



UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA
DOUTORADO EM ENGENHARIA ELÉTRICA

JOSIAS GUIMARÃES BATISTA

MÉTODO DE IDENTIFICAÇÃO COM INTELIGÊNCIA COMPUTACIONAL E
CONTROLE DE UM MANIPULADOR ROBÓTICO

FORTALEZA

2023

JOSIAS GUIMARÃES BATISTA

MÉTODO DE IDENTIFICAÇÃO COM INTELIGÊNCIA COMPUTACIONAL E CONTROLE
DE UM MANIPULADOR ROBÓTICO

Tese apresentada ao Programa de Pós-Graduação em Engenharia Elétrica do Centro de Tecnologia da Universidade Federal do Ceará, como requisito parcial à obtenção do título de doutor em Engenharia Elétrica. Área de Concentração: Automação e Controle.

Orientadora: Profa. Dra. Laurinda Lúcia Nogueira dos Reis

FORTALEZA

2023

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

- B337m Batista, Josias Guimarães.
Método de identificação com inteligência computacional e controle de um manipulador robótico / Josias Guimarães Batista. – 2023.
162 f. : il. color.
- Tese (doutorado) – Universidade Federal do Ceará, Centro de Tecnologia, Programa de Pós-Graduação em Engenharia Elétrica, Fortaleza, 2023.
Orientação: Profa. Dra. Laurinda Lúcia Nogueira dos Reis.
1. Manipulador robótico. 2. Identificação de sistemas. 3. Recursive least squares with particle swarm optimization. 4. Algoritmo genético. 5. Otimização por enxame de partículas. I. Título.
- CDD 621.3
-

JOSIAS GUIMARÃES BATISTA

MÉTODO DE IDENTIFICAÇÃO COM INTELIGÊNCIA COMPUTACIONAL E CONTROLE
DE UM MANIPULADOR ROBÓTICO

Tese apresentada ao Programa de Pós-Graduação em Engenharia Elétrica do Centro de Tecnologia da Universidade Federal do Ceará, como requisito parcial à obtenção do título de doutor em Engenharia Elétrica. Área de Concentração: Automação e Controle.

Aprovada em:

BANCA EXAMINADORA

Profa. Dra. Laurinda Lúcia Nogueira dos
Reis (Orientadora)
Universidade Federal do Ceará (UFC)

Prof. Dr. Fabrício Gonzalez Nogueira
Universidade Federal do Ceará (UFC)

Prof. Dr. Auzuir Ripardo de Alexandria
Instituto Federal do Ceará (IFCE)

Prof. Dr. Antonio Barbosa de Souza Júnior
Instituto Federal do Ceará (IFCE)

Prof. Dr. Antônio da Silva Silveira
Universidade Federal do Pará (UFPA)

AGRADECIMENTOS

A minha orientadora, Prof^ª. Dra. Laurinda Lúcia Nogueira dos Reis, pela orientação efetiva, compreensão, incentivo e valiosa colaboração.

A minha esposa, Amanda, pelo amor, cuidado e atenção.

Aos meus pais pelo exemplo de vida, superação e conquista.

A Deus, por tudo.

Aos meus amigos Darielson, Antônio Barbosa, Clauson, Renê pelas contribuições nas pesquisas, e a todos que de alguma forma contribuíram para a conclusão deste trabalho.

Aos professores participantes da banca examinadora pelo tempo, pelas valiosas colaborações e sugestões.

A todos do GPAR, pelo aprendizado, pela infraestrutura e por todo suporte oferecido, em especial a Vinícius Cortêz, José Raimundo e Júnior Nogueira.

“O sonho é que leva a gente para frente. Se a gente for seguir a razão, fica aquietado, acomodado.”

(Ariano Suassuna)

RESUMO

Neste trabalho apresenta-se a identificação com mínimos quadrados, mínimos quadrados recursivos e mínimos quadrados recursivos com otimização por enxame de partículas *Recursive Least Squares with Particle Swarm Optimization* (RLSPSO) do modelo de uma junta de um manipulador robótico cilíndrico acionado por motores de indução trifásico (MIT). Também é apresentado o modelo da primeira junta, em espaço de estados, encontrado a partir da dinâmica do manipulador, com a utilização de dados experimentais de corrente e velocidade. Os algoritmos de identificação são avaliados pelas métricas, coeficiente de correlação múltipla (R^2) e pela raiz quadrada do erro médio (*root mean squared error* - RMSE). São utilizados os controladores *Linear-Quadratic-Integral* (LQI) e *Linear-Quadratic-Gaussian* (LQG) otimizados com as metaheurísticas Algoritmo Genético (AG) e *Particle Swarm Optimization* (PSO), onde estas encontram as matrizes Q e R dos controladores citados. Nos resultados da identificação são apresentados as curvas dos estágios de treinamento e teste de cada algoritmo e os erros de cada estágio, além das funções de transferências discretas encontradas pelos algoritmos e, para o melhor algoritmo, no caso o RLSPSO, foi apresentada a função de transferência contínua e o modelo em espaço de estados. As principais contribuições deste trabalho são a identificação com um novo algoritmo, RLSPSO, e a melhoria dos controladores LQI e LQG otimizados com as metaheurísticas AG e PSO aplicadas ao modelo identificado pelo RLSPSO e modelo dinâmico da junta do manipulador. Em relação aos controladores, o LQI+PSO apresentou melhores resultados que os demais. Com isso pode-se afirmar que os resultados da identificação e dos controladores propostos, foram satisfatórios para a aplicação na junta do manipulador.

Palavras-chave: Manipulador robótico. Identificação de sistemas. *Recursive least squares with particle swarm optimization* (RLSPSO). Cinemática e dinâmica. Controlador LQI e LQG. Algoritmo genético. Otimização por enxame de partículas.

ABSTRACT

This work presents the identification with least squares, recursive least squares, and recursive least squares with particle swarm optimization RLSPSO of a joint model of a robotic manipulator driven by three phase induction motor. Also presented is the model of the first joint, in state space, found from the dynamics of the manipulator, using experimental data of current and velocity. The identification algorithms are evaluated by the metrics multiple correlation coefficient (R^2) and by the root mean squared error - RMSE. The controllers LQI and LQG optimized with the metaheuristics GA and PSO are used, where they find the Q and R matrices of the mentioned controllers. In the identification results, the curves of the training and test stages of each algorithm and the errors of each stage are presented, in addition to the discrete transfer functions found by the algorithms, and for the best algorithms, in this case, the RLSPSO, the continuous transfer function and the model in space are presented. of states. The main contributions of this work are the identification with a new algorithm, RLSPSO, and the improvement of the controllers LQI and LQG optimized with the metaheuristics GA and PSO applied to the model identified by the RLSPSO and dynamic model of the manipulator's joint. Regarding the controllers, the LQI+PSO presented better results than the others. With this, it can be stated that the results of the identification and the proposed controllers were satisfactory for the application in the manipulator's joint.

Keywords: Robotic manipulator. Systems identification. Recursive least squares with particle swarm optimization (RLSPSO). LQI and LQG controllers. Genetic algorithm. Particle swarm optimization.

LISTA DE FIGURAS

Figura 1 – Instalação de robôs e previsão do número anual de robôs fornecidos para fábricas no mundo.	23
Figura 2 – Configuração do manipulador cilíndrico.	38
Figura 3 – Configuração do acionamento da máquina de indução para a identificação do modelo.	38
Figura 4 – Bancada experimental com os circuitos utilizados para controle e acionamento do manipulador.	39
Figura 5 – Dados de entrada e saída experimentais da planta para o estágio de treinamento.	40
Figura 6 – Dados de entrada e saída experimentais da planta para o estágio de validação.	40
Figura 7 – Representação esquemática do modelo <i>ARX</i>	42
Figura 8 – Representação esquemática do algoritmo de identificação <i>RLSPSO</i>	46
Figura 9 – Curvas de treinamento e validação (teste) para identificação com MQ, para os dados experimentais de velocidade.	51
Figura 10 – Curvas de erro do treinamento e da validação (teste) para o MQ.	51
Figura 11 – Curvas de treinamento e validação (teste) para identificação com MQR, para os dados experimentais de velocidade.	52
Figura 12 – Curvas de erro do treinamento e da validação (teste) para o MQR.	52
Figura 13 – Curvas de treinamento e validação (teste) para identificação com <i>RLSPSO</i> , para os dados experimentais de velocidade.	54
Figura 14 – Curvas de erro do treinamento e da validação (teste) para o <i>RLSPSO</i>	54
Figura 15 – Manipulador cilíndrico.	60
Figura 16 – Projeção da primeira junta $x_0 - y_0$	61
Figura 17 – Estrutura do manipulador cilíndrico - <i>Software Solid Edge</i> [®]	64
Figura 18 – Propriedades físicas da junta 2 do manipulador - <i>Software Solid Edge</i> [®] . . .	64
Figura 19 – Propriedades físicas da junta 3 do manipulador - <i>Software Solid Edge</i> [®] . . .	65
Figura 20 – Posição e velocidade da junta 1 do manipulador, a partir dos dados experimentais.	69
Figura 21 – Aceleração da junta 1 do manipulador e torque calculado a partir dos valores de posição, velocidade e aceleração.	69
Figura 22 – Velocidades da junta 1 a partir dos dados experimentais, e calculada a partir do modelo dinâmico do manipulador.	71

Figura 23 – Estrutura do controlador <i>Linear Quadratic Regulator</i> (LQR) em diagrama de blocos.	76
Figura 24 – Estrutura de realimentação de estado com ação integral.	77
Figura 25 – Estrutura em diagrama de blocos do controlador LQG.	80
Figura 26 – Diagrama de blocos do controlador LQG em malha fechada.	81
Figura 27 – Fluxograma da implementação do AG.	84
Figura 28 – Resposta ao degrau com o controlador LQI aplicado à junta do manipulador.	91
Figura 29 – Resposta ao degrau negativo com o controlador LQI aplicado à junta do manipulador.	92
Figura 30 – Resposta ao degrau com o controlador LQI+AG aplicado à junta do manipulador.	92
Figura 31 – Resposta com degrau negativo com o controlador LQI+AG aplicado à junta do manipulador.	93
Figura 32 – Resposta ao degrau com o controlador LQI+PSO aplicado à junta do manipulador.	93
Figura 33 – Resposta ao degrau negativo com o controlador LQI+PSO aplicado à junta do manipulador.	94
Figura 34 – Resposta ao degrau com o controlador LQI aplicado ao modelo dinâmico da junta do manipulador.	94
Figura 35 – Resposta ao degrau negativo com o controlador LQI aplicado ao modelo dinâmico da junta do manipulador.	95
Figura 36 – Resposta ao degrau com o controlador LQI+AG aplicado ao modelo dinâmico da junta do manipulador.	95
Figura 37 – Resposta com degrau negativo com o controlador LQG+AG aplicado ao modelo dinâmico da junta do manipulador.	96
Figura 38 – Resposta ao degrau com o controlador LQI+PSO aplicado ao modelo dinâmico da junta do manipulador.	96
Figura 39 – Resposta ao degrau negativo com o controlador LQI+PSO aplicado ao modelo dinâmico da junta do manipulador.	97
Figura 40 – Resposta ao degrau com o controlador LQG aplicado à junta do manipulador.	98
Figura 41 – Resposta ao degrau negativo com o controlador LQG aplicado à junta do manipulador.	98

Figura 42 – Resposta ao degrau com o controlador LQG+AG aplicado à junta do manipulador.	99
Figura 43 – Resposta com degrau negativo com o controlador LQG+AG aplicado à junta do manipulador.	99
Figura 44 – Resposta ao degrau com o controlador LQG+PSO aplicado à junta do manipulador.	100
Figura 45 – Resposta ao degrau negativo com o controlador LQG+PSO aplicado à junta do manipulador.	100
Figura 46 – Resposta ao degrau com o controlador LQG aplicado ao modelo dinâmico da junta do manipulador.	101
Figura 47 – Resposta ao degrau negativo com o controlador LQG aplicado ao modelo dinâmico da junta do manipulador.	102
Figura 48 – Resposta ao degrau com o controlador LQG+AG aplicado ao modelo dinâmico da junta do manipulador.	102
Figura 49 – Resposta com degrau negativo com o controlador LQG+AG aplicado ao modelo dinâmico da junta do manipulador.	103
Figura 50 – Resposta ao degrau com o controlador LQG+PSO aplicado ao modelo dinâmico da junta do manipulador.	103
Figura 51 – Resposta ao degrau negativo com o controlador LQG aplicado ao modelo dinâmico da junta do manipulador.	104
Figura 52 – Estruturas de coordenadas anexadas a um manipulador com juntas rotacionais.	123
Figura 53 – Coordenadas dos quadros para a convenção <i>Denavit-Hartenberg</i> (D-H). . .	125
Figura 54 – Manipulador planar de dois links	126
Figura 55 – Manipulador cilíndrico de três links	128
Figura 56 – Desacoplamento cinemático	133
Figura 57 – Desacoplamento cinemático	134
Figura 58 – Projeção do centro do punho no plano $x_0 - y_0$	134
Figura 59 – Sistema com um grau de liberdade	141
Figura 60 – Corpo rígido geral.	142
Figura 61 – Arquitetura da malha interna/controla da malha externa.	150
Figura 62 – Manipulador cilíndrico utilizado na tese.	152
Figura 63 – Trajetória helicoidal no espaço cartesiano.	153

Figura 64 – Trajetória das juntas.	154
Figura 69 – Torque das juntas (modelo linearizado).	154
Figura 65 – Velocidade das juntas.	155
Figura 66 – Aceleração das juntas.	155
Figura 67 – Arranque das juntas.	156
Figura 68 – Torque das juntas.	156
Figura 70 – Configuração do acionamento de campo orientado indireto para uma máquina de indução.	158
Figura 71 – Diagrama do espaço vetorial.	161

LISTA DE TABELAS

Tabela 1 – Comparação do tempo de processamento e as métricas R^2 e $RMSE$ para a identificação com os métodos MQ, MQR e $RLSPSO$	55
Tabela 2 – Complexidade, $C(n)$, dos algoritmos para a identificação com os métodos MQ, MQR e $RLSPSO$	56
Tabela 3 – Parâmetros D-H do manipulador cilíndrico.	59
Tabela 4 – Valores de massas (m) e comprimentos (l) das juntas.	65
Tabela 5 – Valores de R^2 e $RMSE$ para o modelo dinâmico da velocidade.	71
Tabela 6 – Tempo de processamento e função de custo, J , para os controladores LQI, para o modelo do $RLSPSO$	89
Tabela 7 – Tempo de processamento e função de custo, J , para os controladores LQG, para o modelo do $RLSPSO$	90
Tabela 8 – Tempo de processamento e função de custo, J , para os controladores LQI, para o modelo dinâmico.	90
Tabela 9 – Tempo de processamento e função de custo, J , para os controladores LQG, para o modelo dinâmico.	90
Tabela 10 – Comparação através dos critérios de desempenho dos controladores LQI, LQI+AG e LQI+PSO aplicados ao modelo identificado pelo $RLSPSO$	104
Tabela 11 – Comparação através dos critérios de desempenho dos controladores LQG, LQG+AG e LQG+PSO aplicados ao modelo identificado pelo $RLSPSO$	104
Tabela 12 – Comparação através dos critérios de desempenho dos controladores LQI, LQI+AG e LQI+PSO aplicados ao modelo dinâmico da junta.	105
Tabela 13 – Comparação através dos critérios de desempenho dos controladores LQG, LQG+AG e LQG+PSO aplicados ao modelo dinâmico da junta.	105
Tabela 14 – Parâmetros para um manipulador planar de 2 juntas.	127
Tabela 15 – Parâmetros de um manipulador cilíndrico de 3 juntas	129

LISTA DE ALGORITMOS

Algoritmo 1 – MQR.	44
Algoritmo 2 – <i>RLPSO</i>	46
Algoritmo 3 – Pseudocódigo do AG.	85
Algoritmo 4 – Pseudocódigo do PSO.	87

LISTA DE ABREVIATURAS E SIGLAS

AG	Algoritmo Genético
CA	corrente alternada
CC	corrente contínua
D-H	<i>Denavit-Hartenberg</i>
DTC	<i>Direct Torque Control</i>
E-L	<i>Euler-Lagrange</i>
FOC	<i>Field Oriented Control</i>
FPGA	<i>Field Programmable Gate Array</i>
GDL	graus de liberdade
GPC	<i>Generalized Predictive Control</i>
IFR	<i>International Federation of Robotics</i>
IOFL	<i>Input and Output Feedback Linearization</i>
ISO	<i>International Standards Organization</i>
LQG	<i>Linear-Quadratic-Gaussian</i>
LQI	<i>Linear-Quadratic-Integral</i>
LQR	<i>Linear Quadratic Regulator</i>
MIT	motores de indução trifásico
PID	Proporcional-Integral-Derivativo
PSO	<i>Particle Swarm Optimization</i>
PWM	<i>Pulse Width Modulation</i>
RLSPSO	<i>Recursive Least Squares with Particle Swarm Optimization</i>
RPP	<i>Revolute-Prismatic-Prismatic</i>
SMC	<i>Sliding Mode Control</i>
SVM	<i>Space Vector Modulation</i>
SVPWM	<i>Spatial Vector Pulse Width Modulation</i>
VSI	<i>Voltage Source Inverter</i>

LISTA DE SÍMBOLOS

A_i	Matriz de transformação
a	Aceleração (m/s^2)
B	Coefficiente de atrito viscoso
C_i	Cosseno correspondente ao ângulo θ_i
d	Vetor de respostas desejadas
d_i	Junta i linear
$e(t)$	Perturbação
G_l^s	Sistema em malha fechada
H	Matriz de transformação homogênea
i	Junção i associada a uma junta
i_a^*, i_b, i_c^*	Comandos das correntes elétricas em ampères (A)
i_q^*	Corrente de quadratura em ampères (A)
I_r	Corrente do rotor (A)
J	Constante momento de inércia
J_n^0	Jacobiano do manipulador ou Jacobiano
K	Energia cinética
k	Matriz de realimentação de estados
K_d	Ganho derivativo do controlador PID
K_i	Ganho integral do controlador PID
K_p	Ganho proporcional do controlador PID
K_T	Constante de torque do motor
L	Lagrangiano do sistema
l	Comprimento das juntas
m	Massa das juntas
n	Número de articulações
$o_i x_i y_i z_i$	Quadro de coordenadas associado a um elo i

O_n^0	Vetor de coordenadas de referencia do <i>end-effector</i> no que diz respeito ao quadro base
P	Energia potencial
P_x	Ponto no espaço cartesiano
P_y	Ponto no espaço cartesiano
q_i	Articulação i
r_1	Comprimento da normal comum à junta i
R_n^0	Matriz de rotação
S_i	Seno do ângulo θ_i em graus
$T_a, T_b e T_c$	Período para chaveamento dos circuitos de potência
T_e	Torque eletromagnético (Nm)
T_j^i	Matriz de transformação
t_r	Tempo de subida (s)
t_s	Tempo de acomodação (s)
U_{out}	Tensão de referência (V)
$u(t)$	Entrada do sistema
$y(t)$	Saída do sistema
V	Tensão elétrica em volts (V)
v_n^0	Velocidade linear do <i>end-effector</i>
x	Eixo do plano cartesiano
Y	Eixo do plano cartesiano
Z	Eixo do plano cartesiano
x	Ponto de uma trajetória no espaço cartesiano
y	Ponto de uma trajetória no espaço cartesiano
z	Ponto de uma trajetória no espaço cartesiano
z^{-d}	Atraso do sistema
\dot{y}	Velocidade da junta
\ddot{y}	Aceleração da junta

α_i	Ângulo em torno da normal correspondente a junta i
θ	Rotação em torno do eixo x
θ_i	Junta i rotativa
θ_r^*	Comando de posição do rotor
τ_i	Torque da junta i
φ	Rotação em torno do eixo y
ψ	Rotação em torno do eixo z
ω_r^*	Comando de velocidade do rotor
ω_s^l	Frequência de escorregamento
ω_n^0	Velocidade angular do <i>end-effector</i>

SUMÁRIO

1	INTRODUÇÃO	22
1.1	Importância do trabalho	22
1.2	Revisão da literatura	25
1.3	Motivação e problemática	33
1.4	Objetivos e Contribuições	34
1.4.1	<i>Objetivo geral</i>	34
1.4.2	<i>Objetivos específicos</i>	34
1.4.3	<i>Contribuições da tese</i>	35
1.5	Organização geral do trabalho	36
2	IDENTIFICAÇÃO DO MODELO DA JUNTA DO MANIPULADOR	37
2.1	Manipulador Robótico	37
2.1.1	<i>Bancada Experimental</i>	37
2.2	Representação em Tempo Discreto	41
2.2.1	<i>Modelo ARX</i>	41
2.3	Mínimos Quadrados não Recursivo	42
2.4	Mínimos Quadrados Recursivo - MQR	43
2.5	MQR com Otimização por Enxame de Partículas - <i>Recursive Least Square with Particle Swarm Optimization - RLSPSO</i>	44
2.6	Métricas de Avaliação dos Algoritmos de Identificação	47
2.6.1	<i>Coefficiente de Correlação Múltipla - R^2</i>	47
2.6.2	<i>Raiz do Erro Quadrático Médio - RMSE</i>	48
2.6.3	<i>Complexidade do algoritmo</i>	48
2.7	Considerações Finais	49
3	RESULTADOS DA IDENTIFICAÇÃO DO MODELO	50
3.1	Resultados do MQ	50
3.2	Resultados do MQR	51
3.3	Resultados do MQR com PSO - <i>RLSPSO</i>	53
3.4	Comparação dos Algoritmos	55
3.5	Modelo da Junta do Manipulador	56
3.5.1	<i>Função de transferência da junta do manipulador</i>	56

3.5.2	<i>Modelo em espaço de estados da junta</i>	57
3.6	Considerações Finais	58
4	MODELO CINEMÁTICO E DINÂMICO DO MANIPULADOR	59
4.1	Cinemática do Manipulador Cilíndrico	59
4.1.1	<i>Cinemática Direta do Manipulador Cilíndrico</i>	59
4.1.2	<i>Cinemática Inversa do Manipulador cilíndrico</i>	61
4.2	Jacobiano do Manipulador Cilíndrico	62
4.3	Dinâmica do Manipulador Cilíndrico	63
4.3.1	<i>Energia Cinética</i>	65
4.3.2	<i>Energia potencial</i>	66
4.3.3	<i>Equação de Movimento</i>	67
4.4	Identificação do Modelo da Junta 1 a partir da Dinâmica	68
4.5	Considerações Finais	73
5	CONTROLADORES LQI E LQG COM METAHEURÍSTICAS	74
5.1	Controlador <i>LQR</i> - <i>Linear Quadratic Regulator</i>	74
5.2	Realimentação de Estado com Ação Integral	76
5.3	<i>Linear-Quadratic-Gaussian (LQG)</i>	79
5.4	Otimização dos Controladores LQI e LQG com Metaheurísticas	82
5.4.1	<i>Função de Custo</i>	83
5.4.2	<i>Algoritmo Genético - AG</i>	84
5.4.3	<i>Particle Swarm Optimization - PSO</i>	86
5.5	Métricas de Avaliação de Desempenho dos Controladores	87
5.6	Considerações Finais	88
6	RESULTADOS DOS CONTROLADORES	89
6.1	Aplicação dos Controladores LQI e LQG com as Metaheurísticas	89
6.1.1	<i>Modelo Identificado pelo RLSPSO</i>	89
6.1.2	<i>Modelo a partir da Dinâmica do Manipulador</i>	90
6.2	Resultados dos Controladores LQI, LQI+AG e LQI+PSO	90
6.2.1	<i>Controladores Aplicados ao Modelo Identificado com RLSPSO</i>	91
6.2.2	<i>Controladores Aplicados ao Modelo Calculado a partir da Dinâmica</i> . . .	93
6.3	Resultados dos Controladores LQG, LQG+AG e LQG+PSO	97
6.3.1	<i>Controladores Aplicados ao Modelo Identificado com RLSPSO</i>	97

6.3.2	<i>Controladores aplicados ao Modelo Calculado a partir da Dinâmica</i>	101
6.4	Comparação entre os Controladores LQI e LQG com as Metaheurísticas	101
6.4.1	<i>Modelo Identificado com RLSPSO</i>	103
6.4.2	<i>Modelo Calculado a partir da Dinâmica</i>	105
6.5	Discussões	106
6.6	Considerações Finais	107
7	CONCLUSÕES E TRABALHOS FUTUROS	108
7.1	Contribuições	108
7.2	Conclusões do Trabalho	108
7.3	Sugestões de Trabalhos Futuros	109
7.4	Publicações	110
7.4.1	<i>Artigos publicados em congressos e conferências</i>	110
7.4.2	<i>Artigos publicados em periódicos</i>	111
	REFERÊNCIAS	113
	APÊNDICES	121
	APÊNDICE A – Modelagem Cinemática e Dinâmica de Manipuladores .	121
A.1	Introdução	121
A.2	Cinemática Direta de Manipuladores Robóticos	122
A.2.1	<i>Representação Denavit-Hartenberg (DH)</i>	124
A.2.2	<i>Manipulador planar</i>	126
A.2.3	<i>Manipulador cilíndrico</i>	128
A.3	Cinemática inversa	129
A.3.1	<i>Desacoplamento cinemático</i>	131
A.3.2	<i>Abordagem geométrica</i>	133
A.3.3	<i>Configuração articulada</i>	134
A.4	Jacobiano de Manipuladores	135
A.4.1	<i>Derivação do Jacobiano</i>	136
A.4.2	<i>Velocidade angular</i>	137
A.4.3	<i>Velocidade linear</i>	138
A.4.4	<i>Combinando os Jacobianos angular e linear</i>	138
A.5	Dinâmica de Manipuladores	140
A.5.1	<i>Equações de Euler-Lagrange</i>	140

A.5.2	<i>Expressões Gerais para Energia Cinética e Potencial</i>	141
A.5.2.1	<i>Tensor de Inércia</i>	142
A.5.2.2	<i>Energia cinética para um robô de n-Link</i>	143
A.5.2.3	<i>Energia potencial para um robô de n-Link</i>	144
A.5.3	<i>Equações de Movimento</i>	144
A.6	<i>Dinâmica Inversa</i>	146
A.6.1	<i>Dinâmica Inversa do Espaço de Tarefas</i>	150
A.7	<i>Resposta ao Modelo Dinâmico</i>	152
	APÊNDICE B – Modelagem do Motor de Indução Trifásico como Servo posicionador	158
B.0.1	<i>Modelagem dinâmica do campo orientado indireto para uma máquina de</i> <i>indução</i>	158
B.0.2	<i>Modulação por largura de pulso usando vetores espaciais (Spatial Vector</i> <i>Pulse Width Modulation (SVPWM))</i>	160

1 INTRODUÇÃO

Neste capítulo será apresentada a importância deste trabalho, a revisão da literatura sobre as pesquisas com aplicações com algoritmos de identificação de mínimos quadrados, mínimos quadrados recursivo clássico e melhorado com otimização por enxame de partículas (*PSO - Particle Swarm Optimization*), a utilização do modelo dinâmico de manipuladores, o controlador LQR (*Linear Quadratic Regulator*) e LQI (*Linear-Quadratic-Integral*) otimizados pelas metaheurísticas AG (Algoritmo Genético) e PSO, e o controlador *Linear-Quadratic-Gaussian* LQG. Também serão apresentados os objetivos, as contribuições, bem como as produções científicas publicadas e submetidas até o momento, e a organização da tese.

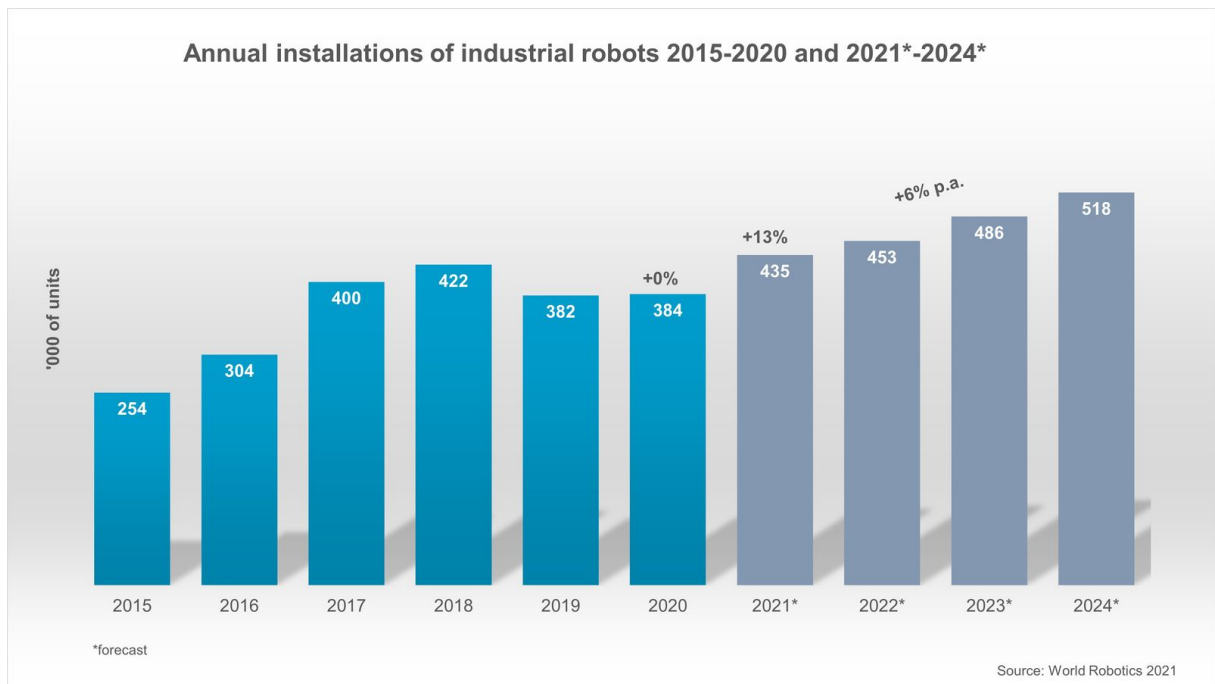
1.1 Importância do trabalho

A área da robótica tem crescido muito ao longo dos anos devido à necessidade do aumento da produção industrial e a busca pela qualidade do produto industrializado. Com isso, a utilização de robôs industriais teve um crescimento significativo no contexto de produção industrial nos últimos anos, fomentando o aumento de investimentos em robôs industriais.

A maioria dos robôs na indústria executam tarefas de manipulação de materiais, montagens, soldagem, pintura, paletização, entre outras. Com os avanços nos sistemas de acionamentos dos robôs manipuladores, sistemas de imagens e sistemas de controle, é possível os robôs dividirem a mesma área de trabalho, realizarem tarefas compartilhadas com outros robôs e interagirem com o homem (HABIB, 2014; KOPPULA; SAXENA, 2015).

A *International Federation of Robotics* (IFR) apresentou relatórios da *World Robotics 2021*, destacando que "as vendas de robôs aumentaram novamente". O relatório *World Robotics 2021* mostra um recorde de 3 milhões de robôs industriais operando em fábricas em todo o mundo – uma previsão de aumento de 13%. As vendas de novos robôs cresceram ligeiramente em 0,5%, apesar de uma pandemia global, com 384.000 unidades enviadas globalmente em 2020. Essa tendência foi dominada pelos desenvolvimentos positivos do mercado na China, compensando as contrações de outros mercados. Este é o terceiro ano de maior sucesso na história da indústria robótica, após 2018 (IFR, 2021). A Figura 1 apresenta a instalação anual de robôs no mundo de 2015 a 2020, e a previsão do número anual de robôs fornecidos para fábricas no mundo de 2021 a 2024.

Figura 1 – Instalação de robôs e previsão do número anual de robôs fornecidos para fábricas no mundo.



Fonte: IFR (2021).

Os robôs industriais colaborativos são uma classe de robôs que executam tarefas em colaboração com trabalhadores e outros robôs em ambientes industriais. A IFR define dois tipos de robôs projetados para uso colaborativo. Um grupo abrange robôs projetados para uso colaborativo que cumprem a norma 10218-1 da *International Standards Organization (ISO)*, que especifica requisitos e diretrizes para o *design* seguro, medidas de proteção e informações para o uso de robôs industriais. O outro grupo abrange robôs projetados para uso colaborativo que não atendem aos requisitos da ISO 10218-1 (MOON; VIRK, 2009). Isso não implica que esses robôs não sejam seguros, eles podem seguir diferentes padrões de segurança, por exemplo, padrões nacionais ou internos. Os robôs que trabalham com seres humanos em outros ambientes comerciais ou não comerciais (por exemplo, saúde, preparação de alimentos e espaços públicos) são cobertos por normas ISO separadas e, portanto, não são incluídos nas estatísticas da IFR sobre robôs industriais colaborativos (IFR, 2021).

Dentro desse contexto do avanço na área da robótica e melhorias nos sistemas industriais de modo geral, surgem novas ideias para melhorar cada vez mais os sistemas robóticos, além de torná-los mais econômicos. Pode-se citar aqui as melhorias relacionadas ao acionamento das juntas dos manipuladores, identificação do modelo mais próximo do real com técnicas de inteligência artificial e sistemas de controle melhorados para torná-los mais eficientes. Muitos

robôs manipuladores ainda são acionados por servomotores de corrente contínua que, apesar da sua facilidade de modelagem e controle, tem uma manutenção dispendiosa devido a grande quantidade de componentes e ao contato mecânico de suas partes (KINGSLEY *et al.*, 2006; KOSOW, 1993).

Com o avanço da eletrônica de potência, hoje é possível realizar o controle de posição e velocidade de motores de indução trifásicos. Esses motores podem ser utilizados em várias aplicações de controle de posição, inclusive em robôs manipuladores. Em Diniz *et al.* (2010), foi realizado o controle de um manipulador acionado por um motor de indução trifásico do tipo gaiola de esquilo usando *Sliding Mode Control* (SMC).

Muitas pesquisas relacionadas à identificação e controle de posição e velocidade de motores de indução trifásicos foram e ainda estão sendo desenvolvidas. Pode-se citar a pesquisa de Krim *et al.* (2019), onde foi proposto um *Direct Torque Control* (DTC) não-linear baseado em *Field Programmable Gate Array* (FPGA) associado a *Space Vector Modulation* (SVM), com *Input and Output Feedback Linearization* (IOFL) e um controlador de velocidade de torção de segunda ordem para controlar um motor de indução.

Em Rebouças *et al.* (2017) foi realizada a implementação de uma estratégia de controle de campo orientado (*Field Oriented Control* (FOC)), usando uma técnica de *Generalized Predictive Control* (GPC) aplicada à malha de velocidade e posição de um motor de indução trifásico para aplicações em baixas velocidades. Souza. *et al.* (2014), apresentaram um estudo e a implementação do controle de posição para um acionamento de um motor de indução usando a estratégia de controle vetorial para melhorar o rastreamento do eixo do motor. Neste, uma nova técnica de controle híbrido foi proposta em contraste com o controlador Proporcional-Integral-Derivativo (PID) com um ganho fixo tradicionalmente usado neste tipo de aplicação. O controle híbrido se baseia em técnicas de erro a partir de sistemas de regras *fuzzy*, que ponderam a ação do controlador PID e GPC previamente ajustados.

Do ponto de vista das necessidades de um sistema de produção industrial de forma eficiente e sem paradas, com as potencialidades de um sistema robótico, torna-se um desafio, em função do acontecimento de falhas nos equipamentos que ainda exigem a necessidade de parada para realizar a manutenção. A utilização de motores de indução trifásicos com gaiola de esquilo, na indústria, reduz a quantidade de paradas para a manutenção porque estes não têm partes móveis, como escovas, comutadores ou anéis comutadores. O item que requer manutenção são os mancais de rolamentos que após um longo período de funcionamento necessitam de

lubrificação ou a realização da troca destes. Outro fator para tornar os sistemas industriais mais eficientes, está relacionado ao modelo para a implementação do controlador. Diante disto um sistema industrial, como um manipulador robótico, necessita de um modelo de identificação preciso e um sistema de controle eficiente.

Dentro deste contexto pode-se afirmar que manipuladores robóticos acionados por motores de indução trifásico terão grandes vantagens, visto que, estes reduzem os custos de manutenção e conseqüentemente tem-se um aumento na produção. Desta forma, explora-se neste trabalho a identificação do modelo de uma junta de um manipulador utilizando o algoritmo de mínimos quadrados recursivo com otimização por enxame de partículas e de controladores LQI e LQG otimizados com metaheurísticas AG e PSO aplicados ao modelo identificado e ao modelo dinâmico de uma junta de um manipulador robótico acionada por motor de indução trifásico.

1.2 Revisão da literatura

A difusão de diversos sistemas em ambientes industriais levou ao longo dos anos ao fato de que vários métodos de identificação fossem desenvolvidos para monitorar e controlar vários modelos de plantas, como robôs móveis ou robôs manipuladores, dando-lhes a capacidade de operar com precisão e eficiência. Os robôs devem realizar as tarefas de forma correta e segura. Para tanto, necessita-se de modelos que os representem da forma mais precisa possível, se aproximando do real (MUSTAFA; AL-SAIF, 2014).

A identificação de um sistema requer que a saída do modelo seja próxima da saída real, a fim de melhorar o projeto do sistema de controle. Alguns métodos de identificação híbridos podem melhorar a estimativa de modelos por meio de técnicas de inteligência computacional, principalmente melhorando as limitações das técnicas lineares. No trabalho de Souza *et al.* (2021b) foi apresentada uma contribuição de um algoritmo híbrido para identificação de manipuladores robóticos industriais baseado no método dos mínimos quadrados recursivo (RLS), que tem sua matriz de regressores e vetor de parâmetros otimizados através do método do filtro de Kalman (KF) (RLS-KF). Também foi utilizado o algoritmo de *Extended Recursive Least Square* (ERLS) e a geração da função de transferência por cada método. Os resultados são comparados com os conhecidos mínimos quadrados recursivo e recursivo estendido considerando os critérios de coeficiente de determinação ajustável (R^2) e custo computacional. O RLS-KF apresentou melhores resultados em relação aos outros dois algoritmos (RLS e ERLS).

A não linearidade e a variação no tempo são características de alguns sistemas e,

para modelá-los e controlá-los, muitas vezes se utilizam modelos lineares. Uma das dificuldades de alguns processos é quando acontece uma mudança das condições de operação, dando assim uma escolha valiosa de partições de modelo durante as mudanças. Algumas metodologias de estimação de parâmetros do modelo foram propostas como o método dos mínimos quadrados recursivo (MQR). De acordo com o trabalho apresentado em Vijaysai *et al.* (2003) o método MQR atualiza um vetor de parâmetros e tem um custo computacional menor que o método dos mínimos quadrados não recursivo. O trabalho de Hafezi e Arefi (2019) aborda dois métodos de identificação recursiva com ruído, modelo autorregressivo de médias móveis (*autoregressive moving averages model - ARMA*) com aplicações na identificação de sistemas bilineares: especificamente, foram propostos os métodos de mínimos quadrados estendidos generalizados (*generalized extended least square - GELS*) e os métodos de máxima verossimilhança recursiva (*recursive maximum likelihood - RML*).

A qualidade dos dados obtidos pelo sistema pode influenciar significativamente na identificação de um modelo de manipulador. Geralmente alguns dados podem ser de baixa qualidade, interferindo no processo de identificação. Problemas nos dados podem incluir, por exemplo, excitação de entrada insuficiente e baixa relação sinal-ruído. Além disso, um breve conhecimento do modelo do sistema pode ajudar no projeto e implementação do método de controle do manipulador. Os sistemas físicos podem conter comportamentos não lineares e estocásticos e apresentar dados discrepantes. Esses sistemas podem interferir no desempenho dos algoritmos de identificação. A abordagem de algoritmos robustos tem relevância quando se trata de sistemas com *outliers* (valores atípicos) de acordo com Stojanovic e Nedic (2016a). O surgimento de *outliers* também pode comprometer o desempenho das técnicas quando há inserção da distribuição Gaussiana do ruído nas amostras de dados conforme apresentado em Stojanovic e Nedic (2016b).

Na pesquisa de Zhang *et al.* (2015), o método de mínimos quadrados (MQ) foi usado para resolver o problema da manobra de navios com 4 graus de liberdade. O algoritmo foi usado para realizar a modelagem de identificação com os dados do teste em escala real. Um novo algoritmo de mínimos quadrados multi-inovação transformado (*transformed multi-innovation least square - TMILS*) foi usado. O artigo de Ma *et al.* (2015) propôs uma nova abordagem para identificar um modelo baseado em *Wiener* no qual o sistema pode ser interpretado por um modelo exógeno autorregressivo acoplado a mínimos quadrados e uma máquina de vetores de suporte (*least square and a support vector machine - LSSVM*). Os parâmetros foram selecionados

por meio de otimização adaptativa por enxame de partículas (*adaptive optimization by particle swarm* - *APSO*) e o método obteve melhor desempenho quando comparado ao PSO (*particle swarm optimization*) clássico.

O trabalho de Zha *et al.* (2019) apresenta uma identificação dos parâmetros dinâmicos da extremidade inferior do exoesqueleto usando a metaheurística *particle swarm optimization* (PSO) em um espaço de busca definido por *recursive least square* (RLS), tornando-o em um método híbrido. Durante a definição do espaço de busca do PSO, o método híbrido não apenas evita a convergência da identificação parental para o mínimo local, mas também apresenta resultados muito precisos. Métodos de identificação baseados em otimização de enxame de partículas (PSO) com algumas variações foram mostrados em Bingül e Karahan (2011). O método de identificação é aplicado a um manipulador robótico, onde os intervalos estimados são usados para prever os torques da junta.

Na pesquisa de Yuan *et al.* (2017) foi utilizado um algoritmo para estimar os parâmetros das juntas de um manipulador a partir do torque exercido por cada junta de um manipulador, ao realizar trajetórias de *Fourier* com excitações periódicas. Os parâmetros foram divididos em uma parte linear e não linear e foi utilizada a estimativa de parâmetro linear de mínimos quadrados (*linear least square* - *LLS*) e a otimização por enxame de partículas com base em enxame duplo (DPSO) para calcular as partes linear e não linear, respectivamente. As configurações utilizadas foram mais simples e permitiu identificar os parâmetros dinâmicos e os coeficientes de atrito das juntas. No artigo de Urrea e Pascal (2017) foram utilizadas técnicas para identificar os parâmetros dinâmicos em um robô manipulador industrial com 5 graus de liberdade. Os parâmetros foram identificados usando MQ, redes neurais artificiais *Adaline*, redes neurais artificiais de *Hopfield* e o filtro de *Kalman* estendido.

Para resolver os problemas de identificação do robô manipulador, Yan *et al.* (2015) apresentaram uma abordagem inteligente com PSO que foi chamada de estratégia de aprendizagem elitista (ELS), e a abordagem hibridizada do controlador proporcional-integral-derivativo (ELPIDSO). A identificação dos parâmetros dos robôs manipuladores foi realizada para avaliar o desempenho da abordagem. O ELPIDSO foi superior ao método de MQ, algoritmo genético (GA) e PSO na estimativa dos parâmetros do modelo de robôs manipuladores. No artigo de Batista *et al.* (2020a) o modelo da cinemática inversa de um manipulador cilíndrico com 3-graus de liberdade (GDL) foi obtido utilizando o algoritmo de MQR com *PSO* chamado de *RLSPSO*. Neste artigo uma trajetória helicoidal foi utilizada como entrada para o modelo proposto e a

partir deste foram calculadas, as trajetórias, velocidades, acelerações, torques do manipulador.

Muitas técnicas de otimização usando metaheurísticas e métodos estocásticos têm sido usadas em muitas aplicações (SOUZA *et al.*, 2021a; PERES *et al.*, 2018; CUI; GAO, 2012; SABIR; KHAN, 2014; DONG *et al.*, 2005). A aplicação de métodos de otimização com diferentes algoritmos para uma melhor sintonia dos controladores PID foi apresentada em diversas pesquisas no controle de máquinas elétricas. Alguns algoritmos comumente considerados são algoritmos genéticos (GA), otimização de enxame de partículas (PSO) e recozimento simulado (*Simulated Annealing - SA*). Esses algoritmos são usados para resolver problemas de otimização considerando operações de teste simples. Eles têm sido aplicados com sucesso em diversos campos de sistemas de controle e os resultados provaram sua supremacia sobre os clássicos (SUSHNIGDHA; JOSHI, 2018; LIU *et al.*, 2008; KAZARLIS *et al.*, 1996). No artigo de Premkumar e Manikandan (2015), o GA, PSO e o algoritmo *bat (BA)* são usados para ajustar os parâmetros de um controlador PID aplicado em experimento a um motor de corrente contínua (CC) sem escova.

O modelo dinâmico de manipuladores é importante para o projeto de controladores aplicados ao sistema. É necessário ter um modelo matemático que revele o comportamento dinâmico de um sistema (LEWIS *et al.*, 2003). As equações dinâmicas de um robô manipulador na forma fechada podem ser obtidas a partir das equações de movimento de Newton ou de Lagrange (KELLY *et al.*, 2006).

A derivação do modelo dinâmico de um manipulador desempenha um papel importante na simulação de movimento, análise de estruturas do manipulador e projetos de algoritmos de controle. A simulação do movimento do manipulador permite que estratégias de controle e técnicas de planejamento de movimento sejam testadas sem a necessidade de usar um sistema disponível fisicamente. A análise do modelo dinâmico é útil para o projeto mecânico de manipuladores. O cálculo das forças e torques necessários para a execução de determinados movimentos fornece informações úteis para projetar juntas, transmissões e atuadores (SICILIANO *et al.*, 2010).

Em Guechi *et al.* (2018) foi realizado o controle de um braço robótico planar de dois elos. O projeto da abordagem de controle foi baseado no modelo dinâmico do robô. O modelo matemático do sistema utilizado foi não linear, e assim um controle de linearização por realimentação foi proposto para obter um sistema linear para o qual um controle preditivo foi desenvolvido. Os parâmetros de controlador foram obtidos analiticamente pela minimização

de uma função de custo. Na pesquisa também foi feito um estudo de simulação comparando a abordagem de controle preditivo com um controle linear quadrático (LQ) baseado na mesma linearização obtida.

Na pesquisa de Swevers *et al.* (2007) foi utilizada uma excitação periódica para realizar a identificação de um robô manipulador. Os resultados da pesquisa mostraram que o modelo dinâmico do manipulador estimado pode prever com precisão os torques do atuador para um determinado movimento do robô. A previsão precisa do torque do atuador foi um requisito fundamental para modelos de robôs usados para programação *offline*, otimização de tarefas e controle avançado baseado em modelo. No artigo de Tang *et al.* (2021) foi apresentada a modelagem dinâmica de um manipulador flexível *single-link* com dois cabos. O atuador final foi fixado na extremidade do elo flexível e os cabos tensionados foram usados para suprimir a vibração. Os resultados da pesquisa mostraram que a vibração foi suprimida pela utilização de cabos no manipulador de elo flexível, o que demonstra que um sistema utilizado pela dinâmica com cabos teve potencial para ser empregado nas áreas de manipuladores industriais e braços robóticos para manipulação de peças.

O controlador LQR com metaheurísticas também é utilizado em muitas aplicações no controle de velocidade e posição de sistemas. Na pesquisa de Souza *et al.* (2020) foi utilizada a metaheurística AG para projetar o regulador LQI e uma comparação foi feita entre esse algoritmo e o controlador PID de modo a selecionar a melhor técnica para o projeto do controlador LQR/LQI e ajuste dos parâmetros Q e R que foram aplicados a um sistema de acionamento com motor de relutância variável. O controlador híbrido (LQI+AG) torna-se adequado no sentido de uma possível extensão do controle de um sistema multivariável.

O artigo de Maghfiroh *et al.* (2022) propôs um controlador LQR aprimorado usando o método *Particle Swarm Optimization* (PSO) e o estimador de filtro de Kalman. O PSO foi usado para encontrar o valor ideal de Q e R , e o filtro de Kalman reduziu o número de sensores usados para medir o estado do sistema. Tanto o desempenho quanto o consumo de energia foram comparados usando a Integral do Erro Absoluto (IAE) e o consumo total de energia. Os resultados da simulação mostraram que o método proposto foi superior em desempenho e energia em comparação com o LQR ajustado manualmente. No entanto, com base na implementação experimental, embora o LQR sintonizado manualmente tenha o menor consumo de energia, o método proposto tem o menor IAE, o que pode reduzir o IAE em 11,28% com apenas 1% a mais de energia em comparação com o sintonizado manualmente LQR.

Carvalho *et al.* (2022) propuseram a aplicação do controle por realimentação de estado utilizando o regulador linear quadrático (LQR) otimizado por metaheurísticas para amortecer oscilações eletromecânicas de baixa frequência em sistemas elétricos de potência. O modelo de sensibilidade foi usado para representar o sistema de barramento infinito de máquina única (SMIB) no domínio do tempo. As matrizes de ponderação, Q e R, do LQR foram ajustadas usando quatro algoritmos diferentes: o algoritmo genético (AG), algoritmo de evolução diferencial (DE), algoritmo de otimização de enxame de partículas (PSO) e algoritmo de otimização do lobo cinzento (GWO). Em cada caso os distúrbios foram aplicados ao sistema elétrico de potência e, em seguida, foram analisadas estatisticamente comparações de desempenho associadas a cada metaheurística. Com os resultados concluiu-se que os algoritmos foram eficientes no ajuste das matrizes de ponderação do LQR e o algoritmo GWO apresentou o melhor desempenho.

