

UNIVERSIDADE ESTADUAL DO MARANHÃO
CENTRO DE CIÊNCIAS E TECNOLOGIAS - CCT
PROGRAMA DE POS-GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO E
SISTEMAS
MESTRADO PROFISSIONAL EM ENGENHARIA DE COMPUTAÇÃO E SISTEMAS
ÁREA DE AUTOMAÇÃO E CONTROLE

CRISTOVAM DERVALMAR RODRIGUES TEIXEIRA FILHO

**HIERARQUIA DE CONTROLADORES ÓTIMOS LQG/LTR NEURONAL
GENÉTICO E BARREIRA DE ROBUSTEZ VIA MODELO DE REDUÇÃO POR
FRAÇÕES PARCIAIS EM DEXTEROUS HAND MASTER**

São Luís
2017

CRISTOVAM DERVALMAR RODRIGUES TEIXEIRA FILHO

**HIERARQUIA DE CONTROLADORES ÓTIMOS LQG/LTR NEURONAL
GENÉTICO E BARREIRA DE ROBUSTEZ VIA MODELO DE REDUÇÃO POR
FRAÇÕES PARCIAIS EM DEXTEROUS HAND MASTER**

Dissertação apresentada no Programa de Pós Graduação em Engenharia de Computação e Sistemas da Universidade Estadual do Maranhão como parte dos requisitos necessários para obtenção do grau de mestre em Engenharia da Computação.

Orientador: Prof. Ivanildo Silva Abreu, PhD

São Luís
2017

Teixeira Filho, Cristovam Dervalmar Rodrigues.

Hierarquia de controladores ótimos LQG/LTR e barreira de robustez via modelo de redução por frações parciais neuronal genético em dexterous hand master / Cristovam Dervalmar Rodrigues Teixeira Filho. – São Luís, 2017.

118 f. : il.

Dissertação (Mestrado) – Programa de Pós-Graduação em Engenharia da Computação e Sistemas, Universidade Estadual do Maranhão, 2017

Orientador: Prof. Ivanildo Silva Abreu PhD

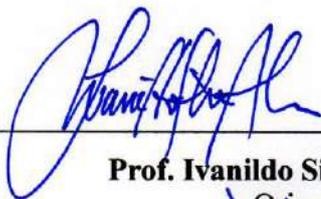
1. LQR. 2. LQG/LTR. 3. Algoritmo genético. 4. Rede neural. 5. Robustez. I. Título.

CRISTOVAM DERVALMAR RODRIGUES TEIXEIRA FILHO

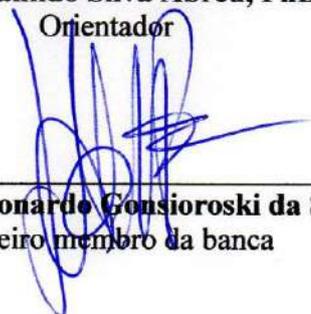
**HIERARQUIA DE CONTROLADORES ÓTIMOS LQG/LTR NEURONAL
GENÉTICO E BARREIRA DE ROBUSTEZ VIA MODELO DE REDUÇÃO POR
FRAÇÕES PARCIAIS EM DEXTEROUS HAND MASTER**

Dissertação apresentada no Programa de Pós
Graduação em Engenharia de Computação e
Sistemas da Universidade Estadual do Maranhão
como parte dos requisitos necessários para obtenção
do grau de mestre em Engenharia da Computação

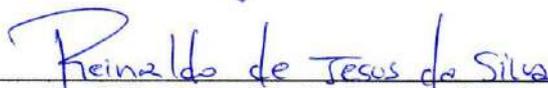
Dissertação aprovada. São Luís-Ma, *29 de Novembro* de 2017.



Prof. Ivanildo Silva Abreu, PhD
Orientador



Prof. Dr. Leonardo Gonsioroski da Silva
Primeiro membro da banca



Prof. Dr. Reinaldo de Jesus da Silva
Segundo membro da banca

"A Deus por me conduzir e fortalecer de forma tão presente em todos os momentos da minha vida."

Dedico este trabalho a meus pais, Cristovam e Therezinha Teixeira, a minha esposa Maristela e a meus três filhos Cristovam Neto, Rafael e Thiago, que sempre estão ao meu lado apoiando e incentivando o meu desenvolvimento pessoal e profissional.

AGRADECIMENTOS

Ao meu bom Deus, a quem busco, em todas as horas, e encontro respostas. Por meio de seus ensinamentos, encontrei forças para continuar a caminhada em busca de me tornar, a cada dia, uma pessoa melhor. Obrigada, meu Senhor!

Ao meu orientador Prof. Ivanildo Silva Abreu, pelo incentivo e aprendizado durante esse período do curso.

Aos meus pais Cristovam e Therezinha Teixeira, a minha esposa Maristela, a meus filhos, Cristovam Neto, Rafael e Thiago, motivos de minha maior inspiração e a quem incansavelmente tento mostrar que o caminho do conhecimento é o que nos leva à verdadeira liberdade e à conquista da própria existência. Obrigada por fazerem parte da minha vida.

A toda minha família, por compreenderem minha ausência ou a pouca participação nos encontros familiares nos últimos tempos.

Aos professores do programa de Pós-graduação da UEMA, que sempre me atenderam com atenção e profissionalismo. Prof. Henrique Mariano, Prof^a Patrícia Helena, Prof. Reinaldo de Jesus, Prof. Rogério Moreira Lima, ao diretor do CCT Prof. Jorge Passinho e todos que participaram do programa de pós-graduação.

Aos meus colegas de turma, Rildenir, Guilherme, Marta, Marco Selner, e todos que sempre me apoiaram e incentivaram ao longo desse período do curso. Obrigado pelos momentos de estudo e convivência saudável com bom humor e alegria.

As secretarias da PPG Sara e Karol, que sempre nos atenderam com atenção e muita presteza, obrigado.

