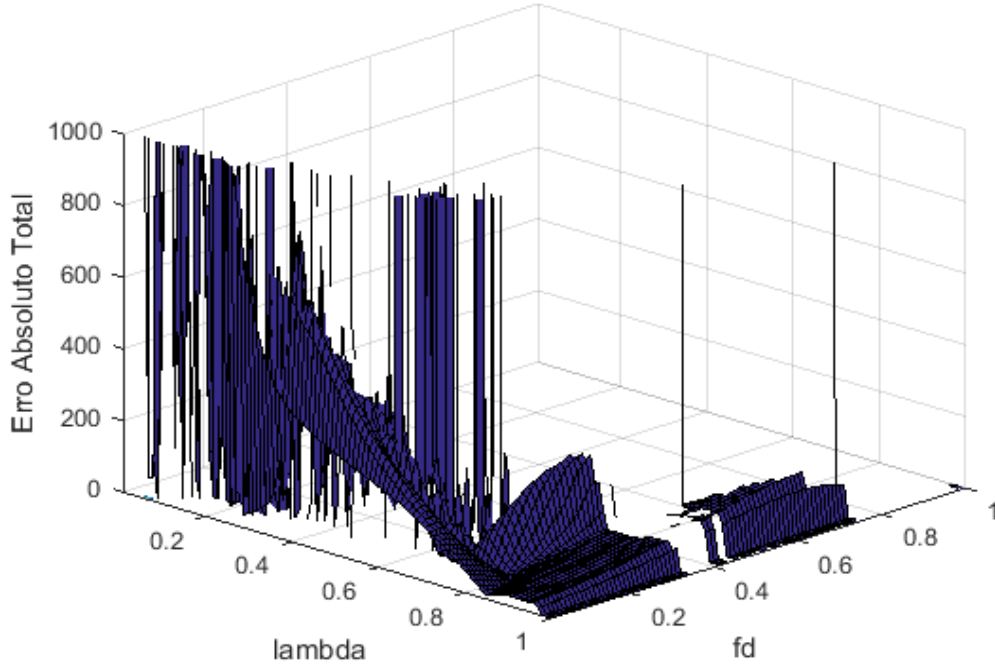


Figura 32 – Figura rotacionada da superfície de erro para variações do fator de desconto e fator de esquecimento do RLS, entre 0.01 – 1, com intervalo de 0.01 e eixo do Erro Total limitado a 1000.

Fonte: Produzido pelos Autores.

Deve-se ter em mente que o não retorno de solução do algoritmo em diversas regiões, está relacionado a diversos fatores, como por exemplo: a sensibilidade do RLS à escolha do fator de esquecimento. Outros problemas estão relacionados ao fator de desconto também e até mesmo para algum conjunto desses valores dos parâmetros, além de possíveis problemas relacionados ao condicionamento numérico e sensibilidade a ruídos.

6.3.3 Variação e Análise dos Parâmetros xk , fd e $prls_k$

Neste trabalho foram usados dois fatores multiplicativos, xk e $prls_k$. Esses fatores foram utilizados para a sintonia do algoritmo e para a análise da influência dos estados iniciais e da matriz de covariância inicial do RLS. O fator xk , foi usado para multiplicar os estados iniciais, sendo escolhidos inicialmente estados iguais, ou seja, com o mesmo fator xk , sendo posteriormente verificado que os melhores resultados seriam encontrados para valores de $x_1 \neq x_2$. Sendo assim não é necessária análise do fator xk posteriormente.

Apresenta-se nesta seção, a influência do fator multiplicativo $prls_k$ e do fator de desconto fd , com relação ao erro absoluto total, a fim de identificar a influência dos mesmos no comportamento do erro absoluto total.

Inicialmente foram avaliados valores de prls_k entre 0 e 10000, com intervalos de variação de tamanho 500 e fator de desconto entre 0.01 e 1, com intervalos de 0.01. A superfície de erro é apresentada pela Figura 33. Por conta dos elevados valores de erro, não é possível observar o seu comportamento por inspeção visual, para isso posteriormente apresenta-se a Figura 34, com o eixo do erro limitado.

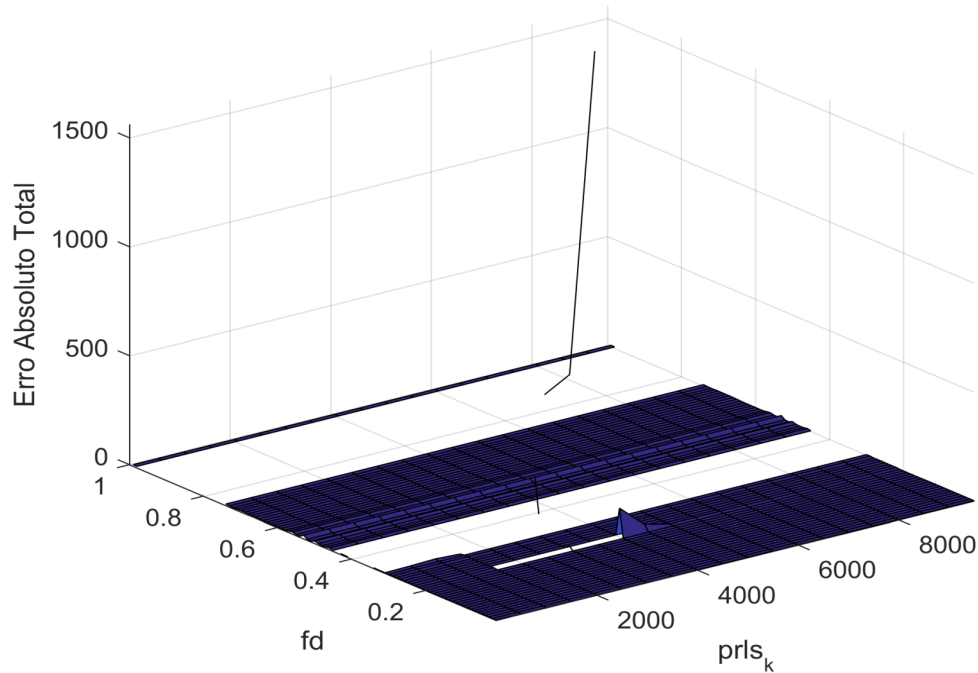


Figura 33 – Superfície de Erro para variações de fd entre 0.01 – 1, com intervalos de 0.01 e prls_k entre 1 – 10000, com intervalos de 500.

Fonte: Produzido pelos Autores.

Na Figura 33, a melhor solução foi obtida para $\text{prls}_k = 1$ e $fd = 0.19$, com erro de 0.0756. Claramente percebe-se que em duas faixas de valores, o fator de desconto não retornou solução, as faixas foram para fd entre 0.3 e 0.44, para valores de fd superiores à 0.7, embora para valores próximos de 1, o algoritmo retornou solução como pôde ser visto uma pequena listra contínua na figura.

A influência da variação dos fatores na Figura 34, limitando o eixo z até 20 é apresentado, ficando assim mais fácil de observar a influência dos parâmetros no comportamento do erro absoluto total. A faixa de valores conjuntos, para o fator de desconto e parâmetro prls_k , que aparentemente produzem os melhores valores foram, para fator de desconto abaixo de 0.2 e aparentemente não existiu uma correlação clara da influência da matriz de covariância inicial do RLS, com relação ao erro avaliado. Nesta faixa ainda tem-se uma região mais estável, visto que para todos as combinações dos parâmetros o algoritmo retornou solução, com pouca dispersão para os valores dos erros.