No trabalho de Hassani e Lee (2014) foi aplicado o algoritmo de otimização de enxame de partículas quânticas (QPSO) para ajustar de forma automática e otimizada as matrizes de ponderação Q e R do LQR. O QPSO é uma extensão do algoritmo PSO convencional, no qual as partículas obedecem à mecânica quântica em vez da mecânica newtoniana. Na pesquisa foi proposta uma abordagem para estabilizar um sistema de pêndulo invertido. Os resultados mostraram que o LQR baseado em QPSO supera o LQR ajustado por tentativa e erro, algoritmo genético e métodos PSO convencionais em termos de tempo de subida, tempo de acomodação e índice de desempenho quadrático. Além disso, foi competitivo com as abordagens mencionadas em termos de porcentagem máxima de ultrapassagem e estabilidade, e erro de estado.

No trabalho de Selamat *et al.* (2015) foi apresentado o estudo de sistemas de tanques acoplados utilizando LQR e controlador PID. Os parâmetros dos controladores foram ajustados usando PSO. O desempenho do sistema foi comparado com base na resposta transitória em termos de tempo de subida (T_r), tempo de estabilização (T_s), erro de estado estacionário (ess) e *overshoot* (OS). A simulação foi realizada no ambiente *Matlab*[®] para verificar o desempenho do sistema. Os resultados mostraram que ambos os controladores podem ser ajustados usando PSO, enquanto o controlador LQR oferece resultados melhores em comparação com o controlador PID.

No trabalho de Jing *et al.* (2020) foi apresentada uma modelagem e controle dinâmico recursivo para o manipulador de braço duplo com juntas elásticas. A abordagem de modelagem dinâmica foi baseada no método clássico recursivo de Newton-Euler e equações de movimento e força. A cinemática inversa de alta ordem que foi necessária para a modelagem do movimento do

braço. Para verificar a eficácia do método proposto, foi realizada a linearização por realimentação baseado no método de torque calculado baseado no modelo dinâmico das juntas rígidas para o controle baseado em modelo. Em Bueno *et al.* (2016), foi aplicado um controlador LQR e H^∞ em um manipulador de espaço flutuante com dois braços. O modelo nominal do robô foi usado para simular o controlador LQR projetado em um cenário sem perturbações e incertezas. Para simular o controlador H^∞ , foram considerados torques de perturbação e incertezas paramétricas. Os resultados das simulações foram analisados e demonstram que o LQR e o H^∞ tiveram desempenhos satisfatórios.

Em Dwivedi e Tayal (2017) um sistema de pêndulo invertido foi modelado principalmente por meio da formulação dinâmica de *Euler-Lagrange* (E-L) ou Newtoniana. A pesquisa utilizou o controlador LQR. Para encontrar os parâmetros do controlador LQR, AG e PSO foram usados para determinar e ajustar os parâmetros de controle do sistema. As simulações foram conduzidas usando *Matlab/Simulink*[®] em diferentes circunstâncias e o desempenho de cada técnica de controle foi analisado e comparado em termos de tempo de subida do sistema, tempo de estabilização, amplitude de pico e erro de estado estacionário. No trabalho de Asadi *et al.* (2016) foi utilizado um algoritmo ótimo de sinalização de movimento baseado em AG com LQR que pode gerar movimentos de alta fidelidade dentro das limitações físicas do simulador de movimento. Tanto a velocidade angular quanto a aceleração linear foram adotadas como entradas para o sistema de movimento. O método é usado para restringir o erro de percepção humana entre as tarefas de condução real e simulada.

No artigo de Grewal *et al.* (2012) foi utilizada uma estratégia de controle aplicada à uma plataforma *Stewart-Gough* acionada pneumaticamente com 6 graus de liberdade (6-DOF). A abordagem para o controle de movimento da plataforma foi apresentada usando uma técnica de controle linear quadrático Gaussiano LQG com rastreamento de referência. O controlador LQG é a combinação de um filtro de Kalman, ou seja, um estimador linear-quadrático (LQE) com um regulador linear-quadrático (LQR). A robustez do esquema de controle foi acessada sob várias condições de carga, e os resultados experimentais foram apresentados.

Na pesquisas de Lee *et al.* (2019) um controlador linear quadrático Gaussiano LQG baseado em modelo com matriz Q adaptativa foi proposto para projetar de forma eficiente e sistemática o controlador de rastreamento de caminho para qualquer veículo alvo, enquanto os algoritmos de planejamento lidavam efetivamente com os problemas de ruído e erro que surgiam na localização e no caminho, respectivamente. O regulador, por sua vez, foi projetado

automaticamente, sem esforços adicionais para sintonização em várias velocidades. Os resultados experimentais mostraram que o LQG proposto com matriz Q adaptativa teve desempenho melhor do que outros métodos convencionais de rastreamento.

Pode-se destacar aqui também que, pesquisas recentes têm sido realizadas com a aplicação do controlador LQG com metaheurísticas e técnicas de inteligência computacional (SINGH *et al.*, 2019). A seguir destacam-se algumas destas pesquisas. No artigo de Lin e Lin (2022) foi proposta uma estratégia de controle inteligente, que combina Linear-Quadrático-Gaussiano, algoritmo metaheurístico de otimização de baleia (*whale optimization algorithm - WOA*) e sistema de inferência neuro-fuzzy adaptativo (*adaptive neuro-fuzzy inference system - ANFIS*). Para a aplicação e comparação, também foram utilizados outros algoritmos metaheurísticos GA, evolução diferencial (*differential evolution - DE*) e colônia artificial de abelhas (*artificial bee colony - ABC*). Por meio de um teste chamado análise de variância na otimização do controle LQG, foi validada a superioridade do WOA sobre outros três algoritmos metaheurísticos, ou seja, GA, DE e ABC.

Remes *et al.* (2021) apresentaram um controlador LQG em malha e em cascata aplicado a um conversor direto de duas chaves (2SFC). Uma vez que as matrizes de ponderação do LQG possuem diversos parâmetros a serem selecionados, elas foram ajustados por PSO, em uma estratégia denominada LQG-PSO. Isso permitiu trabalhar intuitivamente com características complexas, como ruído, incertezas e saturação, apenas inserindo-as na simulação utilizada para PSO. Como o algoritmo é uma metaheurística, o trabalho apresentou uma análise de convergência do PSO para o problema do LQG. Na pesquisa, diferentes ensaios resultaram em ganhos de LQG em uma região bem definida, sugerindo que a otimalidade foi alcançada. Os resultados da simulação também mostraram que a estratégia LQG-PSO, que superou um controlador PI em cascata para diferentes cenários.

Benkhoud e Bouallègue (2018) realizaram a modelagem e ajuste avançado baseado em metaheurísticas de um controlador LQG para uma classe particular de veículos aéreos não tripulados (VANTs) conversíveis, denominado *Quad Tilt-Wing* (QTW). Foi derivado um modelo dinâmico usando o formalismo de Newton-Euler para os voos verticais, de transição e horizontais. Os principais parâmetros de projeto na abordagem de controle LQG são as matrizes de ponderação que geralmente são selecionadas por procedimentos clássicos e baseados em tentativas e erros, um problema de otimização sob restrições operacionais foi formulado para ajustar sistematicamente essas matrizes, no caso Q e R . Para resolver o problema do controlador

de LQG baseado em otimização, foram utilizadas metaheurísticas avançadas e recentes, como o algoritmo de busca harmônica, algoritmo do ciclo da água (WCA) e algoritmo memético baseado em otimização de enxame de partículas fracionárias (FPSOMA). Nos resultados da pesquisa foi verificada a eficiência das estruturas de controle LQG sintonizadas com metaheurísticas propostas.

1.3 Motivação e problemática

Com base na revisão da literatura realizada, muitas pesquisas ainda utilizam o algoritmo de mínimos quadrados recursivo na identificação de modelos, e também o algoritmo de MQR melhorado com outras técnicas, inclusive com metaheurísticas. E também pode-se afirmar que são utilizados controladores LQR/LQI e LQG com metaheurísticas.

Diante do que foi apresentado, este trabalho tem como finalidade identificar o modelo de uma junta de um manipulador robótico utilizando mínimos quadrados recursivo com otimização por enxame de partículas (*recursive least square with particle swarm optimization - RLSPSO*) e comparar com os algoritmos de mínimos quadrados não recursivo (MQ) e mínimos quadrados recursivo (MQR) clássico. Será realizada a identificação do modelo pelos algoritmos citados, e também será utilizado o modelo dinâmico (modelo fenomenológico) da junta da base do manipulador cilíndrico e estes modelos serão utilizados com os controladores LQR e LQG com as metaheurísticas AG e PSO.

Conforme destacado nos trabalhos da Seção 1.2, foram adotadas muitas técnicas de identificação com MQR com metaheurísticas ou técnicas de identificação e inteligência computacional (vide (SOUZA *et al.*, 2021b; ZHANG *et al.*, 2015; ZHA *et al.*, 2019; YUAN *et al.*, 2017)). Alguns trabalhos também utilizaram as metaheurísticas PSO e AG para a sintonia do controlador LQR/LQI (CARVALHO *et al.*, 2022; SOUZA *et al.*, 2020; ASADI *et al.*, 2016; SELAMAT *et al.*, 2015). Pode-se destacar também a utilização do modelo dinâmico de manipuladores para a aplicação do controlador (SWEVERIS *et al.*, 2007; JING *et al.*, 2020). Os algoritmos metaheurísticos também têm sido utilizados para otimizar os parâmetros do controlador LQG, como destacados em (LIN; LIN, 2022; REMES *et al.*, 2021; BENKHOUD; BOUALLÈGUE, 2018). Nestas pesquisas os algoritmos metaheurísticos foram usados para otimizar as matrizes Q e R do controlador LQG.

Visando o contexto da produção industrial, surge a necessidade de sistemas mais eficientes, mais produtivos, e com menos paradas para as manutenções programadas. Pretende-se,

portanto, com o estudo aqui realizado mostrar as contribuições da utilização do algoritmo de mínimos quadrados recursivo melhorado com otimização por enxame de partículas (*RLSPSO*), aplicado na identificação de uma junta de um manipulador robótico acionado por um motor de indução trifásico, além da utilização do modelo dinâmico do manipulador e aplicação dos controladores LQI e LQG otimizados com metaheurísticas AG e PSO aplicados aos modelos citados.

Esta pesquisa está direcionada para a identificação do modelo de uma junta de um manipulador cilíndrico, acionada por motor de indução trifásico, com *RLSPSO* e utilização do modelo dinâmico com dados experimentais, a partir de uma entrada *PRBS* (*pseudorandom binary sequence*). Os modelos serão representados na forma de espaço de estados, e pretende-se aplicar as estratégias de controle LQI e LQG otimizado com metaheurísticas AG e PSO, onde estas irão encontrar as matrizes Q e R otimizadas dos controladores citados. Pesquisas recentes mostram que esta tese apresenta contribuições no ponto de vista da aplicação de melhorias na identificação e controle, como a aplicação de metaheurísticas nos controladores LQI e LQG.

1.4 Objetivos e Contribuições

A seguir são apresentados o objetivo geral, os objetivos específicos, e as contribuições da tese.

1.4.1 Objetivo geral

O objetivo geral deste trabalho é realizar a identificação do modelo de uma junta de um manipulador acionado por motor de indução trifásico com algoritmo de mínimos quadrados recursivo com otimização por enxame de partículas (*RLSPSO*), e a implementação dos controladores LQI e LQG otimizado com as metaheurísticas AG e PSO aplicados ao modelo identificado.

1.4.2 Objetivos específicos

Os objetivos específicos deste trabalho são enumerados a seguir:

1. Utilizar os algoritmos de MQ e MQR para a identificação do modelo da junta do manipulador;
2. Implementar o algoritmo de mínimos quadrados recursivo com otimização por enxame de

partículas;

3. Comparar os algoritmos MQ, MQR e RLSPSO na identificação do modelo da junta do manipulador através das métricas $RMSE$ e coeficiente de correlação múltipla, R^2 ;
4. Desenvolver e calcular o modelo cinemático e dinâmico do manipulador;
5. Desenvolver os controladores LQI e LQG otimizados com as metaheurísticas AG e PSO;
6. Realizar uma comparação entre os controladores LQI e LQG, com relação ao tempo de processamento dos algoritmos e custo para encontrar as matrizes Q e R otimizadas;
7. Realizar uma comparação dos controladores LQI convencional e otimizados com AG e PSO aplicados aos modelos identificado e fenomenológico (dinâmico);
8. Realizar uma comparação dos controladores LQG convencional e o LQG otimizado com AG e PSO, aplicados aos modelos identificado e fenomenológico (modelo dinâmico do manipulador).

1.4.3 Contribuições da tese

Pode-se destacar que este trabalho apresenta as seguintes contribuições:

1. Melhoria do algoritmo de mínimos quadrados recursivo clássico com PSO. O algoritmo PSO é utilizado para otimizar a matriz de covariância do MQR, que é usada como função objetivo no algoritmo PSO;
2. Comparação dos algoritmos MQR com PSO (RLSPSO), com MQ e MQR clássico;
3. Implementação do modelo cinemático e dinâmico do manipulador em estudo;
4. Utilização do modelo dinâmico do manipulador com dados experimentais para a aplicação de controladores ao modelo;
5. A utilização de algoritmos com metaheurísticas para otimizar os controladores LQI e LQG submetidos a uma função custo;
6. Alguns ajustes nos parâmetros dos controladores, sem resultados adequados, podem ter um alto custo computacional ao mudar o ponto de operação da planta e este trabalho apresenta-se como uma alternativa para esse problema;
7. Otimização dos parâmetros (matriz Q e R) dos controladores LQI e LQG, onde serão encontrados os melhores valores;
8. Os parâmetros (Q e R) de ajuste dos controladores são baseados em uma função de custo ($J = \min(IAE + ISE)$);
9. Os controladores híbridos (LQI+GA, LQI+PSO, LQG+AG e LQG+PSO) tornam-se

eficientes no sentido de uma possível extensão do controle de um sistema mais robusto e com mais parâmetros de controle, como em um sistema multivariável;

10. Desenvolvimento de modelos e controladores, com a utilização de técnicas de inteligência computacional, através de metaheurísticas, aplicados a um manipulador robótico acionado por motores de indução.

1.5 Organização geral do trabalho

O trabalho está organizado em sete capítulos, os quais apresentam uma sequência que mostra gradualmente o estudo e o desenvolvimento do mesmo.

No Capítulo 2 é apresentada a descrição do manipulador robótico, a bancada experimental, a coleta dos dados utilizados na identificação e os métodos dos MQ, MQR e RLSPSO.

Já no Capítulo 3 são apresentados os resultados dos algoritmos de identificação e a avaliação (comparação destes) através do custo computacional, coeficiente de correlação múltipla (R^2), raiz do erro quadrático médio, *RMSE*, e da complexidade dos algoritmos.

No Capítulo 4 é apresentado o modelo matemático do manipulador cilíndrico, objeto deste trabalho. São apresentadas as equações que representam a cinemática direta e inversa, o Jacobiano, a dinâmica do manipulador (modelo fenomenológico), e por último é apresentado o modelo da junta 1 do manipulador em espaço de estados, identificado a partir da dinâmica e com dados experimentais.

No Capítulo 5 é apresentado o projeto dos controladores LQI e LQG, bem como o controlador com realimentação de estado com as metaheurísticas AG e PSO. É apresentada a função custo utilizada como métrica para comparação dos controladores.

No Capítulo 6 são apresentados os resultados dos controladores LQI e LQG convencionais e otimizados com as metaheurísticas, aplicados aos modelos da junta do manipulador identificado pelo *RLSPSO* e modelo dinâmico, bem como uma comparação entre estes.

Por fim, no Capítulo 7 são apresentadas as conclusões, as análises dos resultados, e por fim, as sugestões de trabalhos futuros.

2 IDENTIFICAÇÃO DO MODELO DA JUNTA DO MANIPULADOR

Este capítulo apresenta a bancada experimental, a descrição do experimento para a coleta dos dados e os métodos de identificação utilizados neste trabalho: mínimos quadrados não recursivo (MQ), mínimos quadrados recursivo (MQR) e o método de mínimos quadrados recursivo com otimização por enxame de partículas (*recursive least square with particle swarm optimization - RLSPSO*), além das métricas de avaliação dos algoritmos citados. Os métodos de identificação serão utilizados para determinar a função de transferência da junta da base do manipulador cilíndrico, ou seja, da junta rotacional.

A prática de algoritmos de identificação é interessante para muitas aplicações, como supervisão, diagnóstico, filtragem, predição, processamento de sinal, detecção e rastreamento de parâmetros variantes para controle adaptativo (COELHO; COELHO, 2004; LJUNG; SÖDERS-TRÖM, 1983; KJAER *et al.*, 1995). Este capítulo tem como principal objetivo apresentar os métodos de identificação mencionados anteriormente e utilizados para encontrar o modelo da junta 1 do manipulador.

2.1 Manipulador Robótico

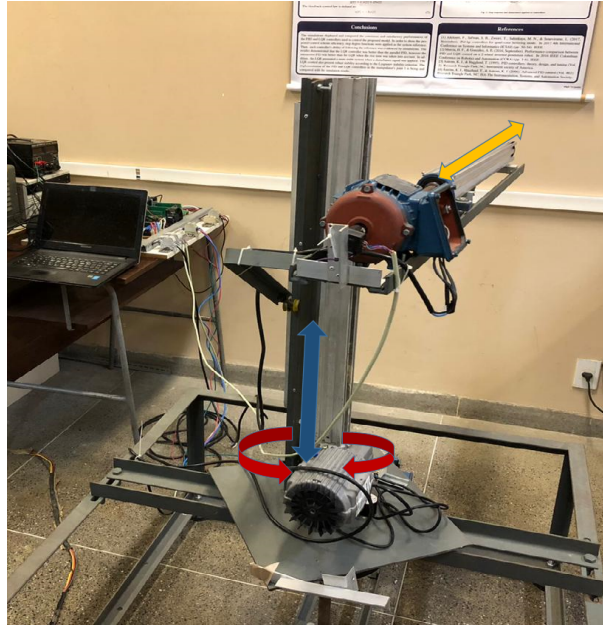
Neste trabalho é utilizado um manipulador robótico tipo cilíndrico acionado por motores de indução trifásicos. Como pode ser observado na Figura 2, a primeira articulação se move em torno do eixo principal da estrutura (movimento rotacional), a segunda e a terceira articulações possuem movimentos lineares (prismáticos), que se definem como um *Revolute-Prismatic-Prismatic* (RPP) (Rotacional-Prismático-Prismático). Os motores de indução trifásicos usados são do tipo gaiola de esquilo, cuja potência foi escolhida de forma que fosse possível movimentar cada junta do manipulador.

2.1.1 Bancada Experimental

O manipulador é acionado por motores de indução trifásico, como apresentado na Figura 2, do tipo gaiola de esquilo. A junta 1, que é de rotação (junta da base) do manipulador é acionada com um motor de 0,5 cv de potência nominal, tensão nominal de 220/380 V, 4 polos, com corrente nominal de 1,18 A e com ligação em delta (220 V). O motor foi dimensionado, através de cálculos, com a potência suficiente para movimentar toda a estrutura do manipulador.

Foi utilizado um processador digital de sinais, DSP (*Digital Signal Processors*), da

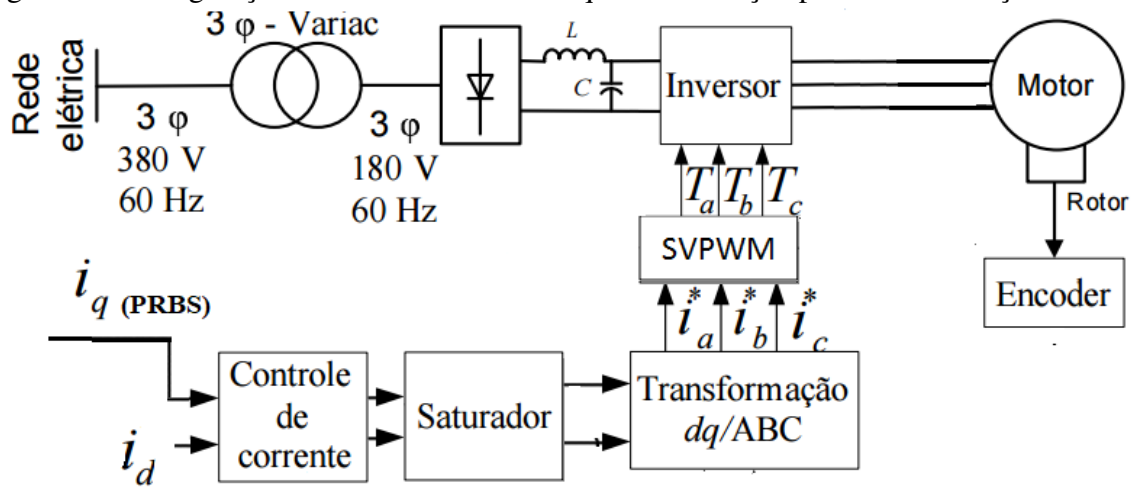
Figura 2 – Configuração do manipulador cilíndrico.



Fonte: Acervo do autor.

Texas Instruments[®], modelo *TMS320F28335*[®] para a coleta de dados e comunicação com o computador. A principal vantagem da utilização deste modelo se deve ao fato de que, além do alto desempenho, sendo capaz de executar 150 milhões de instruções por segundo, e possui o suporte de forma intrínseca à modulação em vetores espaciais (*space vector modulation based on pulse width modulation - SVPWM*). A Figura 3 apresenta o diagrama de blocos do sistema de acionamento da junta 1 através do controle de campo orientado (*field oriented control - FOC*), que representa o modelo utilizado no conversor CC/CA para o acionamento do MIT para a coleta de dados para a identificação da planta. Para mais detalhes ver no Apêndice B, Figura 70.

Figura 3 – Configuração do acionamento da máquina de indução para a identificação do modelo.

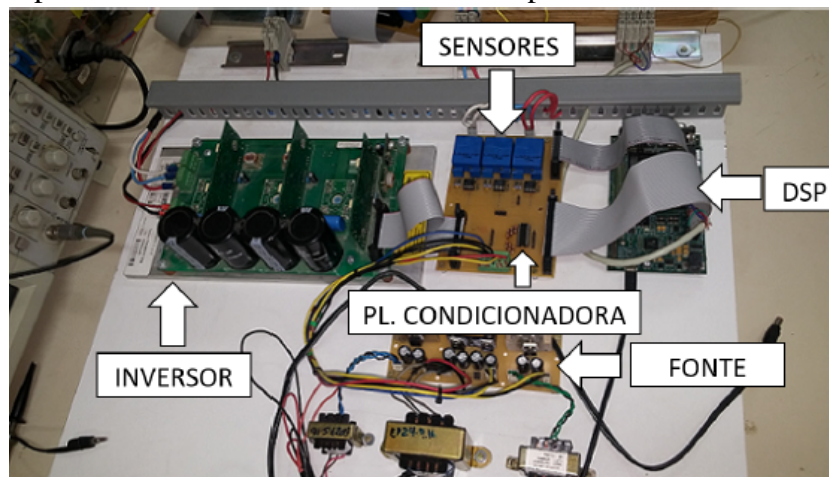


Fonte: Acervo do autor.

Quando o sistema está em operação, a corrente de campo, i_d , é escolhida de modo a manter o fluxo constante, neste caso foi utilizada a corrente, $i_d = 0,4 A$, de modo que o conjugado pode ser controlado pela mudança na corrente de armadura, i_q . Uma vez que i_d é desacoplada de i_q , uma corrente não influencia na outra.

Os outros dispositivos utilizados foram: sensores de corrente; inversor de frequência trifásico (comutação $2,5 kHz$) e um *encoder* de 400 pulsos/revolução para realizar a medição da velocidade da junta do manipulador. Na Figura 4 é apresentada a bancada experimental com os circuitos e placas, onde foi utilizada uma fonte de tensão auxiliar capaz de fornecer 4 níveis de tensão contínua. Sendo $18 V$ a tensão para chavear os gatilhos do inversor trifásico, $15 V$ e $-15 V$ a tensão para os sensores de corrente e $5 V$, a tensão para o painel de alimentação de condicionamento de sinal. A placa é a condicionadora que recebe os sinais de gatilho, corrente e sinais de posição/velocidade do *encoder* (REBOUÇAS *et al.*, 2017; RABELO L, 2017).

Figura 4 – Bancada experimental com os circuitos utilizados para controle e acionamento do manipulador.

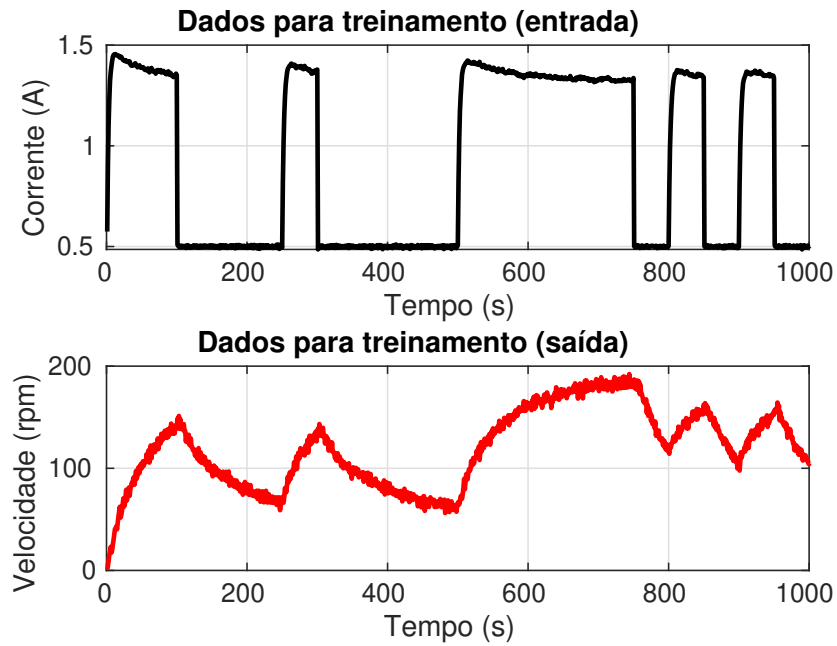


Fonte: Acervo do autor.

Dois experimentos foram realizados a fim de coletar os dados de corrente e velocidade. O tempo total de cada experimento é de $1000 s$ para que o primeiro conjunto de dados seja usado para estimar o modelo e o segundo para testá-lo. A referência utilizada na entrada do sistema foi um sinal *PRBS* (*pseudorandom binary sequence*) de corrente (i_q) para que a dinâmica da junta seja explorada.

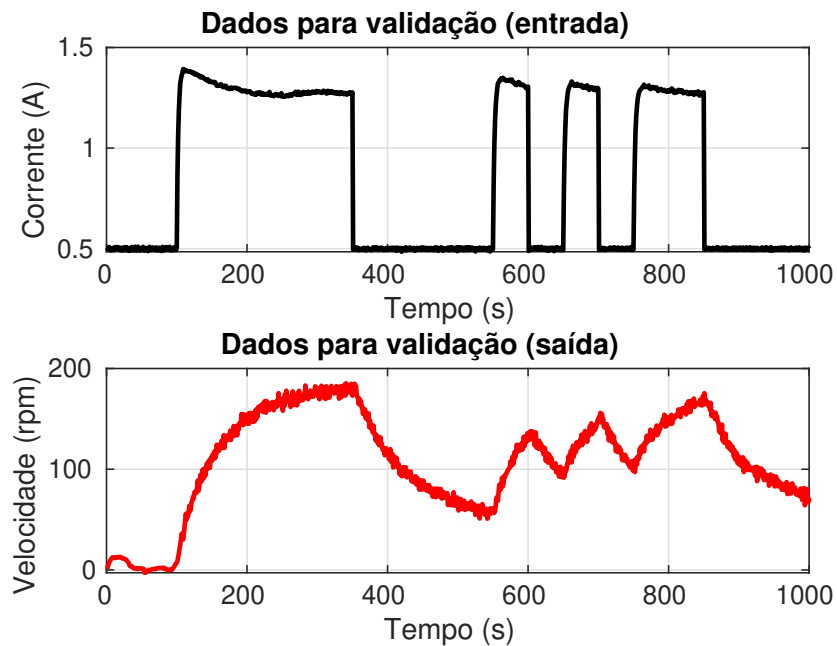
A Figura 5 apresenta os dados experimentais de entrada (corrente i_q) e saída (velocidade) para o estágio de treinamento dos algoritmos. A Figura 6 apresenta os dados utilizados na etapa de validação.

Figura 5 – Dados de entrada e saída experimentais da planta para o estágio de treinamento.



Fonte: Acervo do autor.

Figura 6 – Dados de entrada e saída experimentais da planta para o estágio de validação.



Fonte: Acervo do autor.

Como pode ser observado nas Figuras 5 e 6 foram realizados dois ensaios (um para validação e outro para treinamento com os *PRBS* diferentes. A ideia é verificar se os algoritmos estão convergindo de forma aceitável.

2.2 Representação em Tempo Discreto

Existem algumas representações matemáticas que são especialmente adequadas à identificação de sistemas usando algoritmos conhecidos para a estimação de parâmetros (AGUIRRE, 2007).

Considerando-se o seguinte modelo geral,

$$\begin{aligned} A(q)y(k) &= \frac{B(q)}{F(q)}u(k) + \frac{C(q)}{D(q)}v(k), \\ y(k) &= \frac{B(q)}{F(q)A(q)}u(k) + \frac{C(q)}{D(q)A(q)}v(k), \\ y(k) &= H(q)u(k) + G(q)v(k), \end{aligned} \quad (2.1)$$

sendo q^{-1} o operador de atraso, de forma que $y(k)q^{-1} = y(k-1)$, $v(k)$ ruído branco e $A(q)$, $B(q)$, $C(q)$, $D(q)$ e $F(q)$ são polinômios.

As funções $H(q)$ e $G(q)$ normalmente são referidas às funções de transferência do processo e do ruído, respectivamente. Ou seja, $H(q)$ é o resultado de substituir $q = z$ na transformada unilateral Z da resposta ao impulso do processo, $h(k)$ (AGUIRRE, 2007).

2.2.1 Modelo ARX

O modelo auto-regressivo com entradas externas (*ARX - autoregressive with exogenous inputs*) pode ser obtido a partir do modelo geral, (2.1), tomando-se $C(q) = D(q) = F(q) = 1$ sendo $A(q)$ e $B(q)$ polinômios arbitrários, resultando em:

$$A(q)y(k) = B(q)u(k) + v(k). \quad (2.2)$$

Uma vez que o ruído seja $v(k)$ aparece diretamente na equação, o modelo *ARX* é normalmente classificado como pertencente à classe de modelos de erro na equação. O modelo (2.2) pode ser reescrito como,

$$y(k) = \frac{B(q)}{A(q)}u(k) + \frac{1}{A(q)}v(k), \quad (2.3)$$

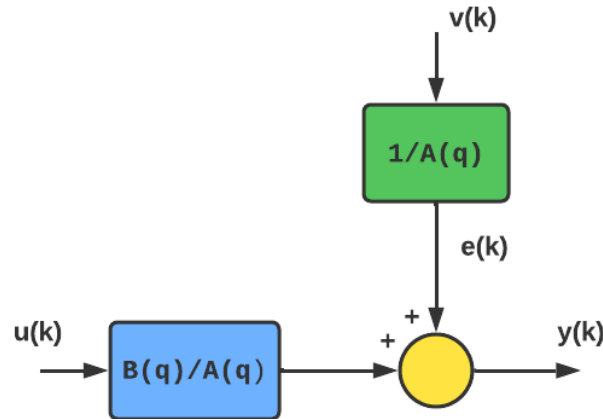
o que coloca em evidência as funções de transferências do sistema,

$$H(q) = \frac{B(q)}{A(q)} \quad (2.4)$$

e de ruído

$$\frac{C(q)}{D(q)A(q)} = \frac{1}{A(q)}, \quad (2.5)$$

Figura 7 – Representação esquemática do modelo ARX.



Fonte: Adaptado de Aguirre (2007).

conforme pode ser apresentado na Figura 7.

A representação na Figura 7 é equivalente à Equação (2.3), apresentada anteriormente.

Diferentemente do modelo *FIR* (*finite impulse response*), o ruído que aparece adicionado a saída,

$$e(k) = \frac{v(k)}{A(q)}, \quad (2.6)$$

não é branco. Ou seja, nesta representação o ruído é modelado como um processo branco filtrado por um filtro auto-regressivo, como polos idênticos aos do processo, que são as raízes do polinômio $A(q)$ (AGUIRRE, 2007; LJUNG; SÖDERSTRÖM, 1983).

2.3 Mínimos Quadrados não Recursivo

O método dos mínimos quadrados não recursivo - MQ (do inglês *Least Square - LS*) é um dos mais conhecidos e utilizados nas mais diversas áreas (AGUIRRE, 2007).

No processo de identificação da junta do manipulador foi caracterizado por uma entrada $u(t)$, uma saída $y(t)$, uma perturbação $e(t)$, e uma função de transferência discreta linear na forma (COELHO; COELHO, 2016):

$$A(z^{-1})y(t) = z^{-d}B(z^{-1}u(t) + e(t)). \quad (2.7)$$

Considerando o vetor de medidas, $\varphi(t)$, e o vetor de parâmetros, $\theta(t)$, a dimensão $(na + nb + 1) \times 1$, [e dada por]:

$$\varphi^T(t) = [-y(t-1) - y(t-2) \cdots - y(t-na)u(t-d) \cdots u(t-d-nb)], \quad (2.8)$$

e

$$\theta^T(t) = [a_1 \ a_2 \ \dots \ a_{na} \ b_0 \ b_1 \ \dots \ b_{nb}]. \quad (2.9)$$

A equação (2.9) pode ser reescrita como:

$$y(t) = \varphi^T(t)\theta(t) + e(t), \quad (2.10)$$

que é denominado de modelo de regressão linear (LJUNG; SÖDERSTRÖM, 1983).

Da Equação (2.10) obtém-se estimador dos mínimos quadrados não recursivo, para amostras variando de $t = [1 \rightarrow n]$ é encontrado por (COELHO; COELHO, 2016)

$$\hat{\theta}(t) = [\phi(t)^T \phi(t)]^{-1} \phi(t)^T Y(t), \quad (2.11)$$

2.4 Mínimos Quadrados Recursivo - MQR

A utilização de algoritmos de identificação é muito importante para vários propósitos, como supervisão, rastreamento de parâmetros variantes para controle adaptativo, filtragem, predição, processamento de sinais, detecção e diagnóstico (COELHO; COELHO, 2016). Vários métodos de identificação baseados em um conjunto de medidas são inadequados para a aplicação em tempo real. Com isso, deseja-se uma formulação eficiente dos algoritmos para proporcionar procedimentos eficazes (WELLSTEAD, 1991).

O método dos mínimos quadrados recursivo - MQR (*recursive least square - RLS*) é frequentemente aplicado quando é necessário encontrar um modelo do sistema em tempo real, enquanto o sistema está em operação (COELHO; COELHO, 2004; KJAER *et al.*, 1995).

O desenvolvimento das equações do estimador MQR pode ser obtido da Equação (2.11) dada por:

$$\left\{ \begin{array}{l} \hat{\theta}(t+1) = \hat{\theta} + K(t+1) \{y(t+1) - \varphi^T(t+1)\hat{\theta}(t)\}; \\ K(t+1) = \frac{P(t)\varphi(t+1)}{1 + \varphi^T(t+1)P(t)\varphi(t+1)}; \\ P(t+1) = P(t) - K(t+1)\varphi^T(t+1)P(t), \end{array} \right. \quad (2.12)$$

onde P é a matriz de covariância, φ é o vetor de regressão, K é o ganho do estimador. A matriz P e o vetor de parâmetros são inicializados com um valor muito próximo de zero. Com o decorrer das iterações, esses valores mudam e os parâmetros finais são obtidos de acordo com um melhor resultado.

O algoritmo do estimador dos MQR pode ser implementado como apresentado a seguir no Algoritmo 1.

Algoritmo 1: MQR.

- 1: **iniciar o algoritmo com as condições iniciais;**
 - 2: **atualizar o vetor de medidas:**
 - 3: **repita**
 - 4: $\varphi^T(t+1) = [-y(t) - y(t-1)u(t-d+1)u(t-d)];$
 - 5: **calcular o erro de predição:**
 - 6: $\varepsilon(t+1) = y(t+1) - \varphi^T(t+1)\hat{\theta}(t);$
 - 7: **calcular o ganho do estimador:**
 - 8: $K(t+1) = \frac{P(t)\varphi(t+1)}{1+\varphi^T(t+1)P(t)\varphi(t+1)};$
 - 9: **calcular o valor dos parâmetros estimados:**
 - 10: $\hat{\theta}(t+1) = \hat{\theta}(t) + K(t+1)\varepsilon(t+1)$
 - 11: **calcular a matriz de covariância:**
 - 12: $P(t+1) = P(t) - K(t+1)\varphi^T(t+1)P(t).$
-

Na etapa de treinamento do algoritmos dos MQR são obtidos os parâmetros de θ e P . Depois disso, os parâmetros são utilizados na etapa de validação do algoritmo.

Em situações mais exigentes, o número de iterações pode ser aumentado. No entanto, pode comprometer o tempo de execução do algoritmo.

2.5 MQR com Otimização por Enxame de Partículas - *Recursive Least Square with Particle Swarm Optimization - RLSPSO*

Particle Swarm Optimization (PSO) é uma metaheurística inspirada no comportamento social proposta por (EBERHART; KENNEDY, 1995). O objetivo principal do algoritmo é pesquisar em um determinado espaço através da permutação de dados das partículas, onde cada partícula será uma trajetória no espaço de busca. O PSO se destaca dos outros algoritmos pela convergência rápida. Similarmente a outros algoritmos de busca, o PSO pode ter partículas presas em localizações mínimas locais (PAIVA *et al.*, 2017).

A metaheurística PSO possui partículas semelhantes a um conjunto de pássaros que buscam a melhor forma de voar levando em consideração a posição e a velocidade de cada partícula. Uma curva de convergência é usada durante a execução do algoritmo. Cada

partícula terá seu objetivo resultante dependendo também do comportamento da população geral de partículas (PAIVA *et al.*, 2017). A posição no momento t é atualizada por $x_i(t)$ e no momento futuro $t + 1$ será dada por

$$x_i(t + 1) = x_i(t) + v_i(t + 1), \quad (2.13)$$

em que $v_i(t)$ é a velocidade (ENGELBRECHT, 2007). Cada partícula apresentará um componente cognitivo que será uma relação da distância entre ela e o melhor (solução ótima), além do componente social que é o entendimento do conjunto sobre a existência de uma dada partícula. Para este problema, usou-se o PSO global (melhor PSO global) em que a velocidade das partículas é atualizada por

$$v_{ij}(t + 1) = v_{ij}(t) + c_1 r_1(t) [y_{ij}(t) - x_{ij}(t)] + c_2 r_2(t) [\hat{y}_{ij}(t) - \hat{x}_{ij}(t)], \quad (2.14)$$

em que $v_{ij}(t)$ representa a velocidade da partícula em uma dada dimensão no tempo t . Novamente, c_1 e c_2 são os parâmetros de aceleração. A melhor informação de partícula é dada por \hat{x}_{ij} , x_{ij} , \hat{y}_{ij} e y_{ij} , é a melhor posição desde o início (ENGELBRECHT, 2007).

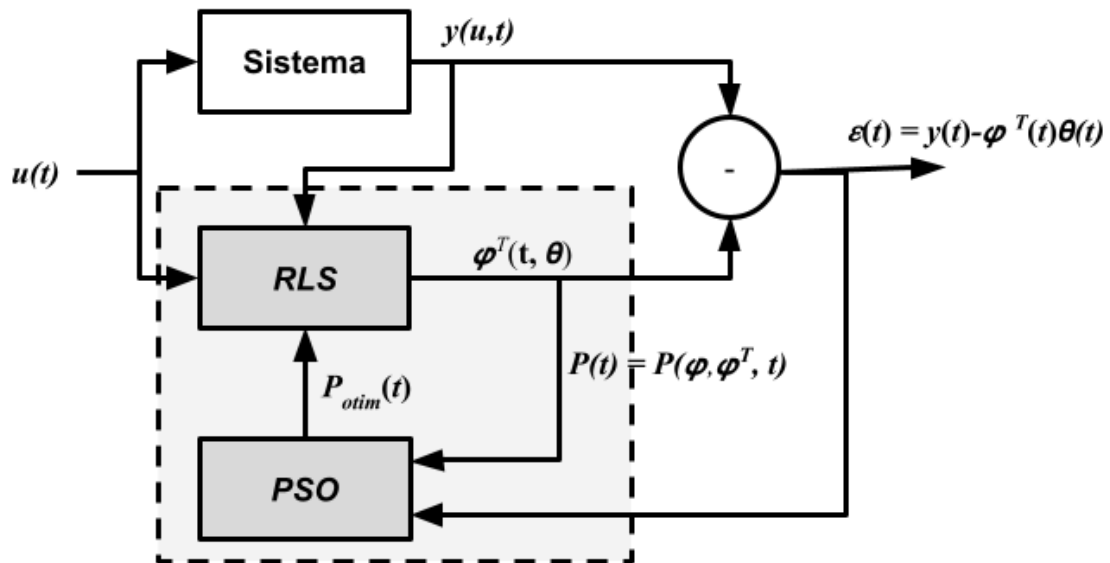
Ao contrário de outras técnicas de computação evolucionária, no PSO, cada partícula está associada a uma velocidade. As partículas voam pelo espaço de busca com velocidades que são ajustadas dinamicamente de acordo com seu comportamento histórico. Finalmente, as partículas tendem a percorrer as melhores áreas de pesquisa para uma solução durante o processo de busca (ENGELBRECHT, 2007).

Para o algoritmo PSO, os seguintes valores dos elementos foram usados:

- Número de partículas = 10 partículas;
- Razão de Amortecimento de Peso de Inércia = 0,99
- Parâmetros cognitivos e sociais (taxas de aprendizagem): $c_1 = 0,8$ e $c_2 = 1,0$;
- Iterações = 10 iterações;
- Fator de inércia (w) = 1,0;
- Geração inicial da população = utilizou-se de forma randômica em uma equação genérica que é restrita ao intervalo $[0,01;0,09]$.

Neste trabalho o algoritmo do PSO vai otimizar a matriz de covariância P do MQR (*RLSPSO*), para encontrar a melhor solução para o problema e minimizar a função objetivo (função de custo) do coeficiente de correlação múltipla. O diagrama de blocos da Figura 8 apresenta o comportamento geral do sistema de identificação *RLSPSO*.

Figura 8 – Representação esquemática do algoritmo de identificação *RLSPSO*.



Fonte: Acervo do autor.

O pseudocódigo do algoritmo *RLSPSO* pode ser implementado com base em algumas etapas apresentadas no Algoritmo 2, a seguir.

Algoritmo 2: *RLSPSO*.

- 1: **inicializa a nuvem de partículas;**
 - 2: **repita**
 - 3: **para** $i = 1$ **até** m
 - 4: **se** $f(x_i) < f(p_i)$ **então**
 - 5: $p_i = x_i$;
 - 6: **se** $f(x_i) < f(g)$ **então**
 - 7: $g = x_i$;
 - 8: **fim se**
 - 9: **fim se**
 - 10: **para** $j = 1$ **até** n
 - 11: $r_1 = rand(), r_2 = rand()$;
 - 12: $v_{ij} = wv_{ij} + c_1r_1(p_i - x_{ij}) + c_2r_2(g_j - x_{ij})$;
 - 13: **fim para**
 - 14: $x(i) = x(i-1) + v(i)$;
 - 15: **fim para**
 - 16: **até satisfazer o critério de parada:**
 - 17: **melhor valor da matriz de covariância, P , avaliado por:**
 - 18: $J_{min} = 1 - mean(R^2)$.
-

A metaheurística PSO tem como missão minimizar a função objetivo dada pela equação (2.15), com o número de iterações utilizado no algoritmo igual a 10, e o algoritmo foi

executado 10 vezes para obtenção do melhor resultado

$$J_{min} = 1 - \text{mean}(R^2), \quad i = 1, 2, 3, \dots, n, \quad (2.15)$$

em que R^2 é o coeficiente de correlação múltipla.

Para o algoritmo *RLSPSO* na etapa de treinamento, foram obtidos os parâmetros θ e P . Estes parâmetros foram utilizados na etapa de validação do algoritmo *RLSPSO*.

2.6 Métricas de Avaliação dos Algoritmos de Identificação

Ao realizar-se uma medição de uma grandeza, não importa o quão confiável ela seja, deve-se estabelecer uma relação entre o valor verdadeiro e o valor medido. O erro é a diferença entre estes valores. Os desvios ou erros de medidas são utilizados para verificar a precisão de um algoritmo, por exemplo (ARNOT, 2002).

Para isso é necessário se utilizar de métricas de avaliação para validar as relevâncias estatísticas de um algoritmo. Uma forma de avaliação é verificar através de gráficos se os resultados preditos estão próximos dos resultados reais. No entanto, mesmo que o resultado pareça visualmente dentro dos padrões exigidos, isso não mostra que o resultado é estatisticamente relevante, ou seja, mesmo que os gráficos estejam visivelmente perfeitos, as métricas podem indicar que a modelagem não foi tão promissora quanto parece (GARETH *et al.*, 2013).

As métricas apresentadas a seguir serão aplicadas para avaliar os algoritmos utilizados para encontrar o modelo identificado, comparado com os dados de identificação reais medidos com a aplicação do sinal *PRBS*.

2.6.1 Coeficiente de Correlação Múltipla - R^2

Para avaliação de algoritmos de regressão numérica, o coeficiente de correlação múltipla (R^2) é uma das métricas mais utilizadas e eficientes, indicando a proporção da variância dos valores preditos que pode ser explicada utilizando os valores de entrada como parâmetros (GARETH *et al.*, 2013). Matematicamente, essa métrica pode ser escrita como

$$R^2 = 1 - \frac{\sum_{i=1}^N (y(i) - \hat{y}(i))^2}{\sum_{i=1}^N (y(i) - \bar{y}(i))^2}, \quad (2.16)$$

em que:

- N é o número de amostras;

- $y_{(i)}$ são as observações;
- $\hat{y}_{(i)}$ são as predições;
- $\bar{y}_{(i)}$ é a média das observações.

2.6.2 Raiz do Erro Quadrático Médio - RMSE

A raiz do erro quadrático médio - *RMSE* (*root mean squared error*) é a medida que calcula a raiz do erro quadrático médio entre valores observados (reais) e predições (hipóteses). O *RMSE* possui uma dimensão igual à dimensão dos valores observados e preditos. Seu valor pode ser interpretado como uma medida do desvio médio entre o observado e o predito (GARETH *et al.*, 2013),

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^n (y_i - \hat{y}_i)^2}, \quad (2.17)$$

em que:

- N é o número de amostras;
- $y_{(i)}$ são as observações;
- $\hat{y}_{(i)}$ são as predições.

2.6.3 Complexidade do algoritmo

A complexidade de uma algoritmo pode ser verificada através do comportamento assintótico do algoritmo, ou seja: a complexidade expressa uma tendência a um limite à medida que cresce o tamanho do problema (n).

Pode-se levar em consideração a quantidade de operações matemáticas e laços, aninhados ou não, no processamento do algoritmo. Dessa forma, pode-se levar a outra simplificação: ao se ter uma expressão polinomial $C(n)$, que representa a complexidade do algoritmo, como os termos de menor grau podem ser desprezados (quando n é grande) diante do termo de maior grau, este é que será adotado como aproximação (WILF, 2002).

Uma boa maneira de tratar o cálculo da complexidade de algoritmos é categorizando-os como sendo $C(n)$, $C(n \log n)$, $C(n^2)$ ou $C(n^3)$. Estes são conceitos que descrevem exatamente quantos passos são necessários para que um algoritmo execute suas tarefas. $C(n)$, refere-se a um algoritmo que executa ações com um número de passos que é diretamente proporcional ao número de itens de entrada. $C(n \log n)$ se refere a um algoritmo que realiza ações com um número de passos que é proporcional ao logaritmo da quantidade de itens de entrada. $C(n^2)$

se refere a um algoritmo que executa ações com um número de passos que é proporcional ao quadrado do número de itens

Pode-se utilizar as seguintes as complexidades (em ordem crescente):

- $C(1)$ ou constante;
- $C(\log n)$ ou logarítmica;
- $C(n)$ ou linear;
- $C(n \log n)$ ou $n \log de n$;
- $C(n^2)$ ou quadrática;
- $C(n^3)$ ou cúbica.

Quando tem-se $C(n^2)$, por exemplo, significa que a complexidade de um algoritmo com dois trechos aninhados, em que o segundo é repetidamente executado pelo primeiro, é dada como o produto da complexidade do trecho mais interno pela complexidade do trecho mais externo (KOŚCIELSKI; PACHOLSKI, 1996).

Regras rígidas sobre o cálculo da complexidade de qualquer algoritmo não existem, cada caso deve ser estudado em suas condições. No entanto, as estruturas de controle clássicas da programação estruturada permitem uma estimativa típica de cada uma. A partir disso, algoritmos construídos com combinações delas podem ter sua complexidade mais facilmente estabelecida.