A meu amigo da Caema Engenheiro Ignácio de Oliveira, que me incentivou e apoiou nos momentos mais difíceis para realizar e concluir esse curso.

A todos que de forma direta ou indireta me ajudaram durante esse período. Obrigado.

RESUMO

Esta dissertação descreve e analisa o projeto de controladores LQG/LTR ótimos, bem como as barreiras de robustez referente à estabilidade e desempenho usando abordagens de inteligência computacional IC, em particular, algoritmo genético e redes neuronais na planta Dexterous Hand Master, DHM. O desenvolvimento se fundamenta em um sistema na forma desacoplada e formado por um conjunto de equações diferenciais lineares e invariante no tempo. Assume-se que o sistema é controlável e observável. O projeto do controlador e observador estão direcionados para o Regulador Linear Quadrático, LQR e Regulador Linear Gaussiano, LQG que tem como objetivo resolver a equação algébrica de Riccati, EAR. Para sua solução, utilizou-se uma rede neural artificial, RNA, de modo que seus parâmetros matriciais são obtidos por um algoritmo genético, AG. Para avaliar o desempenho do AG e da RNA, exibem-se gráficos referente à superfície da energia em função da solução da EAR. A estimação dos estados é feita pela metodologia do filtro de Kalman, FK, enquanto que a recuperação das propriedades de robustez que foram perdidas com a inserção de um estimador estocástico é feita pelo projeto de LTR. A recuperação do ganho de malha é feita pela entrada, sendo fixado o ganho do projeto LQR e em seguida projeta-se os ganhos do FK, uma vez que a sintonia de um parâmetro variável v tende para zero é realizada por um AG. O controlador LQG/LTR realiza uma recuperação assintótica das propriedades de robustez, sendo que as propriedades de estabilidade e desempenho estão garantidas.

Palavras-chave: LQR, LQG/LTR, Algoritmo Genético, Rede neural e Robustez

ABSTRACT

This dissertation aims to describe and analyze the control design LQG/LTR, as well as the robustness of barriers related to stability and performance using computational intelligence approaches IC, in particular, genetic algorithms and neural networks in Dexterous Hand Master plant, DHM . The development is based on a system in uncoupled form and formed by a set of linear differential equations and time-invariant. It is assumed that the system is controllable and observable. The controller design and observer are directed to the Linear Quadratic Regulator, LQR Linear Quadratic Regulator and, LQG which aims to solve the algebraic equation of Riccati, EAR. To a solution was used an artificial neural network ANN, so that its matrix parameters are obtained by a genetic algorithm, AG. To evaluate the performance of AG and RNA-display graphics are related to the surface energy as a function of the RAS solution. The estimation of the states is made by the Kalman filter methodology, FK, while the recovery of robustness properties that were lost with the inclusion of a stochastic estimator is made by the LTR project. The recovery loop gain is made by the entry, and set the gain of the LQR design and then projected to the FK gains, since the line of a variable parameter v tends to zero is performed by a AG. The LQG / LTR controller performs an asymptotic recovery of strength properties, and the stability and performance properties are guaranteed.

Keywords: LQR, LQG / LTR, Genetic Algorithm, Neural Network and Robustness

LISTA DE TABELAS

| | | |
|-------------|--|----|
| Tabela 01 - | Parâmetros duais do LQR e FK..... | 58 |
| Tabela 02 - | Pop. final AG-sensibilidades e função objetivo dos indivíduos viáveis... | 84 |
| Tabela 03 - | Traços das Matrizes de ponderação..... | 84 |

LISTA DE ILUSTRAÇÕES

| | | |
|-----------|---|----|
| Figura 01 | Os sensores Dexterous Hand Máster – DHM..... | 21 |
| Figura 02 | Diagrama do Filtro de Kalman..... | 28 |
| Figura 03 | Estrutura do controlador LQG..... | 30 |
| Figura 04 | LQG com observador de estado..... | 31 |
| Figura 05 | Estrutura básica de um Algoritmo Genético..... | 35 |
| Figura 06 | Modelo de um neurônio Biológico..... | 39 |
| Figura 07 | Modelo Artificial de Neurônio Biológico..... | 40 |
| Figura 08 | Rede denominada de feedforward..... | 41 |
| Figura 09 | Rede denominada de feedforward múltiplas camadas..... | 42 |
| Figura 10 | Modelo de rede Recorrente..... | 43 |
| Figura 11 | Arquitetura gráfica de um perceptron multicamada com duas camadas intermediárias..... | 44 |
| Figura 12 | Ilustração do algoritmo Backpropagation..... | 45 |
| Figura 13 | Arquitetura do rede neural artificial para solução da equação algébrica de Riccati..... | 51 |
| Figura 14 | Síntese do diagrama de bloco do processo dinâmico para o Regulador Linear Quadrático..... | 53 |
| Figura 15 | Diagrama de blocos do sistema Dinâmico Estocástico..... | 56 |
| Figura 16 | Diagrama em Blocos do filtro de Kalman..... | 57 |
| Figura 17 | Esquema para realimentação de estados..... | 59 |
| Figura 18 | Diagrama de blocos do sistema com perturbações..... | 66 |
| Figura 19 | Barreiras de robustez debaixo e alta frequências..... | 73 |
| Figura 20 | Diagrama de blocos do sistema DHM..... | 77 |
| Figura 21 | Sensibilidade total de cada indivíduo..... | 79 |
| Figura 22 | Sensibilidade 1 da população inicial..... | 79 |
| Figura 23 | Sensibilidade 2 da população inicial..... | 80 |
| Figura 24 | Sensibilidade 3 da população inicial..... | 80 |
| Figura 25 | Evolução do processo - função objetivo média e melhor função objetivo das gerações..... | 81 |
| Figura 26 | Fitness normalizada de cada geração e de cada indivíduo..... | 83 |
| Figura 27 | Sensibilidade 1 da Pop.final de cada geração e de cada indivíduo..... | 83 |