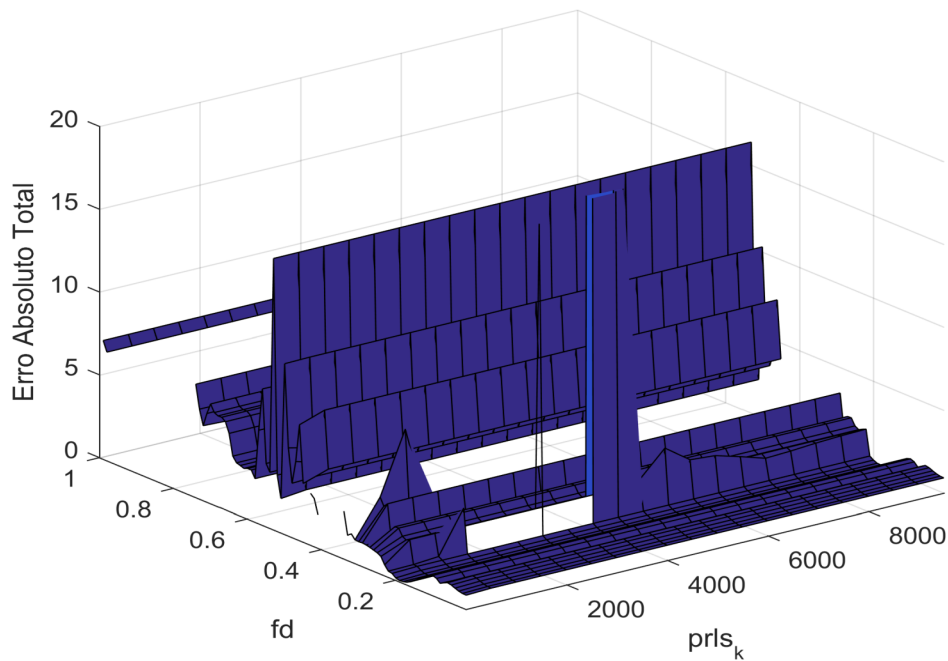


Figura 34 – Superfície de Erro para variações de fd entre 0.01 – 1, com intervalo de amostragem de 0.01 e $prls_k$ entre 0 – 10000, intervalos de 500 e erro limitado a 20.

Fonte: Produzido pelos Autores.

Existem zonas proibidas para diversos valores de fd e $prls_k$ nas superfícies anteriores. Como a melhor solução foi atingida para $prls_k = 1$. Foi construído outra superfície, na Figura 35, variando o fator $prls_k$, entre 0 e 1, com intervalos de 0.01, a fim de se observar a influência do parâmetro com relação ao erro total.

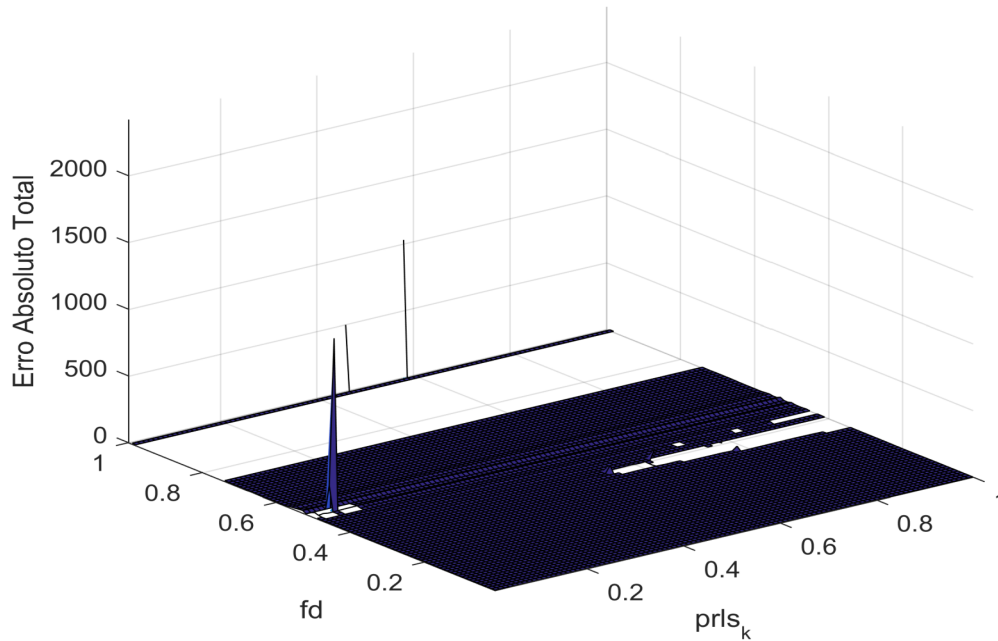


Figura 35 – Superfície de Erro para variações de fd e $prls_k$ entre 0.01 – 1, com intervalo de 0.01.

Fonte: Produzido pelos Autores.

Pode-se observar na Figura 35, que alguns valores de erros foram muito elevados e outras regiões com erros com baixos valores, que não podem ser visualizados na Figura 35, com isso limitou-se o eixo z até 20, sendo mostrado na Figura 36, obtendo-se agora uma superfície de erro mais legível e assim observa-se o comportamento de variação do erro no limiar das regiões proibidas. Percebe-se ainda que é possível rotacionar a figura a fim de se observar melhor em outros ângulos o comportamento do erro.

O melhor resultado foi atingido na Figura 36, para $prls_k = 0.98$ e $fd = 0.19$, com erro de 0.0754. Nesta região limita-se o intervalo no entorno dos parâmetros que retornaram o menor erro, visando refinar ainda mais o resultado, melhorando a solução. O melhor resultado obtido na Figura 37, foi atingido com $fd = 0.1917$ e fator $prls_k = 0.9832$, com erro total de 0.0735.

Para a simulação com o melhor resultado na Figura 37, com as curvas dos coeficientes estimados e o erro dos coeficientes, são apresentados nas Figuras 38 e 39, respectivamente. Observa-se que a convergência de praticamente todos os coeficientes ocorrem com de menos de 600 iterações. Embora não apresentado, foram realizados testes com 2000 e 5000 iterações, a fim de observar se com o aumento do número de iteração acarretaria em alguma mudança no comportamento de convergência, o que não ocorreu, sendo suficientes somente 1000 iterações. Outra questão da limitação do número de iterações é para observarmos o comportamento transitório da convergência dos parâmetros ao longo do processo iterativo.

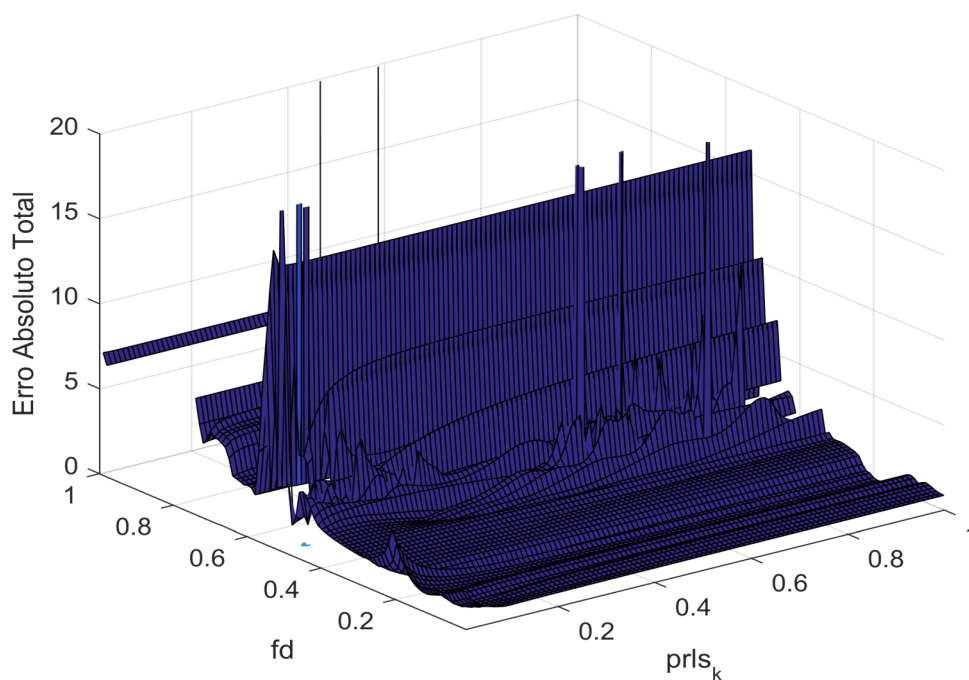


Figura 36 – Superfície de Erro para variações de fd e $prls_k$ entre 0.01 e 1, em intervalos de 0.01, com erro total limitado à 20.

Fonte: Produzido pelos Autores.