2.7 Considerações Finais

Este capítulo apresentou as características do manipulador robótico 3-GDL acionado por motores de indução trifásico. Também foram apresentados os algoritmos dos MQ, MQR e MQR com PSO (*RLSPSO*), e como estes foram implementados para a identificação. Para cada algoritmo utilizados na identificação do modelo da junta 1 do manipulador, foi encontrada uma função de transferência, que foram descritas na forma de tempo discreto discreto e os parâmetros P e θ de cada algoritmo, além dos seus pseudocódigos.

Por fim, e não menos importante, foram apresentadas as métricas de avaliação utilizadas para aferir os algoritmos de identificação aqui apresentados. As métricas foram: o coeficiente de correlação múltipla, R^2 ; e a raiz do erro quadrático médio, *RMSE*.

No próximo capítulo são apresentados os resultados da identificação do modelo através dos algoritmos de MQ, MQR e *RLSPSO*.

3 RESULTADOS DA IDENTIFICAÇÃO DO MODELO

Neste capítulo são apresentados os resultados dos algoritmos MQ, MQR e MQR com PSO - *RLSPSO*, utilizados na identificação do modelo da junta. Os resultados apresentam as curvas de treinamento e validação, as curvas de erro para o treinamento e validação, a função de transferência discreta e a avaliação (comparação) através do tempo de processamento, coeficiente de correlação múltipla (R^2), a raiz do erro quadrático médio, *RMSE*, e a complexidade, $C(n)$, para cada algoritmo. Também é apresentado o modelo identificado pelo algoritmo *RLSPSO*, na forma de função de transferência e espaço de estado para a junta do manipulador.

Conforme mencionado na Seção 2.1 foram utilizados um sinal de entrada de corrente *PRBS* e um sinal de saída de velocidade para treinamento e validação dos algoritmos de MQ, MQR e *RLSPSO*.

3.1 Resultados do MQ

A identificação com MQ é apresentada na Figura 9. Na parte superior da figura é apresentada a etapa de treinamento, onde são obtidos P e θ do algoritmo. Estes parâmetros de treinamento foram usados com os dados experimentais do ensaio para validação, que está ilustrado na parte inferior da Figura 9. Os dados são referentes à velocidade da junta 1 do manipulador.

Na Figura 10 são apresentadas as curvas de erro normalizado para o treinamento e validação do algoritmo de mínimos quadrados para os valores de velocidade.

O vetor de parâmetros θ estimado no treinamento e usado no teste de validação do algoritmo dos MQ foi o seguinte

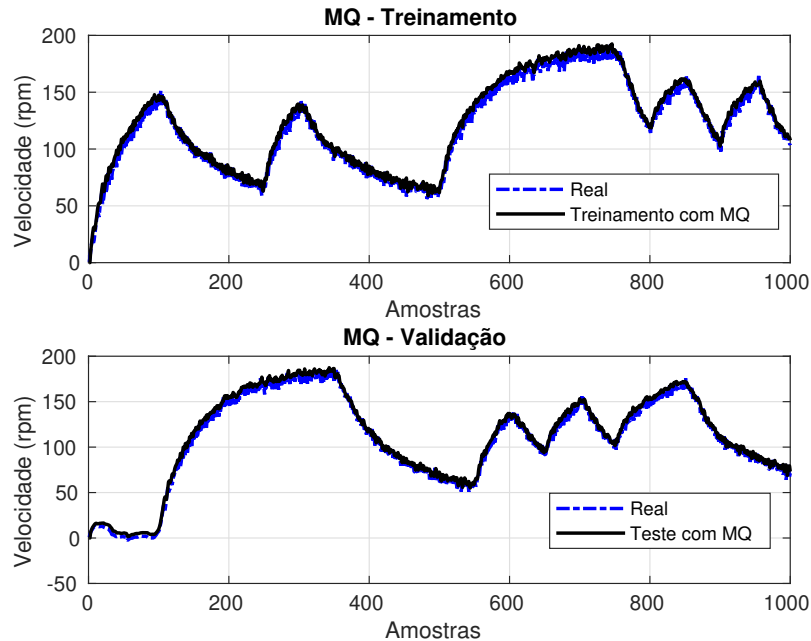
$$\theta = \left[-0,3771 \quad -0,5939 \quad 1,7151 \quad 2,1975 \right]^T. \quad (3.1)$$

A função de transferência de tempo discreto gerada com o modelo de treinamento dos MQ, com um período de amostragem de 0,2 s, pois é o tempo para a resposta entre o sistema elétrico e mecânico, é apresentada como

$$G(z) = \frac{1,715z + 2,198}{z^2 - 0,3771z - 0,5939}. \quad (3.2)$$

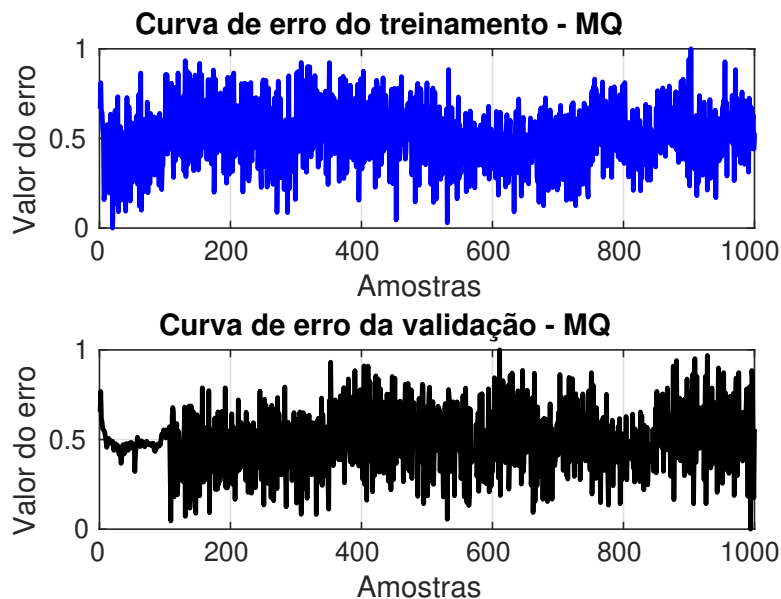
A função de transferência discreta apresentada em (3.2) pode ser utilizada como modelo da planta identificada pelos MQ e para projeto do controlador.

Figura 9 – Curvas de treinamento e validação (teste) para identificação com MQ, para os dados experimentais de velocidade.



Fonte: Acervo do autor.

Figura 10 – Curvas de erro do treinamento e da validação (teste) para o MQ.



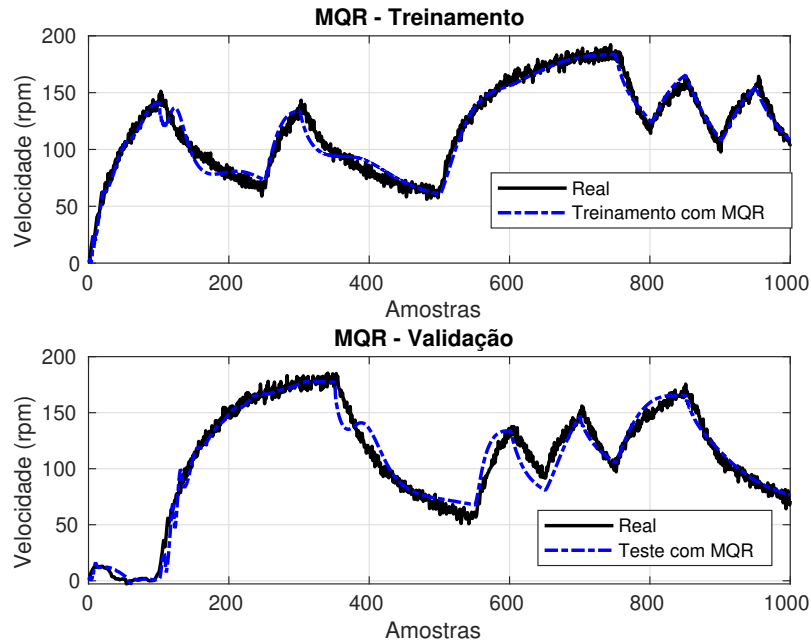
Fonte: Acervo do autor.

3.2 Resultados do MQR

Da mesma forma, na Figura 11 é apresentado o resultado da identificação com MQR. Na parte superior da figura é apresentada a etapa de treinamento, onde são obtidos os parâmetros P e θ do algoritmo. Estes parâmetros de treinamento foram usados com os dados experimentais

do ensaio para validação, que está apresentado na parte inferior da Figura 11. Da mesma forma, os dados são referentes à velocidade da junta 1 do manipulador.

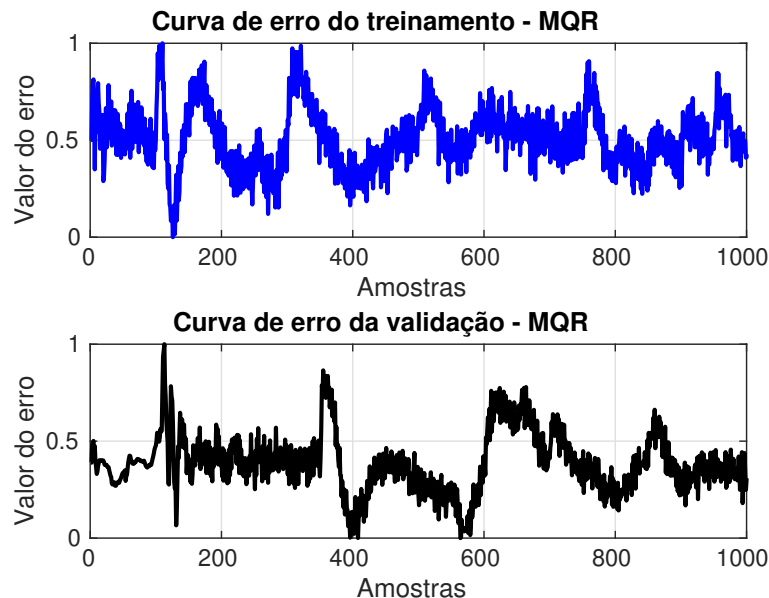
Figura 11 – Curvas de treinamento e validação (teste) para identificação com MQR, para os dados experimentais de velocidade.



Fonte: Acervo do autor.

Na Figura 12 são apresentadas as curvas de erro normalizado para o treinamento e validação do algoritmo de mínimos quadrados recursivo, para os valores de velocidade.

Figura 12 – Curvas de erro do treinamento e da validação (teste) para o MQR.



Fonte: Acervo do autor.

A matriz P obtida no algoritmo MQR por busca exaustiva, pode ser escrita como

$$P = \begin{bmatrix} 0,0009 & -0,0009 & 0,0003 & 0,0018 \\ -0,0009 & 0,0009 & -0,0003 & -0,0017 \\ 0,0003 & -0,0003 & 0,2180 & -0,2177 \\ 0,0018 & -0,0017 & -0,2177 & 0,2284 \end{bmatrix}, \quad (3.3)$$

e o vetor de parâmetros θ gerado no treinamento e usado no teste de validação, é escrito como segue

$$\theta = [-0,1121 \quad -0,8558 \quad 5,8545 \quad -1,6266]^T. \quad (3.4)$$

A função de transferência de tempo discreto, gerada com o modelo de treinamento do MQR, é apresentada a seguir. O tempo de amostragem foi de 0,2 s:

$$G(z) = \frac{5,8545z - 1,6266}{z^2 - 0,1121z - 0,8558}. \quad (3.5)$$

A função de transferência discreta apresentada em (3.5) pode ser utilizada como modelo da planta identificada pelo MQR e para projeto do controlador.

3.3 Resultados do MQR com PSO - *RLSPSO*

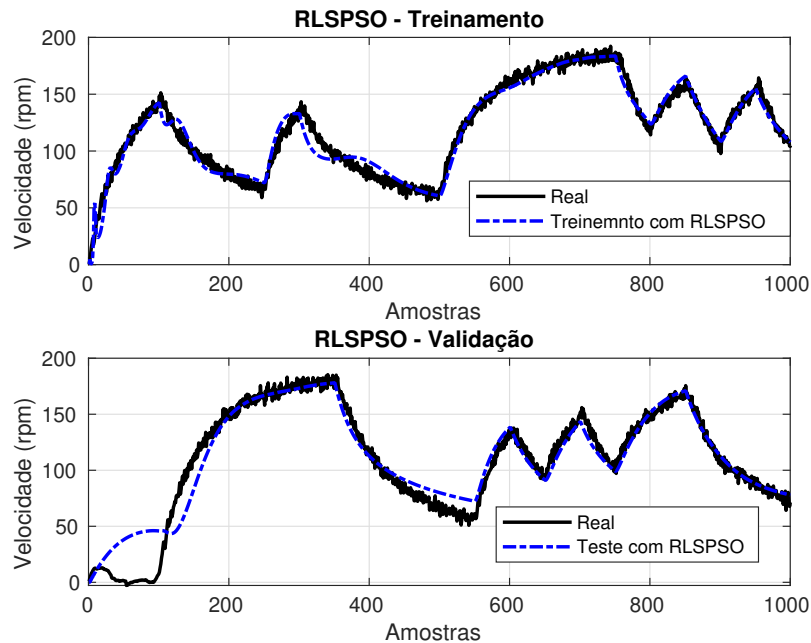
Na Figura 13 é apresentado o resultado da identificação com o *RLSPSO*. Na parte superior da figura é apresentada a etapa de treinamento onde são obtidos os parâmetros P e θ do algoritmo. Estes parâmetros de treinamento foram usados com os dados experimentais de velocidade da junta 1, do ensaio para validação, que está apresentado na parte inferior da Figura 13.

Na Figura 14 são apresentadas as curvas de erro normalizado para o treinamento e validação do algoritmo de mínimos quadrados recursivo com *PSO (RLSPSO)*, para os valores de velocidade.

A matriz P obtida no algoritmo *RLSPSO* foi

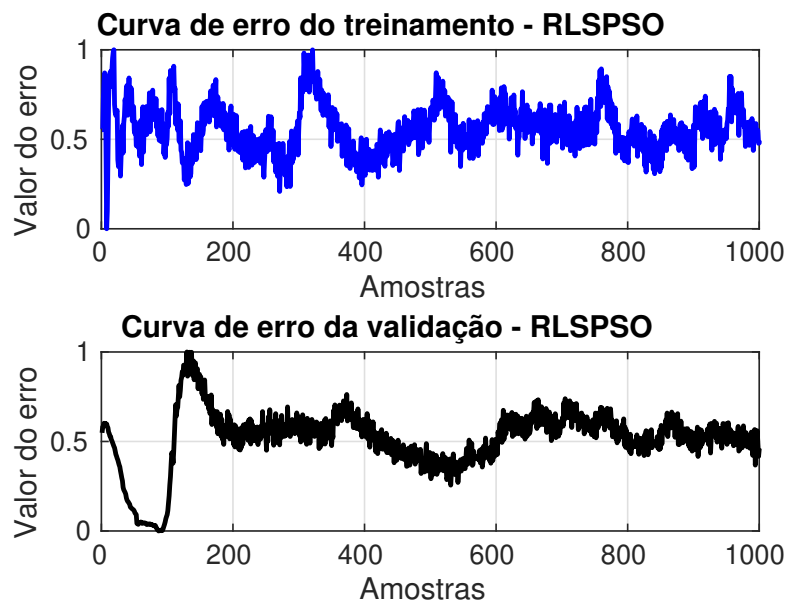
$$P = \begin{bmatrix} 0,0007 & -0,0006 & 0,0004 & 0,0010 \\ -0,0006 & 0,0006 & -0,0004 & -0,0009 \\ 0,0004 & -0,0004 & 0,0239 & -0,0221 \\ 0,0010 & -0,0009 & -0,0221 & 0,0265 \end{bmatrix}. \quad (3.6)$$

Figura 13 – Curvas de treinamento e validação (teste) para identificação com *RLSPSO*, para os dados experimentais de velocidade.



Fonte: Acervo do autor.

Figura 14 – Curvas de erro do treinamento e da validação (teste) para o *RLSPSO*.



Fonte: Acervo do autor.

O vetor de parâmetros θ gerado no treinamento e usado no teste de validação é escrito como segue

$$\theta = \left[-0,1851 \quad -0,7814 \quad 1,4814 \quad 3,2751 \right]^T. \quad (3.7)$$

A função de transferência discreta gerada com o modelo de treinamento do algoritmo

RLSPSO é apresentada a seguir. Novamente, o tempo de amostragem foi de 0,2 s.

$$G(z) = \frac{1,481z + 3,275}{z^2 - 0,1851z - 0,7814}. \quad (3.8)$$

Para o algoritmo *RLSPSO* a função de transferência discreta apresentada na equação (3.8) também pode ser utilizada como modelo identificado da planta e para projeto do controlador.

3.4 Comparação dos Algoritmos

Nesta seção é realizada uma análise comparativa mais detalhada dos diferentes algoritmos. Foi realizada uma análise qualitativa de cada método na identificação (treinamento) da velocidade da junta 1 do manipulador, com resultados evidenciados na Tabela 1 por meio dos índices de desempenho: tempo de processamento, coeficiente de correlação múltipla (R^2) e a raiz quadrada do erro médio (*RMSE* - *root mean squared error*) de cada algoritmo.

Tabela 1 – Comparação do tempo de processamento e as métricas R^2 e *RMSE* para a identificação com os métodos MQ, MQR e *RLSPSO*.

Método	Tempo de Processamento (s)	R^2	<i>RMSE</i>
MQ	2,8168	0,9709	0,1760
MQR	42,1222	0,9656	0,1921
<i>RLSPSO</i>	36,8397	0,9962	0,1546

Fonte: Acervo do autor.

Como pode ser verificado na Tabela 1, o algoritmo de mínimos quadrados recursivo com o PSO (*RLSPSO*) apresentou melhor resultado que o MQR clássico, pois seu tempo de processamento foi de 36,8397 s em comparação com os 42,1222 s para o MQR. Com relação ao coeficiente de correlação múltipla, R^2 , o *RLSPSO* foi de 0,9962, enquanto o MQR o valor foi de 0,9656. E para o *RMSE*, o valor foi de 0,1546 para o *RLSPSO* e 0,1921 para o MQR. O algoritmo de MQ obteve o melhor resultado no tempo de processamento que os demais, mas em relação ao R^2 e *RMSE* o *RLSPSO* foi melhor que o MQ.

Uma comparação importante, dos algoritmos também pode ser apresentada em relação à complexidade, $C(n)$, dos algoritmos MQ, MQR e *RLSPSO*. O número de instruções para descobrir a complexidade do algoritmo é muito importante para se ter uma base teórica mais detalhada. A Tabela 2 apresenta a complexidade de cada algoritmo por valores das amostras de entrada para a identificação com cada métodos.

Tabela 2 – Complexidade, $C(n)$, dos algoritmos para a identificação com os métodos MQ, MQR e *RLSPSO*.

Método (Algoritmo)	Complexidade do Algoritmo $C(n)$
MQ	$8n + 12$
MQR	$8n^2 + 8n + 15$
<i>RLSPSO</i>	$6n^2 + 6n + 15$

Fonte: Acervo do autor.

Em uma comparação teórica, o algoritmo *RLSPSO* mantém uma complexidade menor ao contar o número de instruções como visto na Tabela 2, o MQR tem mais instruções no algoritmo e o MQ apresentou o menor número de instruções. O Algoritmo *RLSPSO* é melhor que o MQR convencional, pois apresenta uma complexidade $C(n)$ menor.

De acordo com as comparações realizadas de cada algoritmo, o *RLSPSO* apresentou o melhor resultado. Diante disto será utilizada a função de transferência apresentada na Equação (3.8) como modelo da planta identificada e a mesma será utilizada no projeto do controlador deste trabalho.

3.5 Modelo da Junta do Manipulador

O algoritmo de identificação *RLSPSO* apresentou o melhor resultado de acordo com as comparações realizadas. Diante disso, será utilizado neste trabalho o modelo identificado por esse algoritmo para o projeto dos controladores LQI e LQG com as metaheurísticas AG e PSO. A seguir são apresentados a função de transferência contínua, e o modelo em espaço de estados da planta identificada.

3.5.1 Função de transferência da junta do manipulador

A função de transferência na forma discreta linear apresentada anteriormente, (3.8), identificada pelo algoritmo de identificação *RLSPSO* é da forma

$$G(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}. \quad (3.9)$$

A equação (3.8), que é a função de transferência discreta, identificada, pode ser convertida para a forma contínua e escrita como

$$G(s) = \frac{b_2 s^2 + b_1 s + b_0}{s^3 + a_2 s^2 + a_1 s + a_0}, \quad (3.10)$$

com isso pode-se apresentar, a seguir, a função de transferência contínua, identificada pelo algoritmo *RLSPSO*

$$G(s) = \frac{13,01s^2 - 51,42s + 3301}{s^3 + 2,184s^2 + 248s + 2324}. \quad (3.11)$$

3.5.2 Modelo em espaço de estados da junta

A representação do modelo no espaço de estado é um sistema de duas equações, em que A , B , C e D são matrizes que descrevem o estado, entrada, saída e passagem direta, respectivamente (ZADEH; DESOER, 2008; CHEN *et al.*, 2004).

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (3.12)$$

e

$$y(t) = Cx(t) + Du(t). \quad (3.13)$$

Existem algumas formas para realizar a transformação de um sistema representado por uma função de transferência para a forma espaço de estados. O vetor de estados não é único para um sistema dinâmico, ou seja, existem inúmeras possibilidades de definir o vetor de estado para um mesmo sistema dinâmico, pois cada vetor de estado vai gerar uma forma diferente de espaço de estados para uma mesma função de transferência (OGATA, 2010).

Para realizar a transformação de função de transferência para espaço de estados utilizou-se o caso onde a ordem do numerador < ordem do denominador

$$G(s) = \frac{b_0s^m + b_1s^{m-1} + \dots + b_{n-1}s + b_n}{s^n + a_1s^{n-1} + \dots + a_{n-1}s + a_n}, \quad (3.14)$$

com $m < n$.

Considerando o sistema apresentado na Equação (3.11), que é um sistema de 3ª ordem tem-se

$$G(s) = \frac{b_2s^2 + b_1s + b_0}{s^3 + a_2s^2 + a_1s + a_0}. \quad (3.15)$$

Assim a equação na forma canônica controlável é da forma

$$\dot{x}(t) = -a_2x_1(t) - a_1x_2(t) - a_0x_3(t) + u(t) \quad (3.16)$$

daí

$$\dot{x}(t) = \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} = \begin{bmatrix} -a_2 & -a_1 & -a_0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u(t) = Ax(t) + Bu(t). \quad (3.17)$$

O vetor de saída é dado por

$$y(t) = \begin{bmatrix} b_2 & b_1 & b_0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} = Cx(t). \quad (3.18)$$

e

$$D = [0]. \quad (3.19)$$

Após realizar as substituições adequadas, a partir da equação (3.11), pode-se representar o modelo da junta do manipulador em espaço de estados:

$$A = \begin{bmatrix} -2,184 & -248 & -2324 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad (3.20)$$

$$C = \begin{bmatrix} 13,010 & -51,420 & 3310 \end{bmatrix} \text{ e } D = \begin{bmatrix} 0 \end{bmatrix}.$$

3.6 Considerações Finais

Este capítulo apresentou os resultados dos algoritmos MQ, MQR e *RLSPSO*, utilizados na identificação do modelo da junta, onde foram apresentadas as curvas de treinamento e validação, as curvas de erro para o treinamento e validação e a função de transferência discreta. Também foram apresentados os resultados da avaliação (comparação) através do tempo de processamento, coeficiente de correlação múltipla (R^2) e a raiz do erro quadrático médio, *RMSE*, para cada algoritmo, onde o algoritmo *RLSPSO* apresentou melhor desempenho.

Por fim, e não menos importante, foi apresentada uma comparação dos algoritmos através da complexidade, $C(n)$, em que o *RLSPSO* mostrou-se menos complexo que o algoritmo de MQR.

No próximo capítulo é apresentado o modelo cinemático e dinâmico do manipulador em estudo, bem como o modelo em espaço de estados da junta 1 do manipulador, encontrado a partir da dinâmica e dos dados experimentais.

4 MODELO CINEMÁTICO E DINÂMICO DO MANIPULADOR

Neste capítulo é apresentado o modelo matemático do manipulador cilíndrico, objeto deste trabalho. São apresentadas as equações que representam a cinemática direta e inversa, a matriz Jacobiana, e a dinâmica do manipulador. A cinemática direta será encontrada através da convenção Denavit-Hartenberg D-H, a cinemática inversa será determinada pelo método geométrico, a matriz Jacobiana será determinado através das velocidades lineares e angulares, e por último, a dinâmica será calculada utilizando as equações de movimento de *Euler-Lagrange*. Por fim, é apresentado o modelo da junta identificada pela dinâmica do manipulador utilizando os dados experimentais.

4.1 Cinemática do Manipulador Cilíndrico

O modelo cinemático estabelece quais são as trajetórias que devem ser seguidas por cada articulação do robô num intervalo de tempo para atingir os objetivos fixados pelo usuário (ponto de destino, trajetória cartesiana da ferramenta do robô, tempo gasto pelo usuário, etc.). Estas trajetórias são selecionadas atendendo às restrições físicas próprias dos acionamentos e a certos critérios de qualidade da trajetória, como suavidade ou precisão da mesma (ANTONIO *et al.*, 1997).

4.1.1 Cinemática Direta do Manipulador Cilíndrico

A Figura 15 apresenta a estrutura do manipulador cilíndrico em estudo e os parâmetros D-H são apresentados na Tabela 3, parâmetros estes que representam as variáveis das juntas do manipulador (ver Apêndice A.2).

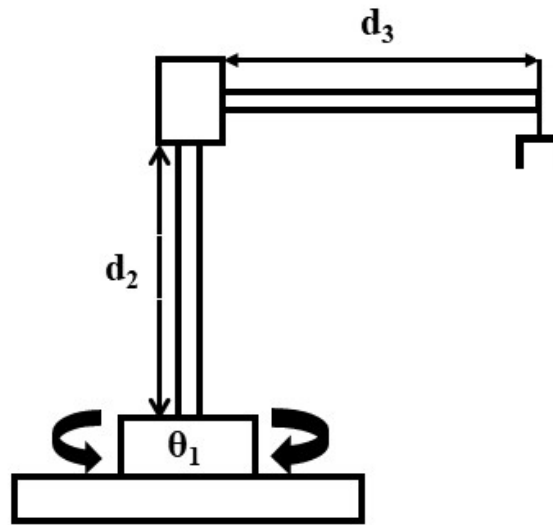
Tabela 3 – Parâmetros D-H do manipulador cilíndrico.

Junta	a_i	α_i	d_i	θ_i
1	0	0	0,245	θ_1^*
2	0,11	-90°	d_2^*	0
3	0	0	d_3^*	0

Fonte: o autor.

Em que * representa os parâmetros que são variáveis de acordo com a posição do manipulador.

Figura 15 – Manipulador cilíndrico.



Fonte: Acervo do autor.

Aplicando-se a conversão D-H a partir dos parâmetros apresentados na Tabela 3, encontram-se as matrizes A e T correspondentes:

$$A_1 = \begin{bmatrix} C_1 & -S_2 & 0 & 0 \\ S_2 & C_2 & 0 & 0 \\ 0 & 0 & 1 & 0,245 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (4.1)$$

$$A_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (4.2)$$

$$A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (4.3)$$

e

$$T_3^0 = A_1 A_2 A_3 = \begin{bmatrix} C_1 & 0 & -S_1 & -S_1(d_3 + 0,35) \\ S_1 & 0 & C_1 & C_1(d_3 + 0,35) \\ 0 & -1 & 0 & 0,245 + d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (4.4)$$

em que as quatro grandezas θ_i , a_i , d_i , α_i são parâmetros associados com o link i e a junta i , $C\theta = \cos(\theta)$ e $S\theta = \sin(\theta)$, $C\alpha = \cos(\alpha)$ e $S\alpha = \sin(\alpha)$.

Qualquer posição, P_x , P_y e P_z , do elemento final do robô pode ser encontrada no espaço de tarefa a partir das coordenadas no espaço das juntas como observado em (4.5),

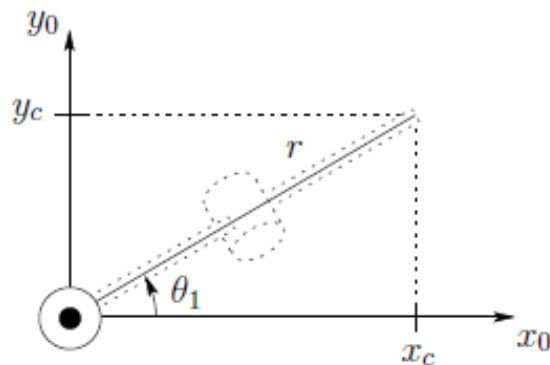
$$\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} -\sin(\theta_1)(d_3 + 0,35) \\ \cos(\theta_1)(d_3 + 0,35) \\ 0,245 + d_2 \end{bmatrix}. \quad (4.5)$$

4.1.2 Cinemática Inversa do Manipulador cilíndrico

Neste trabalho utilizou-se o método geométrico para resolver o problema da cinemática inversa do manipulador cilíndrico.

A partir da Figura 16 pode-se encontrar θ_1 (variável da junta 1) que mostra uma projeção da junta 1 sobre os eixos x e y .

Figura 16 – Projeção da primeira junta $x_0 - y_0$.



Fonte: Spong e Vidyasagar (2008).

A equação que representa a variável da junta 1 é definida como:

$$\theta_1 = \text{Atan2}(P_x, P_y). \quad (4.6)$$

O parâmetro do link 2 é prismático e como pode ser observado na Figura 15, d_2 está no mesmo eixo z_1 ,

$$d_2 = P_z - 0,245. \quad (4.7)$$

No caso do terceiro parâmetro d_3 , se desloca no plano formado por x e y e é determinado por:

$$d_3 = (\sqrt{P_x^2 + P_y^2}) - 0,35. \quad (4.8)$$

As Equações (4.6), (4.7) e (4.8) são as soluções do problema da cinemática inversa do manipulador cilíndrico e serão usadas para realizar o controle de posição e geração de caminho e trajetória do manipulador.

4.2 Jacobiano do Manipulador Cilíndrico

Como pode ser observado na Figura 15, mostrada anteriormente, um manipulador cilíndrico tem as seguintes variáveis das juntas, $q = (\theta_1, d_2, d_3)$. Para mais detalhes ver Apêndice A.4.

Como o manipulador possui uma junta de revolução e duas prismáticas, ou seja três juntas, a matriz jacobiana, neste caso é 6×3 , e é da forma:

$$J(q) = \begin{bmatrix} z_0 \times (o_3 - o_0) & z_1 & z_2 \\ & z_0 & o_0 \quad o_0 \end{bmatrix}, \quad (4.9)$$

onde tem-se $z_o = [0 \ 0 \ 1]^T = z_1$ e $o_0 = [0 \ 0 \ 0]^T$. z_2 e o_3 são dados por:

$$z_2 = \begin{bmatrix} \text{sen}(\theta_1) \\ \text{cos}(\theta_1) \\ 0 \end{bmatrix} \quad (4.10)$$

e

$$o_3 = \begin{bmatrix} -\text{cos}(\theta_1)(l_3 + 0,35) \\ -\text{sen}(\theta_1)(l_3 + 0,35) \\ 0 \end{bmatrix}. \quad (4.11)$$

Substituindo-se cada matriz e realizando as operações necessárias, considerando $d_3 = l_3 + 0,35$, tem-se a matriz jacobiana 6×3 , dada por,

$$J = \begin{bmatrix} -\text{cos}(\theta_1)d_3 & 0 & \text{sen}(\theta_1) \\ -\text{sen}(\theta_1)d_3 & 0 & \text{cos}(\theta_1) \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}. \quad (4.12)$$

O Jacobiano também pode ser representado na forma

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} -\cos(\theta_1)d_3 & 0 & \sin(\theta_1) \\ -\sin(\theta_1)d_3 & 0 & \cos(\theta_1) \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \dot{\theta}_1 \\ \dot{d}_2 \\ \dot{d}_3 \end{bmatrix}. \quad (4.13)$$

A partir da Equação 4.13 observa-se que não é possível se executar uma rotação em torno dos eixos x_0 e y_0 . O Jacobiano em relação à velocidade linear do *end-effector* é obtido considerando-se apenas as três primeiras linhas, ou seja,

$$J = \begin{bmatrix} -\cos(\theta_1)(d_3 + 0,35) & 0 & \sin(\theta_1) \\ -\sin(\theta_1)(d_3 + 0,35) & 0 & \cos(\theta_1) \\ 0 & 1 & 0 \end{bmatrix}. \quad (4.14)$$

4.3 Dinâmica do Manipulador Cilíndrico

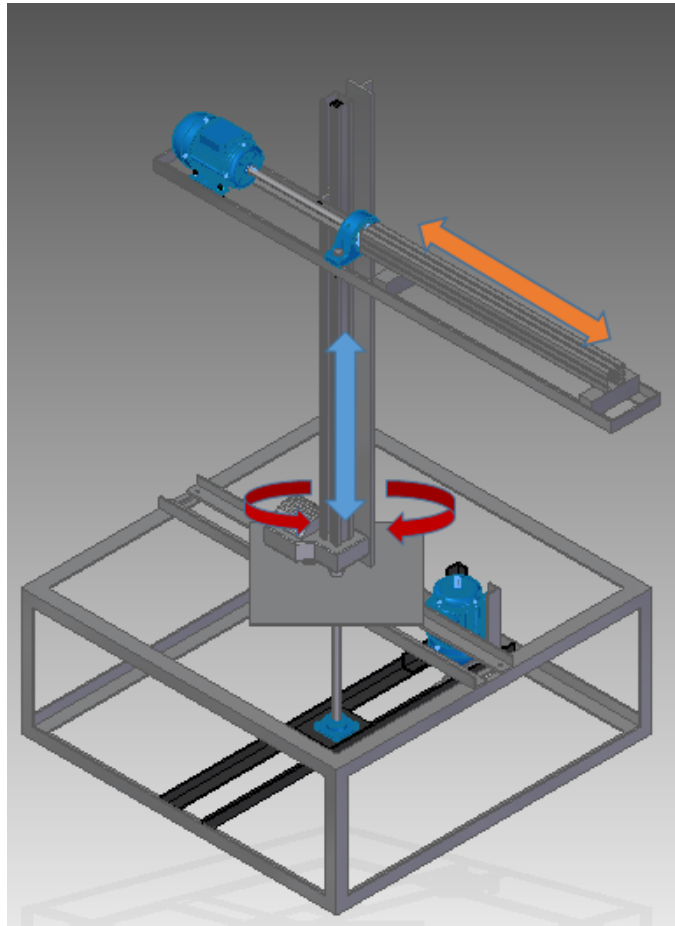
Foi realizada a modelagem do manipulador por meio do *software* de modelagem *Solid Edge*[®] (BATISTA *et al.*, 2020a), através da modelagem pode-se encontrar as massas das juntas do manipulador e posteriormente calcular os torques de cada junta. A Figura 17 apresenta a modelagem computacional do manipulador.

Através do *software* de modelagem computacional foram encontradas as principais propriedades físicas do manipulador como as dimensões, massas e momentos de inércia de cada junta. Importante destacar que a massa de cada junta do manipulador é uma aproximação do valor real, pois foram encontradas pelo *software* de modelagem, que leva em consideração o material e as dimensões das juntas. As Figuras 18 e 19 apresentam as telas do *software* de modelagem, *Software Solid Edge*[®], com as propriedades físicas do manipulador para as juntas 2 e 3, respectivamente.

As informações apresentadas na Tabela 4 serão utilizadas para o cálculo dos torques das juntas 1, 2 e 3. Observe que a massa correspondente à junta 1 é a soma das massas das juntas 2 e 3. Os valores das massas (m) e comprimentos (l) de cada elo do manipulador são mostrados na Tabela 4.

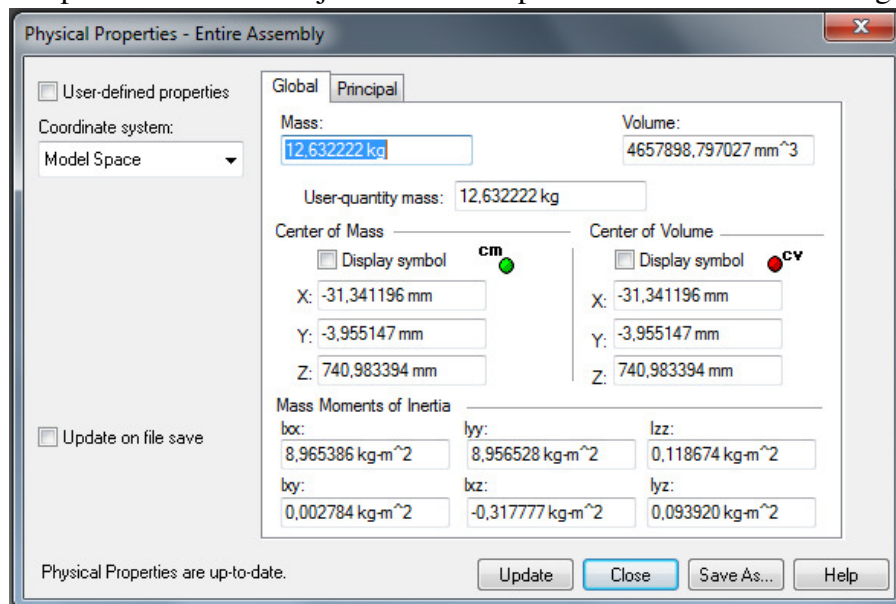
A formulação baseada no Lagrangiano do sistema mecânico é definida como (ver

Figura 17 – Estrutura do manipulador cilíndrico - *Software Solid Edge*®.



Fonte: Acervo do autor.

Figura 18 – Propriedades físicas da junta 2 do manipulador - *Software Solid Edge*®.



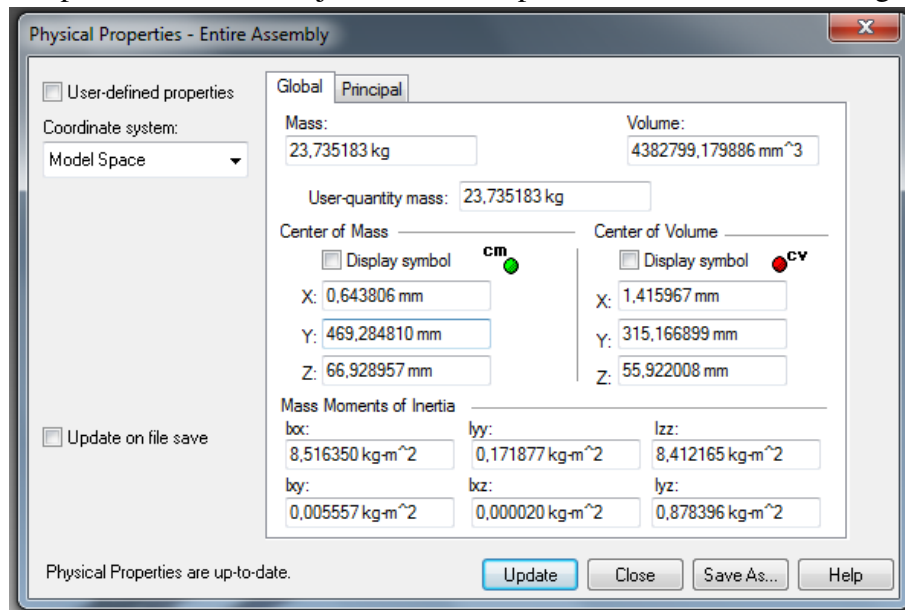
Fonte: Acervo do autor.

Apêndice A.5)

$$L(q, \dot{q}) = K(q, \dot{q}) - P(q).$$

(4.15)

Figura 19 – Propriedades físicas da junta 3 do manipulador - Software Solid Edge®.



Fonte: Acervo do autor.

Tabela 4 – Valores de massas (m) e comprimentos (l) das juntas.

Junta	Massa - m (kg)	Comprimento - l (m)
1	36,367405	0,050
2	12,632222	0,790
3	23,735183	0,900

Fonte: o autor.

Para o manipulador cilíndrico em estudo, apresentado na Figura 15, foi calculado a energia cinética e potencial e em seguida aplicada a formulação baseada no Lagrangiano.

4.3.1 Energia Cinética

A energia cinética total do manipulador com acionamento para as três juntas é dada por (SANZ, 2009)

$$K = K_1 + K_2 + K_3, \quad (4.16)$$

em que

$$K_1 = \frac{1}{2} [m_1 v_{c1}^2 + \omega_1^T I_1 \omega_1], \quad (4.17)$$

$$K_2 = \frac{1}{2} [m_2 v_{c2}^2 + \omega_2^T I_2 \omega_2] \quad (4.18)$$

e

$$K_3 = \frac{1}{2}[m_3 v_{c3}^2 + \omega_3^T I_3 \omega_3]. \quad (4.19)$$

O Jacobiano do manipulador foi apresentado na Equação (4.13). A partir do Jacobiano podem-se determinar as velocidades e as equações da energia cinética. Assim:

$$K_1 = \frac{1}{2}[m_1(-\cos\theta_1 d_3 \dot{\theta}_1 + \sin\theta_1 \dot{d}_3)^2], \quad (4.20)$$

$$K_2 = \frac{1}{2}[m_2(-\sin\theta_1 d_3 \dot{\theta}_1 + \cos\theta_1 \dot{d}_3)^2], \quad (4.21)$$

e

$$K_3 = \frac{1}{2}[m_3 \dot{d}_2^2 + \dot{\theta}_1^2 I_3]. \quad (4.22)$$

Após a realização de todas as operações matemáticas e algumas transformações trigonométricas, tem-se a equação que representa a energia cinética total

$$K = \frac{1}{2}[(m_1 + m_2)(-2\cos\theta_1 \sin\theta_1)(d_3 \dot{d}_3 \dot{\theta}_1) + (m_1 \cos^2\theta_1 + m_2 \sin^2\theta_1)(\dot{d}_3^2 \dot{\theta}_1^2) + (m_1 \sin^2\theta_1 + m_2 \cos^2\theta_1)(\dot{d}_3^2) + m_3 \dot{d}_2^2 + \dot{\theta}_1^2 I_3]. \quad (4.23)$$

4.3.2 Energia potencial

Partindo-se das definições da mecânica clássica de ponto de referência (zero de energia potencial), a energia potencial para cada junta do manipulador é (SCIAVICCO; SICILIANO, 1996)

$$P_1 = m_1 g l_1 \sin\theta_1 = 0, \quad (4.24)$$

pois $l_1 = a_1 = 0$,

$$P_2 = m_2 g d_2, \quad (4.25)$$

e

$$P_3 = m_3 g d_2. \quad (4.26)$$

Como $P = P_1 + P_2 + P_3$, tem-se:

$$P = g d_2 (m_2 + m_3). \quad (4.27)$$

4.3.3 Equação de Movimento

As equações de movimento do sistema são dadas por

$$\frac{d}{dt} \left[\frac{\partial L}{\partial \dot{q}} \right] - \frac{\partial L}{\partial q} = \tau, \quad (4.28)$$

em que $\tau \in \mathfrak{R}^n$ representa os torques aplicados às juntas. Assim, considerando-se a energia cinética do manipulador, a equação dinâmica do manipulador pode ser escrita de forma simplificada como

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau, \quad (4.29)$$

em que $C \in \mathfrak{R}^n$ é a matriz que descreve as forças centrípetas e de Coriolis e $G = \frac{\partial g}{\partial q} \in \mathfrak{R}^n$ é o vetor de gravidade.

Os efeitos de atrito nas juntas e de forças externas no efetuador podem ser incluídos no modelo dinâmico do manipulador

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + F(q)q + G(q) = \tau - f_{ext}, \quad (4.30)$$

em que f_{ext} é a força externa aplicada no *end-effector* e $F(q) \in \mathbb{R}^{n \times n}$ representa os efeitos das forças de atrito dinâmico e estático nas juntas. Esse vetor também representa os distúrbios e dinâmicas não modeladas como folgas nos acoplamentos e transmissões mecânicas.

Aplicando-se a formulação de Lagrange, equação (4.15), o lagrangiano para o sistema será

$$L = \frac{1}{2} [(m_1 + m_2)(-2\cos\theta_1 \sin\theta_1)(d_3 \dot{d}_3 \dot{\theta}_1) + (m_1 \cos^2\theta_1 + m_2 \sin^2\theta_1)(d_3^2 \dot{\theta}_1^2) + (m_1 \sin^2\theta_1 + m_2 \cos^2\theta_1)(\dot{d}_3^2) + m_3 \dot{d}_2^2 + \dot{\theta}_1^2 I_3] - g d_2 (m_2 + m_3). \quad (4.31)$$

A equação do movimento do manipulador a partir da formulação de Lagrange é obtida pelas derivadas parciais do lagrangiano (4.31). Em seguida tem-se a primeira equação de movimento que descreve o torque da junta 1

$$\begin{aligned} \tau_1 = & -[(4m_1 \sin\theta_1 - 4m_2 \cos\theta_1)d_3 + I_3]\ddot{\theta}_1 + [(m_1 + m_2)(\sin\theta_1 \cos\theta_1)d_3]\ddot{d}_3 \\ & + [(m_1 \sin\theta_1 - m_2 \cos\theta_1)d_3]\dot{\theta}_1^2 - [m_1 \cos\theta_1 + m_2 \sin\theta_1]\dot{d}_3^2 \\ & - [(m_1 + m_2)(\sin\theta_1 \cos\theta_1)d_3]\dot{\theta}_1 \dot{d}_3. \end{aligned} \quad (4.32)$$

Seguindo a mesma ideia apresentada para a junta 1, a partir das equações de movimento, descreve-se o torque da junta 2:

$$\tau_2 = m_3 \ddot{d}_2 + g(m_2 + m_3). \quad (4.33)$$

Em relação o torque da junta 3 tem-se a equação de movimento descrita por:

$$\begin{aligned} \tau_3 = [m_1 \text{sen} \theta_1 \cos \theta_1] \ddot{\theta}_1 - [2(m_1 \text{sen} \theta_1 + m_2 \cos \theta_1)] \ddot{d}_3 + [2d_3(m_1 \text{sen} \theta_1 - m_2 \cos \theta_1)] \dot{\theta}_1^2 \\ - [(m_1 + m_2)(\text{sen} \theta_1 \cos \theta_1)] \dot{\theta}_1 \dot{d}_3. \end{aligned} \quad (4.34)$$

Para o manipulador em estudo as equações de torque de cada junta, (4.32), (4.33) e (4.34), foram linearizadas em torno de um ponto de equilíbrio dado por, $\theta_1 = 0^\circ$, que podem ser expressas como:

$$\tau_1 = (4m_2 d_3 + I_3) \ddot{\theta}_1 - m_2 d_3 \dot{\theta}_1 - m_1 \dot{d}_3^2, \quad (4.35)$$

$$\tau_2 = m_3 \ddot{d}_2 + g(m_2 + m_3), \quad (4.36)$$

e

$$\tau_3 = -2m_2 \ddot{d}_3 - 2m_2 \dot{\theta}_1, \quad (4.37)$$

em que τ_1 , τ_2 e τ_3 são as forças externas aplicadas em cada junta. Neste caso, as equações (4.35), (4.36) e (4.37) correspondem a um conjunto de equações diferenciais autônomas lineares.

Vale ressaltar que neste trabalho serão utilizados somente os dados experimentais relacionados à junta 1 do manipulador, ou seja, será utilizada somente a equação (4.35), modelo linearizado, para o modelo de controle da junta 1, em que serão aplicados os controladores propostos neste trabalho.

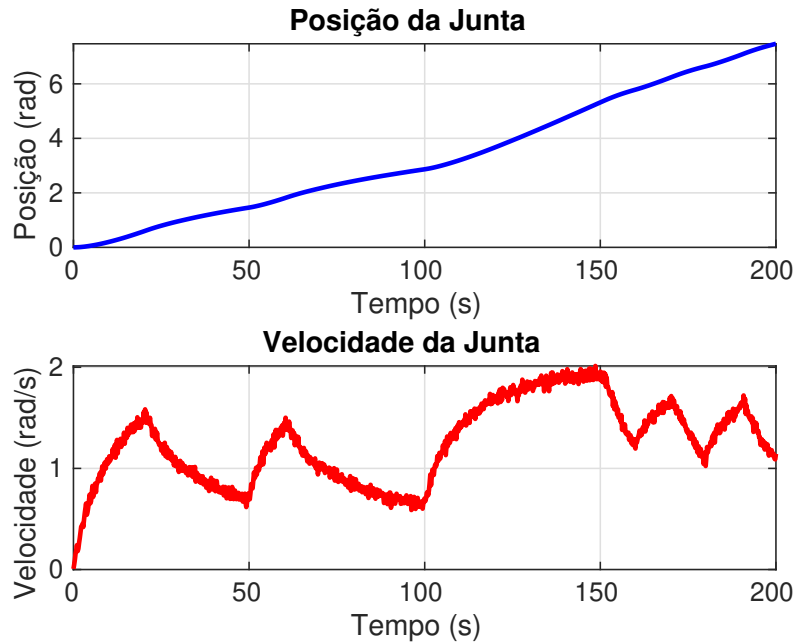
4.4 Identificação do Modelo da Junta 1 a partir da Dinâmica

Nesta seção apresenta-se o modelo da junta 1 do manipulador identificado a partir dos dados experimentais e do modelo dinâmico apresentado na Equação (4.32). Como citado anteriormente este trabalho abordará o modelo da junta 1 do manipulador para aplicação dos controladores.