| | | |
|-----------|--|----|
| Figura 28 | Sensibilidade 2 da Pop.final de cada geração e de cada indivíduo..... | 84 |
| Figura 29 | Sensibilidade 3 da Pop.final de cada geração e de cada indivíduo..... | 84 |
| Figura 30 | Superfície da função de energia para $\eta_v=1000-10000$ e $\eta_z=1-10$ | 85 |
| Figura 31 | Superfície norma do infinito para $\eta_v=1000-10000$ e $\eta_z=1-10$ | 86 |
| Figura 32 | Estimação dos estados e os erros..... | 87 |
| Figura 33 | Valores singulares da recuperação..... | 88 |
| Figura 34 | Barreira de desempenho nominal..... | 89 |
| Figura 35 | Gráfico de Magnitude de bode dos valores singulares planta real..... | 90 |
| Figura 36 | Gráfico de Magnitude de bode dos valores singulares aproximação ordem reduzida..... | 91 |
| Figura 37 | Estabilidade Robusta em alta frequência do erro multiplicativo $m(w)$ | 92 |
| Figura 38 | Estabilidade Robusta em alta frequência com inverso do erro multiplicativo $1/m(w)$ | 92 |

LISTA DE ABREVIATURAS E SIGLAS

| | |
|-----------|--|
| AE - | Autoestrutura |
| AAE - | Alocação de Autoestrutura |
| AG - | Algoritmo Genético |
| DHM - | Dexterous Hand Master (Mão mestre) |
| EAR - | Equação Algébrica de Riccati |
| FK - | Filtro de Kalman |
| K - | Ganho do Regulador Linear Quadrático |
| L - | Ganho do Filtro de Kalman |
| LQG - | Linear Quadratic Gaussian (Gaussiano Linear Quadrático) |
| LQR - | Linear Quadratic Regulator (Regulador Linear Quadrático) |
| LTR - | Loop Transfer Recovery (Recuperação da Malha de Transferência) |
| LQG/LTR - | Linear Quadratic Gaussian /Loop Transfer Recovery |
| MV - | Minimum Variance (Variância Mínima) |
| MIMO - | Multiple-Input and Multiple-Output (Múltiplas-Entradas e Múltiplas-Saídas) |
| SISO - | Single-Input and Single-Output (Simples-Entrada e Simples-Saída) |
| P - | Solução da Equação Algébrica de Riccati |
| S - | Sensibilidade |
| T - | Sensibilidade Complementar |
| ξ - | Solução da Equação Algébrica de Riccati para o Filtro de Kalman |
| Q - | Matriz de Ponderação do Estado |
| R - | Matriz de Ponderação do Controle |

LISTA DE SÍMBOLOS

A - Matriz de ordem $n \times n$

B - Matriz de ordem $n \times 1$

$crom$ - Número de cromossomos

$H(t)$ Hamiltoniano

J Funcional

P_c Matriz de controlabilidade $n \times m$

P_o Matriz de observabilidade $n \times n$

C - Vetor linha $1 \times n$

R_n Conjunto do vetores reais de dimensão n

R_m Conjunto do vetores reais de dimensão m

\vec{x} Vetor coluna $n \times 1$

Ξ Matriz de Covariância da Perturbação no Estado

Θ Matriz de Covariância do Ruído de Medida

$\delta(t - \tau)$ Delta de Dirac

$\bar{\sigma}(G)$ Valor singular máximo

$\underline{\sigma}(G)$ Valor singular mínimo

SUMÁRIO

| | | |
|---------|---|----|
| 1 | INTRODUÇÃO | 17 |
| 1.1 | Objetivo Geral | 18 |
| 1.2 | Objetivos específicos | 18 |
| 1.3 | Justificativa e Motivação | 19 |
| 1.4 | Organização da Dissertação | 20 |
| 1.5 | Produção científica | 21 |
| 2 | DEXTEROUS HAND MASTER-DHM (MÃO MESTRE) | 22 |
| 3 | TEORIA DO CONTROLE ÓTIMO CLÁSSICO - REFERENCIAL TEÓRICO | 23 |
| 3.1 | Regulador Linear Quadrático - LQR | 23 |
| 3.1.1 | Modelagem Clássica do Projeto LQR..... | 24 |
| 3.1.2 | O problema do LQR..... | 25 |
| 3.1.3 | Seleção das matrizes de ponderação Q e R..... | 26 |
| 3.2 | Metodologia do Regulador Linear Gaussiano- LQG | 27 |
| 3.2.1 | O problema LQG..... | 27 |
| 3.2.2 | Solução do Problema do LQG via Filtro de Kalman..... | 28 |
| 3.2.3 | Princípio de separação para o projeto de LQG..... | 29 |
| 3.2.4 | Controlador LQG..... | 30 |
| 3.3 | Metodologia LQG/LTR | 31 |
| 4 | MODELAGEM INTELIGENTE VIA ALGORITMO GENÉTICO E REDE NEURAL | 34 |
| 4.1 | Modelagem Inteligente | 34 |
| 4.2 | Algoritmo Genético | 34 |
| 4.3 | Rede Neural | 40 |
| 4.3.1 | Neurônios Biológicos | 41 |
| 4.3.2 | Modelo Artificial de um Neurônio..... | 41 |
| 4.3.3 | Arquitetura da Rede Neural..... | 42 |
| 4.3.3.1 | Rede Feedforward com Uma Única Camada..... | 43 |
| 4.3.3.2 | Rede Feedforward Múltiplas Camadas..... | 43 |
| 4.3.3.3 | Redes Recorrentes..... | 44 |
| 4.3.3.4 | Rede Perceptron de Múltiplas camadas Recorrente..... | 45 |