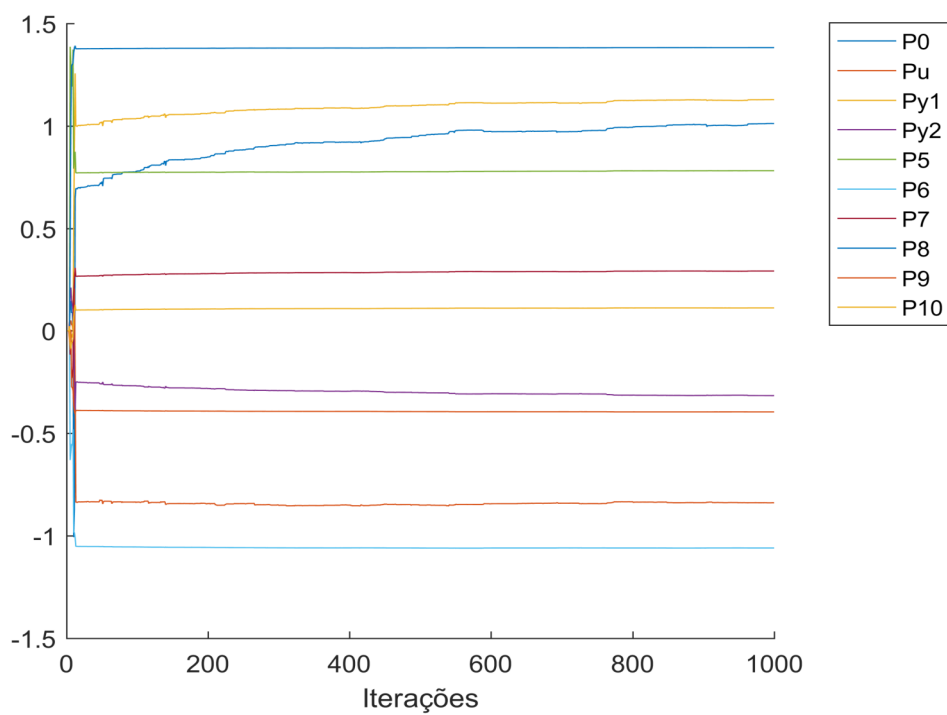


Figura 38 – Coeficientes Estimados e erro total acumulado de 0.0735.

Fonte: Produzido pelos Autores.

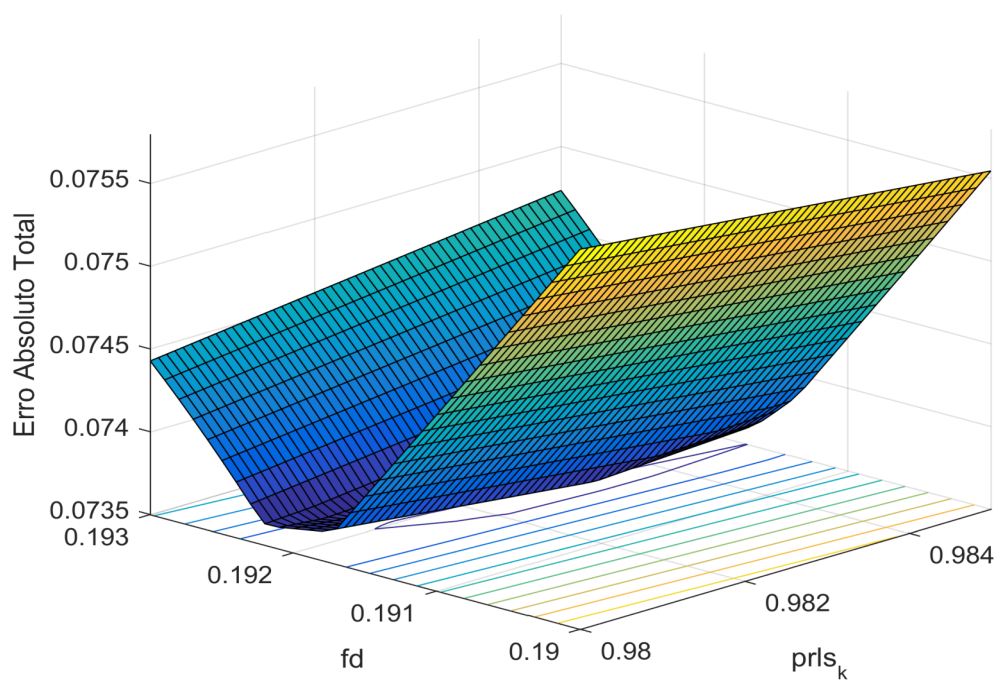


Figura 37 – Superfície de Erro para variações de fd entre 0.1900 – 0.1930 e $prls_k$, entre 0.9800 – 0.9850, com intervalo de amostragem de 0.0001.

Fonte: Produzido pelos Autores.

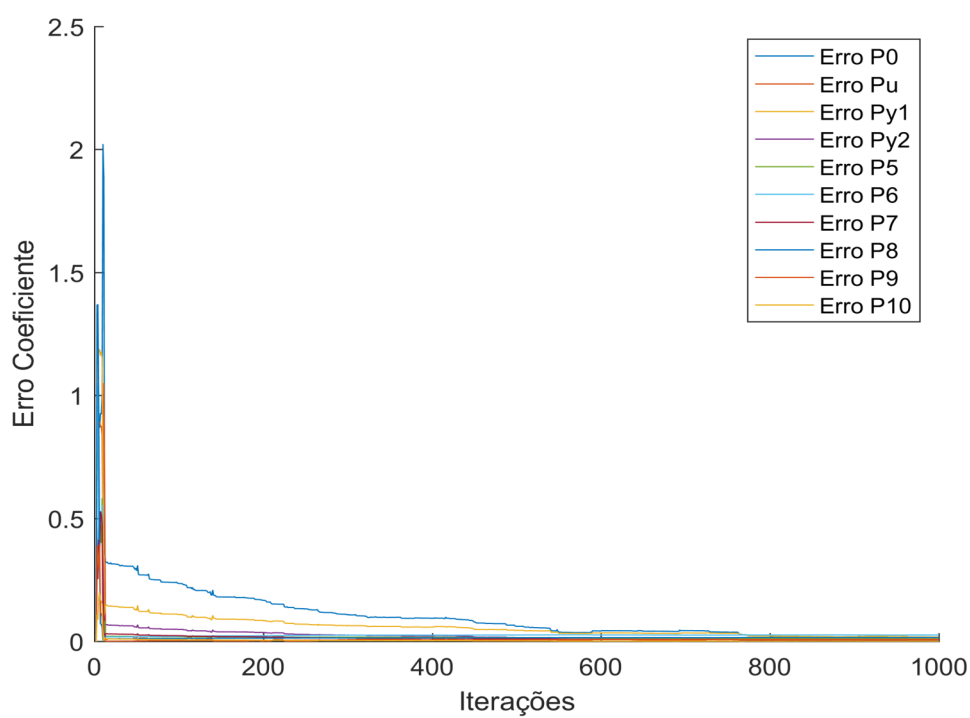


Figura 39 – Erro dos coeficientes estimados, com erro total acumulado de 0.0735.

Fonte: Produzido pelos Autores.

O resultado final das matrizes $\bar{P}_V$ e $\bar{P}_E$, para o menor erro obtido foram:

$$\bar{P}_V = \begin{bmatrix} 1.0150 & -0.8440 & 1.1455 & -0.3165 \\ -0.8440 & 0.7918 & -1.0341 & 0.2969 \\ 1.1455 & -1.0341 & 1.3667 & -0.3878 \\ -0.3165 & 0.2969 & -0.3878 & 0.1113 \end{bmatrix} \quad (6.16)$$

$$\bar{P}_E = \begin{bmatrix} 1.0150 & -0.8438 & 1.1354 & -0.3199 \\ -0.8438 & 0.7886 & -1.0608 & 0.2969 \\ 1.1354 & -1.0608 & 1.3846 & -0.3973 \\ -0.3199 & 0.2969 & -0.3973 & 0.1138 \end{bmatrix} \quad (6.17)$$

Com isso não foi possível melhorar o resultado, pois nenhuma mudança dos parâmetros produziu melhora na diminuição do erro absoluto total e por conseguinte os coeficientes da matriz $\bar{P}_E$ e com isso finalizou-se o processo iterativo para sintonia do algoritmo.

6.4 Experimentos Computacionais para a Solução do Sistema de Equações Não Lineares para Estimação de Estados

Nesta seção, apresenta-se os resultados da aplicação da rede neuronal definida na Seção 4.3, para a resolução do sistema de equações não lineares do sistema (4.20), composto de dez equações e seis incógnitas, para a recuperação das matrizes M_u e M_y e assim realizar a estimação dos estados, pela aplicação do *Lema* na Eq. (3.67).