Com os dados experimentais, de corrente e velocidade, da junta do manipulador apresentados na Figura 5, foi possível encontrar os valores da posição, velocidade e aceleração da junta 1, e com estes valores, foi possível calcular o torque da junta 1, a partir da equação (4.32), para os valores citados.

A Figura 20 apresenta a posição e a velocidade da junta 1 encontrados a partir dos dados experimentais apresentados na Figura 5, onde o valor da velocidade foi convertido para *rad/s* e os valores de posição foram encontrados a partir dos valores de velocidade.

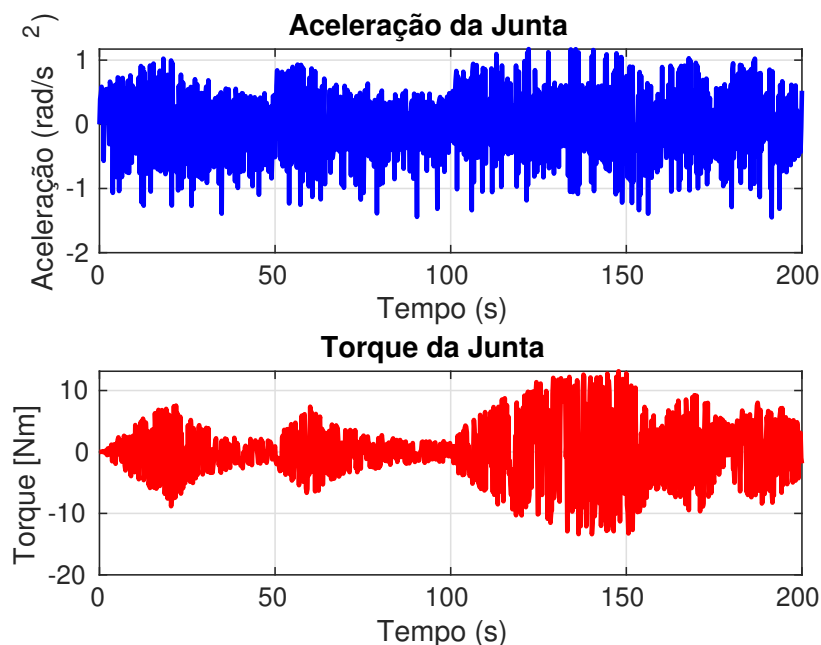
Figura 20 – Posição e velocidade da junta 1 do manipulador, a partir dos dados experimentais.



Fonte: Acervo do autor.

A Figura 21 apresenta a aceleração calculada a partir dos dados experimentais de velocidade (Figura 5), e o torque da junta 1 do manipulador. Os valores de torque foram calculados a partir dos valores de posição, velocidade e aceleração, apresentados anteriormente (Figura 20) e através da equação (4.32).

Figura 21 – Aceleração da junta 1 do manipulador e torque calculado a partir dos valores de posição, velocidade e aceleração.



Fonte: Acervo do autor.

Para encontrar a relação do torque da junta 1 do manipulador com a corrente do motor (I_q), foi necessário realizar alguns cálculos e análises. Como tem-se os valores experimentais de corrente para o valor de velocidade, é possível encontrar uma relação entre o torque da junta e a corrente do motor. No modelo dinâmico o torque é a entrada e a velocidade é a saída do modelo. Da mesma forma que foi aplicada a corrente (entrada) como já citado e encontrado os valores de velocidade (saída) da junta.

Com isso, a ideia é poder realizar uma comparação do modelo dinâmico com os dados experimentais apresentados na Figura 5, onde a corrente é a entrada e a velocidade é a saída. Dessa forma será possível encontrar o modelo em espaço de estados a partir do modelo dinâmico direto do manipulador, que neste caso será utilizada a equação (4.35), que representa o modelo da junta 1 do manipulador.

O torque do motor de indução pode ser obtido através da seguinte equação (CHAPMAN, 2013; KOSOW, 1993)

$$T_m = K_T \varphi I_r \cos \theta_r, \quad (4.38)$$

em que K_T é a constante de torque que relaciona as unidades envolvidas, φ é o fluxo magnético do rotor que é proporcional à tensão aplicada aos terminais da armadura do motor, e $I_r \cos \theta_r$ é a componente da corrente do rotor que está em fase com a tensão do rotor, ou seja, é a componente que produz torque útil.

Como mencionado anteriormente no Capítulo 2, a corrente de campo i_d , foi escolhida de modo a manter o fluxo constante ($i_d = 0,4 A$), ou seja $i_d \approx \varphi$, e o conjugado (torque) do motor pode ser controlado pela mudança na corrente de armadura, i_q . Dessa forma pode-se reescrever a equação (4.38), com uma nova contante de torque K_T' como

$$T_m = K_T' i_d i_q. \quad (4.39)$$

Com os valores de corrente do motor (i_q) obtidos no ensaio experimental e com o valores de torque calculados pelo modelo dinâmico, apresentados na Figura 21, pode-se reescrever a Equação (4.39) para encontrar uma relação entre a corrente i_q e o torque da junta do manipulador, da seguinte forma

$$T_{junta} = K_T'' i_q, \quad (4.40)$$

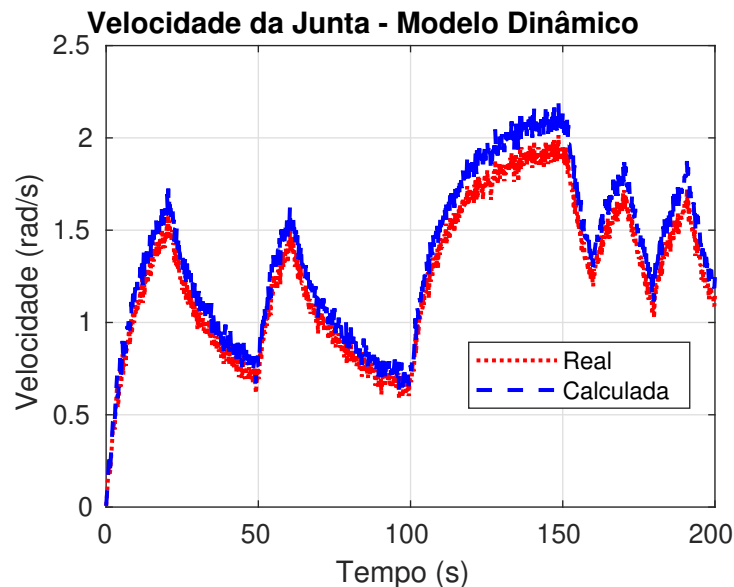
em que K_T'' é a constante que relaciona o torque da junta do manipulador com a corrente i_q .

Dessa forma pode-se relacionar o modelo dinâmico da junta 1 do manipulador com a corrente i_q , onde a entrada do modelo passa também a ser a corrente.

A partir da Equação (4.35), que representa o modelo dinâmico linearizado da junta 1, pode-se calcular a velocidade da junta com o valor de torque calculado pela equação (4.40). Dessa forma pode-se fazer uma comparação do valor calculado com o valor real da velocidade.

A Figura 22 apresenta a velocidade real (experimental) da junta 1 e a velocidade calculada a partir do modelo dinâmico (fenomenológico) do manipulador representado pela Equação (4.35).

Figura 22 – Velocidades da junta 1 a partir dos dados experimentais, e calculada a partir do modelo dinâmico do manipulador.



Fonte: Acervo do autor.

Através de uma comparação entre os dados experimentais e calculados, pelo modelo dinâmico, foram obtidos os seguintes valores de coeficiente de correlação múltipla, e da raiz quadrada do erro médio (*RMSE*), apresentados na Tabela 5. Essa comparação é importante, pois apresenta uma melhor visualização do modelo encontrado.

Tabela 5 – Valores de R^2 e *RMSE* para o modelo dinâmico da velocidade.

Modelo	R^2	<i>RMSE</i>
Velocidade	0,8900	0,2672

Fonte: o autor.

Com os dados de torque encontrados, e considerando que as juntas 2 e 3 do manipu-

lador estão fixas (não se movimentarão), ou seja $\dot{d}_3 = 0$, na Equação (4.35), tem-se

$$\begin{bmatrix} x_1 = \theta \\ x_2 = \dot{\theta} \end{bmatrix} \quad (4.41)$$

e, assim, pode-se reescrever

$$\begin{bmatrix} \dot{x}_1 = x_2 \\ \dot{x}_2 = \ddot{\theta} \end{bmatrix}, \quad (4.42)$$

com isso tem-se

$$\tau_1 = (4m_2d_3 + I_3)\dot{x}_2 - m_2d_3x_2 - m_1\dot{d}_3^2 \quad (4.43)$$

reajustando

$$-(4m_2d_3 + I_3)\dot{x}_2 = -\tau_1 - m_2d_3x_2 - m_1\dot{d}_3^2 \quad (4.44)$$

daí tem-se

$$\dot{x}_2 = \frac{1}{4m_2d_3 + I_3}\tau_1 + \frac{m_2d_3}{4m_2d_3 + I_3}x_2 \quad (4.45)$$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & \frac{m_2d_3}{4m_2d_3 + I_3} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{4m_2d_3 + I_3} \end{bmatrix} \tau_1. \quad (4.46)$$

Escolhe a saída como x_1 e coloca o a entrada $u = \tau_1$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0]u. \quad (4.47)$$

Com isso pode-se escrever o modelo da junta 1 do manipulador em espaço de estados representado por

$$A = \begin{bmatrix} 0 & 1 \\ 0 & \frac{m_2d_3}{4m_2d_3 + I_3} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \frac{1}{4m_2d_3 + I_3} \end{bmatrix} \tau_1, \quad (4.48)$$

$$C = \begin{bmatrix} 1 & 0 \end{bmatrix} e \quad D = \begin{bmatrix} 0 \end{bmatrix} \tau_1.$$

Todos os termos na equação (4.48) foram definidos anteriormente. Os valores das massas foram descritos na Tabela 4, e o valor de $I_3 = 8,5163 \text{ kg.m}^2$, obtido a partir da modelagem através do *software Solid Edge*®.

A equação (4.48) representa o modelo em espaço de estados da junta 1 encontrada através do modelo dinâmico do manipulador. Esse modelo também será utilizado para a implementação dos controladores propostos neste trabalho.

4.5 Considerações Finais

Este capítulo apresentou-se o modelo da cinemática direta e inversa, a matriz jacobiana do manipulador e o modelo dinâmico do manipulador através de equações de movimento que os representam. Foram descritas as equações da cinemática direta e inversa, o modelo da matriz jacobiana a partir das velocidades lineares e angulares, e o por último as equações de movimento do modelo dinâmico para o cálculo do torque das juntas do manipulador.

Também foram apresentados os modelos das juntas linearizados, as curvas da posição, velocidade, aceleração e torque da junta 1. Por fim, e não menos importante, foi descrita a identificação do modelo da junta 1 do manipulador através da dinâmica inversa, e o modelo em espaço de estados.

No próximo capítulo são apresentadas as teorias dos controladores LQR/LQI e LQG e estes controladores otimizados com as metaheurísticas AG e PSO.

5 CONTROLADORES LQI E LQG COM METAHEURÍSTICAS

Neste capítulo é apresentada a teoria do controlador LQR (*Linear Quadratic Regulator*)/LQI (*Linear-Quadratic-Integral*) e do controlador *Linear-Quadratic-Gaussian* (LQG), e o projeto desses controladores. Também é apresentada a metodologia sobre a aplicação dos controladores LQI e LQG com as metaheurísticas AG e PSO para otimização das matrizes Q e R .

5.1 Controlador LQR - *Linear Quadratic Regulator*

O problema do LQR onde todos os estados são conhecidos, é o problema de valor inicial determinístico. Dado um sistema $\dot{x} = Ax + Bu$ com um dado estado inicial diferente de zero $x(0)$, encontra-se o sinal de entrada $u(t)$ que faz o sistema voltar ao estado zero ($x = 0$) de maneira ótima (SKOGESTAD; POSTLETHWAITE, 2007).

De modo geral, pode-se resumir a obtenção do ganho do LQR da seguinte forma (ZHOU; DOYLE, 1998)

$$u = -Kx \quad (5.1)$$

onde,

$$K = R^{-1}B^T P, \quad (5.2)$$

em que P é a solução da Equação de *Riccati*, dada por:

$$-\dot{P} = PA + A^T P - PBR^{-1}B^T P + Q, \quad (5.3)$$

em que Q e R são os pesos associados ao estado e ao sinal de controle e, K é obtida por um processo de otimização, ao invés de ser escolhida para o posicionamento de polos que garante especificações de projeto. Diante disso, K deve ser escolhido de modo a minimizar uma função de custo quadrática que leve em consideração o estado $x(t)$ do sistema e o sinal de controle $u(t)$ da malha fechada. Então, uma das formas mais simples de definir essa função é (SKOGESTAD; POSTLETHWAITE, 2007; ALBERTO *et al.*, 2004)

$$J = \int_0^{\infty} [x(t)^T Q x(t) + u(t)^T R u(t)] dt, \quad (5.4)$$

em que Q e R são as matrizes de peso, cuja dimensões são apropriadas para a compatibilidade dimensional com $x(t)$ e $u(t)$, respectivamente.

Em malha fechada, tem-se o sistema,

$$\dot{x} = (A - BK)x(t) = A_c x(t), \quad (5.5)$$

lembrando que a solução dessa equação é dada por:

$$x(t) = \Phi(t, t_0)x(t_0), \quad (5.6)$$

em que $\Phi(t, t_0)$ é a matriz de transição de estados do sistema em malha fechada.

Escrevendo (5.5) para $t = 0$ tem-se

$$= x^T(0)[A_c^T P + \dot{P} + PA_c]x(0). \quad (5.7)$$

então

$$-L = A_c^T P + PA_c + \dot{P}, \quad (5.8)$$

ou

$$-\dot{P} = A_c^T P + PA_c + L, \quad (5.9)$$

daí

$$-\dot{P} = (A^T - K^T B^T)P + P(A - BK) + K^T R K + Q. \quad (5.10)$$

Suponha que,

$$P = P_0 + P', \quad (5.11)$$

$$K = K_0 + K', \quad (5.12)$$

onde P_0 é a matriz tal que quando aplicada à função custo na Equação (5.4), leva à obtenção do ganho ótimo K_0 . Logo a Equação (5.10) fica,

$$-\dot{P}_0 = (A^T - K_0^T B^T)P_0 + P_0(A - BK_0) + K_0^T R K_0 + Q. \quad (5.13)$$

Diante disso, a Equação (5.13), será da seguinte forma:

$$L = K'^T (-B^T P_0 + R K_0) + (-P_0 B + K_0^T R) K' + K'^T R K'. \quad (5.14)$$

Considerando-se que K' é suficientemente pequeno na Equação (5.14), então os termos lineares podem dominar o resultado para L sobre o termo quadrático $K'^T R K'$, e com isso, existem valores de K' que podem tornar L definida negativa, a menos que os termos lineares não existam na expressão de L (FRIEDLAND, 2012). Logo,

$$\begin{aligned} RK_0 - B^T P_0 &= 0 \\ K_0 &= R^{-1} B^T P_0. \end{aligned} \quad (5.15)$$

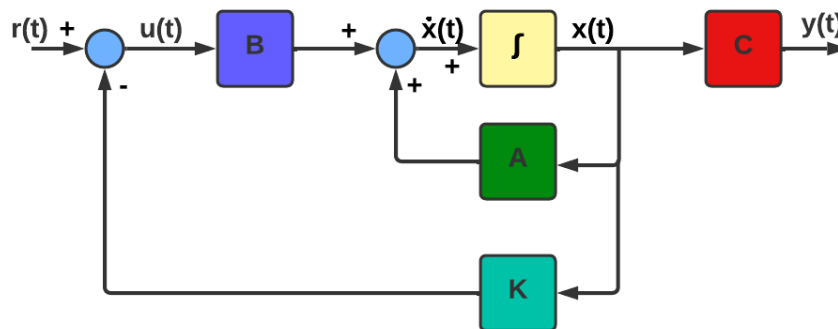
Substituindo (5.15) em (5.13) tem-se,

$$= A^T P_0 + P_0 A - P_0 B R^{-1} B^T P_0 + Q, \quad (5.16)$$

em que a Equação (5.16) é a Equação de Riccati, amplamente mencionada, quando se refere ao problema do LQR.

A Figura 23 apresenta a estrutura em diagrama de blocos do controlador LQR para um sistema em espaço de estados.

Figura 23 – Estrutura do controlador LQR em diagrama de blocos.



Fonte: Acervo do autor.

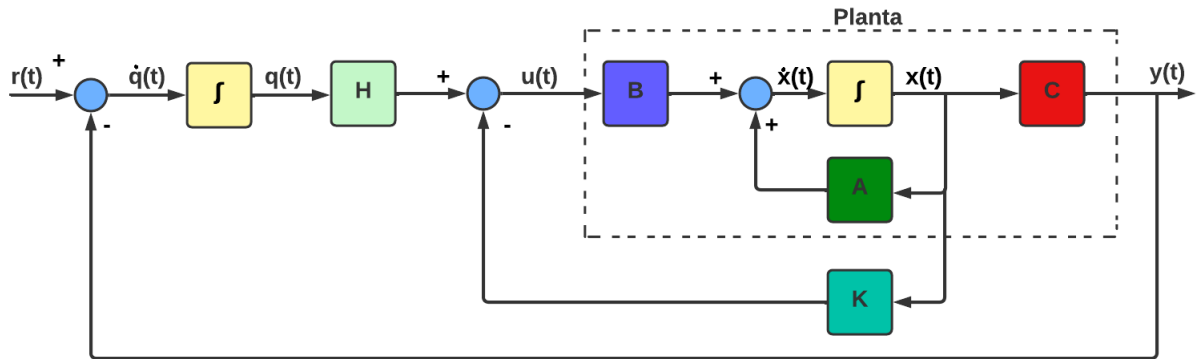
5.2 Realimentação de Estado com Ação Integral

Os sistemas de controle em geral apresentam erros de regime estacionário diferentes de zero e as especificações para um sistema de controle, geralmente, exigem que o erro de regime seja igual a zero para a saída do sistema. Para que isso aconteça é necessário incorporar uma ação integral no sistema antes de calcular os ganhos do controlador, e com isso integrar a saída do sistema para as quais tem especificação de erro de regime igual a zero. Para o novo sistema a matriz de ganhos do controlador é calculada para a planta com a ação integral incorporada.

A partir do modelo do controlador LQR, apresentado na Figura 23, e das equações apresentadas anteriormente, o modelo do sistema em espaço de estados pode ser remodelado

e em seguida, pode-se fechar a malha pelo uso da realimentação de estado. Inserido a ação integral e o estado aumentado, pode-se projetar o controlador com realimentação de estado, desta maneira pode-se adotar como incertezas outros elementos que compõem o a planta. Com isso, pode-se fazer uso da estrutura de realimentação de estado com ação integral LQI, representado na Figura 24.

Figura 24 – Estrutura de realimentação de estado com ação integral.



Fonte: Acervo do autor.

De acordo com a Figura 24 a concepção do estado aumentado pode ser mostrado da seguinte forma (RIOS, 2021)

$$\dot{x} = Ax(t) + Bu(t), \quad (5.17)$$

$$u(t) = Hq(t) - Kx(t) = \begin{bmatrix} -K & H \end{bmatrix} \begin{bmatrix} x(t) \\ q(t) \end{bmatrix}, \quad (5.18)$$

$$\dot{q}(t) = r(t) - y(t) = r(t) - Cx(t), \quad (5.19)$$

$$\dot{x}(t) = Ax(t) + B[Hq(t) - Kx(t)] = (A - BK)x(t) + BHq(t) \quad (5.20)$$

e

$$\begin{bmatrix} \dot{x}(t) \\ \dot{q}(t) \end{bmatrix} = \begin{bmatrix} A - BK & BH \\ -C & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ q(t) \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} r(t). \quad (5.21)$$

A Equação (5.21) mostra que o projeto de realimentação de estado com ação integral pode ser feito como um projeto de realimentação de estado normal para a planta aumentada.

Diante disto, pode-se projetar um controlador $K = [-K \ H]$ de modo que a matriz de malha fechada seja: $(A_a - B_a K)$, onde

$$A_a = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \quad (5.22)$$

e

$$B_a = \begin{bmatrix} 0 \\ I \end{bmatrix}. \quad (5.23)$$

Então, em regime permanente tem-se:

$$\lim_{t \rightarrow \infty} \dot{q}(t) = 0 \Rightarrow \lim_{t \rightarrow \infty} y(t) = r. \quad (5.24)$$

Com isso, pode-se ainda utilizar uma outra abordagem para determinar a matriz aumentada pela inserção da ação integral. Para isso, pode-se considerar $r(t) = 0$, tal que $\dot{q}(t) = -Cx(t)$, e as equações (5.17) e (5.19) podem ser reescritas como,

$$\begin{bmatrix} \dot{x}(t) \\ \dot{q}(t) \end{bmatrix} = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ q(t) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(t). \quad (5.25)$$

Com o modelo pronto, do estado aumentado, pode-se projetar um controlador por realimentação de estado para o novo sistema dado por:

$$\begin{aligned} \dot{x}_a &= A_a x_a(t) + B_a u(t) \quad e \\ y_a(t) &= C_a x_a(t), \end{aligned} \quad (5.26)$$

em que

$$A_a = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix}, \quad \dot{x}_a(t) = \begin{bmatrix} \dot{x}(t) \\ \dot{q}(t) \end{bmatrix}, \quad x_a(t) = \begin{bmatrix} x(t) \\ q(t) \end{bmatrix}, \quad B_a = \begin{bmatrix} B \\ 0 \end{bmatrix}, \quad C_a = \begin{bmatrix} C & 0 \end{bmatrix} \quad e \quad D = 0.$$

Percebe-se que o sistema possui um novo estado, que é o erro de estado estacionário, quando o sinal de referência for ou não considerado. Então, para o projeto do controlador por realimentação de estados, deve-se encontrar $n + 1$ ganhos, onde n representa a quantidade de variáveis de estado (K) e o ganho adicional (H) é devido à ação integral, como mostrado na Figura 24.

Para a implementação do controlador com realimentação de estado com ação integral (controlador LQI) serão utilizados os modelos da junta do manipulador identificado pelo algoritmo RLSPSO apresentado na Equação (3.20) e o modelo a partir da dinâmica do manipulador apresentado na Equação (4.48).

Os valores dos parâmetros do controlador LQI, as matrizes Q e R , para a realimentação de estados serão otimizados com as metaheurísticas AG e PSO.

5.3 *Linear-Quadratic-Gaussian (LQG)*

Os artigos históricos de *Kalman e Bucy* (KALMAN, 1962) do início dos anos 1960 estão entre as contribuições mais importantes para a ciência de controle e sistemas do século XX. O impacto deste trabalho o classifica com o trabalho de *Nyquist* e Bode dos anos 1920 e 1930, e com o trabalho de *Wiener* dos anos 1940. Foi amplamente reconhecido que os algoritmos de processamento de dados baseados nos resultados de *Kalman e Bucy* seriam de enorme benefício para este programa e outros projetos espaciais (FRIEDLAND, 2012).

Só então, a partir daí, os métodos de espaço de estado começaram a receber muita atenção e os resultados de Kalman e Bucy eram perfeitamente adequados a esses métodos. Os cálculos necessários para implementar outros algoritmos de filtragem, por outro lado, teriam sobrecarregado os computadores digitais daquele período e nem foram levados em consideração depois que o filtro de Kalman entrou em cena (FRIEDLAND, 2012).

O controle linear-quadrático-gaussiano (do inglês *Linear-Quadratic-Gaussian LQG*) é uma técnica moderna de espaço de estado para projetar reguladores dinâmicos ideais e servocontroladores com ação integral (também conhecidos como rastreadores de ponto de ajuste). Essa técnica permite-se a compensação do desempenho da regulação/rastreador e o esforço de controle e leve em consideração as perturbações do processo e o ruído de medição. Para projetar reguladores LQG e rastreadores de *setpoint* são necessárias as seguintes etapas: construção do ganho LQ-ótimo; construção do filtro de Kalman (estimador de estado); e projeto do LQG conectando o ganho LQ ótimo e o filtro Kalman (SKOGESTAD; POSTLETHWAITE, 2007).

Ao se considerar um sistema de estimação que tem fundamentação baseada em sinais aleatórios com distribuição gaussiana, têm-se o filtro de Kalman, cujo ganho é obtido a partir da função custo $J = \{e^T\}$, sendo $e = x - \hat{x}$, a distância entre x e o estado estimado \hat{x} . Pode-se considerar que o estado estimado é usado como estado real para a formação de uma lei

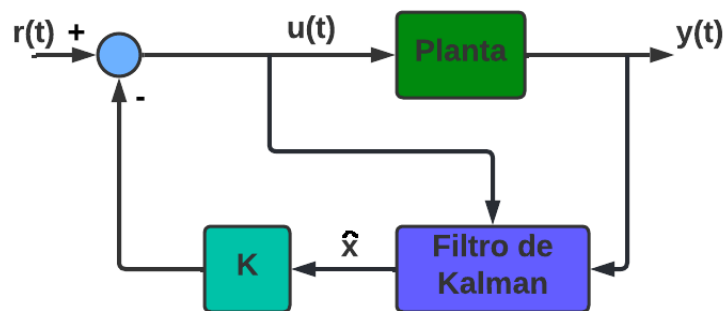
de controle por realimentação de estados (SGRÒ *et al.*, 2022; PINTO *et al.*, 2011),

$$u = -K\hat{x} \quad (5.27)$$

em que K é calculado a partir do problema do LQR.

Com isso, tem-se a otimização no estado e na lei de controle. Esse tipo de estratégia de controle constitui o controle LQG. Nesse contexto, considera-se o diagrama de blocos (PINTO *et al.*, 2011).

Figura 25 – Estrutura em diagrama de blocos do controlador LQG.



Fonte: Acervo do autor.

Considere que esse sistema (planta) é descrito pelo modelo apresentado nas equações (3.12) e (3.13).

Para o caso do estimador ótimo, o sistema a ser considerado é do tipo:

$$\dot{x} = Ax + Bu + Bw(t) \Rightarrow R_w = E\{ww^T\} \quad (5.28)$$

e

$$y = Cx + v(t) \Rightarrow R_v = E\{vv^T\}, \quad (5.29)$$

cuja função custo a ser minimizada é dada por:

$$J = E\{e(t)e^T(t)\}, \quad (5.30)$$

sendo

$$e(t) = x(t) - \hat{x}(t). \quad (5.31)$$

A Equação de *Ricatti* associada ficará da seguinte forma:

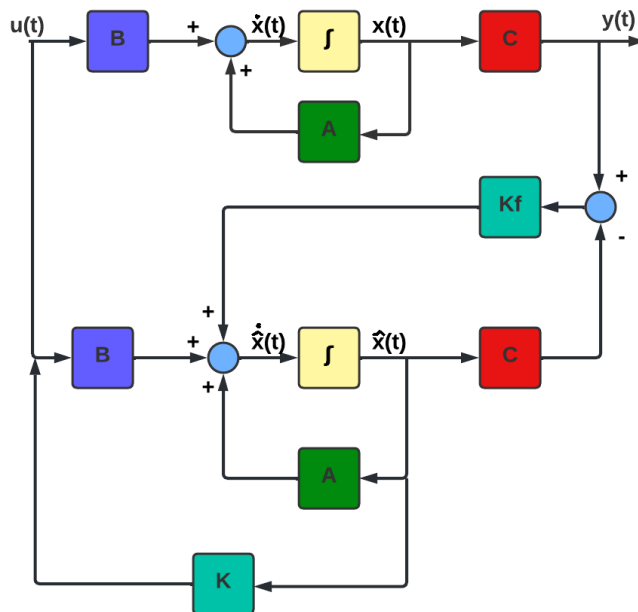
$$\dot{P}_o = AP_o + P_oA^T - P_oC^TR_v^{-1}CP_o + BR_wB^T \quad (5.32)$$

cuja solução leva ao ganho do estimador dado por:

$$K_{fc} = P_o C^T R v^{-1}. \quad (5.33)$$

A implementação do diagrama de blocos do sistema de controle em malha fechada é dado pela representação da Figura 26.

Figura 26 – Diagrama de blocos do controlador LQG em malha fechada.



Fonte: Acervo do autor.

No caso para o sistema em tempo contínuo tem-se

$$\dot{x} = Ax + Bu + Bw, \quad (5.34)$$

$$\dot{\hat{x}} = A\hat{x} + Bu + K_f(y - C\hat{x}), \quad (5.35)$$

sendo $u = -K\hat{x}$, tem-se,

$$\dot{x} = (A - BK)x + BKe + Bw. \quad (5.36)$$

Subtraindo a Equação (5.35) da (5.34) tem-se,

$$\dot{e} = (A - K_f C)e + Bw - K_f v. \quad (5.37)$$

Agrupando as equações (5.36) e (5.37) tem-se,

$$\begin{bmatrix} \dot{x}(t) \\ \dot{e}(t) \end{bmatrix} = \begin{bmatrix} A - BK & BK \\ 0 & A - K_f C \end{bmatrix} \begin{bmatrix} x(t) \\ e(t) \end{bmatrix} + \begin{bmatrix} B & 0 \\ B & -K_f \end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix} \quad (5.38)$$

Os autovalores de A_c são obtidos pela união dos autovalores de $A - BK$ e pelos autovalores de $A - K_f C$, que é conhecido como Teorema da Separação, onde:

$$A_c = \begin{bmatrix} A - BK & BK \\ 0 & A - K_f C \end{bmatrix}. \quad (5.39)$$

Da mesma forma, para a implementação do controlador LQG serão utilizados os modelos da junta do manipulador identificado pelo algoritmo RLSPSO apresentado na Equação (3.20) e o modelo a partir da dinâmica do manipulador apresentado na Equação (4.48).

Para o controlador LQG, as matrizes Q e R também serão otimizadas com as metaheurísticas AG e PSO.

5.4 Otimização dos Controladores LQI e LQG com Metaheurísticas

Metaheurísticas são algoritmos exploratórios com maior nível de busca, usados para resolver problemas de otimização. Eles são capazes de fornecer soluções confiáveis e de qualidade em tempo computacional finito. As metaheurísticas coordenam procedimentos de busca local com estratégias de alto nível, a fim de criar um processo capaz de escapar de mínimos locais e realizar uma busca robusta no espaço de solução de um problema (GLOVER; KOCHENBERGER, 2006).

Existem várias metaheurísticas populacionais (BOUSSAÏD *et al.*, 2013), mas neste trabalho serão utilizados o algoritmo genético (AG), e otimização por enxame de partículas (*PSO - particle swarm optimization*), onde estes algoritmos irão encontrar as matrizes Q e R ótimas para os controladores LQI e LQG, de forma a obter a melhor resposta de acordo com a função de custo que será descrita a seguir. A relevância de usar AG e PSO para melhorar o LQI e o LQG está na busca de uma forma ótima das matrizes Q e R , pois geralmente, são utilizados os métodos empíricos ou por busca exaustiva para encontrar estas matrizes.

O AG é provavelmente o algoritmo mais conhecido e mais utilizado como técnica de computação evolutiva. Muitas variantes do AG foram desenvolvidas e aplicadas a uma ampla variedade de problemas de otimização. São comumente usados para gerar soluções de alta qualidade para otimizar e buscar problemas com base em operadores biologicamente inspirados, como mutação, cruzamento e seleção. Em um algoritmo genético, uma população de soluções candidatas (chamadas de indivíduos, criaturas ou fenótipos) para um problema de otimização é desenvolvido para melhores soluções. Cada solução candidata tem um conjunto de propriedades (seus cromossomos ou genótipos) que podem ser mutados e alterados (SOUZA *et al.*, 2020).

O algoritmo de otimização por enxame de partículas (*Particle Swarm Optimization - PSO*) é um algoritmo heurístico baseado no comportamento social de um bando de pássaros. O método foi proposto por Eberhart e Kennedy em 1995 (EBERHART; KENNEDY, 1995) e tem como objetivo buscar a solução ótima, em um espaço de busca através da troca de informações entre indivíduos de uma população determinando qual trajetória cada um deles deverá tomar no espaço de busca. O PSO possui algumas vantagens tais como facilidade de implementação e rápida convergência, porém muitas vezes enfrenta um problema em que as suas partículas ficam “presas” em ótimos locais. Esse problema é muitas vezes chamado de convergência prematura (PAIVA *et al.*, 2017).

5.4.1 Função de Custo

A ideia neste trabalho, com relação a parte de controle, é utilizar as metaheurísticas AG e PSO para encontrar as matrizes ótimas de ponderação Q e R dos controladores LQI e LQG. A ideia é encontrar essas matrizes de forma mais rápida e otimizada, diferentemente quando estas são encontradas de forma empírica, onde pode-se não obter-se um resultado satisfatório para o controlador. Nesse contexto, pode-se afirmar que as metaheurísticas AG e PSO seriam uma opção para otimizar os controladores. As configurações das plantas que serão utilizadas para os projetos dos controladores foram apresentadas nos Capítulos 3 e 4.

Para o controlador LQG cujo sistema é baseado em sinais aleatórios com distribuição gaussiana, tem-se o filtro de Kalman, cujo ganho é obtido a partir da Equação de *Ricatti* (5.4), que também estão inclusos os ganhos Q e R do controlador LQG e serão otimizados através das metaheurísticas.

Para a implementação das metaheurísticas é necessário a utilização de uma função objetivo (função custo J), esta função deve estar relacionada ao desempenho dos controladores LQI e LQR. Com isto pode-se utilizar algumas métricas de desempenho dos controladores. Neste caso, a função objetivo adotada leva em consideração a minimização do erro que atua na malha de velocidade. A função de custo é definida por

$$J = \min(IAE + ISE) \quad (5.40)$$

em que IAE é a integral do erro absoluto (*integral of absolute error - IAE* = $\int_0^{\infty} |e(t)| dt$) e ISE é a integral do erro quadrado (*integral of squared error - ISE* = $\int_0^{\infty} e(t)^2 dt$).

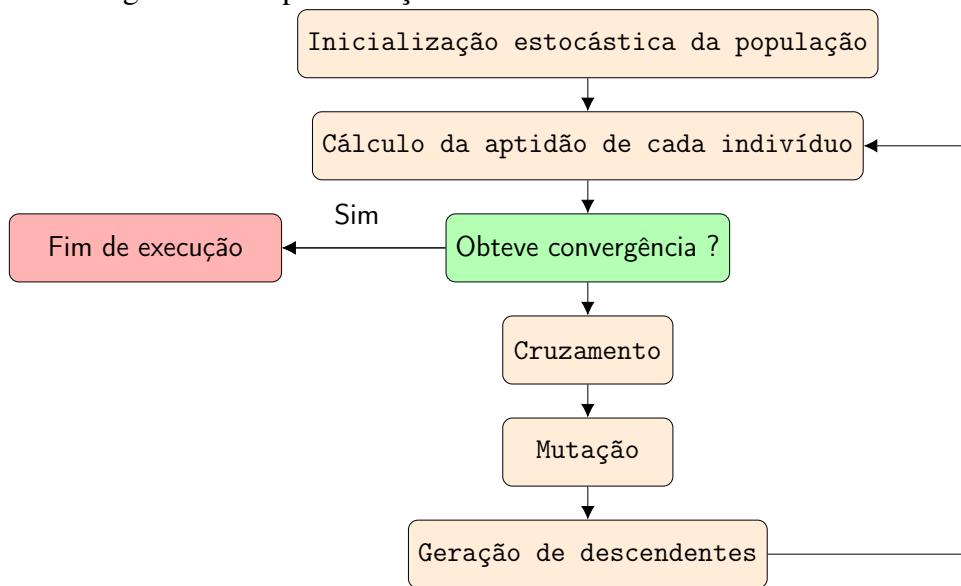
A função custo apresentada na Equação (5.40) será utilizada como critério de parada

nas metaheurísticas AG e PSO aplicadas aos controladores aqui apresentados.

5.4.2 Algoritmo Genético - AG

O Algoritmo Genético (AG) é comumente usado para encontrar soluções de problemas com um grande espaço de busca, problemas com muitas restrições e com diversos parâmetros com grande possibilidade de combinações (LINDEN, 2008). A ideia ao se utilizar AG é que ele seja capaz de evoluir uma população de soluções candidatas de um determinado problema, de modo que um operador inspirado por variação genética natural e seleção natural encontrará melhor solução (LINDEN, 2008). Sendo assim, aplica-se o algoritmo genético para se encontrar os parâmetros Q e R dos controladores LQI e LQG de modo a se obter um desempenho adequado para o controle de velocidade da junta do manipulador. A Figura 27 apresenta um fluxograma da implementação do processo de busca e escolha da melhor solução.

Figura 27 – Fluxograma da implementação do AG.



Fonte: Acervo do autor.

A partir da Figura 27 observa-se que cada indivíduo gera uma solução de acordo com a busca para satisfazer os valores dos parâmetros. Assim, avalia-se como o melhor indivíduo evolui ao longo da busca, e entendendo-se por melhor indivíduo aquele que tem menor custo (*fitness*). Os indivíduos com melhores avaliações são selecionados para gerar novos indivíduos formando desta forma, uma nova geração de soluções (ENGELBRECHT, 2007).

A partir deste conjunto de soluções (população) e utilizando-se algoritmo genético, os resultados são avaliados individualmente recebendo uma nota de acordo com a necessidade e

limitações impostas pelo projeto, como índices de desempenho. Para fazer essa avaliação o AG utiliza uma relação de *fitness* onde serão encontrados os melhores parâmetros (BATISTA *et al.*, 2020b).

Para a aplicação proposta neste trabalho utilizou-se os seguintes valores dos parâmetros no algoritmo AG os quais apresentaram melhor desempenho, após algumas verificações, ou seja, após várias execuções do algoritmo:

- Tamanho da população = 30 indivíduos;
- Probabilidade de cruzamento = 0,8;
- Probabilidade de mutação = 0,01;
- Número de gerações = 50.

O pseudocódigo do AG, para encontrar as matrizes Q e R ótimas, é apresentado a seguir no Algoritmo 3.

Algoritmo 3: Pseudocódigo do AG.

```

1: inicializa população  $P$ ;
2: repita
3: selecione uma subpopulação  $P'$ ;
4: para  $i \leftarrow 1$  até cruzamento CR faça
5: escolha  $S_1, S_2 \in P'$ , aleatoriamente;
6: filho  $\leftarrow$  cruzamento ( $S_1, S_2$ );
7: se  $f(S_1) \geq f(S_2)$  então  $S_{aux} \leftarrow S_1$ ;
8: senão  $S_{aux} \leftarrow S_2$ ;
9: se  $f(S_{aux}) \leq f(\text{filho})$  então
10: filho substitui  $S_{aux}$  em  $P$ ;
11: fim-se;
12: fim-para;
13: para  $i \leftarrow 1$  até número de mutações faça
14: selecione um cromossomo  $S_j$  em  $P$ ;
15:  $S_j \leftarrow$  mutação( $S_j$ )
16: fim-para
17: até que critério de parada seja satisfeito:
18: melhores valores de  $Q$  e  $R$ , avaliado por:
19:  $J_{min} = \min(IAE + ISE)$ .

```

Outros critérios de parada também podem ser utilizados no AG, como o número de gerações do algoritmo, mas nesse caso, para a aplicação nos controladores, a opção mais viável é a minimização da função custo, ou seja, $J_{min} = \min(IAE + ISE)$.

Alguns critérios de parada para o AG:

- número de iterações;
- função objetivo (melhor valor);
- média do valor das funções objetivo em cada iteração em comparação com a precisão do algoritmo.

Outros tipos de seleção foram testados além da seleção por torneio, mas os resultados não foram satisfatórios para o presente problema.

Os modelos das plantas para a implementação do algoritmo AG para os controladores LQI e LQG foram apresentados nas Seções 5.2 e 5.3, respectivamente.

5.4.3 *Particle Swarm Optimization - PSO*

No PSO os indivíduos são chamados agora de partículas que funcionam como um conjunto de pássaros que buscam por um formato de voo, onde é considerada a posição de cada partícula dentro do espaço, isso é em decorrência de uma curva de evolução dentro da sociedade ou conjunto de partículas. Cada partícula terá seu sucesso definido por tendência geral da população (PAIVA *et al.*, 2017).

Cada partícula no PSO apresenta um componente cognitivo que é uma relação da distância entre ela mesma e o melhor (*optimal solution*), além da componente social que é o entendimento do conjunto sobre a existência de uma dada partícula.

Realizou-se várias simulações do algoritmo PSO variando-se o número de iterações (gerações) e também o número de partículas. O melhor resultado do algoritmo foi para 30 partículas e para 50 iterações que apresentaram os melhores valores de Q e R e obteve-se uma convergência entre a melhor partícula, a solução e a média de todas as partículas.

Para o algoritmo do PSO foram utilizados os seguintes valores dos elementos, após algumas execuções do algoritmo:

- Quantidade de partículas = 30 partículas;
- Parâmetros cognitivos e sociais (taxas de aprendizado - c_1 e c_2) = 2;
- Iterações = 50;
- Fator de inércia (w) = 0,5;
- Geração da população inicial = usado um rand numa equação genérica que se restringe ao intervalo [-1, 1].

O pseudocódigo do algoritmo PSO, para encontrar as matrizes Q e R ótimas, pode ser implementado com base em algumas etapas apresentadas a seguir no Algoritmo 4.

Algoritmo 4: Pseudocódigo do PSO.

```

1: inicialize a nuvem de partículas;
2: repita
3: para  $i = 1$  até  $m$ 
4: se  $f(x_i) < f(p_i)$  então
5:  $p_i = x_i$ ;
6: se  $f(x_i) < f(g)$  então
7:  $g = x_i$ ;
8: fim se
9: fim se
10: para  $j = 1$  até  $n$ 
11:  $r_1 = rand(), r_2 = rand()$ ;
12:  $v_{ij} = wv_{ij} + c_1r_1(p_i - x_{ij}) + c_2r_2(g_j - x_{ij})$ ;
13: fim para
14:  $x_i = x_i + v_i$ ;
15: fim para
16: até que critério de parada seja satisfeito:
17: melhores valores de  $Q$  e  $R$ , avaliado por:
18:  $J_{min} = \min(IAE + ISE)$ .

```

Da mesma forma que no AG, os critérios de parada também podem ser utilizados no algoritmo do PSO, como o número de gerações do algoritmo, melhor partícula, mas nesse caso, para a aplicação nos controladores a opção mais viável é a minimização da função de custo, ou seja, $J_{min} = \min(IAE + ISE)$.

Os modelos das plantas para a implementação do algoritmo PSO para os controladores LQI e LQG foram apresentados nas Seções 5.2 e 5.3, respectivamente.

5.5 Métricas de Avaliação de Desempenho dos Controladores

As métricas de avaliação de desempenho são muito importantes para realizar uma comparação entre os controladores. Com base nas respostas dos controladores pode-se analisar alguns critérios de desempenho para os controladores em estudo neste trabalho, estes critérios que são comparações qualitativas. Os critérios de desempenho utilizados foram o tempo de subida (*rise time* - t_r), o tempo de acomodação (*settling time* - t_s) e o sobressinal (*overshoot*). O tempo de acomodação (t_s) é definido como o tempo necessário para a oscilação transitória seja amortecida, ou seja, alcance e permaneça dentro de uma faixa desejada em torno do valor de estado estacionário (SOUZA *et al.*, 2021).

Outros índices de desempenho, que também são comparações qualitativas, que serão

também usados para realizar uma melhor análise dos controladores, são: a integral do erro absoluto (*integral of absolute error - IAE*) e a integral do erro quadrado (*integral of squared error - ISE*).

A equação do *IAE* é da por,

$$IAE = \int_0^{\infty} |e(t)| dt. \quad (5.41)$$

O *ISE* é dado pela equação:

$$ISE = \int_0^{\infty} e(t)^2 dt. \quad (5.42)$$

Outro critério de desempenho muito utilizado é a variação total, *TV (Total Variation)*, utilizado por Skogestad e Postlethwaite (2007) como critério de avaliação da utilização da ação de controle a variação total da entrada do processo u , isto é,

$$TV = \sum_{k=1}^{\infty} |u(k) - u(k-1)|. \quad (5.43)$$

O índice *TV* é uma boa medida de suavidade do controle, ou seja, pode-se verificar o esforço de controle (TORRICO *et al.*, 2016).

5.6 Considerações Finais

Este capítulo apresentou a descrição dos controladores LQR/LQI, além da descrição do controlador LQG para a aplicação nos modelos identificados e dinâmico.

Além disso, foram apresentadas as descrições da função custo para ser utilizada como critério de parada das metaheurísticas AG e PSO e avaliação dos controladores LQI e LQG. Foi explicado como estas metaheurísticas otimizaram as matrizes Q e R dos controladores.

Foram apresentados os pseudocódigos e parâmetros utilizados em cada metaheurística. Por fim, e não menos importante, foram apresentadas as métricas de avaliação dos controladores, t_r , t_s , *overshoot*, *ISE*, *IAE* e *TV*. Neste trabalho serão calculados os valores de *ISE*, *IAE* e *TV* para uma entrada degrau e degrau negativo.

No próximo capítulo serão apresentados os resultados dos controladores LQI e LQG, e estes controladores com as metaheurísticas AG e PSO.

6 RESULTADOS DOS CONTROLADORES

Neste capítulo são apresentados os resultados dos controladores LQI e LQG convencionais e estes controladores otimizados com as metaheurísticas AG e PSO aplicados ao modelo identificado e modelo encontrado a partir da dinâmica do manipulador. Também é apresentada uma comparação através dos critérios de desempenho de cada controlador aplicados aos modelos identificados pelo *RLSPSO* e pela dinâmica do manipulador.

6.1 Aplicação dos Controladores LQI e LQG com as Metaheurísticas

Aqui é apresentada uma comparação dos controladores LQI e LQG com as metaheurísticas, através do tempo de processamento e função de custo para cada modelo da junta. Para os controladores LQI e LQG convencionais, foi utilizada uma busca exaustiva para encontrar os valores de Q e R , a partir da função de custo J ($J = \min(IAE + ISE)$), ou seja, quando foi encontrado o menor valor de J , o algoritmo de busca parou.

6.1.1 Modelo Identificado pelo *RLSPSO*

As tabelas 6 e 7 apresentam os valores do tempo de processamento e o valor da função de custo J ($J = \min(IAE + ISE)$) para os controladores LQI e LQG convencionais e com as metaheurísticas AG e PSO, respectivamente, aplicados ao modelo da junta identificado pelo algoritmo *RLSPSO*.

Tabela 6 – Tempo de processamento e função de custo, J , para os controladores LQI, para o modelo do *RLSPSO*.

Controlador	Tempo de Processamento (s)	J
LQI	4,3150	0,3989
LQI+AG	1,9090	0,3468
LQI+PSO	1,3368	0,2935

Fonte: Acervo do autor.

De acordo com as tabelas 6 e 7 os controladores LQI+PSO e LQG+PSO apresentaram menores valores de tempo de processamento e menor valor da função custo.

Tabela 7 – Tempo de processamento e função de custo, J , para os controladores LQG, para o modelo do *RLSPSO*.

Controlador	Tempo de Processamento (s)	J
LQG	4,6202	0,3994
LQG+AG	1,8421	0,3577
LQG+PSO	1,6369	0,2933

Fonte: Acervo do autor.

6.1.2 Modelo a partir da Dinâmica do Manipulador

Da mesma forma, as tabelas 8 e 9 apresentam os valores de tempo de processamento e o valor da função custo J para os controladores LQI e LQG convencionais e com as metaheurísticas AG e PSO, respectivamente, aplicados ao modelo da junta calculado a partir da dinâmica do manipulador.

Tabela 8 – Tempo de processamento e função de custo, J , para os controladores LQI, para o modelo dinâmico.

Controlador	Tempo de Processamento (s)	J
LQI	2,4689	3,1992
LQI+AG	1,9485	3,1991
LQI+PSO	1,5217	1,4405

Fonte: Acervo do autor.

Tabela 9 – Tempo de processamento e função de custo, J , para os controladores LQG, para o modelo dinâmico.

Controlador	Tempo de Processamento (s)	J
LQG	2,5676	1,6802
LQG+AG	2,0257	1,6812
LQG+PSO	1,7137	0,7476

Fonte: Acervo do autor.

Também, de acordo com as tabelas 8 e 9 os controladores LQI+PSO e LQG+PSO apresentaram menores valores de tempo de processamento e menor valor da função custo.