| | | |
|------------|---|-----------|
| 4.3.4 | Treinamento | 45 |
| 5 | REDE NEURAL APLICADA A CONTROLADORES ÓTIMOS | 47 |
| 5.1 | Controlador Inteligente LQR..... | 48 |
| 5.2 | Solução da Equação Algébrica de Ricatti via Rede Neuronal..... | 49 |
| 5.3 | Formulação do problema..... | 51 |
| 5.3.1 | Arquitetura da rede..... | 52 |
| 5.3.2 | Dinâmica do Diagrama de Blocos..... | 53 |
| 5.3.3 | Análise de Estabilidade e Solvabilidade..... | 54 |
| 5.4 | Filtro de Kalman para estimação dos estados..... | 55 |
| 5.4.1 | Dualidade entre a Estimação e o Controle..... | 58 |
| 5.5 | Controlador LQG/LTR..... | 59 |
| 5.5.1 | Modelo de ajuste de ganho <i>LTR</i> | 60 |
| 5.5.2 | Modelo de Busca <i>LTR</i> | 61 |
| 5.5.3 | Desempenho <i>AG – LQG / LTR</i> | 62 |
| 6 | MODELO HIERÁRQUICO PARA SINTONIA | 64 |
| 6.1 | Modelo de Sintonia do Ganho <i>LQR</i>..... | 65 |
| 6.2 | Modelo de Sintonia do Ganho <i>LQG</i>..... | 65 |
| 6.3 | Modelo de Sintonia do Ganho <i>LTR</i>..... | 65 |
| 7 | SISTEMA DE CONTROLE ROBUSTO | 67 |
| 7.1 | Sensibilidade do sistema..... | 67 |
| 7.2 | Matriz de sensibilidade <i>S</i> e matriz de sensibilidade complementar <i>T</i>:..... | 68 |
| 7.2.1 | Rejeição de perturbações..... | 72 |
| 7.2.2 | Rejeição de erro de medida..... | 72 |
| 7.3 | Análise de Robustez..... | 73 |
| 7.3.1 | Perturbação aditiva..... | 73 |
| 7.3.2 | Perturbação multiplicativa..... | 74 |
| 7.4 | Barreira de robustez..... | 74 |
| 7.5 | Modelo de redução e estabilidade Robusta..... | 75 |
| 7.5.1 | Redução do Modelo por Expansão de Fração Parcial..... | 75 |
| 8 | APLICAÇÃO NEURO-GENÉTICA NA PLANTA DEXTEROUS HAND MASTER – DHM (MÃO MESTRE)..... | 79 |
| 8.1 | Função de transferência e equações de estado..... | 79 |
| 8.2 | Desempenho da população inicial..... | 80 |

| | | |
|------------|--|------------|
| 8.3 | Desempenho da população final..... | 83 |
| 8.4 | Solução da EAR via Rede Neural Recorrente Não Supervisionada..... | 87 |
| 8.5 | Análise do filtro de Kalman..... | 88 |
| 8.6 | Recuperação das propriedades de robustez..... | 90 |
| 8.7 | Barreira de robustez..... | 91 |
| 8.8 | Redução e estabilidade robusta..... | 92 |
| 9 | CONCLUSÃO..... | 96 |
| 9.1 | Trabalhos futuros..... | 96 |
| | REFERÊNCIAS..... | 97 |
| | APÊNDICES..... | 100 |
| A | Controle Ótimo..... | |
| B | Controlabilidade e Observabilidade..... | |
| C | Decomposição de Valor Singular..... | |
| D | Desempenho e Estabilidade Robusta..... | |
| E | Recuperação de malha de Transferência-RFTM..... | |

1 INTRODUÇÃO

Os conceitos de controle robusto para sistemas lineares e invariantes no tempo (LIT), para estimação de alguns parâmetros de estados, recuperação da malha de transferência e as barreiras de desempenho e estabilidade robustas utilizando-se o diagrama de Bode multivariável na metodologia de controle *LQG/LTR*, segundo, (Doyle, 1993) é um dos tópicos de maior atenção da comunidade científica em sistemas de malha fechada. O desenvolvimento desta pesquisa é realizado nos domínios do tempo e da frequência.

A robustez é uma característica desejável dos sistemas de controle por pelo menos duas razões. A primeira diz respeito a preocupação permanente do projetista que os sistemas de controle funcionem de forma satisfatória, ainda que as condições de operação sejam distintas daquelas consideradas no modelo de projeto (nominal) e em segundo lugar, as condições de robustez devem ser utilizadas com o objetivo de se adotar um modelo de projeto intencionalmente simplificado para facilitar a análise, bem como seu impacto sobre a complexidade do controlador resultante.

As ferramentas empregadas para o desenvolvimento do projeto do controlador robusto *LQG/LTR*, são o projeto do Regulador Linear Quadrático (*LQR*) e o Filtro de Kalman (*FK*) que justifica o projeto *LQG* (Linear Quadrático Gaussiano). Estas ferramentas não serão interpretadas de forma usual em controle ótimo estocástico, mas utilizadas em função de suas propriedades. Neste trabalho, apresenta-se uma metodologia de projeto que permite a determinação de um compensador, a ser localizado no ramo direto da malha de controle, de modo que sejam atendidos os requisitos de desempenho e estabilidade

Esta pesquisa está direcionada para recuperação da malha de transferência e análise do desempenho e estabilidade robusta no projeto de mãos robóticas capazes de segurar e manipular objetos que exige um enorme esforço computacional intitulado, Dexterous Hand Master (DHM).

A planta é linear e invariante no tempo e modelada numa representação em espaço de estado. Sendo o sistema controlável todos os autovalores (polos) podem ser deslocados por realimentação de estado para posições convenientes.