Considere o vetor de inicialização dos pesos $X^0 = [1, 1, 1, 1, 1, 1]^T$ e o resultado final da aplicação do algoritmo *OPFB* que gerou a matriz $\bar{P}_E$,

$$\bar{P}_E = \begin{bmatrix} 1.0150 & -0.8438 & 1.1354 & -0.3199 \\ -0.8438 & 0.7886 & -1.0608 & 0.2969 \\ 1.1354 & -1.0608 & 1.3846 & -0.3973 \\ -0.3199 & 0.2969 & -0.3973 & 0.1138 \end{bmatrix}, \quad (6.18)$$

considere a matriz P admissível, utilizada pelo algoritmo *PI*,

$$P = \begin{bmatrix} 1.0150 & -0.8150 \\ -0.8150 & 0.6552 \end{bmatrix}, \quad (6.19)$$

o vetor de estados verdadeiro X_V , produzido pelo modelo,

$$X_V = \begin{bmatrix} 0.0841 \\ 0.5277 \end{bmatrix}, \quad (6.20)$$

o vetor de estado estimado X_E , ao final da resolução do sistema não linear, após 12 iterações do processo de treinamento,

$$X_E = \begin{bmatrix} 0.0903 \\ 0.7231 \end{bmatrix}, \quad (6.21)$$

a seguir, temos o erro, entre os vetores produzidos pelo modelo e estimados pela resolução do sistema não linear,

$$Erro = \begin{bmatrix} 0.0062 \\ 0.1954 \end{bmatrix}. \quad (6.22)$$

Embora o erro da estimação do estado x_2 , sendo de 0.1954, o estado x_1 apresentou um erro de 0.0062, bastante satisfatório, sendo que ainda não foi realizado o ajuste adequado do parâmetro associado a taxa de aprendizagem. Deve-se ter em mente que esse processo pode ser realizado online a cada estimação da matriz $\bar{P}$, podendo assim ser estimado os estados a cada nova iteração do algoritmo OPFB.

Pode-se realizar um estudo sobre a sintonia do parâmetro de aprendizagem associado a Eq. (4.35). Os valores recomendados por Li e Zeng [2008], para a escolha do parâmetro do numerador que será usado para calcular η_{opt} , deve estar entre, 1 e 1.6, como segue,

$$\eta_{opt}(k) = \frac{(1.0 - 1.6)}{\left\| \frac{\partial f_k(\mathbf{x}^k)}{\partial \mathbf{x}^k} \right\|^2}. \quad (6.23)$$

Para as simulações apresentadas, foi usado um valor de 1.0 para o numerador da Eq. (6.23). Esses valores estão associados a velocidade de convergência e nível oscilatório associado a convergência do processo de aprendizagem no treinamento da rede segundo Li e Zeng [2008].

Deve-se ter em mente que ao longo do treinamento, o cálculo de η_{opt} é realizado para cada uma das equações que compõem o sistema a ser solucionado e com isso faz-se necessário ter valores associados a $\eta_{opt}(k)$ diferentes, pois eventualmente algumas das equações irão convergir mais rapidamente que outras e com isso o parâmetro variável ajudaria para um treinamento mais eficiente de modo geral, garantindo velocidade de convergência no início do processo de treinamento e diminuição do erro de regime posteriormente, visando a minimização do desajuste.

Nas Figuras 40, 41 e 42, verificou-se a convergência, com o treinamento por volta de 12 iterações, solucionando assim o sistema em questão. Foram realizadas simulações com número de iterações maiores, mas não foi observado nenhuma melhora, por isso foi suficiente definir somente 50 iterações para o treinamento.

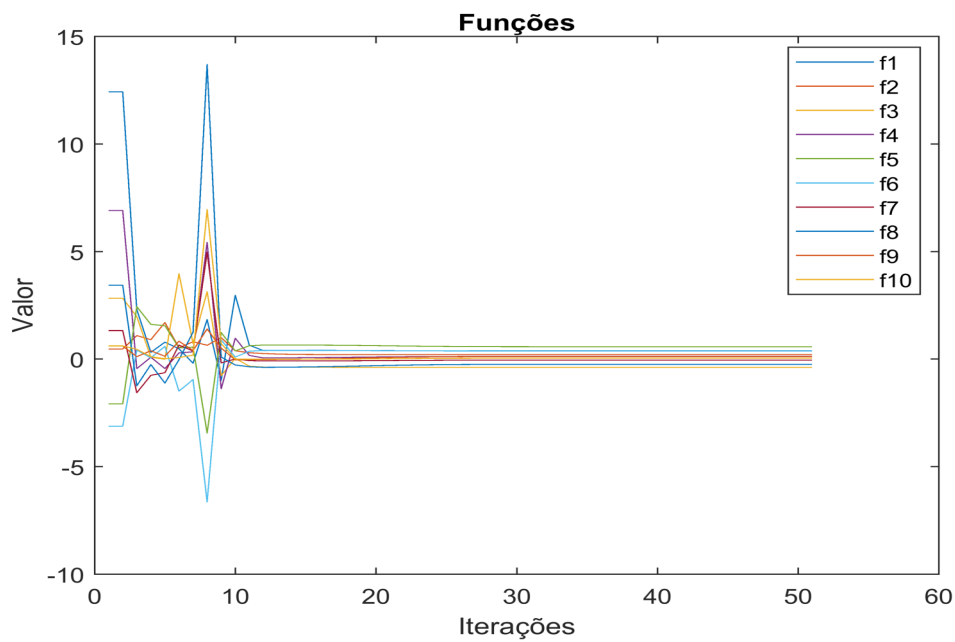


Figura 40 – Trajetória das Funções ao Longo do Processo Iterativo

Fonte: Produzido pelos Autores.

Na Figura 40, percebe-se que todas as funções convergem quase para 0, embora algumas delas converjam para valores acima de zero e outras abaixo de zero, quando calculado o índice de desempenho J , os erros quadráticos médios convergem para o valor total de 0.5364, que não é visível, por que alguns valores do índice J foram muito elevados, como mostrado na Figura 42.

A Figura 41, apresenta o erro total composto pela soma dos valores, que as funções assumiam a cada iteração. Mostrando a convergência e solução do sistema de equações.

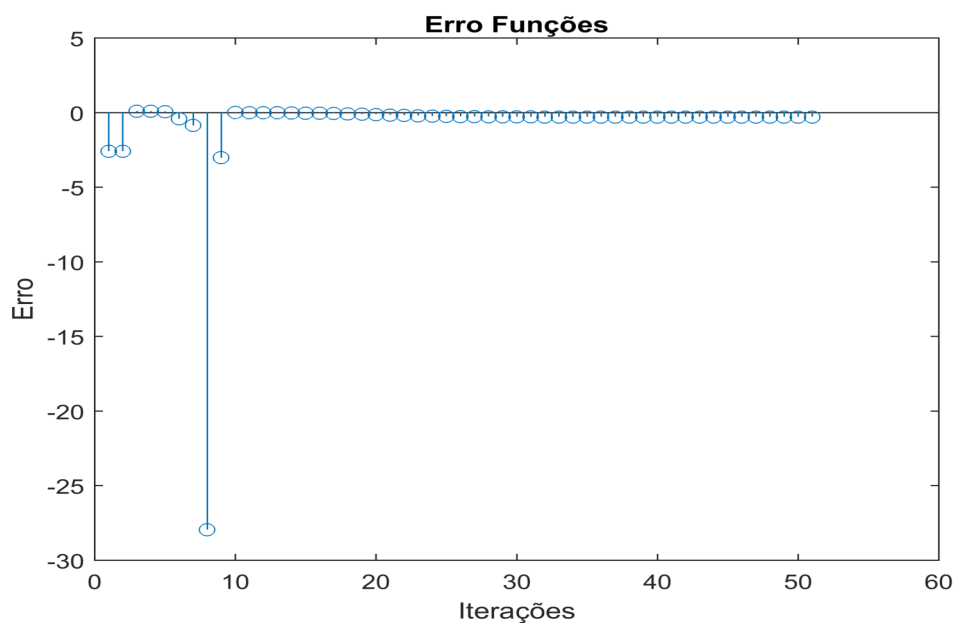


Figura 41 – Erro das funções em função das Iterações

Fonte: Produzido pelos Autores.