6.2 Resultados dos Controladores LQI, LQI+AG e LQI+PSO

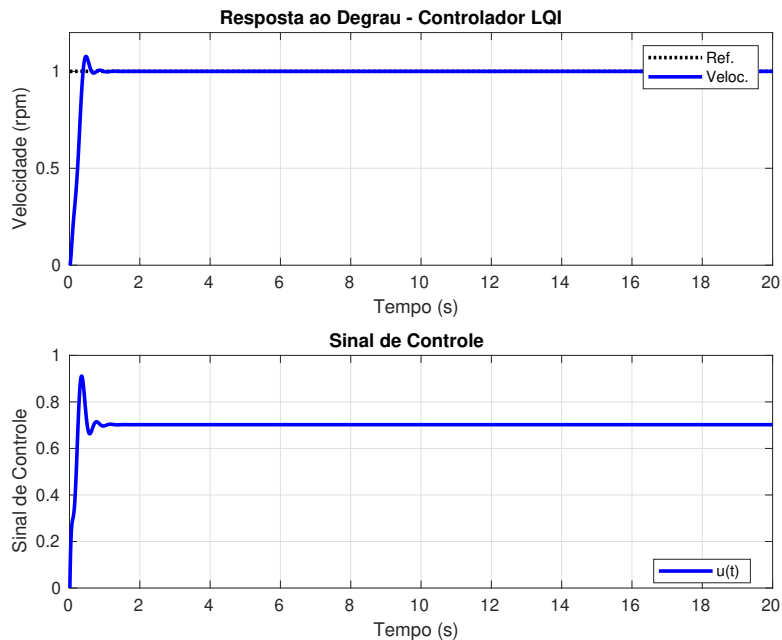
Nesta seção são apresentados os resultados do controlador LQI e LQI com as metaheurísticas AG e PSO para o modelo identificado pelo algoritmo *RLSPSO* e para o modelo

calculado a partir da dinâmica do manipulador. Para cada controlador foram aplicadas uma entrada degrau unitário e uma entrada com degrau negativo de $-0,2$ no intervalo de 10 a 15 s. Também são apresentados os sinais de controle para cada situação e estratégia de controlador.

6.2.1 Controladores Aplicados ao Modelo Identificado com RLSPSO

As Figuras 28 e 29 apresentam as respostas ao degrau e entrada com degrau negativo, respectivamente, para o controlador LQI convencional, aplicado ao modelo identificado com RLSPSO.

Figura 28 – Resposta ao degrau com o controlador LQI aplicado à junta do manipulador.

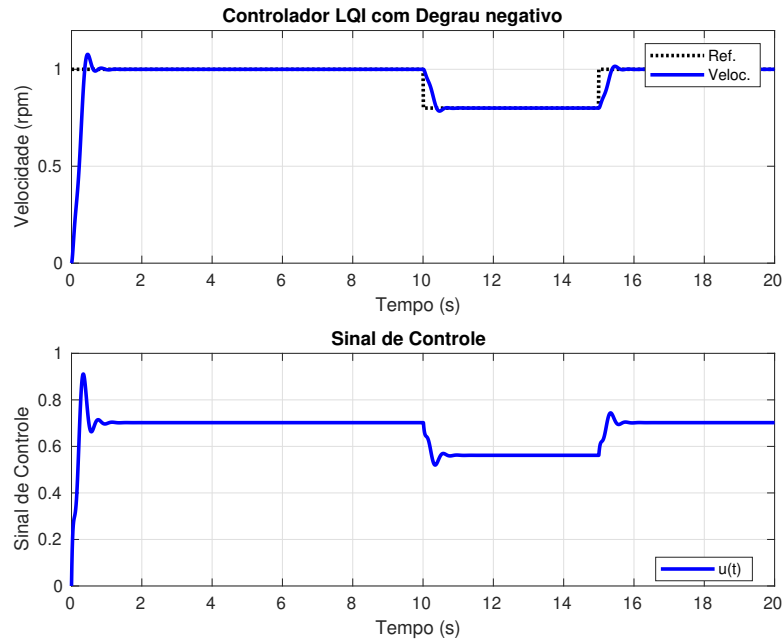


Fonte: Acervo do autor.

As Figuras 30 e 31 apresentam as respostas ao degrau e entrada com degrau negativo, respectivamente, para o controlador LQI+AG.

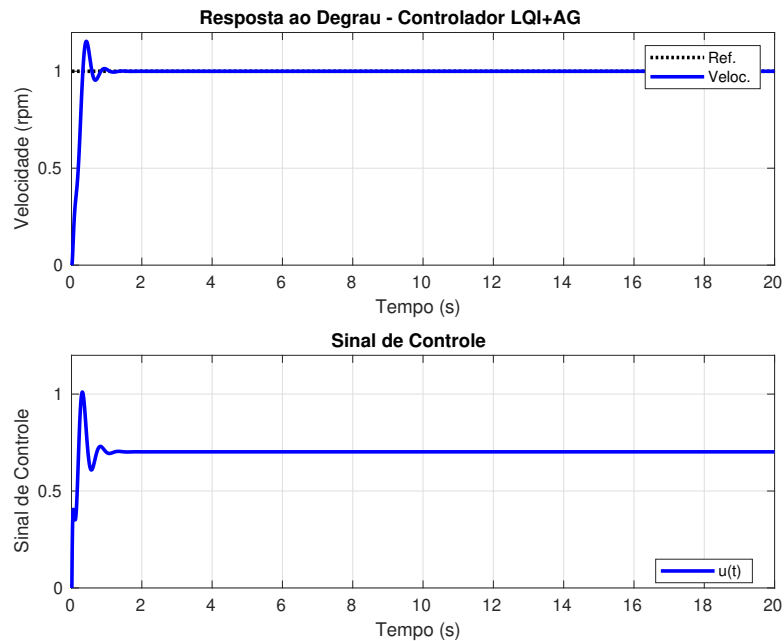
As Figuras 32 e 33 apresentam as respostas ao degrau e entrada com degrau negativo, respectivamente, para o controlador LQI+PSO.

Figura 29 – Resposta ao degrau negativo com o controlador LQI aplicado à junta do manipulador.



Fonte: Acervo do autor.

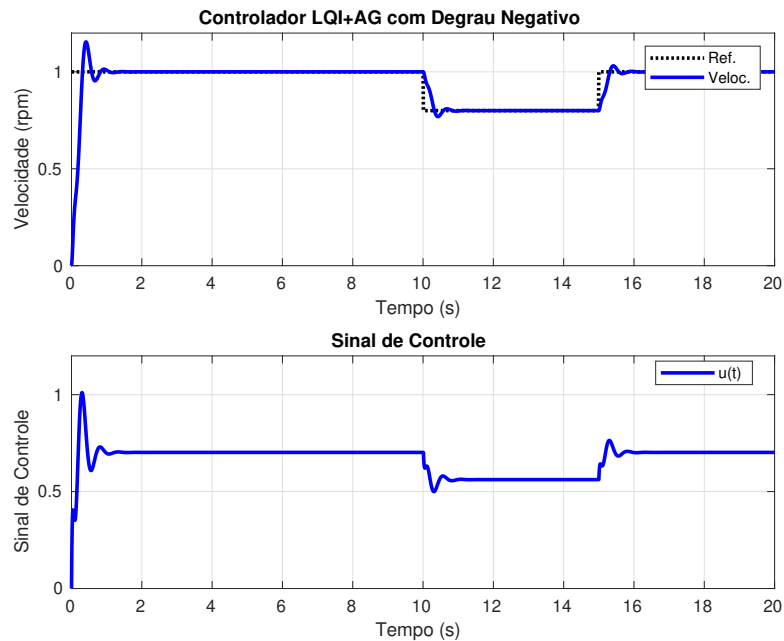
Figura 30 – Resposta ao degrau com o controlador LQI+AG aplicado à junta do manipulador.



Fonte: Acervo do autor.

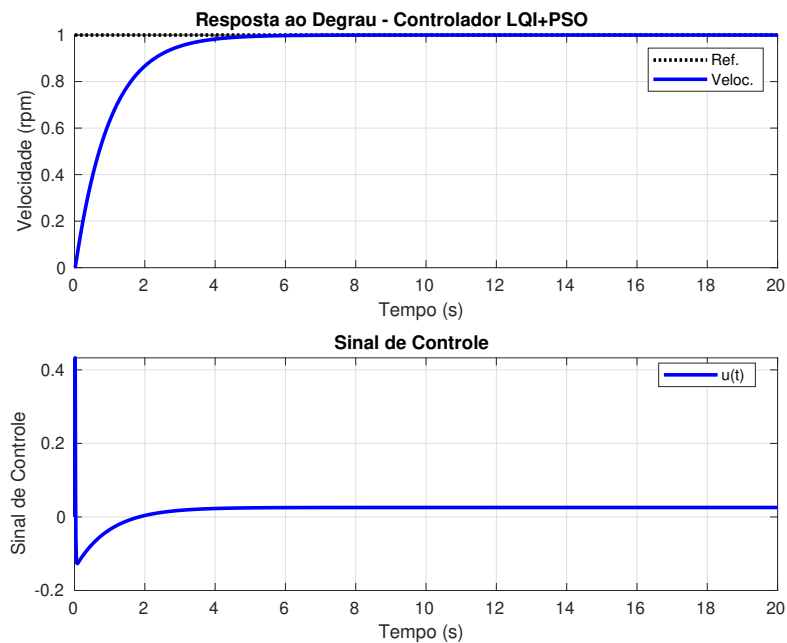
Pode-se observar que todos os controladores apresentaram resultados visualmente semelhantes, no entanto o controlador LQI+PSO, não apresentou *overshoot*.

Figura 31 – Resposta com degrau negativo com o controlador LQI+AG aplicado à junta do manipulador.



Fonte: Acervo do autor.

Figura 32 – Resposta ao degrau com o controlador LQI+PSO aplicado à junta do manipulador.

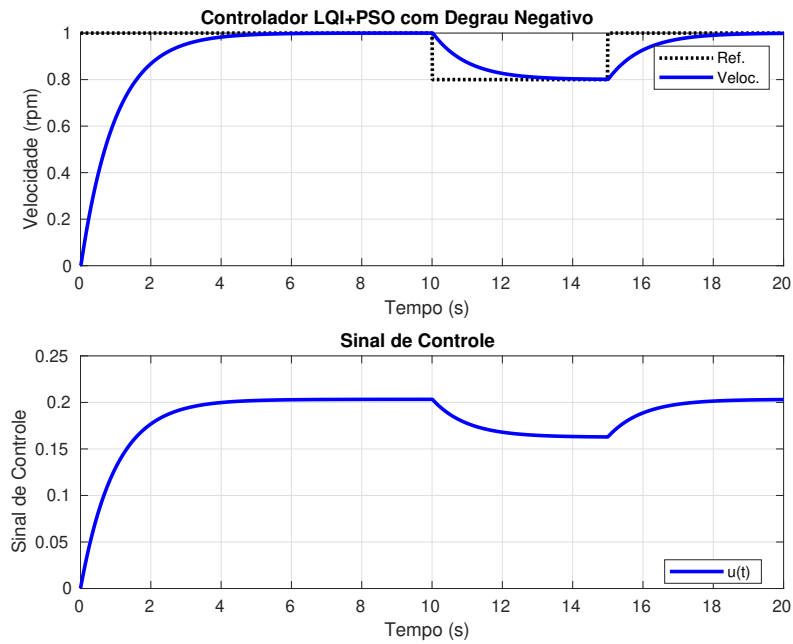


Fonte: Acervo do autor.

6.2.2 Controladores Aplicados ao Modelo Calculado a partir da Dinâmica

As Figuras 34 e 35 apresentam as respostas ao degrau e entrada com degrau negativo, respectivamente, para o controlador LQI convencional, aplicado ao modelo calculado a partir da

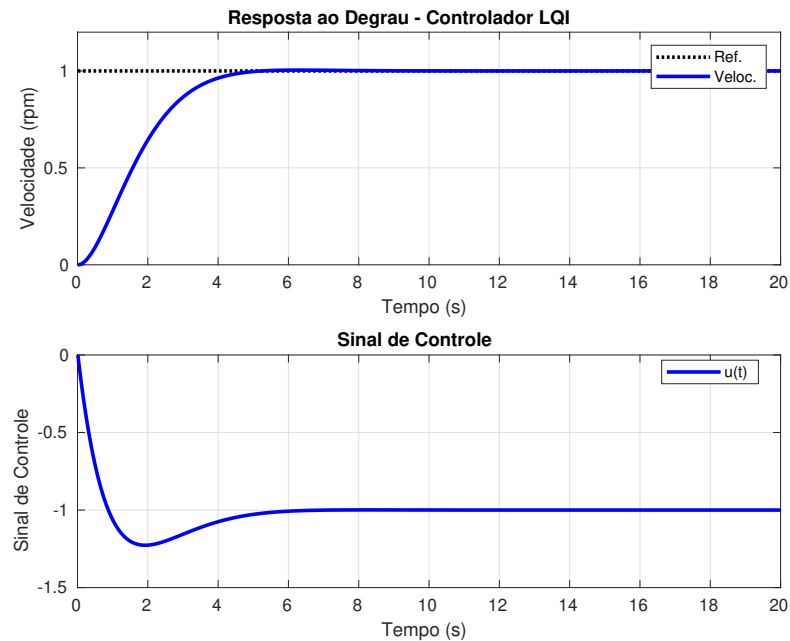
Figura 33 – Resposta ao degrau negativo com o controlador LQI+PSO aplicado à junta do manipulador.



Fonte: Acervo do autor.

dinâmica (Equação (4.48)).

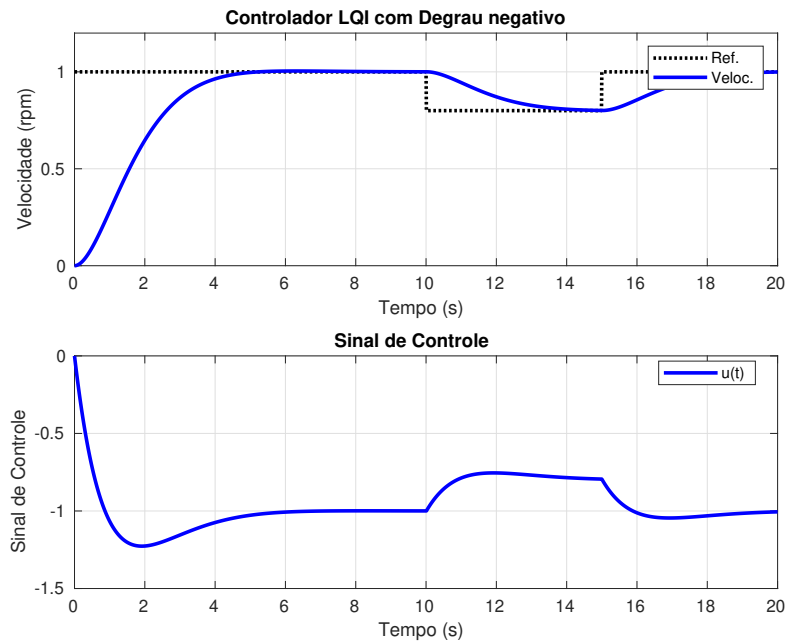
Figura 34 – Resposta ao degrau com o controlador LQI aplicado ao modelo dinâmico da junta do manipulador.



Fonte: Acervo do autor.

As Figuras 36 e 37 apresentam as respostas ao degrau e entrada com degrau nega-

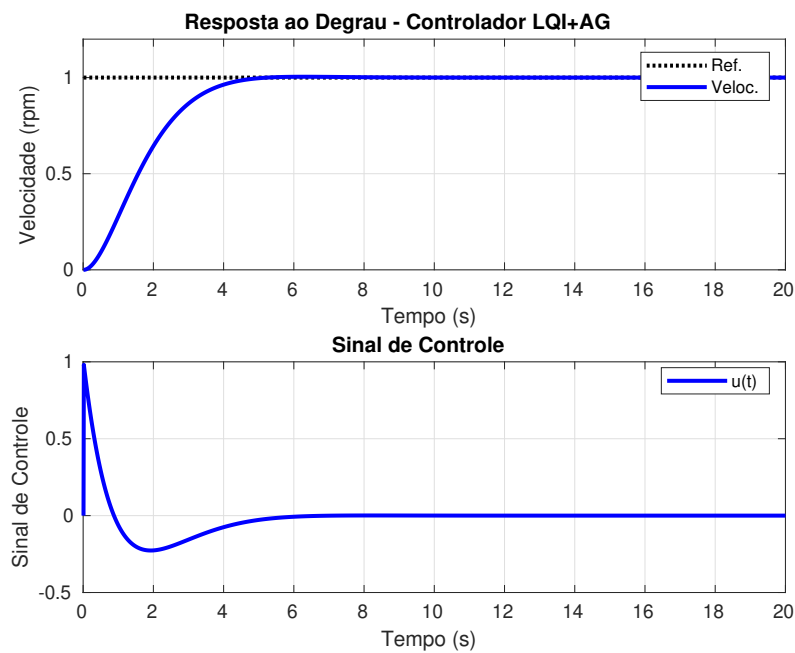
Figura 35 – Resposta ao degrau negativo com o controlador LQI aplicado ao modelo dinâmico da junta do manipulador.



Fonte: Acervo do autor.

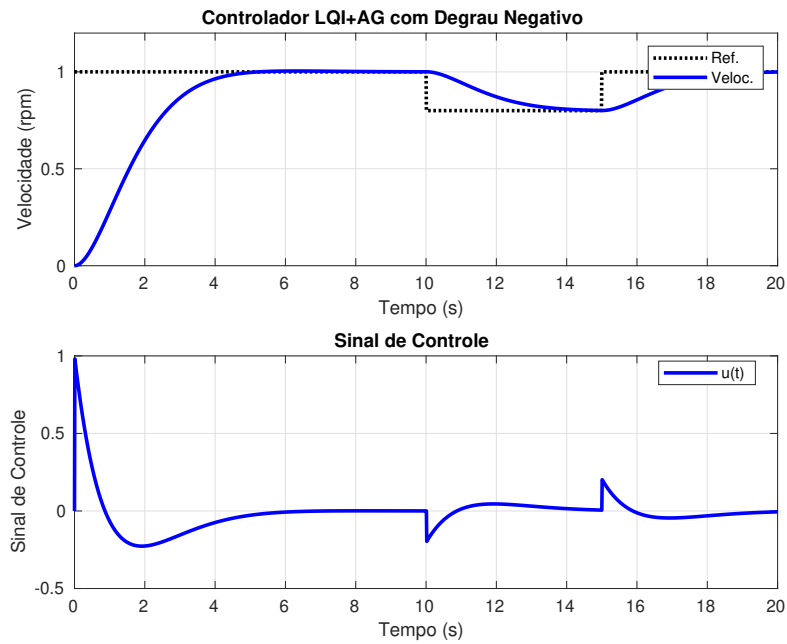
tivo, respectivamente, para o controlador LQI+AG, aplicado ao modelo dinâmico da junta do manipulador.

Figura 36 – Resposta ao degrau com o controlador LQI+AG aplicado ao modelo dinâmico da junta do manipulador.



Fonte: Acervo do autor.

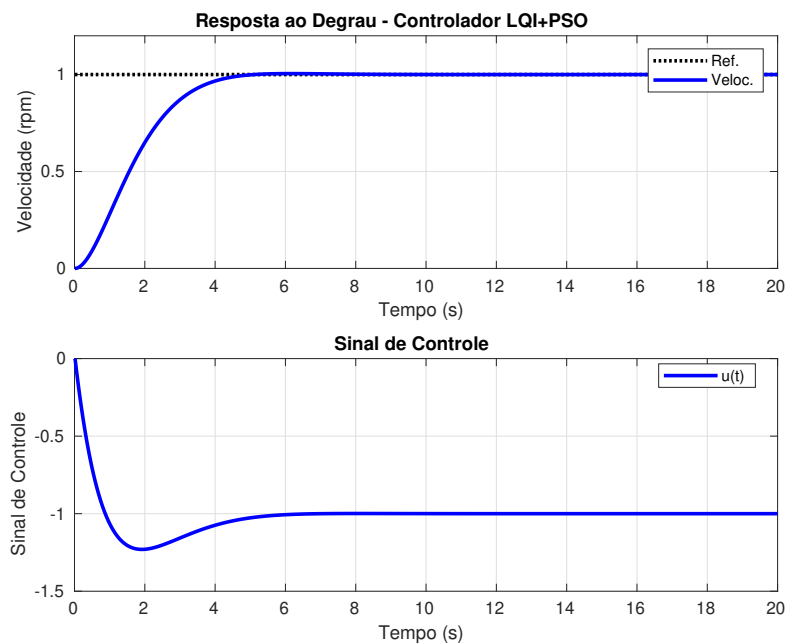
Figura 37 – Resposta com degrau negativo com o controlador LQG+AG aplicado ao modelo dinâmico da junta do manipulador.



Fonte: Acervo do autor.

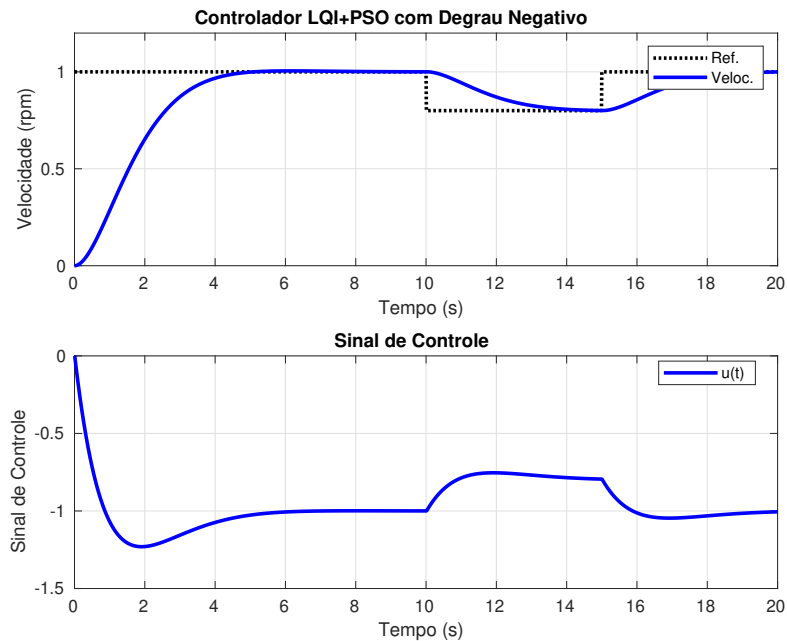
As Figuras 38 e 39 apresentam as respostas ao degrau e entrada com degrau negativo, respectivamente, para o controlador LQI+PSO, aplicado ao modelo dinâmico da junta do manipulador.

Figura 38 – Resposta ao degrau com o controlador LQI+PSO aplicado ao modelo dinâmico da junta do manipulador.



Fonte: Acervo do autor.

Figura 39 – Resposta ao degrau negativo com o controlador LQI+PSO aplicado ao modelo dinâmico da junta do manipulador.



Fonte: Acervo do autor.

Pode-se observar que todos os controladores apresentaram comportamentos semelhantes para a aplicação das entradas.

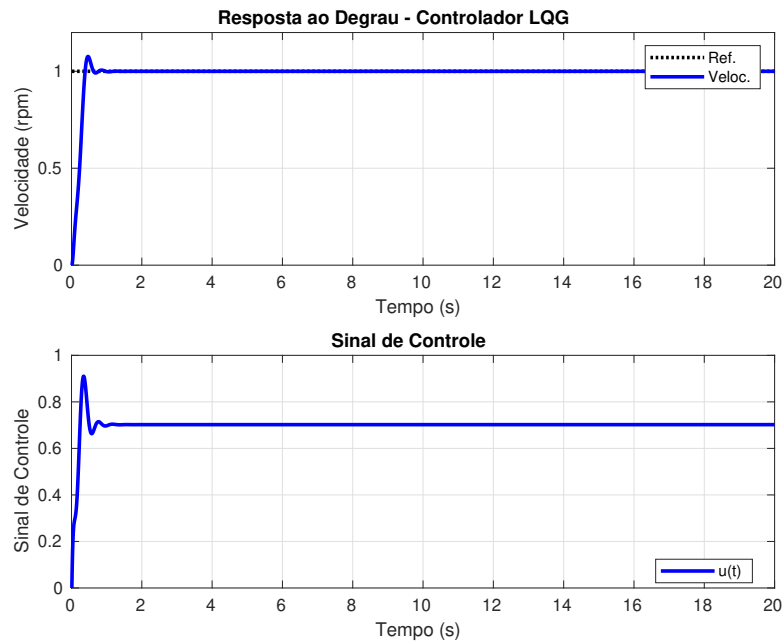
6.3 Resultados dos Controladores LQG, LQG+AG e LQG+PSO

Da mesma forma, aqui são apresentados os resultados dos controladores LQG e LQG com as metaheurísticas AG e PSO para o modelo identificado pelo algoritmo *RLSPSO* e para o modelo calculado a partir da dinâmica do manipulador. Para cada controlador foram aplicadas uma entrada degrau unitário e uma entrada com degrau negativo de $-0,2$ no intervalo de 10 a 15 s. Também são apresentados os sinais de controle para cada situação e estratégia de controlador.

6.3.1 Controladores Aplicados ao Modelo Identificado com *RLSPSO*

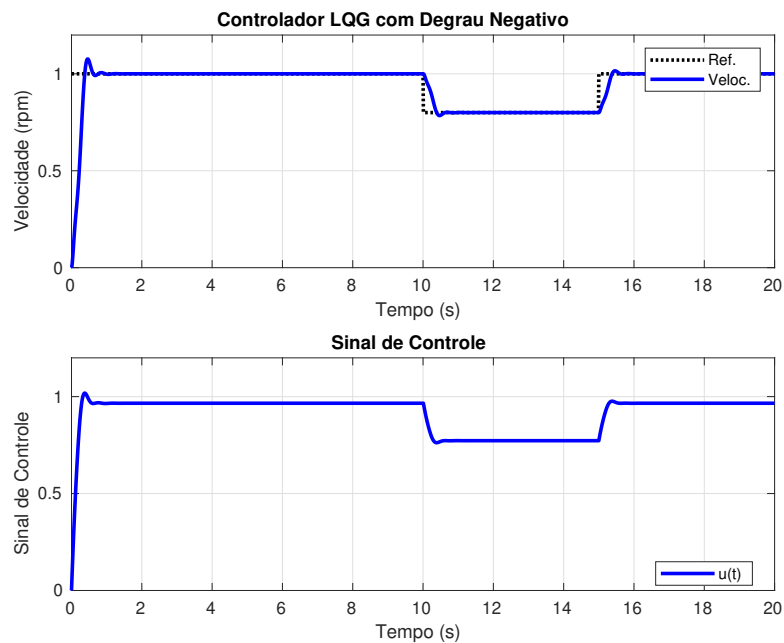
As Figuras 40 e 41 apresentam as respostas ao degrau e entrada com degrau negativo, respectivamente, para o controlador LQG convencional, aplicado ao modelo identificado pelo algoritmo *RLSPSO* (Equação (3.20)).

Figura 40 – Resposta ao degrau com o controlador LQG aplicado à junta do manipulador.



Fonte: Acervo do autor.

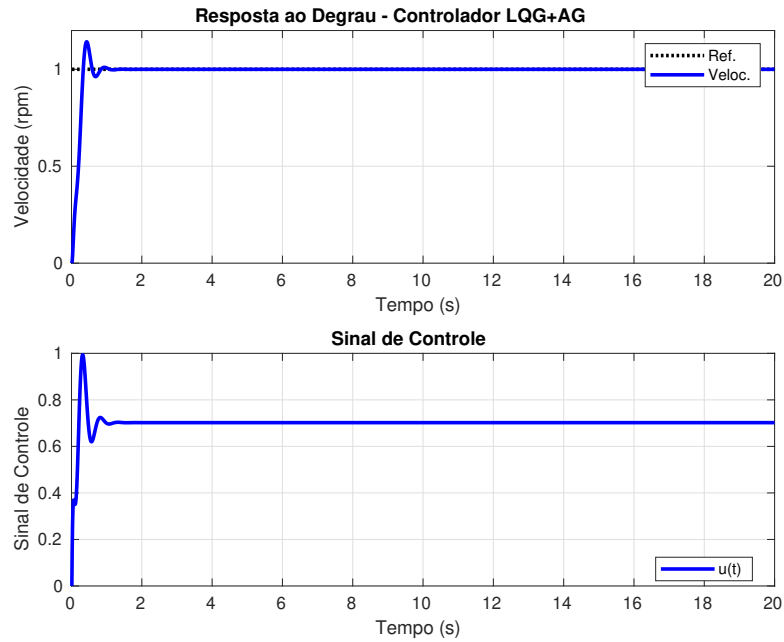
Figura 41 – Resposta ao degrau negativo com o controlador LQG aplicado à junta do manipulador.



Fonte: Acervo do autor.

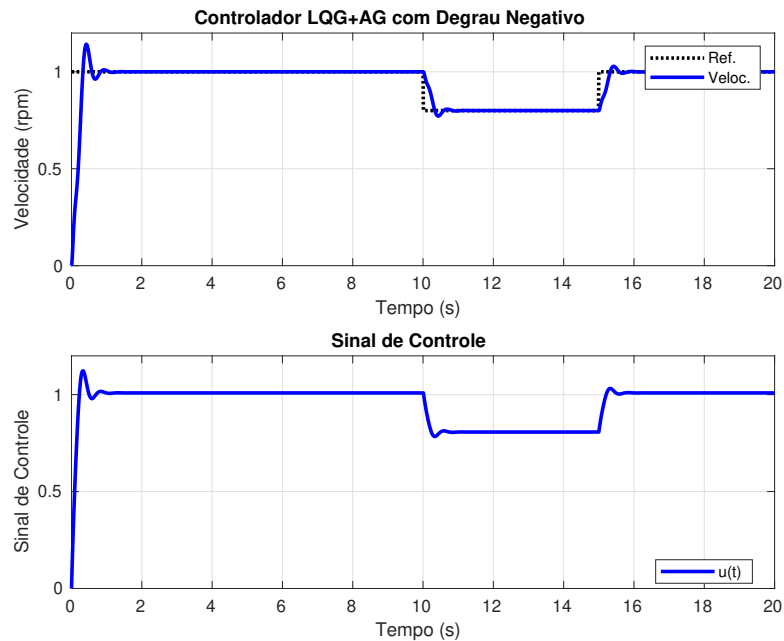
As Figuras 42 e 43 apresentam as respostas ao degrau e entrada com degrau negativo, respectivamente, para o controlador LQG+AG.

Figura 42 – Resposta ao degrau com o controlador LQG+AG aplicado à junta do manipulador.



Fonte: Acervo do autor.

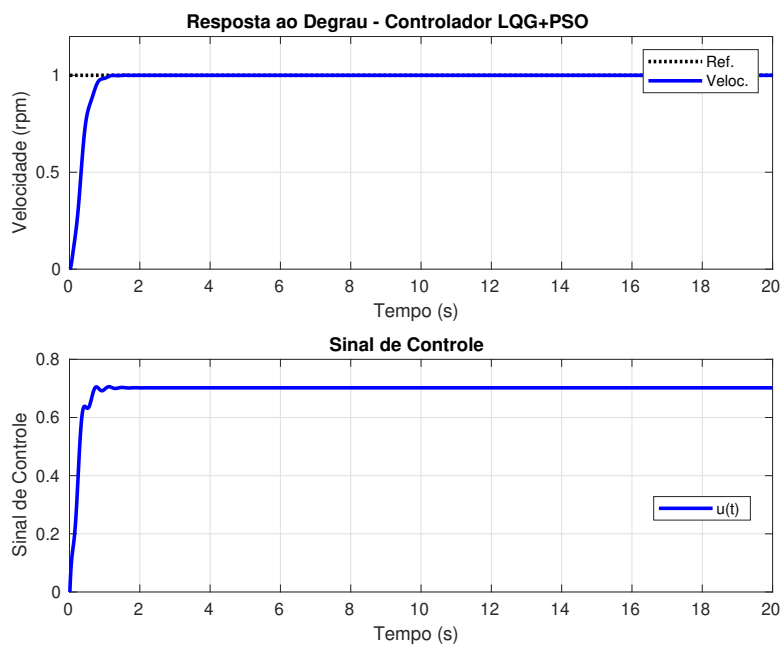
Figura 43 – Resposta com degrau negativo com o controlador LQG+AG aplicado à junta do manipulador.



Fonte: Acervo do autor.

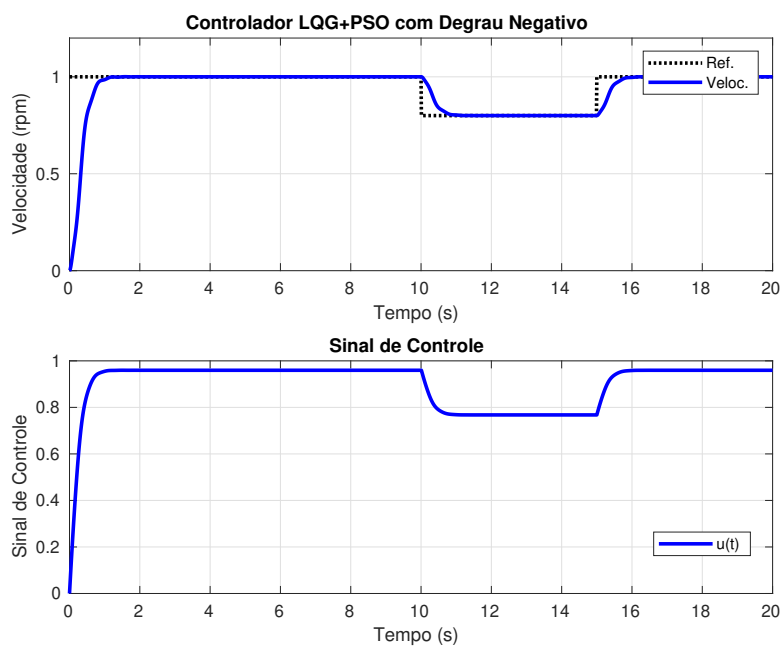
As Figuras 44 e 45 apresentam as respostas ao degrau e entrada com degrau negativo, respectivamente, para o controlador LQG+PSO.

Figura 44 – Resposta ao degrau com o controlador LQG+PSO aplicado à junta do manipulador.



Fonte: Acervo do autor.

Figura 45 – Resposta ao degrau negativo com o controlador LQG+PSO aplicado à junta do manipulador.



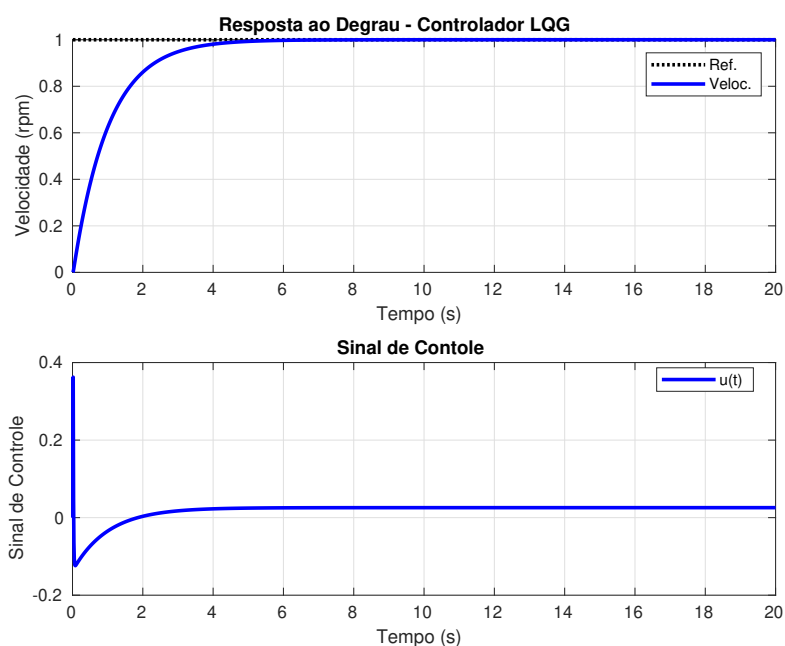
Fonte: Acervo do autor.

Aqui percebe-se também que todos os controladores apresentaram resultados semelhantes, no entanto o controlador LQG+PSO, não apresentou *overshoot*.

6.3.2 Controladores aplicados ao Modelo Calculado a partir da Dinâmica

As Figuras 46 e 47 apresentam as respostas ao degrau e entrada com degrau negativo, respectivamente, para o controlador LQG convencional, aplicado ao modelo calculado a partir da dinâmica.

Figura 46 – Resposta ao degrau com o controlador LQG aplicado ao modelo dinâmico da junta do manipulador.



Fonte: Acervo do autor.

As Figuras 48 e 49 apresentam as respostas ao degrau e entrada com degrau negativo, respectivamente, para o controlador LQG+AG, aplicado ao modelo dinâmico da junta do manipulador.

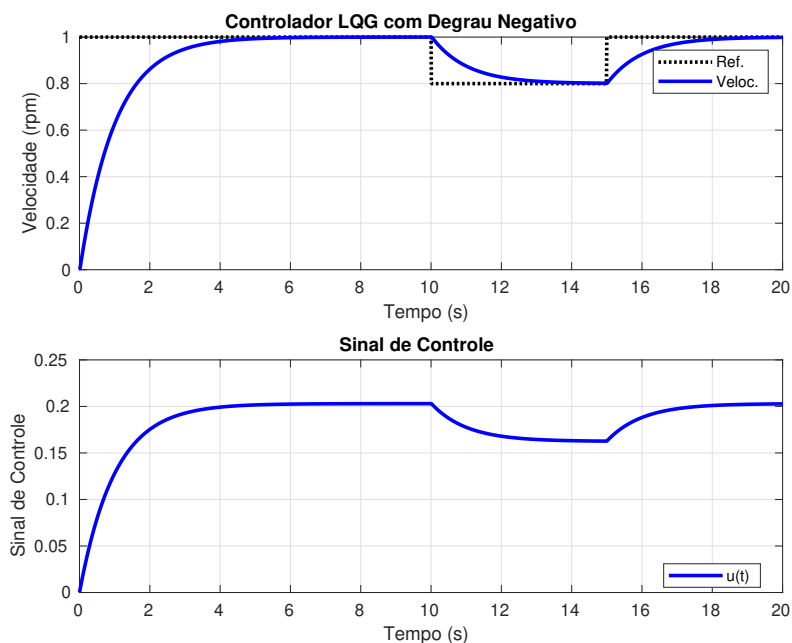
As Figuras 50 e 51 apresentam as respostas ao degrau e entrada com degrau negativo, respectivamente, para o controlador LQG+PSO, aplicado ao modelo calculado a partir da dinâmica.

Pode-se observar, aqui também, que todos os controladores apresentaram comportamentos semelhantes para a aplicação das entradas.

6.4 Comparação entre os Controladores LQI e LQG com as Metaheurísticas

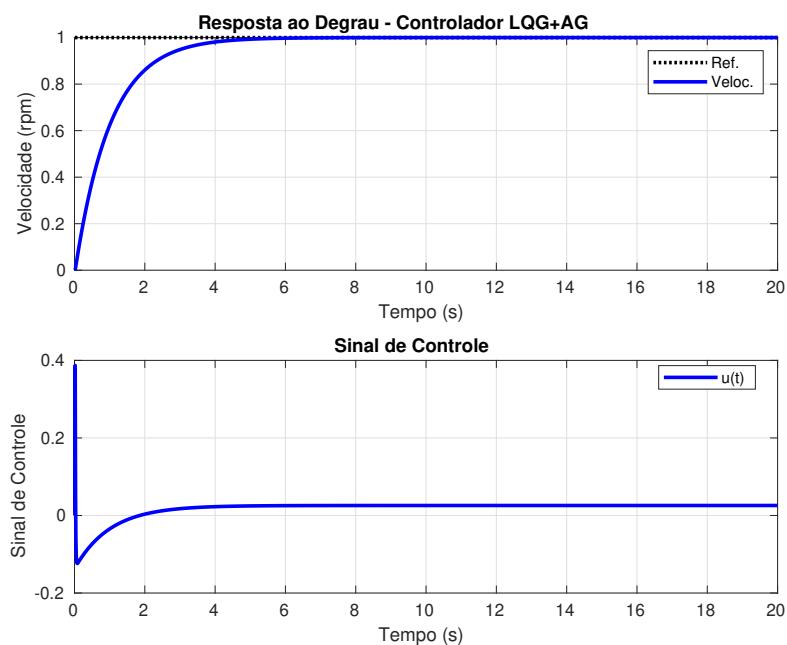
Com base nas respostas dos controladores aplicaram-se os critérios de desempenho apresentados na Seção 5.5, tais como tempo de subida (t_r), tempo de acomodação (t_s), *overshoot*,

Figura 47 – Resposta ao degrau negativo com o controlador LQG aplicado ao modelo dinâmico da junta do manipulador.



Fonte: Acervo do autor.

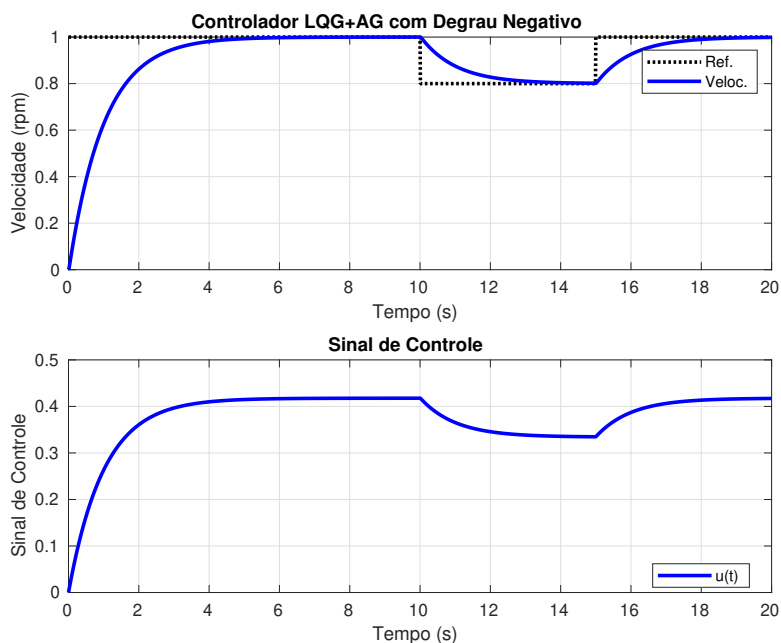
Figura 48 – Resposta ao degrau com o controlador LQG+AG aplicado ao modelo dinâmico da junta do manipulador.



Fonte: Acervo do autor.

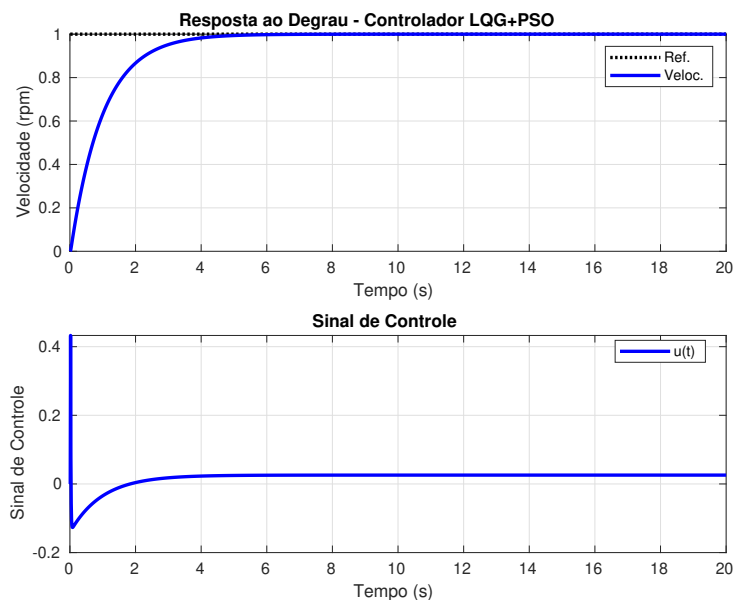
IAE , ISE e TV . Os valores de IAE , ISE , e TV foram calculados para a entrada degrau ($ISE - step$, $IAE - step$ e $TV - step$) e para entrada com degrau negativo ($ISE - neg$, $IAE - neg$ e $TV - neg$). Estes índices foram obtidos para os controladores aplicados aos dois modelos da junta.

Figura 49 – Resposta com degrau negativo com o controlador LQG+AG aplicado ao modelo dinâmico da junta do manipulador.



Fonte: Acervo do autor.

Figura 50 – Resposta ao degrau com o controlador LQG+PSO aplicado ao modelo dinâmico da junta do manipulador.

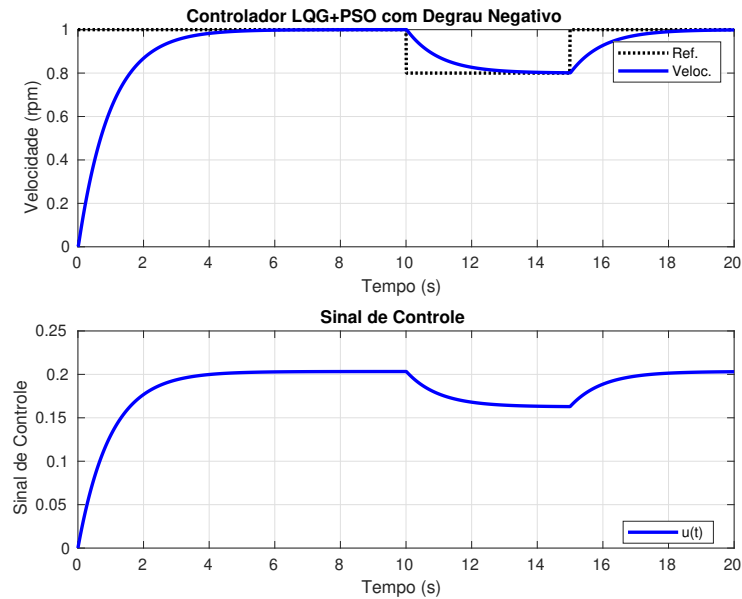


Fonte: Acervo do autor.

6.4.1 Modelo Identificado com RLSPSO

A Tabela 10 apresenta os critérios de desempenho, citados anteriormente, para as respostas dos controladores LQI, LQI+AG e LQI+PSO para o controle de velocidade da junta do

Figura 51 – Resposta ao degrau negativo com o controlador LQG aplicado ao modelo dinâmico da junta do manipulador.



Fonte: Acervo do autor.

manipulador para o modelo identificado pelo *RLSPSO*.

Tabela 10 – Comparação através dos critérios de desempenho dos controladores LQI, LQI+AG e LQI+PSO aplicados ao modelo identificado pelo *RLSPSO*.

Controlador	t_r (s)	t_s (s)	overshoot (%)	IAE_{step}	IAE_{neg}	ISE_{step}	ISE_{neg}	TV_{step}	TV_{neg}
LQI	0,2848	0,5733	7,7287	0,2390	0,3369	0,1599	0,1731	1,2407	1,7370
LQI+AG	0,2522	0,7693	15,4749	0,2204	0,3108	0,1264	0,1369	1,6947	2,3726
LQI+PSO	0,5618	0,8882	0,0000	0,1758	0,2472	0,1176	0,1272	0,7598	1,0637

Fonte: Acervo do autor.

A Tabela 11 apresenta os critérios de desempenho, citados anteriormente, para as respostas do controladores LQG, LQG+AG e LQG+PSO para o controle de velocidade da junta do manipulador.

Tabela 11 – Comparação através dos critérios de desempenho dos controladores LQG, LQG+AG e LQG+PSO aplicados ao modelo identificado pelo *RLSPSO*.

Controlador	t_r (s)	t_s (s)	overshoot (%)	IAE_{step}	IAE_{neg}	ISE_{step}	ISE_{neg}	TV_{step}	TV_{neg}
LQG	0,2851	0,5732	7,6555	0,2392	0,3371	0,1602	0,1734	1,2386	3,0293
LQG+AG	0,2579	0,7610	14,2196	0,2241	0,3159	0,1336	0,1447	1,5493	5,5320
LQG+PSO	0,5615	0,8864	0,0000	0,1757	0,2470	0,1176	0,1270	0,7601	1,6128

Fonte: Acervo do autor.

Através dos critérios de desempenho aplicados aos controladores LQI e LQG com as metaheurísticas AG e PSO apresentados nas tabelas 10 e 11, para a junta do manipulador, pode-se

verificar que os controladores apresentaram resultados satisfatórios para a aplicação proposta. Pode-se observar que os controladores LQI+PSO e LQG+PSO apresentaram os melhores índices de desempenho para ambas as situações de resposta ao degrau.

Observa-se também que o critério de desempenho TV está relacionado ao sinal dos controladores, onde o controlador LQI+PSO e o LQG+PSO apresentaram valores bem inferiores aos demais controladores. Ele avalia a eficiência dos controladores em relação à energia gasta pelo sistema, e sua verificação é muito importante. Com isso, pode-se dizer que os controladores LQI+PSO e LQG+PSO foram melhores dentre os demais aqui comparados, e seria a melhor escolha para a aplicação proposta.

6.4.2 Modelo Calculado a partir da Dinâmica

A Tabela 12 apresenta os critérios de desempenho, citados anteriormente, para as respostas do controladores LQI, LQI+AG e LQI+PSO para o controle de velocidade da junta do manipulador para o modelo encontrado a partir da dinâmica.

Tabela 12 – Comparação através dos critérios de desempenho dos controladores LQI, LQI+AG e LQI+PSO aplicados ao modelo dinâmico da junta.