Na prática, contudo, há limitações de ordem física, pois para que o sistema tenha transitório mais rápido, por exemplo, é necessário dispor de maior potência de acionamento, o que pode não ser viável. A matriz do sistema é diagonalizável, e a matriz de observabilidade possui posto cheio e determinante não nulo, desta forma a reconstituição das variáveis de

estado é possível. Assim, todas as variáveis de estado podem ser determinadas através das variáveis de saída e de entrada do DHM

O procedimento de recuperação das propriedades de robustez, perdidas com a estimação das variáveis de estados é realizada pelo procedimento *LTR* (Recuperação da Malha de Transferência), uma vez que o Dexterous Hand Master, possui zeros de transmissão no semi plano esquerdo (zeros de fase mínima), (Doyle, 1982).

Diante disso, propõe-se neste trabalho analisar os assuntos discutidos anteriormente e mostrar como recuperar a robustez do sistema aplicando a técnica de hierarquia de controladores ótimos LQG/LTR na planta DHM utilizando algoritmo genético e rede neuronal.

1.1 Objetivo Geral

Esta dissertação tem como objetivo aplicar uma metodologia para mostrar a recuperação das propriedades de robustez do sistema de controle da planta Dexterous Hand Master(DHM) utilizando a técnica de hierarquia de controladores ótimos, através de uma modelagem inteligente via algoritmo genético e rede neuronal.

1.2 Objetivos Específicos

- a) Levantar o estado da arte em relação à análise e projetos de controle moderno e robusto via modelo Neuro - Genético;
- b) Transferir o modelo da DHM para termos de variáveis de estado;
- c) Obter o ganho do controlador ótimo através do projeto LQR, selecionando as matrizes de ponderação Q e R através de algoritmo genético, que garanta margem de ganho infinita e fase de pelo menos 60° ;
- d) Encontrar as matrizes de ponderação do estado e da entrada Q e R respectivamente através de algoritmo genético, para solução da equação algébrica de Riccati (EAR) via rede neuronal;
- e) Analisar o projeto do LQG e os impactos da inserção do filtro de Kalman (FK) na malha através da estimação estocástica via algoritmo genético (AG) para busca das matrizes de covariâncias da perturbação do estado e do ruído de medida;

- f) Executar a técnica LTR no projeto LQR e LQG, a fim de recuperar, as propriedades de robustez do sistema modelando como uma estrutura de otimização combinatória;
- g) Utilizar a redução de ordem de um sistema por frações parciais para incertezas multiplicativas ;
- h) Mostrar graficamente o desempenho do AG-LQG/LTR através da barreira de robustez, estabilidade e Análise dos resultados obtidos.

1.3 Justificativa e Motivação

As ferramentas empregadas para o desenvolvimento do projeto de controladores ótimos LQG/LTR e barreira de robustez via modelo neuro-genético em DHM será empregada usando abordagens de inteligência artificial, em particular algoritmo genético e redes neurais utilizando Regulador Linear Quadrático (LQR) e Filtro de Kalman onde serão selecionadas as matrizes de ponderação Q e R via Inteligência Artificial (IA), Resolver a Equação Algébrica de Riccati para obtenção dos pesos ótimos da Rede Neural e índices de desempenho.

A estimação dos estados será feita pela metodologia do filtro de Kalman, FK, enquanto que a recuperação das propriedades de robustez que foram perdidas com a inserção de um estimador estocástico será feita pelo projeto de LTR. A recuperação do ganho de malha é feita pela entrada, sendo fixado o ganho do projeto LQR e em seguida projetam-se os ganhos do FK, uma vez que a sintonia de um parâmetro variável v tende para zero é realizada por um AG. O controlador LQG/LTR realiza uma recuperação assintótica das propriedades de robustez, sendo que as propriedades de estabilidade e desempenho estão garantidas.

O método *LQG/LTR*, em que se trabalha com sistemas dinâmicos estocásticos e levando em consideração ruídos e perturbações na planta sendo recuperadas as propriedades de robustez da malha através da técnica LTR, justifica-se o desenvolvimento desta pesquisa pela necessidade de encontrar soluções ótimas para alocação de autoestruturas hierarquizadas via modelo Neuro-Genético podendo apresentar alto índices de desempenho de robustez.

O que nos leva a realizar esta dissertação é que não encontramos nenhum trabalho que tratasse da redução de ordem por frações parciais para manter a estabilidade, face as incertezas multiplicativas uma vez que as mesmas são mais complexas para serem modeladas.

1.4 Organização da Dissertação

O procedimento realizado para atender o objetivo desta dissertação está organizado em capítulos e apêndices. No capítulo 1, foi descrito um resumo da aplicação de controladores na planta do DHM e a utilização de Algoritmo genético e rede neuronal para solução ótima no sistema robusto.

No capítulo 2, abordam-se explicações técnicas sobre o DHM.

No capítulo 3 é apresentada uma metodologia para o projeto de controladores ótimos LQR , LQG , LGQ/LTR e o método para encontrar as matrizes de ponderação Q e R para solução da equação de Riccati, estimação dos estados via filtro de Kalman, bem como a recuperação das propriedades de robustez pelo LTR por tentativa e erro.

No capítulo 4, faz-se uma abordagem sobre Modelagem Inteligente via Algoritmo Genético e Rede Neural mostrando os principais conceitos do AG, arquitetura da rede neural e seu funcionamento.

No capítulo 5, mostra-se um controlador LQR inteligente para encontrar as matrizes de ponderação de estado e de controle Q e R respectivamente e a arquitetura da rede neuronal para solução da EAR, aborda-se o Filtro de Kalman para a estimação dos estados através da alocação de autovalores e autovetores, na obtenção das matrizes de covariância. E faz-se também a relação de dualidade do FK e o LQR . Apresenta-se o modelo de busca do LTR para recuperação de malha e desempenho do AG comparando-se os valores singulares do projeto LQR e do projeto LTR .

No capítulo 6, define-se o Modelo Hierárquico para sintonia apresenta-se os modelos de busca que são utilizados para sintonizar os ganhos das malhas LQR , LQG e LTR .