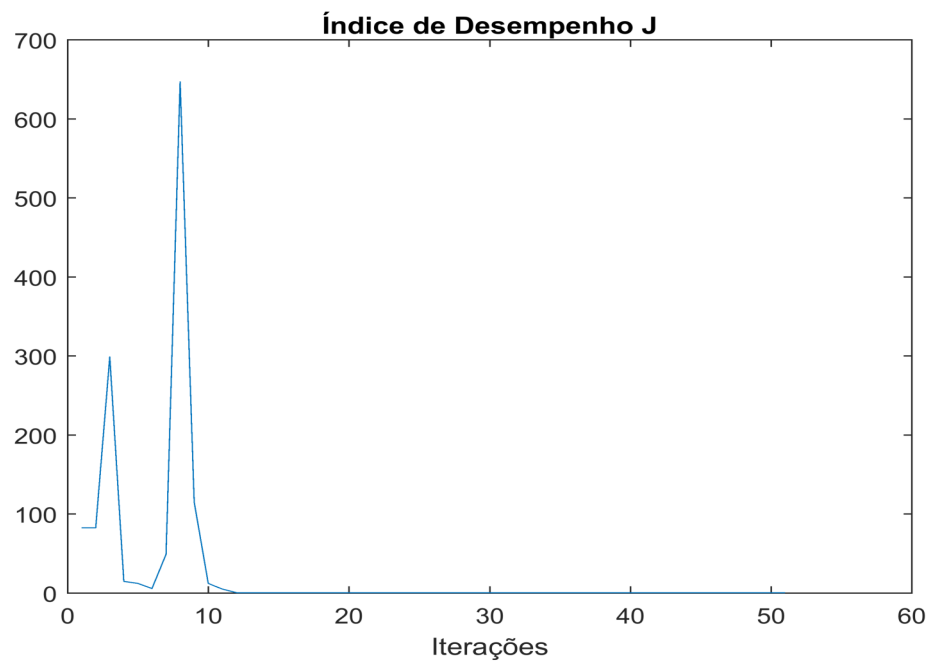


Figura 42 – Trajetória do Índice de Desempenho

Fonte: Produzido pelos Autores.

Esta metodologia baseada na resolução do sistema de equações não lineares, tem a desvantagem associada ao custo computacional para esta aplicação, pois, a cada iteração do algoritmo baseado em RL, é calculado uma nova matriz $\bar{P}_E$ e com isso sendo necessário um novo treinamento da rede para solucionar o sistema de equações não lineares, elevando assim o custo computacional para aplicação desta metodologia.

Para situações em que a constante de tempo do processo é baixa, pode-se avaliar a possibilidade de aplicação desta metodologia, para solucionar o sistema de equações e assim estimar os estados. Outra possibilidade, esta relacionado a inspeção dos estados, para um instante de tempo particular de interesse do analista.

Capítulo 7

Conclusão

Nesta Tese, foram apresentadas propostas, formulações, métodos e heurísticas, para estimação de estados e sintonia de observadores alimentados por dados, baseados em programação dinâmica aproximada e aprendizagem por reforço.

Mais especificamente, foram desenvolvidos heurísticas para sintonia dos observadores alimentados por dados, com aplicação de algoritmos de aprendizagem de máquina, algoritmos de aprendizagem por reforço, aprendizagem-Q e diferenças temporais, para estimação de estados em horizonte de tempo finito.

Foram propostos formulações, para o problema de estimação de estados sem modelo, baseados em programação dinâmica aproximada alimentados por dados. Foram realizados propostas de vários observadores de estados baseados em aprendizagem-Q, além de uma proposta para estimação de estados baseado em programação dinâmica aproximada, que resultou em um sistemas de equações não lineares. Foram ainda propostos métodos para solucionar o sistemas de equações, por meio de uma rede neuronal e uma aproximação, para o cálculo de matrizes necessárias para a estimação de estados.

Outras contribuições deste trabalho estão relacionadas a formulação de métricas, para avaliação de desempenho e métodos de análise de convergência, robustez e seleção de parâmetros em ambiente de simulação na presença de incertezas e gerador de dados de entrada e saída, para as simulações computacionais. Mais especificamente, foram propostos heurísticas para sintonia do processo de aprendizagem dos métodos propostos para estimação de estados, com a realização da seleção dos parâmetros iniciais da metodologia baseada em dados, com realimentação na saída, para sistemas parcialmente observáveis, pelo uso de superfícies de erros. As heurísticas propostas para a geração das superfícies de erros, são utilizadas para a sintonia, análise de

sensibilidade e solvabilidade do algoritmo proposto, auxiliando no processo de análise e sintonia dos observadores.

7.1 Trabalhos Futuros

- Realizar estudos para identificar relações algébricas, para a adequada implementação dos observadores propostos baseados em aprendizagem Q;
- Realizar estudos sobre complexidade computacional da estimação de estados online de modo generalizado, juntamente com o processo de estimação da matriz $\bar{P}$;
- Realizar estudos para a rede neuronal proposta, com relação as funções de ativação com saturação na resolução do sistema de equações não lineares, visando a melhoria no desempenho da rede neuronal;
- Realizar estudos de outras metodologias para estimação dos coeficientes da matriz $\bar{P}$;
- Realizar estudos sobre a possibilidade de generalização da metodologia proposta, para implementação em sistemas de ordem superior, para implementação em sistemas MIMO;
- Propor heurísticas para escolha das taxas de aprendizagem, para melhoria da eficiência no processo de resolução do sistema de equações, com relação a velocidade de convergência e acurácia da solução;
- Realizar estudos para escolha dos parâmetros associados as zonas proibidas, para uma melhor definição das fronteiras de divergência das metodologias propostas.

7.1.1 Trabalhos Publicados

- Data Driven State Reconstruction of Dynamical System Based on Approximate Dynamic Programming and Reinforcement Learning - IEEE Access (Volume 9) May 2021 <https://doi.org/10.1109/ACCESS.2021.3080626> - Qualis A1 (Engenharia IV)
- Tuning Heuristics and Convergence Analysis of Reinforcement Learning Algorithm for Online Data-Based Optimal Control Design N. - vol. 9, n. 2 (2020) - Research, Society and Development - DOI: <http://dx.doi.org/10.33448/rsd-v9i2.2128> - Qualis A3 (Qualis Preliminar 2021)
- Observadores de Estados Baseados em Programação Dinâmica Aproximada - Simpósio Brasileiro de Automação Inteligência - SBAI 2019;

Referências

AANGENENT, W.; KOSTIC, D.; JAGER, B. de; MOLENGRAFT, R. van de; STEINBUCH, M. Data-based optimal control. In: **Proceedings of the 2005, American Control Conference, 2005**. [S.l.: s.n.], 2005. p. 1460–1465 vol. 2. ISSN 0743-1619. Citado 2 vezes nas páginas 32 e 55.

ABERDEEN, D. A (revised) survey of approximate methods for solving partially observable markov decision processes. **National ICT Australia, Canberra, Australia**, 2003. Citado na página 31.

AGHAEI, V. T.; ONAT, A.; EKSIN, I.; GUZELKAYA, M. Fuzzy pid controller design using q-learning algorithm with a manipulated reward function. In: IEEE. **Control Conference (ECC), 2015 European**. [S.l.], 2015. p. 2502–2507. Citado na página 35.

AKBARIMAJD, A. Reinforcement learning adaptive pid controller for an under-actuated robot arm. **International Journal of Integrated Engineering**, v. 7, n. 2, 2015. Citado na página 35.

AL-TAMIMI, A.; LEWIS, F. L.; ABU-KHALAF, M. Discrete-time nonlinear hjb solution using approximate dynamic programming: Convergence proof. **IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)**, v. 38, n. 4, p. 943–949, Aug 2008. ISSN 1083-4419. Citado na página 86.

AL-TAMIMI, A.; VRABIE, D.; ABU-KHALAF, M.; LEWIS, F. L. Model-free approximate dynamic programming schemes for linear systems. In: IEEE. **Neural Networks, 2007. IJCNN 2007. International Joint Conference on**. [S.l.], 2007. p. 371–378. Citado na página 70.

ALONSO, H.; MENDONÇA, T.; ROCHA, P. Hopfield neural networks for on-line parameter estimation. **Neural Networks**, Elsevier, v. 22, n. 4, p. 450–462, 2009. Citado 2 vezes nas páginas 18 e 20.

ANGUELOVA, M. **Nonlinear Observability and Identifiability: 484 General Theory and a Case Study of a Kinetic Model for S. Cerevisiae**. Chalmers tekniska högsk., 2004. (Department of Mathematics, Chalmers University of Technology and Göteborg University). Disponível em: <<https://books.google.com.br/books?id=wIYLYAAACAAJ>>. Citado 3 vezes nas páginas 19, 39 e 40.

ATHANS, M.; FALB, P. L. **Optimal control: an introduction to the theory and its applications**. [S.l.]: Courier Corporation, 2013. Citado na página 70.

BALAKRISHNAN, S.; DING, J.; LEWIS, F. L. Issues on stability of adp feedback controllers for dynamical systems. **Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on**, IEEE, v. 38, n. 4, p. 913–917, 2008. Citado na página 34.

BATTISTELLI, G.; MARI, D.; SELVI, D.; TESI, P. Unfalsified approach to data-driven control design. In: **53rd IEEE Conference on Decision and Control**. [S.l.: s.n.], 2014. p. 6003–6008. ISSN 0191-2216. Citado na página 34.