Controlador	t_r (s)	t_s (s)	<i>overshoot</i> (%)	IAE_{step}	IAE_{neg}	ISE_{step}	ISE_{neg}	TV_{step}	TV_{neg}
LQI	2,7344	4,3456	0,4333	1,9290	2,7000	1,2702	1,3728	2,4169	3,3762
LQI+AG	2,7344	4,3456	0,4333	1,9290	2,7001	1,2702	1,3727	1,4555	2,0303
LQI+PSO	2,6977	4,2716	0,4333	0,8681	1,2151	0,5724	0,6186	1,4429	2,0009

Fonte: Acervo do autor.

A Tabela 13 apresenta os critérios de desempenho, citados anteriormente, para as respostas do controladores LQG, LQG+AG e LQG+PSO para o controle de velocidade da junta do manipulador, a partir da dinâmica.

Tabela 13 – Comparação através dos critérios de desempenho dos controladores LQG, LQG+AG e LQG+PSO aplicados ao modelo dinâmico da junta.

Controlador	t_r (s)	t_s (s)	<i>overshoot</i> (%)	IAE_{step}	IAE_{neg}	ISE_{step}	ISE_{neg}	TV_{step}	TV_{neg}
LQG	2,2080	3,9421	0,0000	1,1165	1,5605	0,5637	0,6089	1,1474	5,82950
LQG+AG	2,2080	3,9416	0,0000	1,1170	1,5612	0,5642	0,6095	1,1281	2,8334
LQG+PSO	2,1575	3,8528	0,0000	0,4966	0,6943	0,2511	0,2712	1,1148	2,8321

Fonte: Acervo do autor.

Com relação aos critérios de desempenho aplicados aos controladores LQI e LQG com as metaheurísticas AG e PSO apresentados nas Tabelas 12 e 13, para o modelo dinâmico

da junta do manipulador, pode-se verificar que o comportamento entre os controladores foram semelhantes com relação ao tempo de subida, tempo de acomodação e *overshoot*. Com relação aos índices de desempenho *ISE*, *IAE* e *TV* os controladores com PSO apresentaram menores valores. No entanto para o LQG convencional, este apresentou-se melhor em alguns índices que o LQG+AG, nos critérios *IAE* e *ISE*. Pode-se destacar também que os valores de *TV*, para o degrau com valor negativo, foram marcados em relação aos demais, para os controladores LQG, LQG+AG e LQG+PSO, sendo que este último apresentou menor valor entre os demais.

6.5 Discussões

Por meio dos resultados apresentados, pode-se afirmar que todos os controladores apresentaram resultados satisfatórios para a aplicação aqui proposta. Na primeira comparação apresentada em relação ao tempo de processamento e função custo, para os controladores LQI, o LQI+PSO apresentou menores valores para essas comparações, tanto para o modelo identificado pelo *RLSPSO* como para o modelo dinâmico. Com relação aos controladores LQG, o LQG+PSO apresentou também os menores valores para o tempo de processamento e função custo. Essa comparação é importante para ter-se uma ideia do tempo necessário para encontrar os valores de Q e R ótimos para a aplicação na junta do manipulador, e para verificar a eficiência dos controladores em relação à função custo J , pois esta é a soma do *IAE* e *ISE*, isto é, quanto menor o valor de J , mais eficiente é o controlador, para aplicação aqui proposta.

Foram realizadas também, comparações através de alguns critérios de desempenho como: tempo de subida (t_r), o tempo de acomodação (t_s), sobressinal, *IAE*, *ISE* e *TV*. Esses critérios são importantes para mostrar o comportamento de cada controlador. Pode-se afirmar que para o modelo identificado com o *RLSPSO*, os controladores apresentaram tempo de subida e de acomodação bem menores que os controladores para o modelo encontrado a partir da dinâmica. No entanto os controladores LQI, LQI+AG, LQG e LQG+AG apresentaram valores maiores de sobressinal maior que zero, diferente dos controladores LQI+PSO e LQG+PSO que não apresentaram valores de sobressinal. Os valores de *IAE*, *ISE* e *TV* para os controladores LQI e LQG com AG e PSO apresentaram valores baixos, de modo geral. Sendo que o LQI+PSO e LQG+PSO apresentaram valores menores que os controladores LQI, LQI+AG, LQG e LQG+AG.

Para o modelo encontrado a partir da dinâmica do manipulador o sistema apresentou-se mais lento que o modelo identificado, apresentando valores de tempo de subida e tempo de acomodação maiores. No entanto, os valores de sobressinal foram muito parecidos para todos os

controladores aplicados ao modelo dinâmico. Em relação ao *IAE*, *ISE* e *TV*, todos os controladores apresentaram valores mais elevados que os controladores aplicados ao modelo identificado pelo *RLSPSO*. Pode-se destacar ainda que os valores de *TV*, para todos os controladores, foram bem mais elevados, para o modelo dinâmico que o para o modelo a partir do *RLSPSO*. Isto significa que para o modelo dinâmico os esforço de controle é bem maior para este modelo encontrado.

No geral, os controladores com o PSO, ou seja, LQI+PSO e LQG+PSO apresentaram melhor desempenho em relação aos demais, tanto para o modelo identificado pelo *RLSPSO*, como para o modelo dinâmico. Esse controlador seria a melhor opção entre os demais para o controle da planta. Pode-se afirmar também que, os controladores aplicados ao modelo identificado pelo algoritmo *RLSPSO*, mostraram-se mais eficientes, através da aplicação dos controladores e critérios de desempenho, que os controladores aplicados ao modelo dinâmico. Diante destes cenários, pode-se afirmar ainda que todos os controladores apresentaram resultados satisfatórios para a aplicação ao qual foram submetidos.

6.6 Considerações Finais

Neste capítulo foram apresentados os resultados dos controladores, onde apresentou-se uma comparação entre cada controlador com relação ao tempo de processamento e função custo. Apresentou-se também as curvas após a aplicação de um degrau e degrau unitário com valor negativo de $-0,2$ no instante de tempo de 10 a 15 s.

Para cada controlador foram encontrados os valores dos índices de desempenho tempo de subida (t_r), o tempo de acomodação (t_s), sobressinal, *IAE*, *ISE* e *TV*. Onde estes valores foram apresentados em tabelas para cada controlador (LQI, LQI+AG, LQI+PSO, LQG, LQG+AG, LQG+PSO) para cada modelo encontrado da junta do manipulador, neste trabalho, ou seja, modelo identificado pelo *RLSPSO* e modelo dinâmico.

No próximo capítulo são apresentadas as conclusões deste trabalho, bem como os trabalhos futuros.

7 CONCLUSÕES E TRABALHOS FUTUROS

Neste trabalho foi apresentada uma proposta de melhoria do algoritmo de MQR com PSO aplicada à identificação do modelo de uma junta de um manipulador robótico. Também foi utilizado o modelo dinâmico do manipulador com os dados experimentais, e a esses modelos identificados foram aplicados os controladores LQI e LQG com as metaheurísticas AG e PSO. A seguir são apresentadas as conclusões deste trabalho e as sugestões de trabalhos futuros.

7.1 Contribuições

Neste trabalho, buscou-se mostrar a implementação do algoritmo de mínimos quadrados recursivos com otimização por enxame de partículas, aplicado à junta de um manipulador robótico, o modelo dinâmico e controladores LQI e LQG com metaheurísticas.

Este trabalho teve como contribuições, apresentar os resultados da implementação do algoritmo RLSPSO, desenvolvimento do modelo dinâmico, e a implementação dos controladores com as metaheurísticas. Foram realizadas os experimentos para a coleta de dados para a aplicação dos algoritmos e cálculo do modelo dinâmico.

Outro ponto de destaque é a aplicação dos algoritmos metaheurísticos para a otimização dos controladores LQI e LQG, onde estes algoritmos encontram os valores das matrizes Q e R dos controladores.

Com a realização deste trabalho foi possível desenvolver modelos de identificação e controladores aplicados à uma junta de um manipulador robótico acionado por motor de indução trifásico.

7.2 Conclusões do Trabalho

Como apresentado, na revisão da literatura, Seção 1.2, muitas pesquisas foram realizadas com aplicações de algoritmos metaheurísticos aplicados para otimizar métodos de identificação e controladores, onde se mostraram eficientes nestas aplicações. Também destacou-se a utilização do modelo dinâmico de manipuladores robóticos. Nesse contexto pode-se justificar que este trabalho está fundamentado em temas atuais e de relevância na comunidade científica.

Com relação a identificação do modelo com a utilização de dados experimentais, foram utilizados os algoritmos de MQ, MQR e RLSPSO, onde este último apresentou melhor resultado. De acordo com as métricas utilizadas, o valor do R^2 foi de **0,9962** para o RLSPSO,

0,9709 para o MQ e 0,9656 para o MQR; para a métrica *RMSE*, o RLSPSO obteve valor **0,1546**, 0,1560 para o MQ, e 0,1921 para o MQR. Foi comparado também o custo computacional onde o RLSPSO (36,8397 s) obteve menor tempo que o MQR (42,1222 s), e o MQ (2,8168 s) obteve o menor tempo que os demais. Realizou-se também uma comparação em relação à complexidade de cada algoritmo, onde o RLSPSO foi menos complexo que o MQR, e o MQ menos complexos que os demais.

Destacou-se também a utilização do modelo dinâmico do manipulador, onde com a utilização de dados experimentais, conseguiu-se encontrar um modelo em espaço de estados onde a corrente foi a entrada, e a velocidade a saída do modelo. Para o ponto de vista da robótica de manipuladores, essa descrição do modelo é importante pois, pode-se aplicar o modelo dinâmico em função da corrente do atuador, que no caso foi substituído pelo torque. Para o modelo encontrado foi traçada a curva de velocidade do modelo e comparou-se com os dados experimentais, onde o valor de R^2 foi 0,8900 e o *RMSE* foi 0.2672.

De acordo com os resultados apresentados aqui, através dos controladores implementados, pode-se afirmar que as metaheurísticas, AG e PSO, utilizadas neste trabalho se mostraram adequadas na otimização dos controladores LQI e LQG. Estes também se mostraram adequados para a aplicação aqui proposta. Pode-se ainda destacar que os controladores LQI+PSO e LQG+PSO, apresentaram os melhores resultados em relação ao tempo para ajustar os parâmetros desses controladores e no valor do custo, o seja, o PSO foi o mais adequado para as aplicações. Para os critérios de desempenho avaliados para os controladores, o LQI+PSO e LQG+PSO, também obtiveram melhores resultados que os demais, tanto para o modelo identificado como para o modelo dinâmico da junta. No entanto, os valores dos critérios de desempenho foram melhores para o modelo identificado pelo RLSPSO.

Com isso, pode-se realizar novas pesquisas no manipulador, como uma identificação de todas as juntas em funcionamento, e aplicação de técnicas de controle inteligente ao manipulador.

7.3 Sugestões de Trabalhos Futuros

Como trabalhos futuros, propõe-se realizar o acionamento de todas as juntas do manipulador, implementando os controladores aqui propostos, tornando o manipulador um sistema de maior complexidade, onde pode ser realizadas melhorias na geração de caminhos e trajetórias livres de colisão aplicadas à manipuladores industriais ou em ambientes com vários

robôs, aplicando os conceitos de robótica colaborativa, tópico da Indústria 4.0. Dentre os quais pode-se citar ainda:

1. Implementar os controladores em todas as juntas do manipulador;
2. Modelar/identificar o manipulador com todas as juntas, tornando um sistema multivariável;
3. Pesquisar novas técnicas de identificação e controle com inteligência computacional;
4. Implementar técnicas de controle adaptativo;
5. Implementar algoritmos de geração de trajetórias livre de colisão como campos potenciais artificiais e rotas probabilísticas;
6. Implementar sistema de visão computacional para realizar trajetórias livres de colisão, com aplicações em tempo real.

7.4 Publicações

Apresenta-se aqui as publicações, até o momento, em congressos/conferências e periódicos.

7.4.1 *Artigos publicados em congressos e conferências*

1. BATISTA J. G., COSTA J. R., SOUZA, D. A., FILGUEIRAS, L. V. O., NOGUEIRA, J. N., BARBOSA, A. S. e DOS REIS, L. L. N. Modelagem Dinâmica e Simulação de um Controlador PID e LQR para um Manipulador Cilíndrico. 14º Simpósio Brasileiro de Automação Inteligente - SBAI, 2019, Ouro Preto. Anais eletrônicos, Campinas, GALOÁ, 2020. Disponível em: <<https://repositorio.ufc.br/handle/riufc/65293>>. Acesso em: 07 jun. 2023.
2. SOUZA, D. A., BATISTA J. G., DOS REIS, L. L. N., NOGUEIRA, J. N., OLIVEIRA, R. J., RAMOS, K. M. e BARBOSA, A. S. Identificação com uma Variação de Mínimos Quadrados Recursivo aplicado a um Manipulador Robótico. 14º Simpósio Brasileiro de Automação Inteligente - SBAI, 2019, Ouro Preto. Anais eletrônicos, Campinas, GALOÁ, 2020. Disponível em: <<https://repositorio.ufc.br/handle/riufc/65296>>. Acesso em: 07 jun. 2023.
3. BATISTA, J. G., SOUZA, D. A., DOS REIS, L. L., FILGUEIRAS, L. V., RAMOS, K. M., BARBOSA, A. S., & CORREIA, W. B. Performance comparison between the PID and LQR controllers applied to a robotic manipulator joint. In IECON 2019-45th Annual

- Conference of the IEEE Industrial Electronics Society (Vol. 1, pp. 479-484). IEEE, 2019. Disponível em: <<https://doi.org/10.1109/IECON.2019.8927059>>. Acesso em: 07 jun. 2023.
4. BATISTA, J. G., SOUZA, D. A., DOS REIS, L. L. N., BARBOSA, A. S., & SANTANA, K. A. Modelo Cinemático e Dinâmico de um Manipulador 3-GDL. DINCON - Conferência Brasileira de Dinâmica, Controle e Aplicações. São Carlos, 2019. Disponível em: <<https://www.editoracientifica.com.br/artigos/modelo-cinematico-e-dinamico-de-um-manipulador>>. Acesso em: 07 jun. 2023.
 5. SOUZA, D., BATISTA, J., DOS REIS, L., SANTANA, K., OLIVEIRA, J. R., NOGUEIRA, J. N., BARBOSA, A. S. & ARAÚJO, R. . Short Paper: Identification by Recursive Least Squares with RMO Applied to a Robotic Manipulator. In 2019 Second International Conference on Artificial Intelligence for Industries (AI4I) (pp. 95-96). IEEE, 2019. Disponível em: <<https://doi.org/10.1109/AI4I46381.2019.00031>>. Acesso em: 07 jun. 2023.

7.4.2 Artigos publicados em periódicos

1. SOUZA, D. A.; BATISTA, J. G.; DOS REIS, L. L. N.; BARBOSA, A. S. PID Controller with novel PSO applied to a joint of a robotic manipulator. Journal of the Brazilian Society of Mechanical Sciences and Engineering, Springer, v. 43, n. 8, p. 1–14, 2021. Disponível em: <<https://doi.org/10.1007/s40430-021-03092-4>>. Acesso em: 07 jun. 2023.
2. SOUZA, D. A.; BATISTA, J. G.; VASCONCELOS, F. J.; DOS REIS, L. L. N.; MACHADO, G. F.; COSTA, J. R.; NOGUEIRA, J.; SILVA, J. L.; RIOS, C. S.; BARBOSA, A. Identification by recursive least squares with kalman filter (RLS-KF) applied to a robotic manipulator. IEEE Access, IEEE, v. 9, p. 63779–63789, 2021. Disponível em: <<https://doi.org/10.1109/ACCESS.2021.3074419>>. Acesso em: 07 jun. 2023.
3. SOUZA, D. A.; MESQUITA, V. A. de; DOS REIS, L. L.; SILVA, W. A.; BATISTA, J. G. Optimal LQI and PID synthesis for speed control of switched reluctance motor using metaheuristic techniques. International Journal of Control, Automation and Systems, Springer, v. 19, n. 1, p. 221–229, 2021. Disponível em: <<https://doi.org/10.1007/s12555-019-0911-x>>. Acesso em: 07 jun. 2023.
4. BATISTA, J.; SOUZA, D.; DOS REIS, L. dos; BARBOSA, A.; ARAÚJO, R. Dynamic model and inverse kinematic identification of a 3-dof manipulator using rlspso. Sensors,

- Multidisciplinary Digital Publishing Institute, v. 20, n. 2, p. 416, 2020. Disponível em: <<https://doi.org/10.3390/s20020416>>. Acesso em: 07 jun. 2023.
5. BATISTA, J., SOUZA, D., SILVA, J., RAMOS, K., COSTA, J., DOS REIS, L., & BRAGA, A. Trajectory Planning Using Artificial Potential Fields with Metaheuristics. *IEEE Latin America Transactions*, 18(05), 914-922, 2020. Disponível em: <<https://doi.org/10.1109/TLA.2020.9082920>>. Acesso em: 07 jun. 2023.
 6. SOUZA, D., BATISTA, J., DOS REIS, L., & BARBOSA, A. Model approach of a robotic manipulator 3DOF using Machine Learning. *Journal on Advances in Theoretical and Applied Informatics* 5.1, 2019. Disponível em: <<https://doi.org/10.26729/jadi.v5i1.3039>>. Acesso em: 07 jun. 2023.
 7. SOUZA, D. A., DOS REIS, L. L., BATISTA, J. G., COSTA, J. R., BARBOSA, A. B., ARAÚJO, J. P., & BRAGA, A. P. Nonlinear Identification of a Robotic Arm Using Machine Learning Techniques. *Advances in Intelligent Systems and Computing*. Springer, v. 2, p. 492-501, 2019. Disponível em: <https://doi.org/10.1007/978-3-030-16184-2_47>. Acesso em: 07 jun. 2023.

REFERÊNCIAS

- AGUIRRE, L. Introdução à identificação de sistemas. editora ufmg, 3a edição. Belo Horizonte, 2007.
- ALBERTO, P.; SALA, P.; SALA, A. **Multi Variable Control Systems**. London: Springer, 2004.
- ANTONIO, B.; FELIPE, P. L.; CARLOS, B.; RAFAEL, A. Fundamentos de robótica. **España: Editorial McGraw-Hill**, 1997.
- ARNOT, A. Estatística fácil. **Saraiva**, São Paulo, 2002.
- ASADI, H.; MOHAMED, S.; LIM, C. P.; NAHAVANDI, S. Robust optimal motion cueing algorithm based on the linear quadratic regulator method and a genetic algorithm. **IEEE Transactions on Systems, Man, and Cybernetics: Systems**, IEEE, v. 47, n. 2, p. 238–254, 2016. Disponível em: <<https://doi.org/10.1109/TSMC.2016.2523906>>. Acesso em: 07 jun. 2023.
- BATISTA, J.; SOUZA, D.; REIS, L. dos; BARBOSA, A.; ARAÚJO, R. Dynamic model and inverse kinematic identification of a 3-dof manipulator using rlspso. **Sensors**, Multidisciplinary Digital Publishing Institute, v. 20, n. 2, p. 416, 2020. Disponível em: <<https://doi.org/10.3390/s20020416>>. Acesso em: 07 jun. 2023.
- BATISTA, J.; SOUZA, D.; SILVA, J.; RAMOS, K.; COSTA, J.; REIS, L. dos; BRAGA, A. Trajectory planning using artificial potential fields with metaheuristics. **IEEE Latin America Transactions**, IEEE, v. 18, n. 05, p. 914–922, 2020. Disponível em: <<https://doi.org/10.1109/TLA.2020.9082920>>. Acesso em: 07 jun. 2023.
- BENKHOUD, K.; BOUALLÈGUE, S. Dynamics modeling and advanced metaheuristics based lqg controller design for a quad tilt wing uav. **International Journal of Dynamics and Control**, Springer, v. 6, n. 2, p. 630–651, 2018. Disponível em: <<https://doi.org/10.1007/s40435-017-0325-7>>. Acesso em: 06 jun. 2023.
- BINGÜL, Z.; KARAHAN, O. Dynamic identification of Staubli RX-60 robot using pso and ls methods. **Expert Systems with Applications: An International Journal**, Pergamon Press, Inc. Elmsford, NY, USA, v. 38, n. 4, p. 4136–4149, 2011. Disponível em: <<https://doi.org/10.1016/j.eswa.2010.09.076>>. Acesso em: 07 jun. 2023.
- BOSE, B. K. Power electronics and motor drives: advances and trends. Academic press, Knoxville, 2020.
- BOUSSAÏD, I.; LEPAGNOT, J.; SIARRY, P. A survey on optimization metaheuristics. **Information sciences**, Elsevier, v. 237, p. 82–117, 2013. Disponível em: <<https://doi.org/10.1016/j.ins.2013.02.041>>. Acesso em: 07 jun. 2023.
- BUENO, J. N. A.; SERRANTOLA, W. G.; BEZERRA, R. A.; GRASSI, V. Lqr and h-infinity controls of a free-floating space manipulator with two arms. In: **IEEE. 2016 XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR)**. 2016. p. 85–90. Disponível em: <<https://doi.org/10.1109/LARS-SBR.2016.21>>. Acesso em: 07 jun. 2023.

CARVALHO, L.; NETO, J. R.; REZENDE, J. C.; COSTA, M. V.; FORTES, E. V.; MACEDO, L. H. Linear quadratic regulator design via metaheuristics applied to the damping of low-frequency oscillations in power systems. **ISA transactions**, Elsevier, 2022. Disponível em: <<https://doi.org/10.1016/j.isatra.2022.08.024>>. Acesso em: 06 jun. 2023.

CHAPMAN, S. J. **Fundamentos de máquinas elétricas**. Porto Alegre: AMGH editora, 2013.

CHEN, B. M.; LIN, Z.; SHAMASH, Y. **Linear systems theory: a structural decomposition approach**. Berlin: Springer Science & Business Media, 2004.

CHENG, C.-A.; HUANG, H.-P.; HSU, H.-K.; LAI, W.-Z.; CHENG, C.-C. Learning the inverse dynamics of robotic manipulators in structured reproducing kernel hilbert space. **IEEE transactions on cybernetics**, IEEE, v. 46, n. 7, p. 1691–1703, 2015. Disponível em: <<https://doi.org/10.1109/TCYB.2015.2454334>>. Acesso em: 07 jun. 2023.

COELHO, A. A. R.; COELHO, L. dos S. **Identificação de sistemas dinâmicos lineares**. Florianópolis: [s.n.], 2004.

COELHO, A. A. R.; COELHO, L. dos S. **Identificação de sistemas dinâmicos lineares**. Florianópolis: [s.n.], 2016. v. 2.

CUI, Z.; GAO, X. **Theory and applications of swarm intelligence**. Taiyuan: Springer, 2012. 205–206 p. Disponível em: <<https://doi.org/10.1007/s00521-011-0523-8>>. Acesso em: 07 jun. 2023.

DAS, A.; SUBBARAO, K.; LEWIS, F. Dynamic inversion with zero-dynamics stabilisation for quadrotor control. **IET control theory & applications**, IET, v. 3, n. 3, p. 303–314, 2009. Disponível em: <<https://doi.org/10.1049/iet-cta:20080002>>. Acesso em: 07 jun. 2023.

DINIZ, E.; HONÓRIO, A.; ALMEIDA, D.; BARRETO, O. Simplified approach for modelling and control a 3-dof rrr type robotic manipulator using squirrel-cage induction motors. In: IEEE. **2010 9th IEEE/IAS International Conference on Industry Applications-INDUSCON 2010**. 2010. p. 1–7. Disponível em: <<https://doi.org/10.1109/INDUSCON.2010.5740067>>. Acesso em: 07 jun. 2023.

DONG, Y.; TANG, J.; XU, B.; WANG, D. An application of swarm optimization to nonlinear programming. **Computers & Mathematics with Applications**, Elsevier, v. 49, n. 11-12, p. 1655–1668, 2005. Disponível em: <<https://doi.org/10.1016/j.camwa.2005.02.006>>. Acesso em: 07 jun. 2023.

DWIVEDI, Y.; TAYAL, V. K. Dynamic stability improvement of alkali fuel cell integrated system using pso optimized pid control design. In: IEEE. **2017 Recent Developments in Control, Automation & Power Engineering (RDCAPE)**. 2017. p. 499–504. Disponível em: <<https://doi.org/10.1109/RDCAPE.2017.8358322>>. Acesso em: 07 jun. 2023.

EBERHART, R.; KENNEDY, J. A new optimizer using particle swarm theory. In: IEEE. **MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science**. 1995. p. 39–43. Disponível em: <<https://doi.org/10.1109/MHS.1995.494215>>. Acesso em: 07 jun. 2023.

ENGELBRECHT, A. P. **Computational intelligence: an introduction**. West Sussex: John Wiley & Sons, 2007.

FRIEDLAND, B. **Control system design: an introduction to state-space methods**. Ney York: Courier Corporation, 2012.

GARETH, J.; DANIELA, W.; TREVOR, H.; ROBERT, T. **An introduction to statistical learning: with applications in R**. Berlin: Springer, 2013.

GLOVER, F. W.; KOCHENBERGER, G. A. **Handbook of metaheuristics**. New York: Springer Science & Business Media, 2006. v. 57.

GREWAL, K. S.; DIXON, R.; PEARSON, J. Lqg controller design applied to a pneumatic stewart-gough platform. **International Journal of Automation and Computing**, Springer, v. 9, n. 1, p. 45–53, 2012. Disponível em: <<https://doi.org/10.1007/s11633-012-0615-7>>. Acesso em: 06 jun. 2023.

GUECHI, E.-H.; BOUZOUALEGH, S.; ZENNIR, Y.; BLAŽIČ, S. Mpc control and lq optimal control of a two-link robot arm: A comparative study. **Machines**, MDPI, v. 6, n. 3, p. 37, 2018. Disponível em: <<https://doi.org/10.3390/machines6030037>>. Acesso em: 06 jun. 2023.

HABIB, M. K. **Handbook of Research on Advancements in Robotics and Mechatronics**. Cairo: IGI Global, 2014.

HAFEZI, Z.; AREFI, M. M. Recursive generalized extended least squares and rml algorithms for identification of bilinear systems with arma noise. **ISA transactions**, Elsevier, v. 88, p. 50–61, 2019. Disponível em: <<https://doi.org/10.1016/j.isatra.2018.12.015>>. Acesso em: 07 jun. 2023.

HARTENBERG, R.; DANAVIT, J. **Kinematic synthesis of linkages**. New York: McGraw-Hill, 1964.

HASSANI, K.; LEE, W.-S. Optimal tuning of linear quadratic regulators using quantum particle swarm optimization. Ottawa, 2014. Disponível em: <https://avestia.com/CDSR2014_Proceedings/papers/59.pdf>. Acesso em: 07 jun. 2023.

IFR, I. F. of R. **Foreword World Robotics Industrial Robots**. Frankfurt: IFR, 2021.

JING, X.; GAO, H.; CHEN, Z.; WANG, Y. A recursive dynamic modeling and control for dual-arm manipulator with elastic joints. **IEEE Access**, IEEE, v. 8, p. 155093–155102, 2020. Disponível em: <<https://doi.org/10.1109/ACCESS.2020.3018470>>. Acesso em: 07 jun. 2023.

KALMAN, R. Discussion:“on the existence of optimal controls”(markus, l., and lee, eb, 1962, asme j. basic eng., 84, pp. 13–20). 1962. Disponível em: <<https://doi.org/10.1115/1.3657258>>. Acesso em: 07 jun. 2023.

KAZARLIS, S. A.; BAKIRTZIS, A.; PETRIDIS, V. A genetic algorithm solution to the unit commitment problem. **IEEE transactions on power systems**, IEEE, v. 11, n. 1, p. 83–92, 1996. Disponível em: <<https://doi.org/10.1109/59.485989>>. Acesso em: 07 jun. 2023.

KELLY, R.; DAVILA, V. S.; PEREZ, J. A. L. **Control of robot manipulators in joint space**. London: Springer Science & Business Media, 2006.

KINGSLEY, C. J.; UMANS, S. D.; FITZGERALD, A. E. **Máquinas Elétricas com Introdução à Eletrônica de Potência**. Porto Alegre: Bookman, 2006.

KJAER, A.; HEATH, W.; WELLSTEAD, P. Identification of cross-directional behaviour in web production: techniques and experience. **Control Engineering Practice**, Elsevier, v. 3, n. 1, p. 21–29, 1995. Disponível em: <[https://doi.org/10.1016/0967-0661\(94\)00060-T](https://doi.org/10.1016/0967-0661(94)00060-T)>. Acesso em: 07 jun. 2023.

KOPPULA, H. S.; SAXENA, A. Anticipating human activities using object affordances for reactive robotic response. **IEEE transactions on pattern analysis and machine intelligence**, IEEE, v. 38, n. 1, p. 14–29, 2015. Disponível em: <<https://doi.org/10.1109/TPAMI.2015.2430335>>. Acesso em: 07 jun. 2023.

KORKMAZ, O.; IDER, S. K. Hybrid force and motion control of flexible joint parallel manipulators using inverse dynamics approach. **Advanced Robotics**, Taylor & Francis, v. 28, n. 18, p. 1221–1230, 2014. Disponível em: <<https://doi.org/10.1080/01691864.2014.920719>>. Acesso em: 07 jun. 2023.

KOŚCIELSKI, A.; PACHOLSKI, L. Complexity of makanin's algorithm. **Journal of the ACM (JACM)**, ACM New York, NY, USA, v. 43, n. 4, p. 670–684, 1996. Disponível em: <<https://doi.org/10.1145/234533.234543>>. Acesso em: 06 jun. 2023.

KOSOW, I. L. **Máquinas elétricas e transformadores**. São Paulo: Globo, 1993.

KRIM, S.; GDAIM, S.; MIMOUNI, M. F. Robust direct torque control with super-twisting sliding mode control for an induction motor drive. **Complexity**, Hindawi, v. 2019, 2019. Disponível em: <<https://doi.org/10.1155/2019/7274353>>. Acesso em: 07 jun. 2023.

LEE, K.; JEON, S.; KIM, H.; KUM, D. Optimal path tracking control of autonomous vehicle: Adaptive full-state linear quadratic gaussian (lqg) control. **IEEE Access**, IEEE, v. 7, p. 109120–109133, 2019. Disponível em: <<https://doi.org/10.1109/ACCESS.2019.2933895>>. Acesso em: 06 jun. 2023.

LEWIS, F. L.; DAWSON, D. M.; ABDALLAH, C. T. **Robot manipulator control: theory and practice**. New York: CRC Press, 2003.

LIN, X.; LIN, W. Whale optimization algorithm-based lqg-adaptive neuro-fuzzy control for seismic vibration mitigation with mr dampers. **Shock and Vibration**, Hindawi, v. 2022, 2022. Disponível em: <<https://doi.org/10.1155/2022/4060660>>. Acesso em: 06 jun. 2023.

LINDEN, R. **Algoritmos genéticos (2a edição)**. Rio de Janeiro: Brasport, 2008.

LIU, L.; LIU, W.; CARTES, D. A. Particle swarm optimization-based parameter identification applied to permanent magnet synchronous motors. **Engineering Applications of Artificial Intelligence**, Elsevier, v. 21, n. 7, p. 1092–1100, 2008. Disponível em: <<https://doi.org/10.1016/j.engappai.2007.10.002>>. Acesso em: 07 jun. 2023.

LJUNG, L.; SÖDERSTRÖM, T. **Theory and Practice of Recursive**. Cambridge: MIT Press, 1983.

MA, J.; ZHAO, L.; HAN, Z.; TANG, Y. Identification of wiener model using least squares support vector machine optimized by adaptive particle swarm optimization. **Journal of Control, Automation and Electrical Systems**, Springer, v. 26, n. 6, p. 609–615, 2015. Disponível em: <<https://doi.org/10.1007/s40313-015-0207-1>>. Acesso em: 07 jun. 2023.

MAGHFIROH, H.; NIZAM, M.; ANWAR, M.; MA'ARIF, A. Improved lqr control using pso optimization and kalman filter estimator. **IEEE Access**, IEEE, v. 10, p. 18330–18337, 2022. Disponível em: <<https://doi.org/10.1109/ACCESS.2022.3149951>>. Acesso em: 07 jun. 2023.

MITTAL, R.; NAGRATH, I. **Robotics and control**. New Delhi: Tata McGraw-Hill, 2003.

MOON, S.; VIRK, G. S. Survey on iso standards for industrial and service robots. In: IEEE. **2009 ICCAS-SICE**. 2009. p. 1878–1881. Disponível em: <<https://ieeexplore.ieee.org/abstract/document/5334289>>. Acesso em: 07 jun. 2023.

MUSTAFA, A. M.; AL-SAIF, A. Modeling, simulation and control of 2-r robot. **Global Journal of Research In Engineering**, King Fahd, 2014. Disponível em: <https://globaljournals.org/GJRE_Volume14/5-Modeling-Simulation-and-Control.pdf>. Acesso em: 07 jun. 2023.

OGATA, K. **Modern control engineering**. Upper Saddle River: Prentice hall, 2010.

PAIVA, F. A. P.; COSTA, J. A. F.; SILVA, C. R. M. A serendipity-based approach to enhance particle swarm optimization using scout particles. **IEEE Latin America Transactions**, IEEE, v. 15, n. 6, p. 1101–1112, 2017. Disponível em: <<https://doi.org/10.1109/TLA.2017.7932698>>. Acesso em: 07 jun. 2023.

PERES, W.; CHAVES, S.; PASSOS, F.; ALBERTO, J. Gradient based hybrid metaheuristics for robust tuning of power system stabilizers. **International Journal of Electrical Power & Energy Systems**, Elsevier, v. 95, p. 47–72, 2018. Disponível em: <<https://doi.org/10.1016/j.ijepes.2017.08.014>>. Acesso em: 07 jun. 2023.

PINTO, V.; CAMPOS, J.; REIS, L. D.; JACOBINA, C.; ROCHA, N. Robustness and performance analysis for the linear quadratic gaussian/loop transfer recovery with integral action controller applied to doubly fed induction generators in wind energy conversion systems. **Electric Power Components and Systems**, Taylor & Francis, v. 40, n. 2, p. 131–146, 2011. Disponível em: <<https://doi.org/10.1080/15325008.2011.629331>>. Acesso em: 07 jun. 2023.

PREMKUMAR, K.; MANIKANDAN, B. Speed control of brushless dc motor using bat algorithm optimized adaptive neuro-fuzzy inference system. **Applied Soft Computing**, Elsevier, v. 32, p. 403–419, 2015. Disponível em: <<https://doi.org/10.1016/j.asoc.2015.04.014>>. Acesso em: 07 jun. 2023.

RABELO L, R. **Controle de um motor de indução trifásico como servoposicionador aplicado a um manipulador robótico utilizando controle de campo orientado** 2017. 107 f. Dissertação (Mestrado em Engenharia Elétrica) — Centro de Tecnologia, Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal do Ceará, Fortaleza, 2017. Disponível em: <<https://repositorio.ufc.br/handle/riufc/29307>>. Acesso em: 07 jun. 2023.

REBOUÇAS, L. R.; BARBOSA, A.; REIS, L.; NOGUEIRA, F. *et al.* Predictive control applied in 3-phase squirrel cage induction motor for zero speed. In: **Proc. Brazilian Symp. Intell. Automat.(SBAI)**. Porto Alegre: SBAI, 2017. Disponível em: <<https://repositorio.ufc.br/handle/riufc/68343>>. Acesso em: 07 jun. 2023.

REMES, C. L.; ROSA, M. B.; HEERDT, J. A.; OLIVEIRA, S. V. Lqg controller in cascade loop tuned by pso applied to a dc–dc converter. **Asian Journal of Control**, Wiley Online Library, v. 23, n. 5, p. 2370–2380, 2021. Disponível em: <<https://doi.org/10.1002/asjc.2373>>. Acesso em: 07 jun. 2023.

RIOS, C. Controle robusto aplicado à conversores dc-dc multiportas isolados para sistemas de suprimento de energia tese (doutorado em engenharia elétrica) – universidade federal do ceará, centro de tecnologia, programa de pós-graduação em engenharia elétrica, fortaleza. 2021. Disponível em: <<https://repositorio.ufc.br/handle/riufc/60073>>. Acesso em: 07 jun. 2023.

ROMANO, V. F. **Robótica industrial: aplicação na indústria de manufaturatura e de processos**. São Paulo: Edgard Blucher, 2002.

SABIR, M. M.; KHAN, J. A. Optimal design of pid controller for the speed control of dc motor by using metaheuristic techniques. **Advances in artificial neural systems**, Hindawi, v. 2014, 2014. Disponível em: <<https://doi.org/10.1155/2014/126317>>. Acesso em: 07 jun. 2023.

SANZ, P. Robotics: Modeling, planning, and control (siciliano, b. et al; 2009)[on the shelf]. **IEEE Robotics & Automation Magazine**, IEEE, v. 16, n. 4, p. 101–101, 2009. Disponível em: <<https://doi.org/10.1109/MRA.2009.934833>>. Acesso em: 07 jun. 2023.

SCIAVICCO, L.; SICILIANO, B. **Modeling and control of robot manipulators**. New York: McGraw-Hill, 1996. v. 8.

SELAMAT, N.; DAUD, F.; JAAFAR, H.; SHAMSUDIN, N. Comparison of lqr and pid controller tuning using pso for coupled tank system. In: IEEE. **2015 IEEE 11th International Colloquium on Signal Processing & Its Applications (CSPA)**. 2015. p. 46–51. Disponível em: <<https://doi.org/10.1109/CSPA.2015.7225616>>. Acesso em: 07 jun. 2023.

SGRÒ, D.; CORREIA, W. B.; LEÃO, R. P. S.; TOFOLI, F. L.; TIBÚRCIO, S. A. S. Nonlinear current control strategy for grid-connected voltage source converters. **International Journal of Electrical Power & Energy Systems**, Elsevier, v. 142, p. 108349, 2022. Disponível em: <<https://doi.org/10.1016/j.isatra.2015.10.018>>. Acesso em: 06 jun. 2023.

SICILIANO, B.; SCIAVICCO, L.; VILLANI, L.; ORIOLO, G. **Robotics: modelling, planning and control**. London: Springer Science & Business Media, 2010.

SINGH, M.; PATEL, R. N.; NEEMA, D. Robust tuning of excitation controller for stability enhancement using multi-objective metaheuristic firefly algorithm. **Swarm and evolutionary computation**, Elsevier, v. 44, p. 136–147, 2019. Disponível em: <<https://doi.org/10.1016/j.swevo.2018.01.010>>. Acesso em: 07 jun. 2023.

SKOGESTAD, S.; POSTLETHWAITE, I. **Multivariable feedback control: analysis and design**. West Sussex: Citeseer, 2007. v. 2.

SOUZA., A. B. d.; DINIZ, E. d. C.; HONORIO, D. d. A.; BARRETO, L. H. S. C.; REIS, L. L. N. dos. Hybrid control robust using logic fuzzy applied to the position loop for vector control to induction motors. **Electric Power Components and Systems**, Taylor & Francis, v. 42, n. 6, p. 533–543, 2014. Disponível em: <<https://doi.org/10.1080/15325008.2014.880967>>. Acesso em: 07 jun. 2023.

SOUZA, A. B. J. **Estratégia híbrida aplicada no controle de posição de um sistema de acionamento com motor de indução usando controle de campo orientado**. Tese (Doutorado em Engenharia Elétrica) — Centro de Tecnologia, Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal do Ceará, Fortaleza, 2014. Disponível em: <<https://repositorio.ufc.br/handle/riufc/10671>>. Acesso em: 07 jun. 2023.

SOUZA, D. A.; BATISTA, J. G.; REIS, L. L. dos; BARBOSA, A. Pid controller with novel pso applied to a joint of a robotic manipulator. **Journal of the Brazilian Society of Mechanical Sciences and Engineering**, Springer, v. 43, n. 8, p. 1–14, 2021. Disponível em: <<https://doi.org/10.1007/s40430-021-03092-4>>. Acesso em: 07 jun. 2023.

SOUZA, D. A.; MESQUITA, V. A. de; REIS, L. L.; SILVA, W. A.; BATISTA, J. G. Optimal lqi and pid synthesis for speed control of switched reluctance motor using metaheuristic techniques. **International Journal of Control, Automation and Systems**, Springer, p. 1–9, 2020. Disponível em: <<https://doi.org/10.1007/s12555-019-0911-x>>. Acesso em: 07 jun. 2023.

SOUZA, D. A.; MESQUITA, V. A. de; REIS, L. L.; SILVA, W. A.; BATISTA, J. G. Optimal lqi and pid synthesis for speed control of switched reluctance motor using metaheuristic techniques. **International Journal of Control, Automation and Systems**, Springer, v. 19, n. 1, p. 221–229, 2021. Disponível em: <<https://doi.org/10.1007/s12555-019-0911-x>>. Acesso em: 07 jun. 2023.

SOUZA, D. A. D.; BATISTA, J. G.; VASCONCELOS, F. J.; REIS, L. L. D.; MACHADO, G. F.; COSTA, J. R.; NOGUEIRA, J. N.; SILVA, J. L.; RIOS, C. S.; BARBOSA, A. S. Identification by recursive least squares with kalman filter (rls-kf) applied to a robotic manipulator. **IEEE Access**, IEEE, v. 9, p. 63779–63789, 2021. Disponível em: <<https://doi.org/10.1109/ACCESS.2021.3074419>>. Acesso em: 07 jun. 2023.

SPONG, M. W.; VIDYASAGAR, M. **Robot dynamics and control**. New York: John Wiley & Sons, 2008.

STOJANOVIC, V.; NEDIC, N. Identification of time-varying oe models in presence of non-gaussian noise: Application to pneumatic servo drives. **International Journal of Robust and Nonlinear Control**, Wiley Online Library, v. 26, n. 18, p. 3974–3995, 2016. Disponível em: <<https://doi.org/10.1002/rnc.3544>>. Acesso em: 07 jun. 2023.

STOJANOVIC, V.; NEDIC, N. Joint state and parameter robust estimation of stochastic nonlinear systems. **International Journal of Robust and Nonlinear Control**, Wiley Online Library, v. 26, n. 14, p. 3058–3074, 2016. Disponível em: <<https://doi.org/10.1002/rnc.3490>>. Acesso em: 07 jun. 2023.

SUSHNIGDHA, G.; JOSHI, A. Evolutionary method based integrated guidance strategy for reentry vehicles. **Engineering Applications of Artificial Intelligence**, Elsevier, v. 69, p. 168–177, 2018. Disponível em: <<https://doi.org/10.1016/j.engappai.2017.11.010>>. Acesso em: 07 jun. 2023.

SWEVERS, J.; VERDONCK, W.; SCHUTTER, J. D. Dynamic model identification for industrial robots. **IEEE control systems magazine**, IEEE, v. 27, n. 5, p. 58–71, 2007. Disponível em: <<https://doi.org/10.1109/MCS.2007.904659>>. Acesso em: 06 jun. 2023.

TANG, L.; GOUTTEFARDE, M.; SUN, H.; YIN, L.; ZHOU, C. Dynamic modelling and vibration suppression of a single-link flexible manipulator with two cables. **Mechanism and Machine Theory**, Elsevier, v. 162, p. 104347, 2021. Disponível em: <<https://doi.org/10.1016/j.mechmachtheory.2021.104347>>. Acesso em: 06 jun. 2023.

TORRICO, B. C.; CORREIA, W. B.; NOGUEIRA, F. G. Simplified dead-time compensator for multiple delay siso systems. **ISA transactions**, Elsevier, v. 60, p. 254–261, 2016. Disponível em: <<https://doi.org/10.1016/j.isatra.2015.10.018>>. Acesso em: 06 jun. 2023.

URREA, C.; PASCAL, J. Parameter identification methods for real redundant manipulators. **Journal of applied research and technology**, UNAM, Centro de Ciencias Aplicadas y Desarrollo Tecnológico, v. 15, n. 4, p. 320–331, 2017. Disponível em: <<https://doi.org/10.1016/j.jart.2017.02.004>>. Acesso em: 07 jun. 2023.

VIJYASAI, P.; GUDI, R.; LAKSHMINARAYANAN, S. Identification on demand using a blockwise recursive partial least-squares technique. **Industrial & engineering chemistry research**, ACS Publications, v. 42, n. 3, p. 540–554, 2003. Disponível em: <<https://doi.org/10.1021/ie020042r>>. Acesso em: 07 jun. 2023.

WELLSTEAD, P. Self-tuning systems. **Control and signal processing**, John Wiley & Sons Ltd, West Sussex, 1991.

WILF, H. S. **Algorithms and complexity**. New York: AK Peters/CRC Press, 2002.

YAN, D.; LU, Y.; LEVY, D. Parameter identification of robot manipulators: A heuristic particle swarm search approach. **PLoS one**, Public Library of Science San Francisco, CA USA, v. 10, n. 6, p. e0129157, 2015. Disponível em: <<https://doi.org/10.1371/journal.pone.0129157>>. Acesso em: 07 jun. 2023.

YU, Z. Space vector pwm with tms320c24x using hardware and software determined switching patterns. **Application Report SPRA524, Texas Instruments**, 2001. Disponível em: <https://www.ti.com/lit/an/spra524/spra524.pdf?ts=1686147323515&ref_url=https%253A%252F%252Fwww.google.com%252F>. Acesso em: 07 jun. 2023.

YUAN, J.-j.; WAN, W.; FU, X.; WANG, S.; WANG, N. A novel llsdpso method for nonlinear dynamic parameter identification. **Assembly Automation**, Emerald Publishing Limited, 2017. Disponível em: <<https://doi.org/10.1108/AA-08-2016-106>>. Acesso em: 07 jun. 2023.

ZADEH, L.; DESOER, C. **Linear system theory: the state space approach**. New York: Courier Dover Publications, 2008.

ZHA, F.; SHENG, W.; GUO, W.; QIU, S.; DENG, J.; WANG, X. Dynamic parameter identification of a lower extremity exoskeleton using rls-pso. **Applied Sciences**, Multidisciplinary Digital Publishing Institute, v. 9, n. 2, p. 324, 2019. Disponível em: <<https://doi.org/10.3390/app9020324>>. Acesso em: 07 jun. 2023.

ZHANG, G.; ZHANG, X.; PANG, H. Multi-innovation auto-constructed least squares identification for 4 dof ship manoeuvring modelling with full-scale trial data. **ISA transactions**, Elsevier, v. 58, p. 186–195, 2015. Disponível em: <<https://doi.org/10.1016/j.isatra.2015.04.004>>. Acesso em: 07 jun. 2023.

ZHOU, K.; DOYLE, J. C. **Essentials of robust control**. Saddle River: Prentice hall Upper, 1998. v. 104.

APÊNDICE A – MODELAGEM CINEMÁTICA E DINÂMICA DE MANIPULADORES

A.1 Introdução

O modelo cinemático tem como objetivo fundamental estabelecer as estratégias de controle adequadas que serão aplicadas ao manipulador robótico que resultem em uma melhor qualidade de seus movimentos. Levando-se em conta a variedade e complexidade das estruturas que compõem os robôs manipuladores, a cinemática descreve o movimento relativo dos vários sistemas de referência à medida que a estrutura se movimenta associando sistemas de referência às várias partes da estrutura (ANTONIO *et al.*, 1997), (SPONG; VIDYASAGAR, 2008).

O movimento relativo a uma tarefa a ser executado pelo robô deve ser representado como um sinal de referência, o qual é enviado ao sistema de controle do manipulador. Este atua no sentido de fazer com que o robô acompanhe o sinal de referência ativando o atuador adequado. Se o sinal de referência muda abruptamente, o sistema pode não acompanhar o sinal visto que o projeto do sistema de controle tem limitações em relação às dinâmicas do sistema. Algoritmos computacionais são projetados para calcular o sinal de referência apropriado baseado na trajetória da tarefa a ser executada e pode ser definido como um conjunto de pontos no espaço operacional ou no espaço das articulações no qual o tempo de aplicação é especificado (SCIAVICCO; SICILIANO, 1996).

Geralmente, o controle cinemático deverá realizar as seguintes funções (ANTONIO *et al.*, 1997):

1. Converter as características do movimento de um programa em uma trajetória analítica no espaço cartesiano.
2. Mostrar a trajetória cartesiana para obter um número finito de pontos dessa trajetória, em que cada um desses pontos serão dados por uma 6-upla $(x, y, z, \alpha, \beta, \gamma)$.
3. Com o uso da cinemática inversa pode-se converter cada um dos pontos no espaço cartesiano em suas correspondentes coordenadas articulares $(q_1, q_2, q_3, q_4, q_5, q_6)$. Necessita-se saber a quantidade de soluções da cinemática inversa, assim como a possibilidade de ausência de soluções e pontos singulares, de modo que a continuidade da trajetória seja garantida.
4. Interpolação dos pontos articulares obtidos gerando para cada variável articular uma expressão $q_i(t)$ que passe ou se aproxime das juntas de modo que uma trajetória realizada pelas juntas se transforme em uma trajetória cartesiana que se aproxime da trajetória

especificada pelo usuário (quanto a precisão, velocidade, etc.).

5. Apresentar a trajetória articular para gerar as referências do controle dinâmico.

A dinâmica se ocupa da relação entre as forças que atuam sobre um corpo e o movimento que se origina. Portanto, o modelo dinâmico de um robô industrial tem por objetivo conhecer a relação entre o movimento do robô e as forças aplicadas ao mesmo. Esta relação se obtém a partir do modelo dinâmico que relaciona matematicamente (ANTONIO *et al.*, 1997), (SCIAVICCO; SICILIANO, 1996):

1. A localização do robô definida suas variáveis articulares ou pelas coordenadas de localização da sua ferramenta e suas derivadas como velocidade e aceleração.
2. As forças e torques aplicados nas articulações (ou na ferramenta do robô).
3. Os parâmetros dimensionais do robô, como comprimento, massas e inércias de seus elementos.
4. Efeitos das forças centrífugas e de Coriolis.