No capítulo 7, explica-se o sistema de controle robusto, a influência da sensibilidade e cosensibilidade nas equações de entrada, saída e erro. Define-se os requisitos de acompanhamento do sinal de referência, rejeição de perturbações e insensibilidade a variações na planta. Barreira de robustez e o modelo de redução e estabilidade por frações parciais.

No capítulo 8, aplica-se às técnicas Neuro-Genética na planta DHM, descritas nesta dissertação, apresentando e analisando os resultados obtidos pelo AG e da rede neural através de gráficos e tabelas.

Finalmente, no Capítulo 9, mostra-se conclusão, comentários, contribuições e perspectivas futuras. Alguns conceitos matemáticos mais detalhados, utilizados nesta dissertação, encontram-se no apêndice.

1.5 Produção Científica

- CONEM-2016 - CON-2016-0993 - **UMA HIERARQUIA DE CONTROLADORES ÓTIMOS LQG/LTR E BARREIRA DE ROBUSTEZ VIA MODELO NEUROGENÉTICO EM DEXTEROUS HAND MASTER-DHM** (16/07/2016). O ISSN do CONEM é 2178180 OX;
- COBEM-2015 - doi://10.20906/CPS/COB-2015-0799 - **ROBUSTNESS BARRIERS IN LQG/LTR CONTROLLER VIA HYBRID MODEL GENETIC-NEURAL IN ROBOTIC MANIPULATOR** (19/10/2015);
- Journal of the Brazilian Society of Mechanical Sciences and Engineering - BMSED-16-01287 - **ROBUSTNESS BARRIERS IN LQG/LTR CONTROLLER VIA HYBRID MODEL GENETIC-NEURAL IN ROBOTIC MANIPULATOR** (05/12/2016);
- COBEM-2017 **A NEURAL GENETIC MODEL FOR EIGENSTRUCTURE ALLOCATION IN THE LQR PROJECT IN DFIG ABCM – 2017;**
- Artigo submetido para revista Optimal Control Applications and Methods **SÍNTESE DE CONTROLADORES ÓTIMOS INTELIGENTES, LQG/LTR E ESTABILIDADE ROBUSTA VIA MODELO DE REDUÇÃO EM DEXTEROUS HAND MASTER.**

2 DEXTEROUS HAND MASTER – DHM OU MÃO MESTRE

Dexterous Hand Master (DHM) é um dos dispositivos de controle exoesqueleto usado para medir as posições da mão de um operador e utilizar essas posições para controlar o movimento de um manipulador remoto ou escravo. Um modelo cinemático simplificado de uma mão humana foi desenvolvido para atender a capacidade de uma luva DHM proposta. Somente os movimentos de flexão / extensão para as articulações dos dedos foram levados em consideração no modelo projetado. Muitos laboratórios de universidades e governamentais construíram mãos robóticas capazes de segurar e manipular objetos, mas ensinar os dispositivos artificiais a realizarem até mesmo tarefas simples requer tremenda programação computacional. Agora, porém o DHM pode ser vestido em uma mão humana para registrar os movimentos laterais e de flexão das articulações dos dedos. Cada articulação é provida com um sensor que muda seu sinal dependendo da posição. Os sinais de todos os sensores são convertidos em dados de computador e usados para operar mãos robóticas (DORF; BISHOP, 2013). A planta é linear e invariante no tempo.

O mesmo pode ser envolvido por uma mão humana para registrar os movimentos laterais e de flexão das articulações dos dedos provida com um sensor que muda seu sinal angular. Os sinais são convertidos em dados e usados para operar mãos robóticas (figura 1).

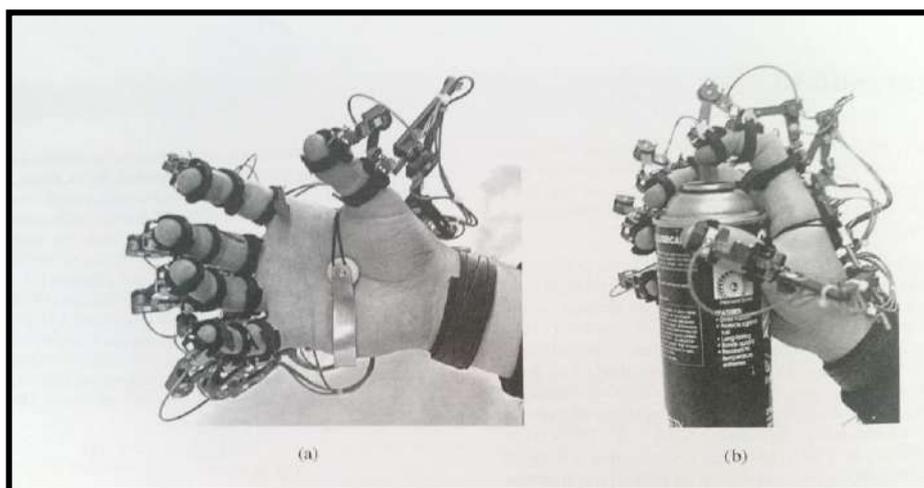


Figura 1- Os sensores Dexterous Hand Máster – DHM Fonte Dorf, 2013

O dispositivo mestre da DHM é usado na mão do operador de forma precisa e mede as posições com os dedos e as utiliza para controlar o movimento de um manipulador remoto capaz de atuar eficientemente executando as tarefas requeridas de forma ótima.

3 TEORIA DO CONTROLE ÓTIMO CLÁSSICO - REFERENCIAL TEÓRICO

No controle ótimo as técnicas de otimização são de extrema importância, pois as mesmas são usadas para se determinar as condições necessárias ou suficientes que possam garantir a existência de uma solução ótima para o controle se ela existir. A compreensão do comportamento das dinâmicas dos sistemas é uma das etapas fundamentais, pois é necessário modelar o processo de modo a se construir a função objetivo que será otimizada e quais as possíveis restrições relacionadas ao problema.