BAUMEISTER, J.; LEITAO, A. **Topics in inverse problems**. [S.l.]: IMPA, 2005. Citado na página 21.

BAUMEISTER, J.; LEITÃO, A. **Introdução à teoria de controle e programação dinâmica**. [S.l.]: IMPA, 2008. Citado na página 20.

BERTSEKAS, D. P.; BERTSEKAS, D. P.; BERTSEKAS, D. P.; BERTSEKAS, D. P. **Dynamic programming and optimal control**. [S.l.]: Athena scientific Belmont, MA, 2005. v. 1. Citado 2 vezes nas páginas 57 e 70.

BIAN, T.; JIANG, Z. P. Data-driven robust optimal control design for uncertain cascaded systems using value iteration. In: **2015 54th IEEE Conference on Decision and Control (CDC)**. [S.l.: s.n.], 2015. p. 7610–7615. Citado na página 35.

BOUTILIER, C.; POOLE, D. Computing optimal policies for partially observable decision processes using compact representations. In: **Proceedings of the National Conference on Artificial Intelligence**. [S.l.: s.n.], 1996. p. 1168–1175. Citado na página 34.

BRADTKE, S. J.; YDSTIE, B. E.; BARTO, A. G. Adaptive linear quadratic control using policy iteration. In: **American Control Conference, 1994**. [S.l.: s.n.], 1994. v. 3, p. 3475–3479 vol.3. Citado 4 vezes nas páginas 57, 66, 87 e 89.

BREWER, J. Kronecker products and matrix calculus in system theory. **IEEE Transactions on Circuits and Systems**, v. 25, n. 9, p. 772–781, Sep 1978. ISSN 0098-4094. Citado na página 87.

BROCKETT, R. The early days of geometric nonlinear control. **Automatica**, Elsevier, v. 50, n. 9, p. 2203–2224, 2014. Citado na página 25.

CAMPESTRINI, L.; ECKHARD, D.; CHÍA, L. A.; BOEIRA, E. Unbiased mimo vrft with application to process control. **Journal of Process Control**, Elsevier, v. 39, p. 35–49, 2016. Citado na página 33.

CHEN, C.-T. **Linear system theory and design**. [S.l.]: Oxford University Press, Inc., 1995. Citado 8 vezes nas páginas 16, 20, 29, 37, 45, 46, 47 e 48.

CHEN, J.; YANG, F.; HAN, Q.-L. Model-free predictive control for grid-connected solar power generation systems. **Control Systems Technology, IEEE Transactions on**, IEEE, v. 22, n. 5, p. 2039–2047, 2014. Citado 2 vezes nas páginas 32 e 33.

CHEN, S.; BILLINGS, S.; GRANT, P. Non-linear system identification using neural networks. **International journal of control**, Taylor & Francis, v. 51, n. 6, p. 1191–1214, 1990. Citado 4 vezes nas páginas 18, 19, 20 e 29.

CHENG, G. S. Model-free adaptive (mfa) control. **Computing and Control Engineering, IET**, v. 15, n. 3, p. 28–33, 2004. Citado na página 31.

- CHIERA, B. A.; WHITE, L. B. Application of model-free lqg subspace predictive control to tcp congestion control. **International Journal of Adaptive Control and Signal Processing**, Wiley Online Library, v. 22, n. 6, p. 551–568, 2008. Citado na página 33.
- CHU, S. R.; SHOURESHI, R.; TENORIO, M. Neural networks for system identification. **Control Systems Magazine, IEEE**, IEEE, v. 10, n. 3, p. 31–35, 1990. Citado 2 vezes nas páginas 18 e 20.
- DAVIS, M.; VARAIYA, P. Dynamic programming conditions for partially observable stochastic systems. **SIAM Journal on Control**, SIAM, v. 11, n. 2, p. 226–261, 1973. Citado na página 31.
- DONG, N.; CHEN, Z. A novel adp based model-free predictive control. **Nonlinear Dynamics**, Springer Netherlands, v. 69, n. 1-2, p. 89–97, 2012. ISSN 0924-090X. Disponível em: <<http://dx.doi.org/10.1007/s11071-011-0248-3>>. Citado 3 vezes nas páginas 31, 32 e 34.
- ELLIS, G. **Observers in control systems: a practical guide**. [S.l.]: Academic press, 2002. Citado 2 vezes nas páginas 16 e 20.
- FAN, Q. Y.; YANG, G. H. Adaptive actor-critic design-based integral sliding-mode control for partially unknown nonlinear systems with input disturbances. **IEEE Transactions on Neural Networks and Learning Systems**, v. 27, n. 1, p. 165–177, Jan 2016. ISSN 2162-237X. Citado na página 35.
- FAVOREEL, W.; MOOR, B. D.; GEVERS, M.; OVERSCHEE, P. V. Closed loop model-free subspace-based lqg-design. In: **Proc. of the 7th IEEE Mediterranean Conference on Control and Automation, June**. [S.l.: s.n.], 1999. p. 28–30. Citado na página 33.
- FAVOREEL, W.; MOOR, B. D.; OVERSCHEE, P. V.; GEVERS, M. Model-free subspace-based lqg-design. In: IEEE. **American Control Conference, 1999. Proceedings of the 1999**. [S.l.], 1999. v. 5, p. 3372–3376. Citado na página 33.
- FLEMING, W. H. Optimal control of partially observable diffusions. **SIAM Journal on Control**, SIAM, v. 6, n. 2, p. 194–214, 1968. Citado na página 31.
- GENC, A. U.; AUDDY, S.; ERDOGMUS, D.; DAS, S. P.; PRINCIPE, J. C. Adaptive extended luenberger observer in state feedback control of nonlinear dynamical systems. Citado na página 20.
- GOULIANAS, K.; MARGARIS, A.; ADAMOPOULOS, M. Finding all real roots of 3×3 nonlinear algebraic systems using neural networks. **Applied Mathematics and Computation**, Elsevier, v. 219, n. 9, p. 4444–4464, 2013. Citado na página 68.
- GREVILLE, T. Solutions of the matrix equation $xax=x$, and relations between oblique and orthogonal projectors. **SIAM Journal on Applied Mathematics**, SIAM, v. 26, n. 4, p. 828–832, 1974. Citado na página 66.
- GUILDAS, B. **Nonlinear Observers and Applications**. [S.l.]: Springer-Verlag Berlin Heidelberg, 2007. Citado 12 vezes nas páginas 19, 20, 25, 37, 38, 39, 40, 42, 43, 49, 51 e 52.
- HAUSKRECHT, M. Value-function approximations for partially observable markov decision processes. **Journal of Artificial Intelligence Research**, v. 13, p. 33–94, 2000. Citado na página 34.

- HERMANN, R.; KRENER, A. J. Nonlinear controllability and observability. **Automatic Control, IEEE Transactions on**, v. 22, n. 5, p. 728–740, Oct 1977. ISSN 0018-9286. Citado 3 vezes nas páginas 16, 22 e 41.
- HINNEN, K.; VERHAEGEN, M.; DOELMAN, N. A data-driven h_2 -optimal control approach for adaptive optics. **IEEE Transactions on Control Systems Technology**, v. 16, n. 3, p. 381–395, May 2008. ISSN 1063-6536. Citado na página 33.
- HIRSCHORN, R. M.; LEWIS, A. D. Geometric local controllability: Second-order conditions. 2002. Citado na página 21.
- HOU, Z.-S.; WANG, Z. From model-based control to data-driven control: Survey, classification and perspective. **Information Sciences**, Elsevier, v. 235, p. 3–35, 2013. Citado na página 32.
- HOU, Z.-S.; XU, J.-X. On data-driven control theory: the state of the art and perspective. **Acta Automatica Sinica**, v. 35, n. 6, p. 650–667, 2009. Citado na página 32.
- HOU ZHONGSHENG; JIN, S. **Model Free Adaptive Control : Theory and Applications**. [S.l.]: CRC Press, 2013. ISBN 9781466594180,9781466594197. Citado 2 vezes nas páginas 16 e 33.
- III, C. C. W. A survey of solution techniques for the partially observed markov decision process. **Annals of Operations Research**, Springer, v. 32, n. 1, p. 215–230, 1991. Citado na página 35.
- ISIDORI, A. **Nonlinear Control Systems**. 3rd. ed. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1995. ISBN 3540199160. Citado 5 vezes nas páginas 16, 21, 37, 39 e 41.
- JIANG, Y.; JIANG, Z.-P. Global adaptive dynamic programming for continuous-time nonlinear systems. IEEE, 2013. Citado na página 34.
- KAILATH, T. **Linear systems**. [S.l.]: Prentice-Hall Englewood Cliffs, NJ, 1980. v. 1. Citado 4 vezes nas páginas 20, 29, 37 e 47.
- KAPIO, J.; SOMERSALO, E. Nonstationary inverse problems and state estimation. **Journal of Inverse and Ill Posed Problems**, VSP, v. 7, p. 273–282, 1999. Citado na página 20.
- KRESSNER, D.; LIU, X. Structured canonical forms for products of skew-symmetric matrices and the matrix equation $xax = b$. **The Electronic Journal of Linear Algebra**, v. 26, 2013. Citado na página 66.
- LEE, J. Y.; PARK, J. B.; CHOI, Y. H. Model-free approximate dynamic programming for continuous-time linear systems. In: IEEE. **Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on**. [S.l.], 2009. p. 5009–5014. Citado na página 70.
- LEWIS, F. L.; SYRMOS, V. L. **Optimal control**. [S.l.]: John Wiley & Sons, 1995. Citado 2 vezes nas páginas 53 e 70.
- LEWIS, F. L.; VAMVOUDAKIS, K. G. Reinforcement learning for partially observable dynamic processes: Adaptive dynamic programming using measured output data. **IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)**, v. 41, n. 1, p. 14–25, Feb 2011. ISSN 1083-4419. Citado 9 vezes nas páginas 36, 52, 55, 56, 59, 81, 87, 88 e 89.