A.2 Cinemática Direta de Manipuladores Robóticos

A cinemática de um robô estuda o movimento do mesmo em relação a um sistema de referência. Portanto, a cinemática trata da descrição analítica do movimento espacial do robô como uma função do tempo, e em particular das relações entre a posição e a orientação da ferramenta do robô com os valores que tornam suas coordenadas articulares. O problema da cinemática direta consiste em determinar qual a posição e a orientação do atuador do manipulador em relação a um sistema de coordenadas de referência fixo conhecido os valores das articulações. Em relação a cinemática inversa resolve a configuração que deve adotar o robô para uma posição e orientação do extremo conhecido (ROMANO, 2002).

O problema da cinemática direta está relacionado com a relação entre as articulações individuais do robô manipulador e a posição e orientação da ferramenta ou *end-effector*. Mais formalmente, o problema da cinemática direta é determinar a posição e orientação do *end-effector* dados os valores das variáveis das juntas do robô. As variáveis das juntas são os ângulos entre os elos no caso de juntas de revolução ou juntas rotacionais e a extensão da junta no caso destas serem prismáticas ou deslizantes. O problema da cinemática direta deve ser contrastado com o problema da cinemática inversa que se preocupa com a determinação dos valores para as variáveis das juntas que atingem uma posição e orientação desejadas para o *end-effector* do robô (SPONG; VIDYASAGAR, 2008).

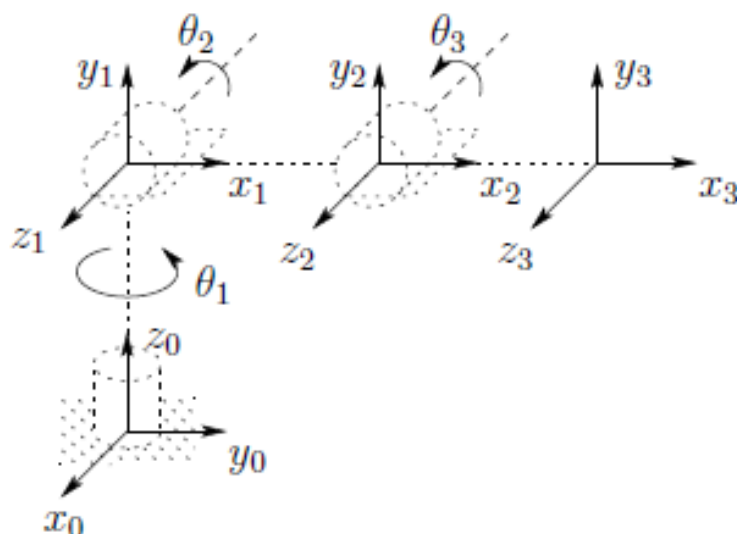
Um robô manipulador com n articulações terá $n + 1$ links já que cada junta conecta dois links. Enumerou-se as juntas de 1 a n , e numeramos os links de 0 a n , começando da base. Por essa convenção, a junção i conecta o link $i - 1$ ao link i . Considerando-se a localização da junta i a ser fixada em relação ao link $i - 1$. Quando a junta i é acionada, move o link i . Portanto, o link 0 (o primeiro link) é fixo e não se move quando as juntas são atuadas. O robô manipulador pode ser móvel (por exemplo, montado numa plataforma móvel ou num veículo autônomo), mas não será considerado este caso (SPONG; VIDYASAGAR, 2008).

Com a junta $i - \text{ésima}$ associa-se uma variável da junta por q_i . No caso de uma junta de revolução a articulação q_i é o ângulo de rotação e no caso de uma junta prismática, q_i é a articulação prismática (deslocamento):

$$q_i = \begin{cases} \theta_i : \text{junta } i \text{ de revol.}, \\ d_i : \text{junta } i \text{ prism.} \end{cases} \quad (\text{A.1})$$

Para realizar a análise cinemática anexa-se rigidamente um quadro de coordenadas a cada link. Em particular, atribuímos $o_i x_i y_i z_i$ para um link i , ou seja, qualquer que seja o movimento que o robô executa, as coordenadas de cada ponto no link i são constantes quando expressas no i -ésimo quadro de coordenadas. Além disso, quando a junta i é acionada o link i e sua estrutura anexada, $o_i x_i y_i z_i$, executa um movimento resultante. O quadro $o_0 x_0 y_0 z_0$, que está fixo na base do robô é referido como o referencial inercial. A Figura 52 ilustra a ideia de anexar quadros rigidamente aos links no caso de um manipulador de juntas rotacionais (SPONG; VIDYASAGAR, 2008).

Figura 52 – Estruturas de coordenadas anexadas a um manipulador com juntas rotacionais.



Fonte: Spong e Vidyasagar (2008).

Suponha que A_i é a matriz de transformação homogênea que expressa a posição e orientação do $o_i x_i y_i z_i$ em relação ao $o_{i-1} x_{i-1} y_{i-1} z_{i-1}$. A matriz A_i não é constante mas varia conforme a configuração do robô é alterada. No entanto, a suposição de que todas as articulações são de revolução ou prismáticas significa que A_i é uma função somente de única articulação variável, ou seja, q_i , isto é,

$$A_i = A_i(q_i) \quad (\text{A.2})$$

A matriz de transformação homogênea que expressa a posição e orientação de $o_j x_j y_j z_j$ em relação ao $o_i x_i y_i z_i$ é chamada por convenção, uma matriz de transformação, T_j^i .

$$\begin{aligned} T_j^i &= A_{i+1} A_{i+2} \dots A_{j-1} A_j \text{ se } i < j \\ T_j^i &= I \text{ se } i = j \\ T_j^i &= (T_j^i)^{-1} \text{ se } j > i \end{aligned} \quad (\text{A.3})$$

A posição e orientação do *end-effector* em relação ao referencial inercial ou de base é definido por um vetor tridimensional O_n^0 (que fornece as coordenadas da origem do referencial do *end-effector* no que diz respeito ao quadro base) e uma matriz de rotação 3×3 , R_n^0 , que define a matriz de transformação homogênea

$$H = \begin{bmatrix} R_n^0 & O_n^0 \\ 0 & 1 \end{bmatrix} \quad (\text{A.4})$$

A posição e orientação do *end-effector* no quadro inercial são dadas por

$$H = T_n^0 = A_i(q_i) \dots A_n(q_n) \quad (\text{A.5})$$

A.2.1 Representação Denavit-Hartenberg (DH)

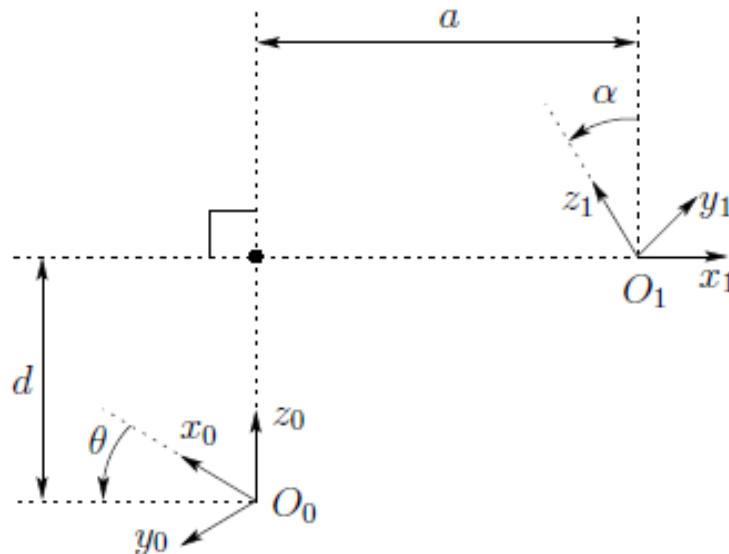
A análise pode ser feita usando um quadro arbitrário anexado a cada link (ver Figura 52). Entretanto, é conveniente ser sistemático na escolha desses quadros. Desta forma, uma convenção utilizada para selecionar quadros de referência em aplicações robóticas é a convenção Denavit-Hartenberg D-H (HARTENBERG; DANAVIT, 1964). Nesta, cada transformação homogênea A_i é representada como um produto de quatro transformações:

$$A_i = Rot_{z, \theta_i} Trans_{z, d_i} Trans_{x, a_i} Rot_{x, \alpha_i} \quad (\text{A.6})$$

$$\begin{aligned}
 A_i &= \begin{bmatrix} C\theta_i & -S\theta_i & 0 & 0 \\ S\theta_i & C\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha_i & -S\alpha_i & 0 \\ 0 & S\alpha_i & C\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} C\theta_i & -S\theta_i C\alpha_i & S\theta_i S\alpha_i & a_i C\theta_i \\ S\theta_i & C\theta_i C\alpha_i & -C\theta_i S\alpha_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{A.7}
 \end{aligned}$$

em que as quatro grandezas θ_i , a_i , d_i , α_i são parâmetros associados com link o i e a junta i , $C\theta = \cos\theta$ e $S\theta = \sin\theta$, $C\alpha = \cos\alpha$ e $S\alpha = \sin\alpha$. Os quatro parâmetros a_i , α_i , d_i e θ_i em (A.7) geralmente recebem o nome comprimento do link, torção do link, desvio de ligação e ângulo de junta, respectivamente, como observado na Figura 53.

Figura 53 – Coordenadas dos quadros para a convenção D-H.



Fonte: Spong e Vidyasagar (2008).

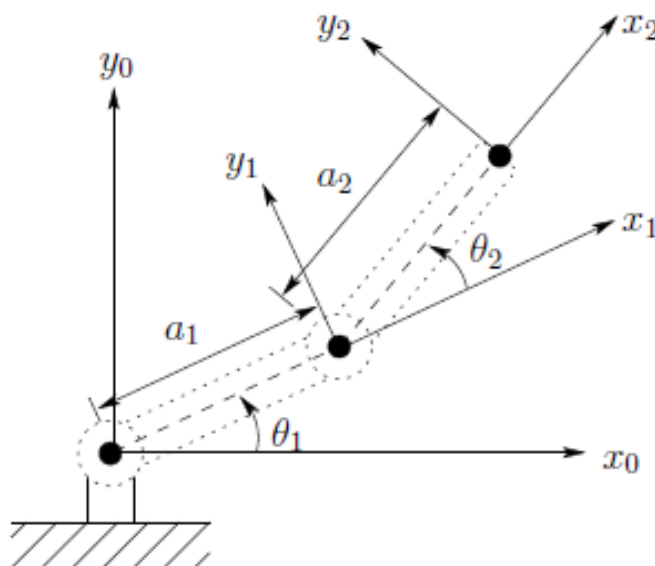
Esses parâmetros mostrados na Figura 53, derivam de aspectos específicos da relação geométrica entre dois quadros de coordenadas. Desde que a matriz A_i é uma função de uma única variável, verifica-se que três dos quatro parâmetros acima são constantes para um determinado link, enquanto o quarto parâmetro, θ_i que representa uma revolução da junta e d_i é a variável para uma junta prismática.

Na convenção D-H o único ângulo variável é θ , assim simplifica-se a notação escrevendo-se C_i para $\cos\theta_i$, etc. Denota-se $\theta_1 + \theta_2$ por θ_{12} e $\cos(\theta_1 + \theta_2)$ por C_{12} . É importante lembrar que a convenção D-H embora sistemática, ainda permite considerável liberdade na escolha de alguns dos parâmetros do manipulador, que corresponde ao caso de eixos de juntas paralelas ou quando as juntas prismáticas estão envolvidas.

A.2.2 Manipulador planar

Considere o braço planar de dois elos da Figura 54. Os eixos comuns z_0 e z_1 são normais à página.

Figura 54 – Manipulador planar de dois links



Fonte: Spong e Vidyasagar (2008).

Estabelece-se o quadro base $o_0x_0y_0z_0$ como mostrado na Figura 54. A origem é escolhida no ponto de intersecção do eixo z_0 com a página e a direção do eixo x_0 é completamente arbitrária. Uma vez que o frame da base é estabelecido, o frame $o_1x_1y_1z_1$ é fixado como mostrado pela convenção D-H, onde a origem o_1 foi localizada na intersecção de z_1 e da página. O quadro final $o_2x_2y_2z_2$ é fixado escolhendo a origem o_2 no final do link 2, como mostrado. Os parâmetros do link são mostrados na Tabela 14. As matrizes A são determinadas a partir de (A.7) como o * é para informar que θ é a grandeza variável.

Tabela 14 – Parâmetros
para um
manipula-
dor planar
de 2 juntas.

Junta	a_i	α_i	d_i	θ_i
1	a_1	0	0	θ_1^*
2	a_2	0	0	θ_2^*

Fonte: o autor.

$$A_1 = \begin{bmatrix} C_1 & -S_1 & 0 & a_1 C_1 \\ S_1 & C_1 & 0 & a_1 S_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{A.8})$$

$$A_2 = \begin{bmatrix} C_2 & -S_2 & 0 & a_2 C_2 \\ S_2 & C_2 & 0 & a_2 S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{A.9})$$

A matriz T , será

$$T_1^0 = A_1 \quad (\text{A.10})$$

$$T_2^0 = A_1 A_2 = \begin{bmatrix} C_{12} & -S_{12} & 0 & a_1 C_1 + a_2 C_{12} \\ S_{12} & C_{12} & 0 & a_1 S_1 + a_2 S_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{A.11})$$

Observe que as duas primeiras entradas da última coluna de T_2^0 são os componentes x e y da origem o_2 no quadro base; isto é,

$$x = a_1 C_1 + a_2 C_{12} \quad (\text{A.12})$$

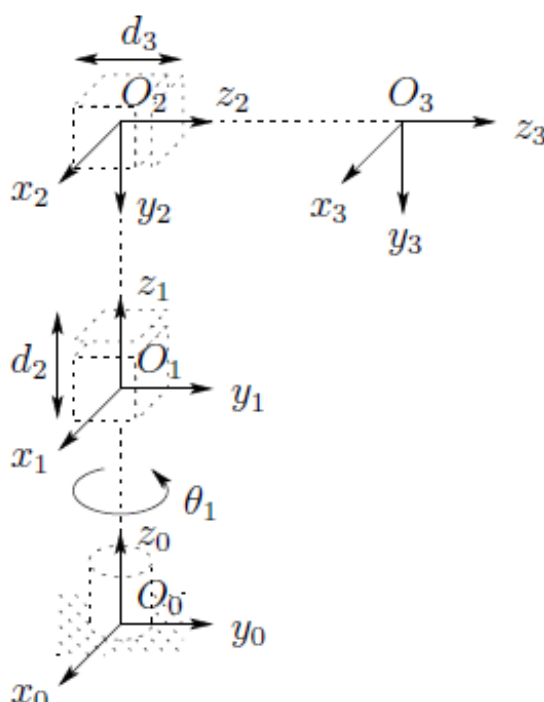
$$y = a_1 S_1 + a_2 S_{12}$$

são as coordenadas do *end-effector* no quadro de base. A parte rotacional de T_2^0 dá a orientação do quadro $o_2 x_2 y_2 z_2$ em relação ao quadro base.

A.2.3 Manipulador cilíndrico

Considere agora o robô cilíndrico de três elos representado simbolicamente pela Figura 55.

Figura 55 – Manipulador cilíndrico de três links



Fonte: Spong e Vidyasagar (2008).

Pode-se estabelecer o_0 como mostrado na junta 1. Note que o posicionamento da origem o_0 ao longo de z_0 , assim como a direção do eixo x_0 , são arbitrários. Nossa escolha de o_0 é a mais natural, mas o_0 também poderia ser colocado na junta 2. O eixo x_0 é escolhido como normal na página. Em seguida, como z_0 e z_1 coincidem, a origem o_1 é escolhida na junta 1, como mostrado. O eixo x_1 é normal para a página quando $\theta_1 = 0$ mas, sua direção mudará desde que θ_1 é variável. Como z_2 e z_1 se cruzam, a origem o_2 é colocada nessa interseção. A direção de x_2 é escolhida paralela a x_1 , de modo que θ_2 é igual a zero. Finalmente, o terceiro quadro é escolhido no final do link 3, como mostrado.

Os parâmetros de link agora são mostrados na Tabela 15.

Tabela 15 – Parâmetros de um manipulador cilíndrico de 3 juntas

Junta	a_i	α_i	d_i	θ_i
1	0	0	d_1	θ_1^*
2	0	-90°	d_2^*	0
3	0	0	d_3^*	0

Fonte: o autor.

As matrizes A e T correspondentes são

$$A_1 = \begin{bmatrix} C_1 & -S_1 & 0 & 0 \\ S_1 & C_1 & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{A.13})$$

$$A_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{A.14})$$

$$A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{A.15})$$

$$T_3^0 = A_1 A_2 A_3 = \begin{bmatrix} C_1 & 0 & -S_1 & -S_1 d_3 \\ S_1 & 0 & C_1 & C_1 d_3 \\ 0 & -1 & 0 & d_1 + d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{A.16})$$

A.3 Cinemática inversa

O problema da cinemática inversa pode ser resolvido através do princípio do desacoplamento cinemático e pode ser usado para simplificar a cinemática inversa da maioria dos manipuladores modernos. Usando o desacoplamento cinemático, podemos considerar os

problemas de posição e orientação de forma independente. Uma outra forma, de resolver o problema, é a abordagem geométrica que resolve o problema de posicionamento, enquanto exploramos a parametrização dos ângulos de Euler para resolver o problema de orientação (SPONG; VIDYASAGAR, 2008).

O problema geral da cinemática inversa pode ser declarado da seguinte maneira. Dada uma transformação homogênea 4×4 .

$$H = \begin{bmatrix} R & o \\ 0 & 1 \end{bmatrix} \quad (\text{A.17})$$

$$T_n^0(q_1, \dots, q_n) = H \quad (\text{A.18})$$

onde

$$T_n^0(q_1, \dots, q_n) = A_1(q_1) \dots A_n(q_n) \quad (\text{A.19})$$

Aqui, H representa a posição desejada e orientação do *end-effector*, e o problema é encontrar os valores para as variáveis das juntas q_1, \dots, q_n para que $T_n^0(q_1, \dots, q_n) = H$. A equação (A.18) resulta em doze equações não-lineares em n variáveis desconhecidas, que podem ser escritas como

$$T_{ij}(q_1, \dots, q_n) = h_{ij}, \quad i = 1, 2, 3, \quad j = 1, \dots, 4 \quad (\text{A.20})$$

onde T_{ij}, h_{ij} se refere às doze entradas não triviais de T_n^0 e H , respectivamente. Como a linha de baixo de ambos T_n^0 e H são $(0,0,0,1)$, quatro das dezesseis equações representadas por (A.18) são triviais.

As equações obtidas a partir de (A.18), são muito difíceis de resolver diretamente na forma fechada. Este é o caso da maioria dos braços robóticos. Portanto, precisamos desenvolver técnicas eficientes e sistemáticas que explorem a estrutura cinemática específica do manipulador. Enquanto o problema da cinemática direta sempre tem uma solução única que pode ser obtida simplesmente pela avaliação das equações avançadas, o problema da cinemática inversa pode ou não ter uma solução. Mesmo que exista uma solução, ela pode ou não ser única.

Além disso, porque estas equações cinemáticas avançadas são em geral funções não lineares complicadas das variáveis das juntas, as soluções podem ser difíceis de obter

mesmo quando elas existem. Na resolução do problema da cinemática inversa, estamos mais interessados em encontrar uma solução de forma fechada das equações do que uma solução numérica. Encontrar uma solução fechada significa encontrar uma solução explícita:

$$q_k = f_k(h_{11}, \dots, h_{34}), k = 1, \dots, n \quad (\text{A.21})$$

Soluções fechadas são preferíveis por dois motivos. Primeiro, em certas aplicações, como o rastreamento de uma emenda de solda cuja localização é fornecida por um sistema de visão, as equações cinemáticas inversas devem ser resolvidas rapidamente, digamos a cada 20 milissegundos, e ter expressões de forma fechada, em vez de uma pesquisa iterativa. Em segundo lugar, as equações cinemáticas em geral possuem múltiplas soluções. O fechamento de soluções fechadas permite desenvolver regras para a escolha de uma solução específica entre várias. A questão prática da existência de soluções para o problema da cinemática inversa depende da engenharia, bem como as considerações matemáticas.

Assumiremos que a posição e orientação dadas são tais que pelo menos uma solução de (A.18) existe. Uma vez que uma solução para as equações matemáticas é identificada, ela deve ser verificada mais detalhadamente para ver se ela satisfaz ou não todas as restrições nos intervalos de possíveis movimentos articulares. Assumimos que a dada matriz homogênea H em (A.18) corresponde a uma configuração dentro de o espaço de trabalho do manipulador com uma orientação atingível. Isto garante então que as soluções matemáticas obtidas correspondem a configurações alcançáveis.

A.3.1 Desacoplamento cinemático

Embora o problema geral da cinemática inversa seja bastante difícil, verifica-se que, para manipuladores com seis articulações, com as três últimas articulações se interceptando em um ponto, é possível desacoplar o problema da cinemática inversa em dois problemas mais simples, conhecidos respectivamente, como cinemática da posição inversa e cinemática da orientação inversa. Em outras palavras, para um manipulador de seis-DOF com um pulso esférico, o problema da cinemática inversa pode ser separado em dois problemas mais simples, a saber, primeiro encontra-se a posição da interseção dos eixos do pulso, daqui em diante chamada de centro do punho e depois encontra-se a orientação do pulso.

Vamos supor que exista exatamente seis graus de liberdade e que os últimos três eixos articulados se cruzam em um ponto o_c . Expressamos (A.18) como dois conjuntos de

equações representando as equações de rotação e posição

$$R_6^0(q_1, \dots, q_6) = R \quad (\text{A.22})$$

$$o_6^0(q_1, \dots, q_6) = o \quad (\text{A.23})$$

onde o e R são a posição e orientação desejadas do quadro do *end-effector*, expressas em relação ao sistema de coordenadas mundiais. Assim, temos o e R , e o problema da cinemática inversa é resolver para q_1, \dots, q_6 .

A suposição de um pulso esférico significa que os eixos z_3, z_4 e z_5 se cruzam em o_c e, portanto, as origens o_4 e o_5 atribuídas pela convenção DH estarão sempre no centro do pulso o_c . Muitas vezes o_3 também estará no o_c , mas isso não é necessário para o nosso desenvolvimento subsequente. O ponto importante desta suposição para a cinemática inversa é que o movimento dos últimos três elos sobre esses eixos não mudará a posição de o_c , e assim, a posição do centro do punho é, portanto, uma função apenas das três primeiras variáveis das juntas. A origem do quadro de ferramentas (cujas coordenadas desejadas são dadas por o) é simplesmente obtida por uma translação da distância d_6 ao longo de z_5 a partir de o_c . No nosso caso, z_5 e z_6 são o mesmo eixo, e a terceira coluna de R expressa a direção de z_6 em relação ao quadro de base. Portanto, temos

$$o = o_c^0 + d_6 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (\text{A.24})$$

Assim, para ter o *end-effector* do robô no ponto com coordenadas dadas por o e com a orientação do *end-effector* dado por $R = (r_{ij})$, é necessário e suficiente que o centro do punho, o_c tenha coordenadas dadas por

$$o_c^0 = o - d_6 R \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (\text{A.25})$$

e orientação do *frame* $o_6x_6y_6z_6$ em relação à base seja dada por R . Se os componentes da posição do *end-effector* o forem denotados por o_x, o_y, o_z e os componentes do centro do pulso o_c^0 são

chamados de x_c, y_c, z_c então (A.25) dá a relação

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} o_x - d_6 r_{13} \\ o_y - d_6 r_{23} \\ o_z - d_6 r_{33} \end{bmatrix} \quad (\text{A.26})$$

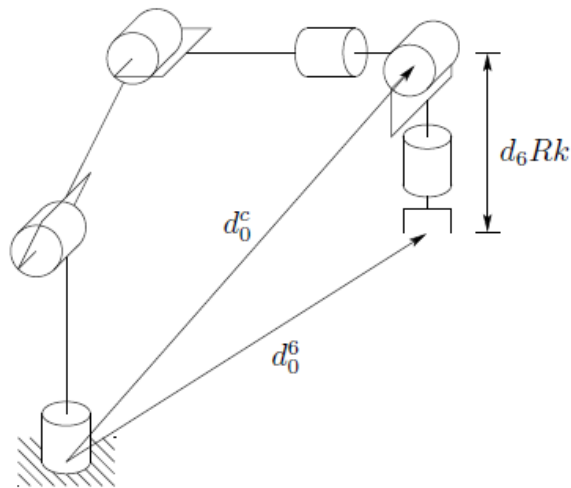
Usando a equação (A.26), podemos encontrar os valores das três primeiras variáveis das juntas. Esta determina a transformação de orientação R_3^0 que depende apenas destas três primeiras variáveis das juntas. Podemos agora determinar a orientação do *end-effector* em relação ao quadro $o_3x_3y_3z_3$ da expressão a seguir

$$R = R_3^0 R_6^3 \quad (\text{A.27})$$

$$R_6^3 = (R_3^0)^{-1} R = (R_3^0)^T R \quad (\text{A.28})$$

Os três ângulos finais podem ser encontrados como um conjunto de ângulos de Euler correspondentes a R_6^3 . Observe que o lado direito de (A.28) é completamente conhecido, uma vez que R é dado e R_3^0 pode ser calculado uma vez que as três primeiras variáveis das juntas são conhecidas. A ideia do desacoplamento cinemático é ilustrada na Figura 56.

Figura 56 – Desacoplamento cinemático



Fonte: Spong e Vidyasagar (2008).

A.3.2 Abordagem geométrica

Para os arranjos cinemáticos comuns que consideramos, podemos usar uma abordagem geométrica para encontrar as variáveis, q_1, q_2, q_3 correspondendo a o_c^0 dada por (A.25).

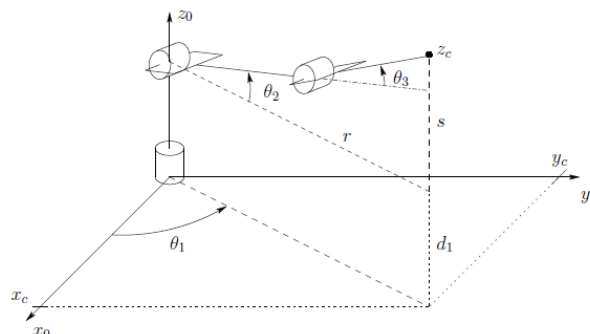
Restringimos nosso tratamento à abordagem geométrica por dois motivos. Primeiro, como já dissemos, os designs de manipulador mais atuais são cinematicamente simples, geralmente consistindo de uma das cinco configurações básicas do Capítulo 1 com um pulso esférico. De fato, é em parte devido à dificuldade do problema da cinemática inversa geral que os projetos de manipulador evoluíram ao seu estado atual. Em segundo lugar, existem poucas técnicas que podem lidar com o problema da cinemática inversa geral para configurações arbitrárias.

Em geral, a complexidade do problema da cinemática inversa aumenta com o número de parâmetros de ligação não nulos. Para a maioria dos manipuladores, muitos dos parâmetros a_i , d_i são zero, os α_i são zero ou $\pm\pi/2$, etc. Nesses casos, especialmente, uma abordagem geométrica é a mais simples e natural. Vamos ilustrar isso com vários exemplos importantes.

A.3.3 Configuração articulada

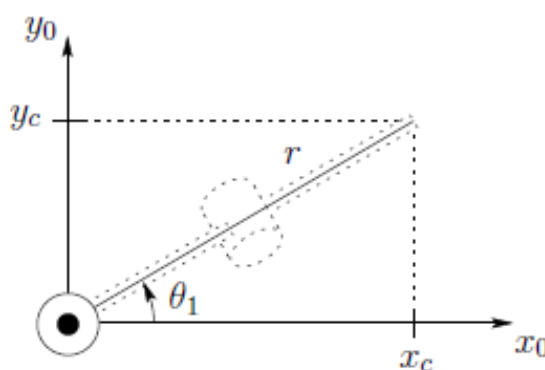
Considere o manipulador articulado mostrado na Figura 57, com os componentes de o_c^0 denotados por x_c , y_c , z_c . Nós projetamos o_c no plano $x_0 - y_0$ como mostrado na Figura 58.

Figura 57 – Desacoplamento cinemático



Fonte: Spong e Vidyasagar (2008).

Figura 58 – Projeção do centro do pulso no plano $x_0 - y_0$



Fonte: Spong e Vidyasagar (2008).

Podemos verifica nesta projeção que

$$\theta_1 = \text{Atan2}(x_c, y_c) \quad (\text{A.29})$$

em que $\text{Atan2}(x, y)$ é a função arco-tangente de dois argumentos. $\text{Atan2}(x, y)$ é definido para todos $(x, y) \neq (0, 0)$ e é igual ao único ângulo θ de tal modo que

$$\cos\theta = \frac{x}{(x^2 + y^2)^{\frac{1}{2}}}, \quad \sin\theta = \frac{y}{(x^2 + y^2)^{\frac{1}{2}}} \quad (\text{A.30})$$

Por exemplo $\text{Atan2}(1, -1) = -\frac{\pi}{4}$, enquanto $\text{Atan2}(-1, 1) = +\frac{3\pi}{4}$.

Observe que uma segunda solução válida para θ_1 é

$$\theta_1 = \pi + \text{Atan2}(x, y) \quad (\text{A.31})$$

A.4 Jacobiano de Manipuladores

O Jacobiano fornece as relações entre as velocidades lineares e angulares do *end-effector* (ou qualquer outro ponto no manipulador) com as velocidades das juntas. Em particular, deriva-se a velocidade angular da estrutura do *end-effector* (que fornece a taxa de rotação da estrutura) e a velocidade linear da origem. Essas velocidades estão relacionadas com as velocidades das juntas, q_i . Matematicamente, as equações cinemáticas avançadas definem uma função entre o espaço das posições e orientações cartesianas e o espaço das posições das articulações. As relações de velocidade são então determinadas pelo Jacobiano dessa função (SPONG; VIDYASAGAR, 2008).

O Jacobiano é uma função de valor matricial e pode ser considerada como a versão vetorial da derivada de uma função escalar. Esta matriz Jacobiana ou Jacobiana é uma das grandezas importante na análise e controle do movimento do robô. Surge em praticamente todos os aspectos da robótica tais como, no planejamento e execução de trajetórias suaves, na determinação de configurações singulares, na execução de movimento antropomórfico coordenado, na derivação das equações dinâmicas de movimento e na transformação de forças e torques do *end-effector* para as articulações do manipulador (SPONG; VIDYASAGAR, 2008).

Para determinar o Jacobiano de um manipulador de n-link primeiro deduz-se o jacobiano representando a transformação instantânea entre o vetor n das velocidades de junção e o vetor 6 consistindo das velocidades lineares e angulares do *end-effector*, obtendo-se assim uma matriz $6 \times n$. A mesma abordagem é usada para determinar a transformação entre as velocidades

das juntas e a velocidade linear e angular de qualquer ponto no manipulador, que é importante para a derivação das equações dinâmicas do movimento. Com o jacobiano pode-se encontrar as configurações singulares nas quais o manipulador perde um ou mais graus de liberdade, as quais são determinadas geometricamente. Pode-se obter também os problemas inversos da determinação das velocidades e acelerações da junta para velocidades e acelerações de atuadores finais especificados (SPONG; VIDYASAGAR, 2008).

A.4.1 Derivação do Jacobiano

Considere um manipulador n-link com variáveis das juntas q_1, \dots, q_n

$$T_n^0(q) = \begin{bmatrix} R_n^0(q) & o_n^0(q) \\ 0 & 1 \end{bmatrix}, \quad (\text{A.32})$$

que é a transformação do quadro do *end-effector* para o quadro base, onde $q = (q_1, \dots, q_n)^T$ é o vetor de variáveis das juntas. À medida que o robô se move, tanto as variáveis das juntas q_i como a posição do *end-effector* o_n^0 e a orientação R_n^0 serão funções do tempo. O objetivo agora é relacionar a velocidade linear e angular do *end-effector* ao vetor de velocidades de junção $\dot{q}(t)$. Desta forma, tem-se:

$$S(\omega_n^0) = \dot{R}_n^0 (R_n^0)^T, \quad (\text{A.33})$$

o vetor velocidade angular ω_n^0 , do *end-effector* é definido como

$$v_n^0 = \dot{o}_n^0, \quad (\text{A.34})$$

é a velocidade linear do *end-effector*. Procurou-se expressões da forma:

$$v_n^0 = Jv\dot{q}, \quad (\text{A.35})$$

$$\omega_n^0 = J\omega\dot{q}, \quad (\text{A.36})$$

onde Jv e $J\omega$ são matrizes $3 \times n$. Pode-se escrever (A.35) e (A.36) juntos como,

$$\begin{bmatrix} v_n^0 \\ \omega_n^0 \end{bmatrix} = J_n^0 \dot{q}, \quad (\text{A.37})$$

onde J_n^0 é dado por:

$$J_n^0 = \begin{bmatrix} J_v \\ J_\omega \end{bmatrix}. \quad (\text{A.38})$$

A matriz J_n^0 é chamada de Jacobiano do manipulador ou Jacobiano. Note que J_n^0 é uma matriz $6 \times n$ onde n é o número de links. Em seguida, deduz-se uma expressão simples para o Jacobiano de qualquer manipulador (SPONG; VIDYASAGAR, 2008).

A.4.2 Velocidade angular

As velocidades angulares podem ser adicionadas vetorialmente desde que sejam expressas em relação a um quadro de coordenadas comum. Assim, determina-se a velocidade angular do *end-effector* em relação à base expressando a velocidade angular contribuída por cada junta na orientação do referencial e, em seguida, somando-os (SPONG; VIDYASAGAR, 2008).

Se a i -ésima articulação é de revolução, então a i -ésima variável conjunta q_i é igual a θ_i e o eixo de rotação é z_{i-1} . Seguindo a convenção que foi apresentada acima, tem-se ω_i^{i-1} que representa a velocidade angular da junta i que é transmitida pela rotação da junta i , expressa em relação ao quadro $o_{i-1}x_{i-1}y_{i-1}z_{i-1}$. Esta velocidade angular é expressa no quadro $i-1$ por:

$$\omega_i^{i-1} = \dot{q}_i z_{i-1}^{i-1} = \dot{q}_i k, \quad (\text{A.39})$$

em que, k é o vetor de coordenadas unitárias $(0, 0, 1)^T$.

Se a i -ésima articulação é prismática, então o movimento do quadro i em relação ao quadro $i-1$ é uma translação e

$$\omega_i^{i-1} = 0. \quad (\text{A.40})$$

Assim, se a junta i é prismática, a velocidade angular do *end-effector* não depende de q_i , que agora é d_i .

Portanto, a velocidade angular total do *end-effector*, ω_n^0 , na estrutura da base é determinado por

$$\begin{aligned} \omega_n^0 &= \rho_1 \dot{q}_1 k + \rho_2 \dot{q}_2 R_1^0 k + \dots + \rho_n \dot{q}_n R_{n-1}^0 k \\ &= \sum_{i=1}^n \rho_i \dot{q}_i z_{i-1}^0 \end{aligned} \quad (\text{A.41})$$

em que ρ_i é igual a 1 se a articulação i é de revolução e 0 se a articulação i é prismática, uma vez que

$$z_{i-1}^0 = R_{i-1}^0 k. \quad (\text{A.42})$$

Observa-se que $z_0^0 = k = (0, 0, 1)^T$.

A metade inferior do Jacobiano $J\omega$, em (A.38) é assim dada como

$$J\omega = [\rho_1 z_0 \dots \rho_n z_{n-1}]. \quad (\text{A.43})$$

Da equação A.42, omitiu-se os sobrescritos para os vetores unitários ao longo dos eixos z , pois todos são referenciados ao quadro inercial.

A.4.3 Velocidade linear

A velocidade linear do *end-effector* é de apenas \dot{o}_n^0 . Pela regra da cadeia para diferenciação

$$\dot{o}_n^0 = \sum_{i=1}^n \frac{\partial o_n^0}{\partial q_i} \dot{q}_i. \quad (\text{A.44})$$

Assim, a i -ésima coluna de J_v , foi chamada de J_{v_i} , é dada por

$$J_{v_i} = \frac{\partial o_n^0}{\partial q_i}. \quad (\text{A.45})$$

Além disso, essa expressão é apenas a velocidade linear do *end-effector* que resultaria se \dot{q}_i fosse igual a um e o outro \dot{q}_j fosse zero. Em outras palavras, a i -ésima coluna do Jacobiano pode ser gerada segurando todas as juntas fixas e a i -ésima atuando com a i -ésima velocidade unitária. Considera-se agora os dois casos (juntas prismáticas e revolucionárias) separadamente.

A.4.4 Combinando os Jacobianos angular e linear

Como apresentado na seção anterior, a metade superior do Jacobiano, J_v é dada como

$$J_v = [J_{v_1} \dots J_{v_n}], \quad (\text{A.46})$$

onde a i -ésima coluna J_{v_i} é

$$J_{v_i} = z_{i-1} \times (o_n - o_{i-1}), \quad (\text{A.47})$$

se a junta i é de revolução e

$$J_{v_i} = z_{i-1}, \quad (\text{A.48})$$

se a junta i é prismática.

A metade inferior do jacobiano é dada como

$$J_{\omega} = [J_{\omega_1} \dots J_{\omega_n}], \quad (\text{A.49})$$

onde a i -ésima coluna J_{ω_i} é

$$J_{\omega_i} = z_{i-1}. \quad (\text{A.50})$$

se a junta i é de revolução e

$$J_{\omega_i} = 0, \quad (\text{A.51})$$

se a junta i é prismática.

Colocando-se as metades superior e inferior do Jacobiano juntos, mostramo-se que o Jacobiano para um manipulador n -link é da forma

$$J = [J_1 J_2 \dots J_n], \quad (\text{A.52})$$

onde a i -ésima coluna J_i é dada por

$$J_i = \begin{bmatrix} z_{i-1} \times (o_n - o_{i-1}) \\ z_{i-1} \end{bmatrix}, \quad (\text{A.53})$$

se a junta i é de revolução e

$$J_i = \begin{bmatrix} z_{i-1} \\ 0 \end{bmatrix}, \quad (\text{A.54})$$

se a junta i é prismática.

As equações A.53 e A.54 determinam o Jacobiano de qualquer manipulador, já que todas as quantidades necessárias estão disponíveis uma vez que a cinemática direta é calculada. ou seja, as únicas quantidades necessárias para calcular o jacobiano são os vetores unitários z_i e as coordenadas das origens o_1, \dots, o_n . Observa-se que as coordenadas para z_i com relação ao quadro da base é dado pelos três primeiros elementos da terceira coluna de T_i^0 , enquanto o_i é dado pelos três primeiros elementos da quarta coluna de T_i^0 . Assim, tem-se que a terceira e quarta colunas das matrizes T são necessárias para avaliar o Jacobiano de acordo com as fórmulas acima (SPONG; VIDYASAGAR, 2008).

O procedimento acima funciona não apenas para calcular a velocidade do *end-effector*, mas também para calcular a velocidade de qualquer ponto no manipulador. Isto será importante quando necessita-se calcular a velocidade do centro de massa dos vários elos para derivar as equações dinâmicas do movimento.

A.5 Dinâmica de Manipuladores

A dinâmica estuda o movimento de um sistema levando-se em consideração as forças que produzem o movimento. A dinâmica de um manipulador rígido de n elos é convenientemente descrita pela formulação Lagrangiana (MITTAL; NAGRATH, 2003). No enfoque Lagrangiano as variáveis das juntas q são um conjunto adequado de coordenadas generalizadas.

Para o propósito de projetar um controlador é necessário ter um modelo matemático que revele as propriedades dinâmicas do sistema. Desta forma, deduz-se as equações dinâmicas do movimento para o manipulador. Em seguida, calcula-se a equação da energia cinética e potencial do manipulador e aplica-se a equação de Lagrange para o movimento (SANZ, 2009).

A.5.1 Equações de Euler-Lagrange

As equações de movimento de Euler-Lagrange são encontradas a partir de um conjunto geral de equações diferenciais que descrevem a evolução temporal dos sistemas mecânicos sujeitos a restrições holonômicas quando as forças de restrição satisfazem o princípio do trabalho virtual. Note que há pelo menos duas maneiras distintas de derivar essas equações. O método apresentado aqui é baseado no método de deslocamentos virtuais; mas também é possível derivar as mesmas equações baseadas no princípio de menor ação de Hamilton (SPONG; VIDYASAGAR, 2008).

Para motivar a derivação subsequente, mostra-se primeiro como as equações Euler-Lagrange podem ser derivadas da Segunda Lei de Newton para um único grau de liberdade de um sistema consistindo de uma partícula de massa constante m , restrito a move-se na direção y , e sujeito a uma força f e à força gravitacional mg , como mostrado na Figura 59.

Pela segunda lei de Newton, a equação de movimento da partícula é

$$F = ma, \tag{A.55}$$

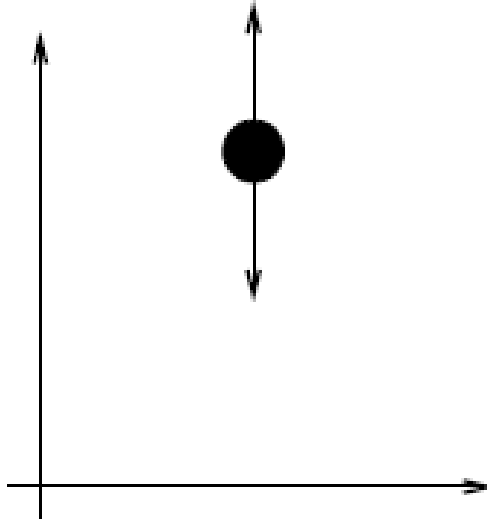
em que F é força, m a massa e a aceleração.

$$m\ddot{y} = f - mg. \tag{A.56}$$

Observe que o lado esquerdo da equação (A.56) pode ser escrito como a seguir, em que o símbolo, $\dot{}$, representa a derivada dessa variável em função do tempo, ou seja,

$$\frac{d}{dt}(m\dot{y}) = \frac{d}{dt} \frac{\partial}{\partial \dot{y}} \left(\frac{1}{2} m \dot{y}^2 \right) = \frac{d}{dt} \frac{\partial K}{\partial \dot{y}}, \tag{A.57}$$

Figura 59 – Sistema com um grau de liberdade



Fonte: Spong e Vidyasagar (2008).

em que $K = \frac{1}{2}m\dot{y}^2$ é a energia cinética. Utiliza-se a notação derivada parcial na expressão acima para ser consistente com sistemas considerados mais tarde quando a energia cinética será uma função de diversas variáveis. Da mesma forma, pode-se expressar a força gravitacional na equação (A.56) como

$$mg = \frac{\partial}{\partial y}(mgy) = \frac{\partial P}{\partial y}, \quad (\text{A.58})$$

em que $P = mgy$ é a energia potencial devido a gravidade. Definindo-se

$$L = K - P = \frac{1}{2}m\dot{y}^2 - mgy, \quad (\text{A.59})$$

nota-se que

$$\frac{\partial L}{\partial \dot{y}} = \frac{\partial K}{\partial \dot{y}} \text{ e } \frac{\partial L}{\partial y} = -\frac{\partial P}{\partial y}, \quad (\text{A.60})$$

a equação (A.56) pode ser escrita como,

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{y}} - \frac{\partial L}{\partial y} = f. \quad (\text{A.61})$$

A função L , que é a diferença da energia cinética e potencial é chamada de **Lagrangiana** do sistema e a equação (A.61) é chamada de **Equação de Euler-Lagrange**.

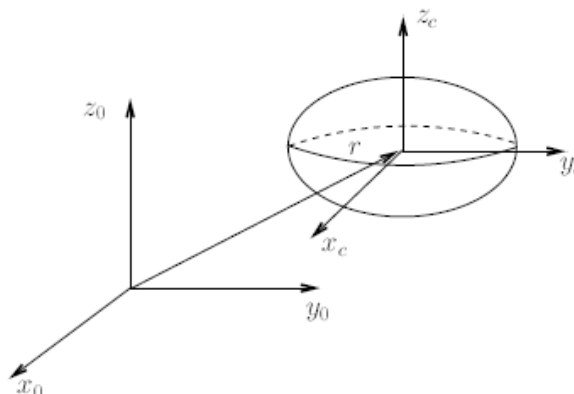
A.5.2 Expressões Gerais para Energia Cinética e Potencial

As equações de Euler-Lagrange podem ser usadas para derivar as equações dinâmicas de maneira direta, desde que seja possível expressar a energia cinética e potencial do sistema em

termos de um conjunto de coordenadas generalizadas. Para que este resultado seja útil em um contexto prático, necessita-se computar esses termos para um manipulador robótico de n-link. Para isso, é preciso deduzir-se fórmulas para a energia cinética e a energia potencial de um robô rígido usando as variáveis das juntas encontradas através da conversão D-H (SICILIANO *et al.*, 2010).

A energia cinética de um objeto rígido é a soma de dois termos: a energia translacional obtida pela concentração de toda a massa do objeto no centro de massa e a energia cinética rotacional do corpo sobre o centro de massa. Referindo-se à Figura 60, anexa-se um quadro de coordenadas no centro de massa (chamado de quadro anexado ao corpo), como apresentado (SPONG; VIDYASAGAR, 2008).

Figura 60 – Corpo rígido geral.



Fonte: Spong e Vidyasagar (2008).

A energia cinética do corpo rígido é então dada como,

$$K = \frac{1}{2}mv^T v + \frac{1}{2}\omega^T I \omega, \quad (\text{A.62})$$

em que m é a massa total do objeto, v e ω são os vetores de velocidade linear e angular, respectivamente, e I é uma matriz simétrica 3×3 chamada de Tensor de Inércia.

A.5.2.1 Tensor de Inércia

Entende-se que os vetores de velocidade linear e angular, v e ω , respectivamente, na expressão acima para a energia cinética são expressos no referencial inercial. Neste caso, sabemos que ω é encontrado a partir da matriz simétrica de inclinação

$$S(\omega) = \dot{R}R^T \quad (\text{A.63})$$

onde R é a transformação de orientação entre o quadro anexado ao corpo e o referencial inercial. Portanto, é necessário expressar o tensor de inércia, I , também no referencial inercial, para calcular o produto triplo $\omega^T I \omega$. O tensor de inércia relativo ao referencial inercial dependerá da configuração do objeto. O tensor de inércia, I , expresso em vez do quadro anexado ao corpo, então as duas matrizes são relacionadas através de uma transformação de similaridade de acordo com

$$I = RIR^T \quad (\text{A.64})$$

Esta é uma observação importante porque a matriz de inércia expressa no quadro anexado ao corpo é uma matriz constante independente do movimento do objeto e facilmente computada. Em seguida, mostramos como calcular essa matriz explicitamente.

A densidade de massa do objeto é representada como uma função da posição, $\rho(x, y, z)$. Então o tensor de inércia no quadro anexado ao corpo é computado como

$$I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \quad (\text{A.65})$$

Os elementos diagonais do tensor de inércia, I_{xx} , I_{yy} , I_{zz} , são chamados de Momentos Principais de Inércia em relação aos eixos x, y, z , respectivamente. Os termos fora da diagonal I_{xy} , I_{xz} , etc., são chamados de Produtos Cruzados de Inércia. Se a distribuição de massa do corpo é simétrica em relação ao corpo emoldurado, então os produtos cruzados de inércia são identicamente iguais a zero.

A.5.2.2 Energia cinética para um robô de n -Link

Considerando-se um manipulador que consiste em n links. Verifica-se que as velocidades lineares e angulares de qualquer ponto em qualquer elo podem ser expressas em termos da matriz Jacobiana e a derivada das variáveis das juntas. Como no nosso caso as variáveis das juntas são de fato as coordenadas generalizadas, segue-se que, para as matrizes Jacobianas apropriadas J_{v_i} e J_{ω_i} , tem-se que

$$v_i = J_{v_i}(q)\dot{q}, \quad \omega_i = J_{\omega_i}(q)\dot{q}. \quad (\text{A.66})$$

A massa de um link i é m_i e a matriz de inércia do link i , em torno de um quadro de coordenadas paralelo ao quadro i , mas cuja origem está no centro de massa, seja igual a I_i .

Então, de (A.62), segue-se que a energia cinética geral do manipulador é igual a

$$K = \frac{1}{2} \dot{q}^T \sum_{i=1}^n [m_i J_{v_i}(q)^T J_{v_i}(q) + J_{\omega_i}(q)^T R_i(q) I_i R_i(q^T) J_{\omega_i}(q)] \dot{q}. \quad (\text{A.67})$$

Ou seja, a energia cinética do manipulador é da forma

$$K = \frac{1}{2} \dot{q}^T D(q) \dot{q}, \quad (\text{A.68})$$

em que $D(q)$ é uma matriz definida positiva simétrica que é em geral dependente da configuração.

A matriz D é chamada de matriz de inércia.