Para se realizar o controle é necessário o conhecimento das leis que regem o sistema, deste modo uma das etapas seria a modelagem matemática, pois, um modelo matemático que descreva a dinâmica do sistema torna possível a aplicação do controle de modo mais eficiente.

3.1 Regulador Linear Quadrático - *LQR*

A metodologia de controle *LQR* pode ser visualizada como uma aplicação particular de um problema de otimização. Este problema de otimização é caracterizado pela busca de vetor de entrada u que minimiza um funcional $J(\cdot)$ e tem como restrição o modelo do sistema descrito no espaço de estados,

$$\min_u J(x, Q, u, R) \quad (3.1)$$

s.a

$$\dot{x} = Ax + Bu \quad (3.2)$$

Esta formulação é chamada de problema de controle ótimo. No caso do Regulador Linear Quadrático, o índice de desempenho J é um mapeamento dos espaços dos vetores de estado e de controle que são ponderados pelas matrizes constantes Q e R , respectivamente.

Nas metodologias de projeto de controle, existe uma série de requisitos que representam uma série de questões desejáveis e que nos dão mais informação a cerca do processo de controle e resposta dinâmica do sistema, tais como: Tempo de subida, tempo de resposta, tempo de acomodação, ganho de margem e ganho de fase e largura de banda, são ferramentas que auxiliam no processo de controle e fornecem várias informações sobre o comportamento do processo e como o controle está atuando.

Métodos clássicos são usados e foram formulados com o propósito de solucionar problemas de controle de sistemas lineares com uma entrada e uma saída. Para o controle de sistemas mais complexos, não lineares e com múltiplas entradas e saídas às metodologias clássicas não são adequadas. A teoria de controle ótimo tem sido usada para o controle de sistemas mais complexos lineares, não lineares e multivariáveis.

A aplicação das técnicas de controle ótimo necessita de uma clara descrição dos índices de desempenho a serem alcançados pelo sistema, de um modelo matemático bem descrito que é representado frequentemente por equações diferenciais ordinárias, da compreensão e determinação das restrições do sistema e do projeto de controle completo. Na prática, a dinâmica do processo de controle ótimo é frequentemente um processo iterativo, em que o índice de desempenho é alterado a cada observação atual e avaliado e então atua na planta do sistema de modo a restabelecer ou mesmo manter a estabilidade do sistema, para um estado de operação próximo do especificado pelo projetista.

3.1.1 Modelagem Clássica do Projeto LQR

Introduzindo o índice de desempenho J temos:

$$J = \frac{1}{2} x^T(t_f) S x(t_f) + \frac{1}{2} \int_{t_0}^{t_f} [x^T(t) Q(t) x(t) + u^T(t) R(t) u(t)] dt \quad (3.3)$$

Onde Q e R são as matrizes de ponderação para as variáveis de estado e variáveis de entrada respectivamente, e t_f é o tempo final para a ação de controle, o que significa que a ação está em um intervalo finito de tempo. $S \geq 0$ é a matriz de ponderação para os estados finais. Este problema é referido como problema de controle ótimo linear quadrático (LQ). Para resolver esse problema ótimo LQ vamos primeiro construir a função Hamiltoniana:

$$H = -\frac{1}{2} [x^T(t) Q x(t) + u^T(t) R u(t)] + \lambda^T(t) [A x(t) + B u(t)] \quad (3.4)$$

Onde não existem restrições no sinal de entrada, o valor ótimo (neste caso, o mínimo) pode ser resolvido tomando-se a derivada de H em relação à u e então resolvendo a seguinte equação:

$$\frac{\partial H}{\partial x} = -Ru(t) + B^T \lambda(t) = 0 \quad (3.5)$$

Denotando por $u^*(t)$ o sinal de controle ótimo $u(t)$. Então, $u^*(t)$ pode ser explicitamente escrito como segue:

$$u^*(t) = R^{-1}B^T \lambda(t) \quad (3.6)$$

Por outro lado, pode ser mostrado que o multiplicador Lagrangiano $\lambda(t)$ pode ser escrito como $\lambda(t) = -P(t)x(t)$, onde $P(t)$ é a matriz solução da equação diferencial de Ricatti (EDR):

$$\dot{P}(t) = -P(t)A - A^T P(t) + P(t)BR^{-1}B^T P(t) - Q \quad (3.7)$$

Com seu valor final $P(t_f) = S$. Então, o sinal de controle ótimo pode ser escrito como:

$$u^*(t) = -R^{-1}B^T P(t)x(t) \quad (3.8)$$

3.1.2 O problema do LQR

O problema do Regulador Linear Quadrático (LQR) é apresentado sob o ponto de vista da natureza das matrizes de ponderação do estado e do controle. O procedimento de solução da *Equação Algébrica de Riccati* (EAR) é estável, pois as matrizes de ponderação e outras condições obedecem rigorosamente às restrições que garantem a existência de uma solução ótima. O LQR é formulado por meio de um índice de desempenho quadrático para minimizar a energia e resultará no valor ótimo e tem como restrição a equação de estado linear invariante no tempo,

$$\begin{aligned} \min J &= \frac{1}{2} \int_{t_0}^T [x^T Q x + u^T R u] \\ \text{s.a } \dot{x} &= Ax + Bu \end{aligned} \quad (3.12)$$

Sendo $x \in R^n$ e $u \in R^m$, o par (A,B) é controlável, (A,C) é observável, $P(T) \geq 0$, $Q \geq 0$ (semi definida positiva) e $R > 0$ (definida positiva). São matrizes simétricas, com $Q \in$

$\mathbb{R}^{n \times n}$ e $R \in \mathbb{R}^{m \times m}$ As metodologias de Cálculo Variacional, (Athans, 2007) são utilizadas para resolver o problema de otimização.