- LEWIS, F. L.; VRABIE, D.; SYRMOS, V. L. **Optimal control**. [S.l.]: John Wiley & Sons, 2012. Citado na página [70](#).
- LI, G.; ZENG, Z. A neural-network algorithm for solving nonlinear equation systems. In: IEEE. **Computational Intelligence and Security, 2008. CIS'08. International Conference on**. [S.l.], 2008. v. 1, p. 20–23. Citado 3 vezes nas páginas [68](#), [70](#) e [121](#).
- LI, H.; LIU, D.; WANG, D.; LI, C. Data-driven iterative adaptive dynamic programming algorithm for approximate optimal control of unknown nonlinear systems. In: **2014 International Joint Conference on Neural Networks (IJCNN)**. [S.l.: s.n.], 2014. p. 3265–3271. ISSN 2161-4393. Citado na página [35](#).
- LI, H.; YAMAMOTO, S. A model-free predictive control method based on polynomial regression. In: IEEE. **2016 SICE International Symposium on Control Systems (ISCS)**. [S.l.], 2016. p. 1–6. Citado na página [16](#).
- LI, Y.; HOU, Z.; FENG, Y. Data-driven optimal stabilization for discrete-time nonlinear systems by approximate value iteration. In: **53rd IEEE Conference on Decision and Control**. [S.l.: s.n.], 2014. p. 5077–5082. ISSN 0191-2216. Citado na página [34](#).
- LIMING, W.; SHAN, F.; DAN, H. Unfalsified adaptive controller design for pilot-aircraft system(iccas 2015). In: **Control, Automation and Systems (ICCAS), 2015 15th International Conference on**. [S.l.: s.n.], 2015. p. 1494–1499. ISSN 2093-7121. Citado na página [34](#).
- LITTMAN, M. L. A tutorial on partially observable markov decision processes. **Journal of Mathematical Psychology**, Elsevier, v. 53, n. 3, p. 119–125, 2009. Citado na página [35](#).
- LIU, Y.; ZHANG, H.; LUO, Y.; HAN, J. Adp based optimal tracking control for a class of linear discrete-time system with multiple delays. **Journal of the Franklin Institute**, Elsevier, v. 353, n. 9, p. 2117–2136, 2016. Citado na página [35](#).
- LOVEJOY, W. S. A survey of algorithmic methods for partially observed markov decision processes. **Annals of Operations Research**, Springer, v. 28, n. 1, p. 47–65, 1991. Citado na página [35](#).
- LUENBERGER, D. Observers for multivariable systems. **IEEE Transactions on Automatic Control**, IEEE, v. 11, n. 2, p. 190–197, 1966. Citado na página [28](#).
- LUENBERGER, D. Canonical forms for linear multivariable systems. **IEEE Transactions on Automatic Control**, IEEE, v. 12, n. 3, p. 290–293, 1967. Citado na página [28](#).
- LUENBERGER, D. An introduction to observers. **IEEE Transactions on automatic control**, IEEE, v. 16, n. 6, p. 596–602, 1971. Citado na página [29](#).
- LUENBERGER, D. G. Observing the state of a linear system. **IEEE transactions on military electronics**, IEEE, v. 8, n. 2, p. 74–80, 1964. Citado na página [28](#).
- LUENBERGER, D. G. **Introduction to dynamic systems: theory, models, and applications**. [S.l.]: Wiley New York, 1979. v. 1. Citado na página [29](#).
- LUO, B.; LIU, D.; HUANG, T.; WANG, D. Model-free optimal tracking control via critic-only q-learning. **IEEE transactions on neural networks and learning systems**, 2016. Citado na página [35](#).

MARGARIS, A.; ADAMOPOULOS, M. Solving nonlinear algebraic systems using artificial neural networks. In: **Proceedings of the 10th international conference on engineering applications of artificial neural networks, August, Thessaloniki, Greece**. [S.l.: s.n.], 2007. Citado na página 68.

MARGARIS, A.; GOULIANAS, K. Finding all roots of 2×2 nonlinear algebraic systems using back-propagation neural networks. **Neural Computing and Applications**, Springer, v. 21, n. 5, p. 891–904, 2012. Citado na página 68.

MAYNE, D.; BROCKETT, R. **Geometric Methods in System Theory: Proceedings of the NATO Advanced Study Institute held at London, England, August 27-September 7, 1973**. Springer Netherlands, 1973. (Nato Science Series C:). ISBN 9789027704153. Disponível em: <<https://books.google.com.br/books?id=3UkyVjRGHU0C>>. Citado 2 vezes nas páginas 21 e 25.

MODARES, H.; PEEN, G. O.; ZHU, L.; LEWIS, F. L.; YUE, B. Data-driven optimal control with reduced output measurements. In: **Intelligent Control and Automation (WCICA), 2014 11th World Congress on**. [S.l.: s.n.], 2014. p. 1775–1780. Citado na página 34.

MONAHAN, G. E. State of the art—a survey of partially observable markov decision processes: theory, models, and algorithms. **Management Science**, INFORMS, v. 28, n. 1, p. 1–16, 1982. Citado na página 35.

MORGANSEN, K. A. **Geometric methods for nonlinear control systems**. 2004. Citado 2 vezes nas páginas 21 e 39.

NECHYBA, M. C.; XU, Y. Neural network approach to control system identification with variable activation functions. In: IEEE. **Intelligent Control, 1994., Proceedings of the 1994 IEEE International Symposium on**. [S.l.], 1994. p. 358–363. Citado 2 vezes nas páginas 18 e 20.

NISHIGUCHI, J.; KONDA, T.; DAZAI, R. Data-driven optimal control for building energy conservation. In: **SICE Annual Conference 2010, Proceedings of**. [S.l.: s.n.], 2010. p. 116–120. Citado na página 18.

OGATA, K. **Modern Control Engineering**. [S.l.]: Pearson Education, 2010. Citado 5 vezes nas páginas 29, 45, 46, 47 e 48.

O'REILLY, J. **Observers for linear systems**. [S.l.]: Academic Press, 1983. v. 170. Citado na página 20.

POWELL, W. B. **Approximate Dynamic Programming: Solving the curses of dimensionality**. [S.l.]: John Wiley & Sons, 2007. v. 703. Citado na página 70.

ROMAN, R.-C.; RADAC, M.-B.; PRECUP, R.-E.; PETRIU, E. M. Data-driven model-free adaptive control tuned by virtual reference feedback tuning. **Acta Polytech. Hung.**, v. 13, n. 1, p. 83–96, 2016. Citado na página 16.

SAEKI, M.; KONDO, K.; WADA, N.; SATOH, S. Data-driven online unfalsified control by using analytic center. In: **53rd IEEE Conference on Decision and Control**. [S.l.: s.n.], 2014. p. 2026–2031. ISSN 0191-2216. Citado na página 34.