A.5.2.3 Energia potencial para um robô de n -Link

Considerando-se agora o termo de energia potencial, no caso da dinâmica rígida, a única fonte de energia potencial é a gravidade. A energia potencial da i -ésima ligação pode ser calculada assumindo que a massa de todo o objeto está concentrada em seu centro de massa e é dada por:

$$P_i = g^T r_{ci} m_i, \quad (\text{A.69})$$

em que g é o vetor dado na direção da gravidade no referencial inercial e o vetor r_{ci} são as coordenadas do centro de massa do *link* i . A energia potencial total do robô com n -link é, portanto,

$$P = \sum_{i=1}^n P_i = \sum_{i=1}^n g^T r_{ci} m_i. \quad (\text{A.70})$$

No caso de o robô conter elasticidade, por exemplo, juntas flexíveis, a energia potencial incluirá termos contendo a energia armazenada nos elementos elásticos. Observa-se que a energia potencial é uma função das coordenadas generalizadas e não de suas derivadas, ou seja, a energia potencial depende da configuração do robô, mas não da sua velocidade.

A.5.3 Equações de Movimento

As equações de Euler-Lagrange mostradas anteriormente para o caso especial quando duas condições são válidas, isto é: primeiro, a energia cinética é uma função quadrática do vetor \dot{q} da forma

$$K = \frac{1}{2} \sum_{ij} d_{ij}(q) \dot{q}_i \dot{q}_j := \frac{1}{2} \dot{q}^T D(q) \dot{q}, \quad (\text{A.71})$$

em que $D(q)$ ($n \times n$ é a matriz de inércia) e é simétrica e positiva definida para cada $q \in R^n$, e segundo, a energia potencial $P = P(q)$ é independente de \dot{q} . Já observou-se que os manipuladores robóticos satisfazem essa condição.

As equações de Euler-Lagrange para tal sistema podem ser derivadas como segue (SPONG; VIDYASAGAR, 2008). Desde que

$$L = K - P = \frac{1}{2} \sum_{ij} d_{ij}(q) \dot{q}_i \dot{q}_j - P(q), \quad (\text{A.72})$$

tem-se que,

$$\frac{\partial L}{\partial \dot{q}_k} = \sum_j d_{kj} \dot{q}_j, \quad (\text{A.73})$$

e

$$\begin{aligned} \frac{d}{dt} \frac{\partial L}{\partial \dot{q}_k} &= \sum_i d_{kj} \ddot{q}_j + \sum_j \frac{d}{dt} d_{kj} \dot{q}_j \\ &= \sum_j d_{kj} \ddot{q}_j + \sum_{ij} \frac{\partial d}{\partial q_i} d_{kj} \dot{q}_i \dot{q}_j. \end{aligned} \quad (\text{A.74})$$

Também, tem-se:

$$\frac{\partial L}{\partial q_k} = \frac{1}{2} \sum_{ij} \frac{\partial d_{ij}}{\partial q_k} \dot{q}_i \dot{q}_j - \frac{\partial P}{\partial q_k}. \quad (\text{A.75})$$

Assim, as equações de Euler-Lagrange podem ser escritas

$$\sum_j d_{kj} \ddot{q}_j + \sum_{ij} \left\{ \frac{\partial d_{kj}}{\partial q_i} - \frac{1}{2} \frac{\partial d_{ij}}{\partial q_k} \right\} \dot{q}_i \dot{q}_j - \frac{\partial P}{\partial q_k} = \tau_k. \quad (\text{A.76})$$

Ao trocar a ordem do somatório e tirar proveito da simetria, pode-se mostrar que:

$$\sum_{ij} \left\{ \frac{\partial d_{kj}}{\partial q_i} \right\} \dot{q}_i \dot{q}_j = \frac{1}{2} \sum_{ij} \left\{ \frac{\partial d_{kj}}{\partial q_i} + \frac{\partial d_{ki}}{\partial q_j} \right\} \dot{q}_i \dot{q}_j. \quad (\text{A.77})$$

Consequentemente,

$$\sum_{ij} \left\{ \frac{\partial d_{kj}}{\partial q_i} - \frac{1}{2} \frac{\partial d_{ij}}{\partial q_k} \right\} \dot{q}_i \dot{q}_j = \sum_{ij} \frac{1}{2} \left\{ \frac{\partial d_{kj}}{\partial q_i} + \frac{\partial d_{ki}}{\partial q_j} - \frac{\partial d_{ij}}{\partial q_k} \right\} \dot{q}_i \dot{q}_j. \quad (\text{A.78})$$

Os termos

$$c_{ijk} := \frac{1}{2} \left\{ \frac{\partial d_{kj}}{\partial q_i} + \frac{\partial d_{ki}}{\partial q_j} - \frac{\partial d_{ij}}{\partial q_k} \right\}, \quad (\text{A.79})$$

são conhecidos como símbolos Christoffel (do primeiro tipo). Note que para um k fixo tem-se $c_{ijk} = c_{jik}$, que reduz o esforço envolvido na computação desses símbolos por um fator de cerca da metade. Finalmente, se definindo-se,

$$\phi_k = \frac{\partial P}{\partial q_k}, \quad (\text{A.80})$$

então pode-se escrever as equações de Euler-Lagrange como,

$$\sum_i d_{kj}(q)\ddot{q}_j + \sum_{ij} c_{ijk}(q)\dot{q}_i\dot{q}_j + \phi_k(q) = \tau_k, \quad k = 1, \dots, n. \quad (\text{A.81})$$

Na equação acima, existem três tipos de termos. O primeiro, envolve a segunda derivada das coordenadas generalizadas e o segundo são termos quadráticos nas primeiras derivadas de q , onde os coeficientes podem depender de q . Estes são ainda classificados em dois tipos. Termos envolvendo um produto do tipo \dot{q}_i^2 são chamados de centrífugos, enquanto aqueles envolvendo um produto do tipo $\dot{q}_i\dot{q}_j$ onde $i \neq j$ são chamados termos de Coriolis. O terceiro tipo de termos são aqueles envolvendo apenas q , mas não seus derivados. Observa-se que os últimos termos surgem da diferenciação da energia potencial. É comum escrever a equação (A.81) em forma de matriz como,

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau, \quad (\text{A.82})$$

em que o k, j –ésimo elemento da matriz $C(q, \dot{q})$ é definido como:

$$\begin{aligned} c_{kj} &= \sum_{i=1}^n c_{ijk}(q)\dot{q}_i \\ &= \sum_{i=1}^n \frac{1}{2} \left\{ \frac{\partial d_{kj}}{\partial q_j} + \frac{\partial d_{ki}}{\partial q_j} - \frac{\partial d_{ij}}{\partial q_k} \right\} \dot{q}_i. \end{aligned} \quad (\text{A.83})$$

Examinando-se um caso especial onde a matriz de inércia é diagonal e independente de q , segue-se neste caso que da equação (A.79) tem-se todos os símbolos de Christoffel são zero, uma vez que cada d_{ij} é uma constante. Além disso, a quantidade d_{kj} é diferente de zero se e somente se $k = j$, de forma que a equação (A.81) se dissociem na forma,

$$d_{kk}\ddot{q} - \phi_k(q) = \tau_k, \quad k = 1, \dots, n. \quad (\text{A.84})$$

Resumindo-se tem-se nesta seção o desenvolvimento geral que se aplica a qualquer sistema mecânico cuja energia cinética esteja na forma da equação (A.71) e cuja energia potencial seja independente de \dot{q} .

A.6 Dinâmica Inversa

O cálculo da dinâmica inversa de um robô manipulador é essencial para o sistema de controle. As abordagens comumente usadas para formular modelos dinâmicos de robôs manipuladores são os métodos de Newton-Euler e Lagrange-Euler. Como os dois métodos são

métodos recursivos numéricos, eles são computacionalmente caros e não são adequados para serem usados diretamente em aplicações em tempo real (CHENG *et al.*, 2015).

A abordagem do controle da dinâmica inversa faz uso da relação entre os torques das juntas como as forças de contato das entradas e do atuador final e as variáveis de posição do *end-effector* ao longo das superfícies de restrição como as saídas (KORKMAZ; IDER, 2014).

A dinâmica inversa é uma abordagem em que um *loop* de linearização de *feedback* é aplicado às saídas de rastreamento de interesse. A dinâmica residual, não controlada diretamente, é conhecida como dinâmica interna. Se as dinâmicas internas são estáveis, a inversão dinâmica é bem sucedida. O uso típico requer a seleção das variáveis de controle de saída para que a dinâmica interna seja garantida como estável. Isso significa que o rastreamento nem sempre pode ser garantido para as saídas originais de que se deseja (DAS *et al.*, 2009).

A ideia da dinâmica inversa é buscar uma lei de controle de feedback não-linear

$$u = f(q, \dot{q}, t) \quad (\text{A.85})$$

que resulta em um sistema linear de malha fechada. Para sistemas não lineares gerais, tal lei de controle pode ser bastante difícil ou impossível de encontrar.

Considere novamente as equações dinâmicas de um robô n-link em forma de matriz (SPONG; VIDYASAGAR, 2008)

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + B(\dot{q}) + g(q) = u \quad (\text{A.86})$$

onde $M(q) = D(q) + J$, onde J é uma matriz diagonal com elementos diagonais $r_k^2 J_{m_k}$. O vetor de entrada u tem componentes

$$u_k = r_k \frac{K_{m_k}}{R_k} V_k \quad (\text{A.87})$$

Note que o u_k tem unidades de torque. Para sistemas não lineares gerais, tal lei de controle pode ser bastante difícil ou impossível de encontrar. No caso da equação dinâmica do manipulador A.87, no entanto, o problema é menos complicado. Ao inspecionar A.87 vemos que se escolhermos o controle u de acordo com a equação

$$u = M(q)a_q + C(q, \dot{q})\dot{q} + g(q) \quad (\text{A.88})$$

então, como a matriz de inércia M é invertível, o sistema combinado A.86 e A.88 reduz para

$$\ddot{q} = a_q \quad (\text{A.89})$$

O termo a_q representa uma nova entrada para o sistema que ainda está para ser escolhido. A equação A.89 é conhecida como sistema de integração dupla, pois representa dois integradores duplos desacoplados. A lei de controle não-linear A.88 é chamada de *inverse dynamics control* alcança um resultado bastante notável, a saber, que o “novo” sistema A.89 é linear e desacoplado. Isto significa que cada entrada a_{qk} pode ser projetada para controlar um sistema linear escalar. Além disso, assumindo que a_{qk} é uma função apenas de q_k e suas derivadas, então a_{qk} afetará q_k independentemente do movimento dos outros links (SPONG; VIDYASAGAR, 2008).

Como o a_{qk} agora pode ser projetado para controlar um sistema linear de segunda ordem, a escolha óbvia é definir

$$a_q = -K_0q - K_1\dot{q} + r \quad (\text{A.90})$$

onde K_0 e K_1 são matrizes diagonais com elementos diagonais consistindo de ganhos de posição e velocidade, respectivamente. O sistema de circuito fechado é então o sistema linear

$$\dot{q} + K_1\dot{q} + K_0q = r \quad (\text{A.91})$$

Agora, dada uma trajetória desejada

$$t \rightarrow (q^d(t), \dot{q}^d(t)) \quad (\text{A.92})$$

se se escolher a entrada de referência $r(t)$ como

$$r(t) = \ddot{q}^d(t) + K_0q^d(t) + K_1\dot{q}^d(t) \quad (\text{A.93})$$

então o erro de rastreamento $e(t) = q - q^d$ satisfaz

$$\ddot{e}(t) + K_1\dot{e}(t) + K_0e(t) = 0 \quad (\text{A.94})$$

Uma escolha simples para as matrizes de ganho K_0 e K_1 é

$$K_0 = \text{diag}\{\omega_1^2, \dots, \omega_n^2\} \quad (\text{A.95})$$

$$K_1 = \text{diag}\{2\omega_1, \dots, 2\omega_n\} \quad (\text{A.96})$$

que resulta em um sistema de malha fechada que é dissociado globalmente, com cada resposta da junta igual à resposta de um sistema linear de segunda ordem criticamente amortecido com

frequência natural ω_i . Como antes, a frequência natural ω_i determina a velocidade de resposta da articulação ou, equivalentemente, a taxa de decaimento do erro de rastreamento.

A abordagem de dinâmica inversa é extremamente importante para o controle de robôs manipuladores e vale a pena tentar vê-la de pontos de vista alternativos. Pode-se dar uma segunda interpretação da lei de controle A.88 da seguinte maneira. Considere novamente as equações dinâmicas do manipulador A.86. Como $M(q)$ é invertível para $q \in \mathbb{R}^n$, podemos resolver a aceleração \ddot{q} do manipulador como

$$\ddot{q} = M^{-1}\{u - C(q, \dot{q})\dot{q} - g(q)\}. \quad (\text{A.97})$$

Suponha que pudéssemos especificar a aceleração como a entrada para o sistema. Ou seja, suponha que tivéssemos atuadores capazes de produzir diretamente uma aceleração comandada (em vez de indiretamente, produzir uma força ou torque). Então a dinâmica do manipulador, que é afinal um dispositivo de controle de posição, seria dada como:

$$\ddot{q} = a_q(t) \quad (\text{A.98})$$

onde $a_q(t)$ é o vetor de aceleração de entrada. Este é novamente o sistema familiar de integrador duplo. Note que A.98 não é uma aproximação em nenhum sentido; ao contrário, representa a dinâmica real do circuito aberto do sistema, desde que a aceleração seja escolhida como entrada. O problema de controle do sistema A.98 agora é fácil e a entrada de aceleração a_q pode ser escolhida como antes, de acordo com A.90.

Na realidade, no entanto, esses "atuadores de aceleração" não estão disponíveis para nós e devemos nos contentar com a capacidade de produzir uma força generalizada (torque) u_i em cada junta i . Comparando as equações A.97 e A.98, vemos que o torque u e a aceleração a_q do manipulador são relacionados por

$$M^{-1}\{u(t) - C(q, \dot{q})\dot{q} - g(q)\} = a_q \quad (\text{A.99})$$

Pela invertibilidade da matriz de inércia, podemos resolver o torque de entrada $u(t)$ como

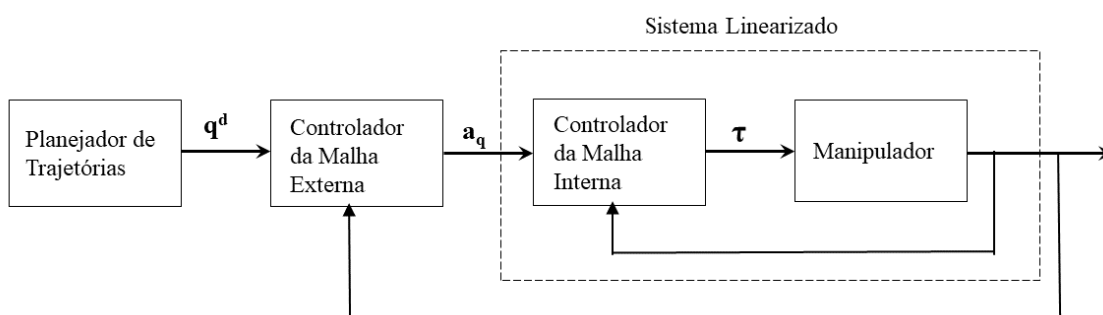
$$u = M(q)a_q + C(q, \dot{q})\dot{q} + g(q) \quad (\text{A.100})$$

que é igual à expressão derivada anteriormente A.88. Assim, a dinâmica inversa pode ser vista como uma transformação de entrada que transforma o problema de uma escolha de comandos de

entrada de torque, que é difícil, para uma escolha de comandos de entrada de aceleração, o que é fácil.

A implementação desse esquema de controle requer o cálculo em cada instante da amostra da matriz de inércia $M(q)$ e o vetor de Coriolis, centrífugo e gravitacional. A dinâmica inversa deve ser calculada *on-line*. Como uma lei de controle de realimentação, ela não pode ser pré-computada *off-line* e armazenada como o torque calculado. Uma questão importante, portanto, na implementação do sistema de controle é o projeto do controlador. O esquema de controle é apresentado na Figura 61.

Figura 61 – Arquitetura da malha interna/controle da malha externa.



Fonte: Adaptado de Spong e Vidyasagar (2008).

A Figura 61 ilustra a noção da malha de controle interna/externa. Com isso, o controle não-linear é realizada em uma malha interna. A malha externa no sistema é então o cálculo do termo de entrada adicional a_q . O controle da malha externa a_q está mais alinhado com a noção de controle de *feedback* no sentido usual de ser acionado por erro. O projeto do controlador de realimentação da malha externa é, em teoria, bastante simplificado, pois pode ser projetado para a planta representada pelas linhas pontilhadas na Figura 61, que agora é um sistema linear ou quase linear.

A.6.1 Dinâmica Inversa do Espaço de Tarefas

O rastreamento no espaço de tarefas pode ser encontrado modificando a escolha do controle da malha externa, \ddot{q} em A.89, mantendo o controle da malha interna inalterada. Seja $X \in \mathbb{R}^6$ a posição do *end-effector*, como X é uma função das variáveis das juntas $q \in C$, temos

$$\dot{X} = J(q)\dot{q} \quad (\text{A.101})$$

$$\dot{X} = J(q)\ddot{q} + \dot{J}(q)\dot{q}. \quad (\text{A.102})$$

onde $J = J_a$ é o Jacobiano analítico. Dado o sistema de integrador duplo, A.89, no espaço das juntas, tem-se a_q que é escolhido como

$$a_q = J^{-1}\{a_X - \dot{J}\dot{q}\} \quad (\text{A.103})$$

o resultado é um sistema integrador duplo nas coordenadas do espaço de tarefas

$$\ddot{X} = a_X \quad (\text{A.104})$$

Dada uma trajetória no espaço de tarefa $X^d(t)$, satisfazendo as mesmas suposições de suavidade e delimitação que a trajetória de espaço das juntas $q^d(t)$, pode-se escolher a_X como

$$a_X = \ddot{X}^d + K_P(X^d - X) + K_D(\dot{X}^d - \dot{X}) \quad (\text{A.105})$$

para que o erro de rastreamento do espaço cartesiano, $\tilde{X} = X - X_d$, satisfaça

$$\ddot{\tilde{X}} + K_D\dot{\tilde{X}} + K_P\tilde{X} = 0. \quad (\text{A.106})$$

Portanto, uma modificação do controle da malha externa atinge um sistema linear e desacoplado diretamente nas coordenadas do espaço de tarefas, sem a necessidade de calcular uma trajetória no espaço das juntas e sem a necessidade de modificar o controle da malha interna não linear.

Usando-se uma representação mínima para a orientação do *end-effector*, a fim de especificar uma trajetória $X \in \mathbb{R}^6$. Em geral, se as coordenadas do *end-effector* são dadas em função do jacobiano, J , a formulação acima será o Jacobiano geométrico J . Nesse caso

$$V = \begin{pmatrix} v \\ \omega \end{pmatrix} = \begin{pmatrix} \dot{x} \\ \dot{\omega} \end{pmatrix} = J(q)\dot{q} \quad (\text{A.107})$$

e o controle da malha externa é

$$a_q = J^{-1}(q)\left\{\begin{pmatrix} a_x \\ a_\omega \end{pmatrix} - J(q)\dot{q}\right\} \quad (\text{A.108})$$

aplicado em A.89 resulta no sistema

$$\ddot{x} = a_x \in \mathbb{R}^3 \quad (\text{A.109})$$

$$\dot{\omega} = a_{\omega} \in \mathbb{R}^3 \quad (\text{A.110})$$

Embora, neste último caso, a dinâmica não tenha sido linearizada para um integrador duplo, os termos da malha externa a_v e a_{ω} ainda pode ser usado para definir as leis de controle para rastrear trajetórias do *end-effector*.

Nos dois casos, vemos que a não singularidade do jacobiano é necessária para implementar o controle de malha externa. Se o robô tiver menos que seis articulações, os jacobianos não são quadrados. Nesse caso, o problema pode ser resolvido resolvendo a pseudo-inverso no lugar do inverso do jacobiano. A abordagem de controle de dinâmica inversa pode ser proposta em várias formas diferentes, como controle de aceleração resolvido, controle de espaço operacional (SPONG; VIDYASAGAR, 2008).

A.7 Resposta ao Modelo Dinâmico

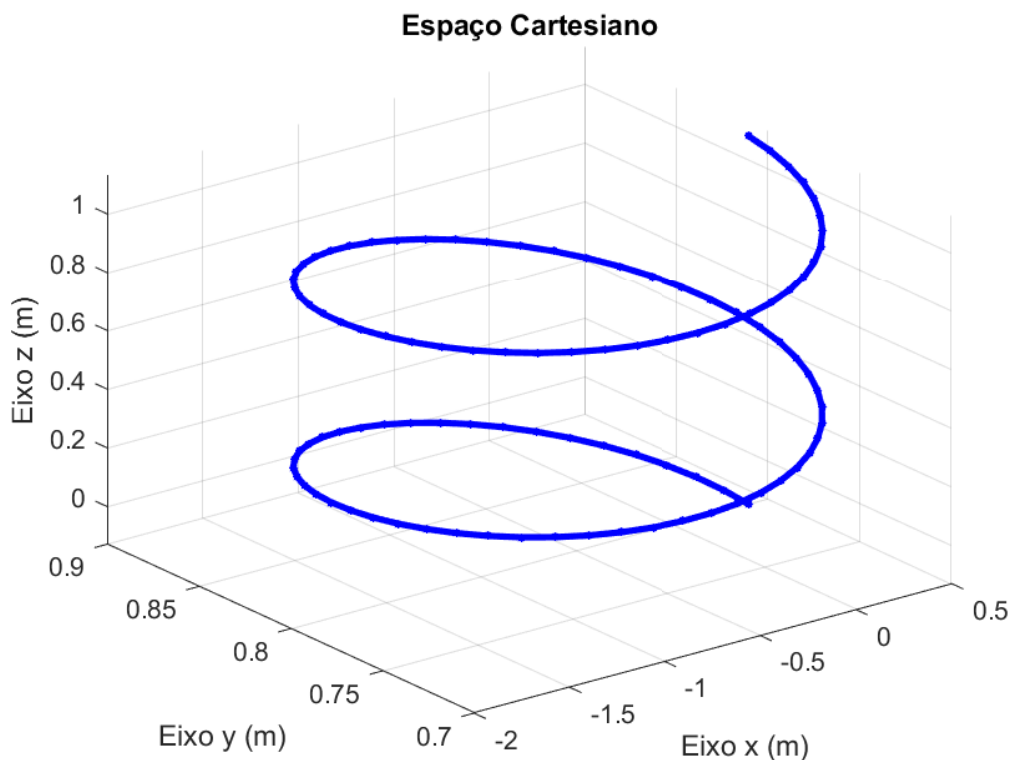
Para executar uma trajetória do manipulador foi desenvolvido um algoritmo que descreve uma trajetória helicoidal, no espaço cartesiano aplicado ao manipulador cilíndrico apresentado na Figura 62. Diante disso, foi realizado um deslocamento de cada junta do manipulador, conforme mostrada na Figura 63.

Figura 62 – Manipulador cilíndrico utilizado na tese.



Fonte: Acervo do autor.

Figura 63 – Trajetória helicoidal no espaço cartesiano.



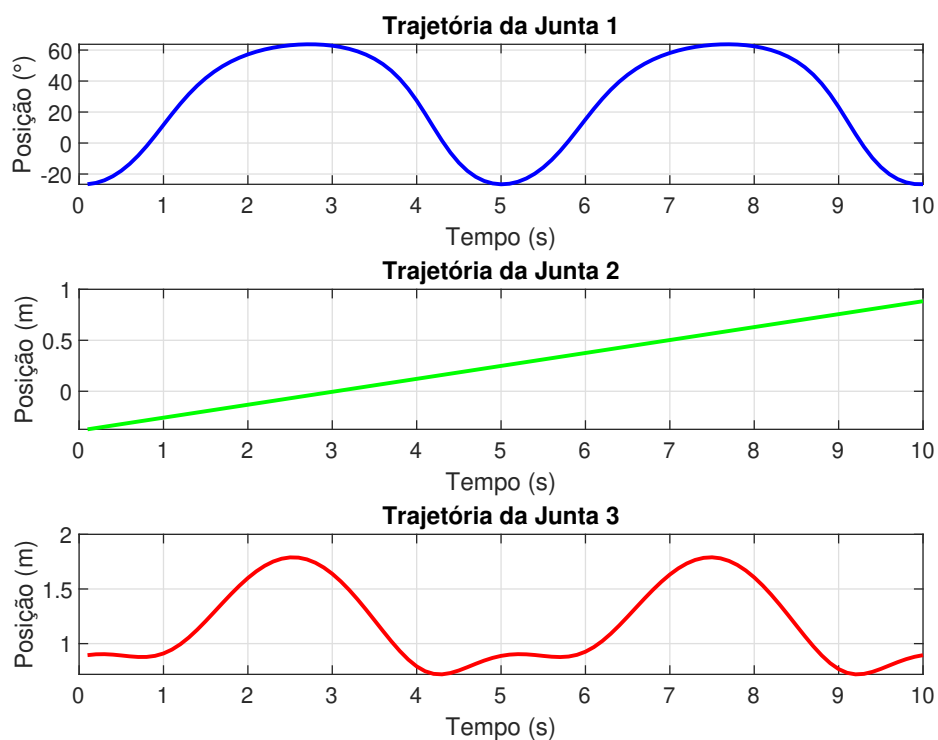
Fonte: Acervo do autor.

A partir da trajetória na Figura 63 são apresentadas as trajetórias das juntas e suas velocidades, acelerações, arranques e torques para as juntas 1, 2 e 3 do manipulador. Estas trajetórias foram obtidas no espaço das juntas a partir da solução da cinemática inversa usando-se como entrada os pontos da trajetória no espaço cartesiano. Desta forma, as trajetórias das velocidades, acelerações e arranques das juntas 1, 2 e 3 são apresentadas nas figuras 64, 65, 66 e 67, respectivamente.

Os torques das juntas foram obtidos a partir do modelo dinâmico obtidos nas equações (4.32), (4.33) e (4.34) do manipulador e são mostrados na Figura 68. Estes foram calculados com os valores das trajetórias, velocidades e acelerações apresentadas nas figuras 64, 65 e 66.

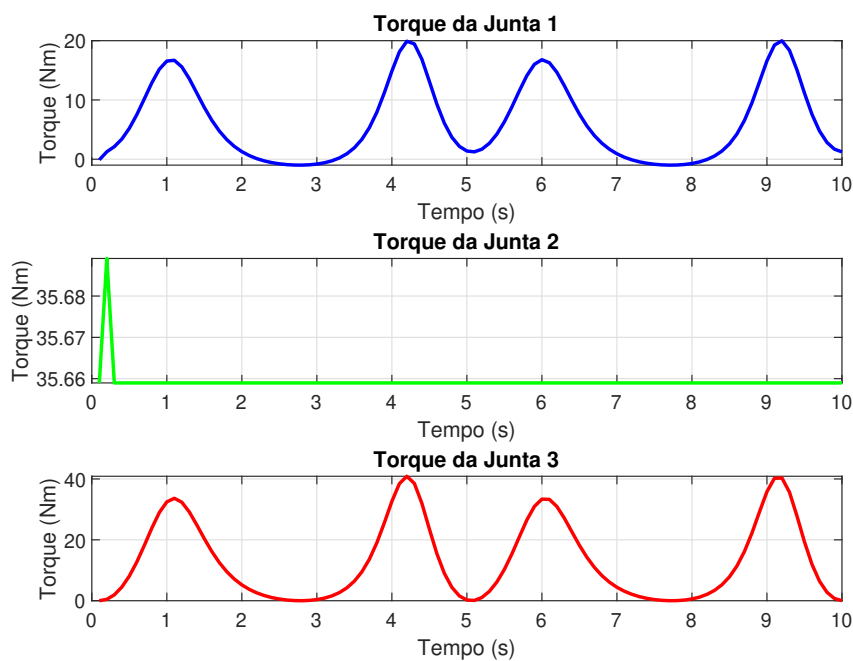
Apresenta-se também na Figura 69 os torques das juntas que foram obtidos a partir do modelo dinâmico linearizado, conforme as equações (4.35), (4.36) e (4.37).

Figura 64 – Trajetória das juntas.



Fonte: Acervo do autor.

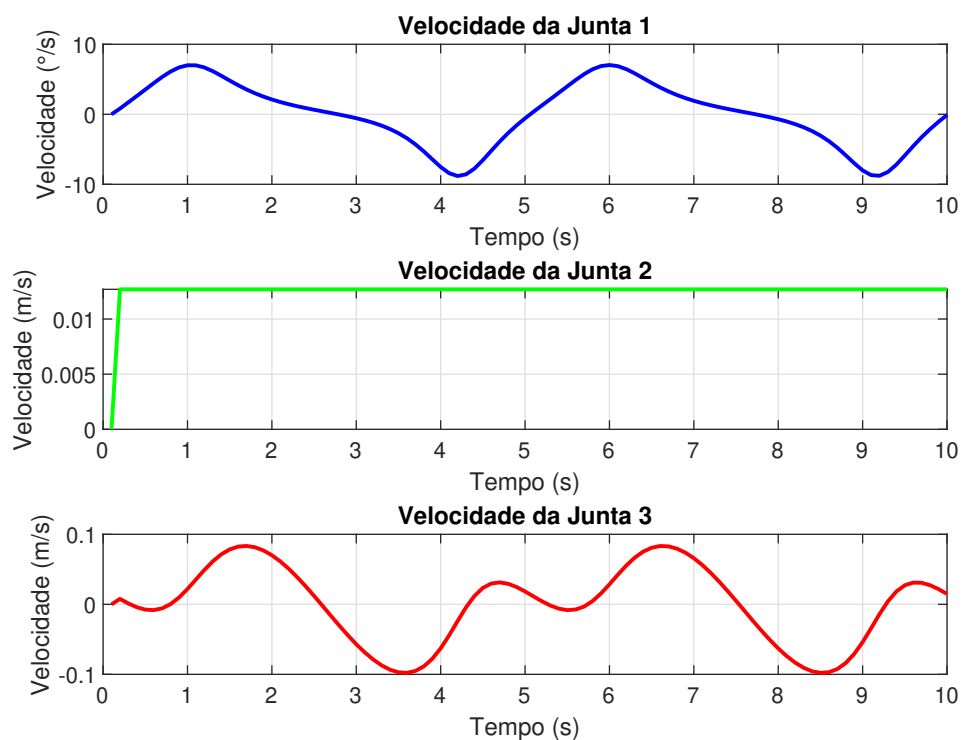
Figura 69 – Torque das juntas (modelo linearizado).



Fonte: Acervo do autor.

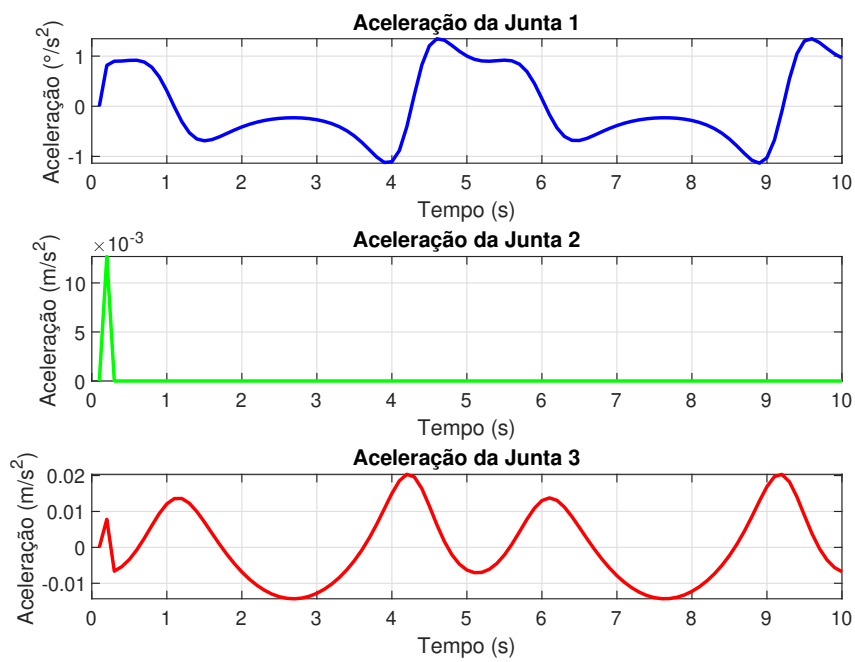
A partir do modelo da dinâmica direta apresentada no Capítulo 4, foram realizadas

Figura 65 – Velocidade das juntas.



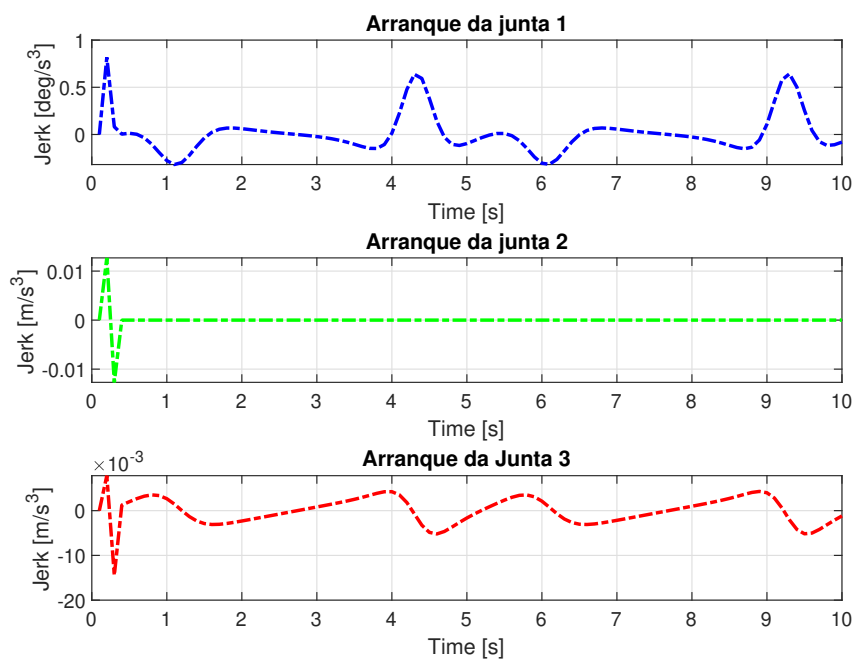
Fonte: Acervo do autor.

Figura 66 – Aceleração das juntas.



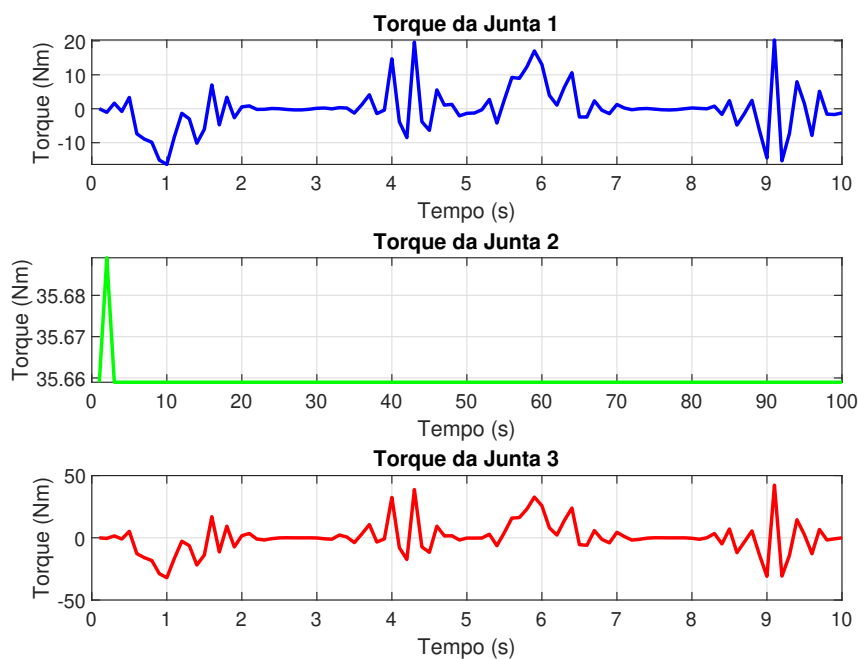
Fonte: Acervo do autor..

Figura 67 – Arranque das juntas.



Fonte: Acervo do autor.

Figura 68 – Torque das juntas.



Fonte: Acervo do autor.

simulações do manipulador para verificar que o movimento do sistema físico real seja descrito em termos das acelerações das juntas quando um conjunto de torques são aplicados ao manipulador. Assim, as velocidades e posições das juntas também podem ser obtidas integrando-se o sistema

de equações diferenciais não lineares conforme em (SICILIANO *et al.*, 2010). Com isso pode-se afirmar que o modelo dinâmico é válido para a aplicação deste trabalho e o controle do manipulador pode ser executável.

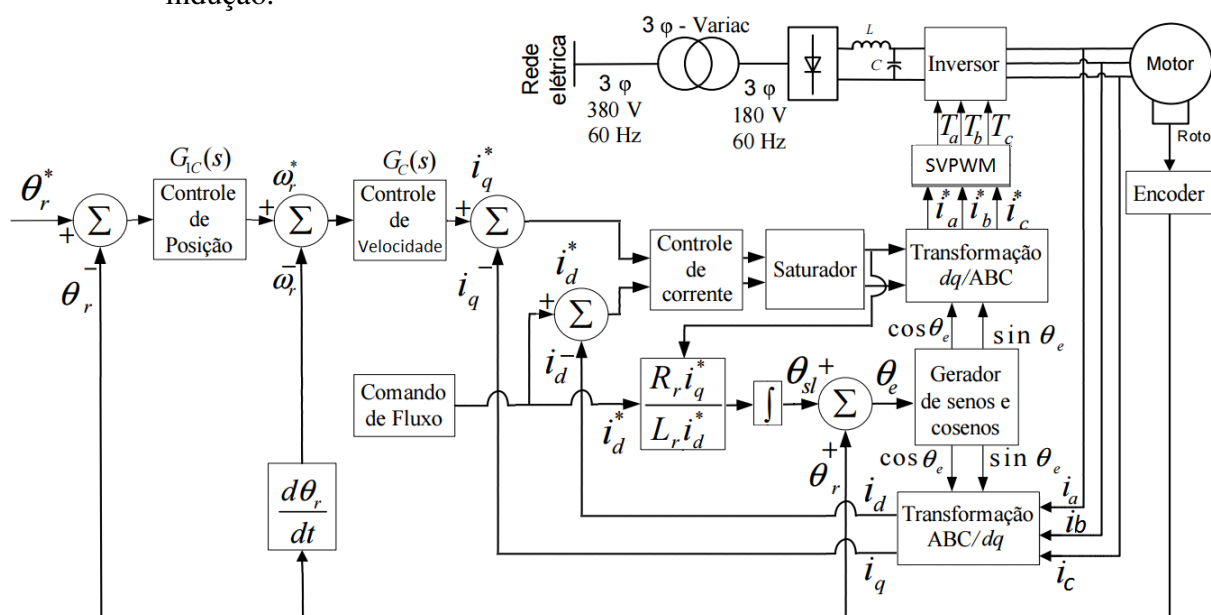
APÊNDICE B – MODELAGEM DO MOTOR DE INDUÇÃO TRIFÁSICO COMO SERVO POSICIONADOR

Como apresentado em 2.1, as juntas do manipulador são acionadas por motores de indução trifásico MIT utilizado como servo posicionador. Para se obter a modelagem do motor de indução foi aplicado a teoria de controle de campo orientado indireto, através da modelagem dinâmica do campo orientado apresentada em (RABELO L, 2017).

B.0.1 Modelagem dinâmica do campo orientado indireto para uma máquina de indução

O diagrama de blocos de controle de campo orientado FOC é mostrado na Figura 70. Este diagrama ilustra os blocos do inversor fonte de tensão de pulso (*Voltage Source Inverter* (VSI)), o modulador de espaço vetorial baseado na técnica de modulação de largura de pulso (SVPWM) e o bloco de controle com a posição de referência compõem a configuração do sistema. O modulador SVPWM converte os sinais de controle de corrente nas funções específicas de chaveamento do VSI. As correntes do motor e a posição do eixo são usadas em sua respectiva malha de realimentação e seus controladores. A descrição de cada um dos blocos do sistema da 70 está apresentada em (RABELO L, 2017).

Figura 70 – Configuração do acionamento de campo orientado indireto para uma máquina de indução.



Fonte: Adaptado de RABELO L (2017).

O modelo de espaço de estado do motor de indução no referencial dq rotativo é dado

como segue: (BOSE, 2020):

$$\frac{d}{dt}[A] = [B][C] + \frac{1}{\sigma L_s}[D], \quad (\text{B.1})$$

onde:

$$[A] = \begin{bmatrix} i_{ds} \\ i_{qs} \\ \lambda_{dr} \\ \lambda_{qr} \end{bmatrix}; \quad [C] = \begin{bmatrix} i_{ds} \\ i_{qs} \\ \lambda_{dr} \\ \lambda_{qr} \end{bmatrix}; \quad [D] = \begin{bmatrix} v_{ds} \\ v_{qs} \\ 0 \\ 0 \end{bmatrix}; \quad (\text{B.2})$$

e

$$B = \begin{bmatrix} -\frac{R_s}{\sigma L_s} - \frac{R_r(1-\sigma)}{\sigma L_r} & \omega_e & \frac{L_m R_r}{\sigma L_s L_r^2} & \frac{P \omega_r L_m}{2 \sigma L_s L_r^2} \\ \omega_2 & -\frac{R_s}{\sigma L_s} - \frac{R_r(1-\sigma)}{\sigma L_r} & -\frac{P \omega_r L_m}{2 \sigma L_s L_r^2} & \frac{L_m R_r}{\sigma L_s L_r^2} \\ \frac{L_m R_r}{L_r} & 0 & -\frac{R_r}{L_r} & \omega_e - \frac{P}{2} \omega_r \\ 0 & \frac{L_m R_r}{L_r} & -(\omega_e - \frac{P}{2} \omega_r) & -\frac{R_r}{L_r} \end{bmatrix} \quad (\text{B.3})$$

onde

$$\sigma = 1 - \frac{L_m^2}{L_s L_r}; \quad \lambda_{qr} = L_m i_{qs} + L_r i_{dr}; \quad \lambda_{dr} = L_m i_{ds} + L_r i_{qr}, \quad (\text{B.4})$$

O torque eletromagnético T_e é dado por:

$$T_e = \frac{3P L_m}{4 L_r} \left(i_{qs} \lambda_{dr} - i_{ds} \lambda_{qr} \right) \quad (\text{B.5})$$

Em um campo orientado ideal de um motor de indução, o desacoplamento ocorre entre o motor e eixos de quadratura, e o fluxo do rotor de dispersão é alinhado ao eixo direto (SOUZA, 2014). Portanto, o fluxo de dispersão e sua derivada no eixo da quadratura são zero, ou seja:

$$\lambda_{qr} = 0 \quad e \quad \frac{d\lambda_{qr}}{dt} = 0. \quad (\text{B.6})$$

O fluxo rotórico de dispersão pode ser calculado através da terceira linha da matriz equação (B.1). Usando a equação (B.6):

$$\lambda_{dr} = \frac{L_m i_{ds}}{1 + s \frac{L_r}{R_r}}. \quad (\text{B.7})$$

Fazendo-se a constante de tempo elétrica do sistema desprezível em relação à constante mecânica, a constante de tempo da equação (B.7) torna-se próxima de zero e a i_{ds} de

corrente torna-se constante ($i_{ds} = i_{ds}^*$) de modo a se ter um fluxo de rotor desejado constante. Portanto, tem-se:

$$\lambda_{dr} = L_m i_{ds}^*. \quad (\text{B.8})$$

Reescrevendo-se a equação (B.5) o torque eletromagnético é dado por:

$$T_e^* = \frac{3P L_m^2}{4 L_r} i_{qs}^* i_{ds}^*. \quad (\text{B.9})$$

A corrente i_{qs}^* indica que o controle de torque é controlado pela corrente do estator do eixo de quadratura, que é controlado pelo controlador $G_c(s)$ apresentado no diagrama da Figura 70. No método de campo orientado indireto a frequência deve ser calculada nas coordenadas $dq0$. Usando a quarta linha da equação (B.2) juntamente com a equação (B.6), a frequência de escorregamento pode ser calculada por:

$$\omega_{sl} = \frac{L_m R_r i_{qs}^*}{L_r \lambda_{dr}} = \frac{R_r i_{qs}^*}{L_r i_{ds}^*}. \quad (\text{B.10})$$

O torque T_e , a velocidade do rotor ω_r e a posição angular θ_r são relacionados por:

$$\omega_r = s \theta_r = \frac{\frac{1}{j}}{s + \frac{B}{j}} \left[T_e(s) - T_L(s) \right], \quad (\text{B.11})$$

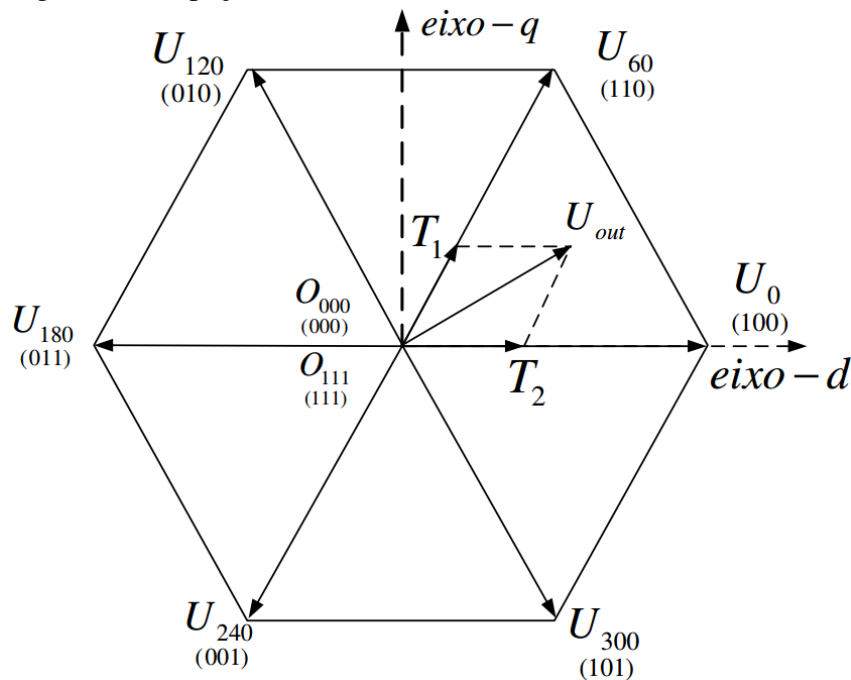
em que B é o coeficiente de atrito viscoso e J o momento de inércia. A equação (B.9) comprova o controle de campo orientado indireto, ou seja, é possível obter a posição da ligação do fluxo do rotor indiretamente através da posição do eixo da máquina e do ângulo de escorregamento entre os campos magnéticos do estator e do rotor.

B.0.2 Modulação por largura de pulso usando vetores espaciais (SVPWM)

A Figura 70 apresenta o bloco de chaveamento do inversor sendo executado pelo bloco SVPWM através da modulação de largura de pulso usando o espaço vetorial, SVPWM. Este método tornou-se uma técnica muito popular de modulação por largura de pulso (*Pulse Width Modulation* (PWM)) para aplicações em conversores corrente contínua (CC)-corrente alternada (CA) trifásicos, como o controle de máquinas de indução e motores trifásicos de ímãs permanentes. O processador digital utilizado neste trabalho, modelo TMS320F228335[®] da *Texas Instrument*[®], possui suporte às funções de modulação em espaço vetorial, facilitando, assim, o acionamento do inversor trifásico utilizado no manipulador robótico. A principal vantagem de se utilizar a técnica de acionamento por SVPWM, se deve ao fato da mínima geração de distorção harmônica nos enrolamentos da máquina de indução trifásica (SOUZA, 2014).

O objetivo do SVPWM é aproximar a tensão de referência U_{out} instantaneamente através de uma combinação de chaveamentos mapeados correspondentes aos vetores de base do espaço vetorial, de modo que cada vetor corresponda a um padrão de comutação, como mostrado na Figura 71 (YU, 2001) e (RABELO L, 2017).

Figura 71 – Diagrama do espaço vetorial.



Fonte: Adaptado de RABELO L (2017).

O ângulo entre quaisquer dois vetores adjacentes não nulos é de 60 graus. Logo o vetor U_{out} é uma combinação linear dos vetores espaciais de base, U_x e $U_{((x+60))}$ (ou $U_{((x+60))}$), que formam o setor que contém o vetor U_{out} . Os períodos T_1 e T_2 correspondem respectivamente à duração da comutação em o estado correspondente aos vetores de base. Desta forma, tem-se a expressão:

$$U_{out}(nT) = \frac{1}{T}(T_1U_x + T_2U_{x+60}) \quad (\text{B.12})$$

Como a soma de T_1 e T_2 deve ser menor ou igual ao período total T_{pwm} , o inversor deve permanecer como o período restante nos estados O_{000} ou O_{111} . A escolha destes vetores deve satisfazer o menor esforço de comutação do inversor (YU, 2001). Assim sendo:

$$T_{pwm}U_{out} = T_1U_x + T_2U_{x+60} + T_0(O_{000,111}), \quad (\text{B.13})$$

em que $T_0 = T_{pwm} - T_1 - T_2$.

Portanto, resolvendo a equação (B.13) para T_1 e T_2 os temporizadores no processador digital são configurados com os valores encontrados, atribuindo os valores aos respectivos registros do processador.