- SAFONOV, M. G.; TSAO, T.-C. The unfalsified control concept and learning. **IEEE Transactions on Automatic Control**, v. 42, n. 6, p. 843–847, Jun 1997. ISSN 0018-9286. Citado na página 34.
- SAJJANSHETTY, K. S.; SAFONOV, M. G. Adaptive dwell-time switching in unfalsified control. In: **2015 IEEE Conference on Control Applications (CCA)**. [S.l.: s.n.], 2015. p. 1773–1778. ISSN 1085-1992. Citado na página 34.
- SCHRÖDER, D.; LENZ, U.; BEUSCHEL, M.; HANGL, F.; FRENZ, T.; STROBL, D.; STRAUB, S.; FISCHLE, K.; RAU, M.; ANGERMANN, A. et al. **Intelligent observer and control design for nonlinear systems**. [S.l.]: Springer Science & Business Media, 2013. Citado 2 vezes nas páginas 18 e 20.
- SHOUSONG, H.; QIXIN, Z. Stochastic optimal control and analysis of stability of networked control systems with long delay. **Automatica**, v. 39, n. 11, p. 1877 – 1884, 2003. ISSN 0005-1098. Citado na página 18.
- SILVA, F. N. da; NETO, J. V. F. Tuning heuristics and convergence analysis of reinforcement learning algorithm for online data-based optimal control design. **Research, Society and Development**, v. 9, n. 2, 2020. Citado na página 87.
- SINGH, S. P.; JAAKKOLA, T. S.; JORDAN, M. I. Learning without state-estimation in partially observable markovian decision processes. In: **ICML**. [S.l.: s.n.], 1994. p. 284–292. Citado na página 34.
- SMALLWOOD, R. D.; SONDIK, E. J. The optimal control of partially observable markov processes over a finite horizon. **Operations Research, INFORMS**, v. 21, n. 5, p. 1071–1088, 1973. Citado na página 35.
- SONDIK, E. J. **The optimal control of partially observable Markov processes**. [S.l.], 1971. Citado na página 31.
- SONDIK, E. J. The optimal control of partially observable markov processes over the infinite horizon: Discounted costs. **Operations research, INFORMS**, v. 26, n. 2, p. 282–304, 1978. Citado na página 35.
- SOROUSH, M. Nonlinear state-observer design with application to reactors. **Chemical Engineering Science**, Elsevier, v. 52, n. 3, p. 387–404, 1997. Citado na página 20.
- SPALL, J. C. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. **Automatic Control, IEEE Transactions on, IEEE**, v. 37, n. 3, p. 332–341, 1992. Citado na página 32.
- SPALL, J. C. **Introduction to stochastic search and optimization: estimation, simulation, and control**. [S.l.]: John Wiley & Sons, 2005. v. 65. Citado 2 vezes nas páginas 19 e 32.
- SPALL, J. C.; CRISTION, J. A. Direct adaptive control of nonlinear systems using neural networks and stochastic approximation. In: **IEEE. Decision and Control, 1992., Proceedings of the 31st IEEE Conference on**. [S.l.], 1992. p. 878–883. Citado na página 32.
- SPALL, J. C.; CRISTION, J. A. Model-free control of general discrete-time systems. In: **IEEE. Decision and Control, 1993., Proceedings of the 32nd IEEE Conference on**. [S.l.], 1993. p. 2792–2797. Citado 2 vezes nas páginas 31 e 32.

- SPOONER, J. T.; MAGGIORE, M.; ORDONEZ, R.; PASSINO, K. M. **Stable adaptive control and estimation for nonlinear systems: neural and fuzzy approximator techniques**. [S.l.]: John Wiley & Sons, 2004. v. 43. Citado 2 vezes nas páginas 19 e 20.
- SUTTON, R. S.; BARTO, A. G. **Reinforcement learning: An introduction**. [S.l.]: MIT press, 2018. Citado 2 vezes nas páginas 57 e 71.
- TERÁN, F. D. Canonical forms for congruence of matrices: a tribute to hw turnbull and ac aitken. In: **Actas del II congreso de la red ALAMA, Valencia**. [S.l.: s.n.], 2010. p. 2010–1. Citado na página 61.
- TSINIAS, J. Observer design for nonlinear systems. **Systems & Control Letters**, Elsevier, v. 13, n. 2, p. 135–142, 1989. Citado na página 20.
- WANG, J.-S.; YANG, G.-H. Data-driven output-feedback fault-tolerant compensation control for digital pid control systems with unknown dynamics. **IEEE Transactions on Industrial Electronics**, IEEE, v. 63, n. 11, p. 7029–7039, 2016. Citado na página 16.
- WANG, J.-S.; YANG, G.-H. Data-driven output-feedback fault-tolerant control for unknown dynamic systems with faults changing system dynamics. **Journal of Process Control**, Elsevier, v. 43, p. 10–23, 2016. Citado na página 35.
- WEI, Q.; LEWIS, F. L.; SUN, Q.; YAN, P.; SONG, R. Discrete-time deterministic q -learning: A novel convergence analysis. **IEEE transactions on cybernetics**, IEEE, v. 47, n. 5, p. 1224–1237, 2017. Citado na página 35.
- WEI, Q.; LIU, D. A new discrete-time iterative adaptive dynamic programming algorithm based on q -learning. In: SPRINGER. **International Symposium on Neural Networks**. [S.l.], 2015. p. 43–52. Citado na página 35.
- WEI, Q.; LIU, D. A novel policy iteration based deterministic q -learning for discrete-time nonlinear systems. **Science China Information Sciences**, Springer, v. 58, n. 12, p. 1–15, 2015. Citado na página 35.
- WERBOS, P. J. Neural networks for control and system identification. In: IEEE. **Decision and Control, 1989., Proceedings of the 28th IEEE Conference on**. [S.l.], 1989. p. 260–265. Citado 2 vezes nas páginas 18 e 20.
- WERBOS, P. J. Approximate dynamic programming for real-time control and neural modeling. **Handbook of intelligent control: Neural, fuzzy, and adaptive approaches**, Nostrand, New York, v. 15, p. 493–525, 1992. Citado na página 58.
- WOODLEY, B. R.; HOW, J. P.; KOSUT, R. L. Model free subspace based $\mathcal{H}_\infty$ control. In: IEEE. **American Control Conference, 2001. Proceedings of the 2001**. [S.l.], 2001. v. 4, p. 2712–2717. Citado na página 33.
- XIA, Z.; ZHAO, D. Online reinforcement learning control by bayesian inference. **IET Control Theory & Applications**, IET, 2016. Citado na página 35.
- XU, D.; JIANG, B.; SHI, P. Adaptive observer based data-driven control for nonlinear discrete-time processes. **Automation Science and Engineering, IEEE Transactions on**, IEEE, v. 11, n. 4, p. 1037–1045, 2014. Citado na página 16.

XU, D.; LIU, Q. Acis: An improved actor-critic method for pomdps with internal state. In: IEEE. **Tools with Artificial Intelligence (ICTAI), 2015 IEEE 27th International Conference on**. [S.l.], 2015. p. 369–376. Citado na página [31](#).

XU, H.; JAGANNATHAN, S. Model-free q-learning over finite horizon for uncertain linear continuous-time systems. In: IEEE. **Adaptive Dynamic Programming and Reinforcement Learning (ADPRL), 2014 IEEE Symposium on**. [S.l.], 2014. p. 1–6. Citado 2 vezes nas páginas [16](#) e [34](#).

YOUNES, Y. A.; NOURA, H.; MUFLEHI, M.; RABHI, A.; HAJJAJI, A. E. Model-free observer for state estimation applied to a quadrotor. In: IEEE. **Unmanned Aircraft Systems (ICUAS), 2015 International Conference on**. [S.l.], 2015. p. 1378–1384. Citado na página [32](#).

ZHANG, J.; ZHANG, H.; WANG, B.; CAI, T. Nearly data-based optimal control for linear discrete model-free systems with delays via reinforcement learning. **International Journal of Systems Science**, Taylor & Francis, v. 47, n. 7, p. 1563–1573, 2016. Citado na página [35](#).

ZHANG, N. L.; ZHANG, W. Speeding up the convergence of value iteration in partially observable markov decision processes. **Journal of Artificial Intelligence Research**, v. 14, p. 29–51, 2001. Citado na página [35](#).

ZHANG, W. **Algorithms for partially observable Markov decision processes**. Tese (Doutorado) — Citeseer, 2001. Citado na página [35](#).

ZHAO, S. A general observer based data-driven control for nonlinear discrete system. In: IEEE. **Control Conference (CCC), 2015 34th Chinese**. [S.l.], 2015. p. 1075–1079. Citado na página [16](#).

ZHU, Y.; HOU, Z.; JIN, S. Full form dynamic linearization controller based data-driven model free adaptive control. In: IEEE. **Control and Decision Conference (CCDC), 2015 27th Chinese**. [S.l.], 2015. p. 1032–1037. Citado na página [16](#).