Quando t_f é finito, resolver o problema de controle ótimo LQ significa resolver a EDR, o que pode ser muito difícil. Porém muitas aplicações se preocupam com o desempenho regulatório que implica $t_f \rightarrow \infty$, o que implica no problema do regulador linear quadrático (LQR). No problema LQR, $t_f = \infty$ e o sistema de malha fechada será assintoticamente estabilizado. A solução da matriz $P(t)$ para a EDR tenderá à uma matriz constante, nesse caso a EDR se reduz para a equação algébrica de Ricatti (EAR) como segue:

$$PA + A^T P - PBR^{-1}B^T P + Q = 0 \quad (3.9)$$

Então o problema LQR pode ser resolvido usando uma realimentação linear de estados com uma matriz de ganho constante.

$$u(t) = -Kx(t) \quad (3.10)$$

Sendo K , o ganho do controlador.

$$K = -R^{-1}B^T P \quad (3.11)$$

Claramente, o sistema de malha fechada é simplesmente $[(A-BK), B, C, D]$. O sistema LQR resultante com realimentação de estados garante propriedades de robustez importantes, como margem de ganho infinita e 60° de margem de fase.

3.1.3 Seleção das matrizes de ponderação Q e R

O desempenho do sistema LQR é fortemente dependente da seleção das matrizes de ponderação. Então, mais precisamente, devemos dizer que o LQR é ótimo em relação às matrizes de ponderação Q e R escolhidas. Assim uma solução LQR ótima com uma escolha de Q e R , normalmente, não será ótima para outras escolhas de matrizes Q e R . O problema é que a especificação para o desempenho de um sistema prático de controle não será em termos de Q e R , logo enfrentamos o problema de tentar encontrar os valores de Q e R que irão atender as especificações (XUE; CHEN; ATHERTON, 2007). Em casos SISO, como a matriz

R é um escalar não negativo, uma prática popular é fazer $Q = C^T C$ e ajustar apenas a matriz R .

3.2 Metodologia do Regulador Linear Gaussiano- LQG

A metodologia LQG (“Linear *Quadratic Gaussian*”) para projeto de sistemas de controle já era conhecida antes da década de 70 (Athans, 1971) quando os aspectos de robustez e possíveis variações desta técnica passaram a ser vastamente investigados (Gilman and Rhoads, 1973; Houpisand Constantinides, 1973). Safonov e Athans (1977) mostram como um projeto LQG multivariável podem satisfazer quatro requisitos:

1. Estabilizar sistemas insuficientemente estáveis;
2. Reduzir a sensibilidade da resposta do sistema a ruídos;
3. Realizar uma relação entrada / saída desejada;
4. Prover ao sistema robustez com relação a variações na dinâmica de malha.

O controle LQG foi desenvolvido para superar as desvantagens do LQR. Certas variáveis de estado não podem ser medidas ou são muito ruidosas. Por meio do controle LQG é possível estimar estes estados. É considerado um método de controle robusto, pois ruídos nas equações de estado e saída são explicitamente considerado. Além disso, informação quantitativa sobre o ruído é utilizado no projeto do controlador (XUE; CHEN; ATHERTON, 2007).

3.2.1 O problema LQG

Considere o modelo da planta em espaço de estados,

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

$$y(t) = Cx(t) + \theta(t) \quad (3.13)$$

Onde $\xi(t)$ e $\theta(t)$ são ruídos aleatórios nas medidas das equações de estado e saída respectivamente. Assumindo que $\xi(t)$ e $\theta(t)$ são processos Gaussianos aleatórios de média zero com matrizes de covariância dadas por:

$$E[\xi(t)\xi^T(t)] = \Xi \geq 0, \quad E[\theta(t)\theta^T(t)] = \Theta \geq 0 \quad (3.14)$$

Onde $E[x]$ denota o valor médio de x e $E[x x^T]$ é a matriz de covariância do sinal x . Assume-se que os sinais aleatórios $\xi(t)$ e $\theta(t)$ são mutualmente independentes. O índice de desempenho para controle ótimo é definido como,

$$J = E \left\{ \int_0^{\infty} [z^T(t)Qz(t) + u^T(t)R(t)]dt \right\} \quad (3.15)$$

Onde $z(t) = Mx(t)$ é a combinação linear do vetor de estado $x(t)$ com M definido pelo usuário para medir o desempenho. As matrizes constantes Q e R são, respectivamente simétrica semi-positiva-definida e simétrica positiva-definida, que significa $Q = Q^T \geq 0, R = R^T > 0$. Note que R é um escalar, quando é uma única entrada-Sistema de controle de feedback de saída única (SISO).

3.2.2 Solução do Problema do LQG via Filtro de Kalman

O problema LQG pode ser dividido em dois subproblemas:

- a) Controle ótimo LQ e
- b) A estimação dos estados com perturbações.

Os estados podem ser estimados de forma ótima se um filtro de Kalman, em vez de um observador, for utilizado. Podemos, primeiramente, encontrar um sinal ótimo de estimação dos estados $\hat{x}(t)$ que minimiza a covariância $E[(x - \hat{x})(x - \hat{x})^T]$ e então usar o sinal estimado $\hat{x}(t)$ para substituir as reais variáveis de estado de modo que o problema original possa ser reduzido para um problema de controle ótimo LQ ordinário. O diagrama de blocos do filtro de Kalman é mostrado na figura 2, onde a matriz de ganho do filtro de Kalman K_f é dada por:

$$K_f = P_f C^T \Theta^{-1}, \quad (3.16)$$

Onde P_f Satisfaz a equação algébrica de Ricatti (EAR):

$$P_f A^T + A P_f - P_f C^T \Theta^{-1} C P_f + \Gamma \Xi \Gamma^T = 0 \quad (3.17)$$

$E P_f$ é uma matriz simétrica semi positiva-definida, o que significa que $P_f = P_f^T \geq 0$

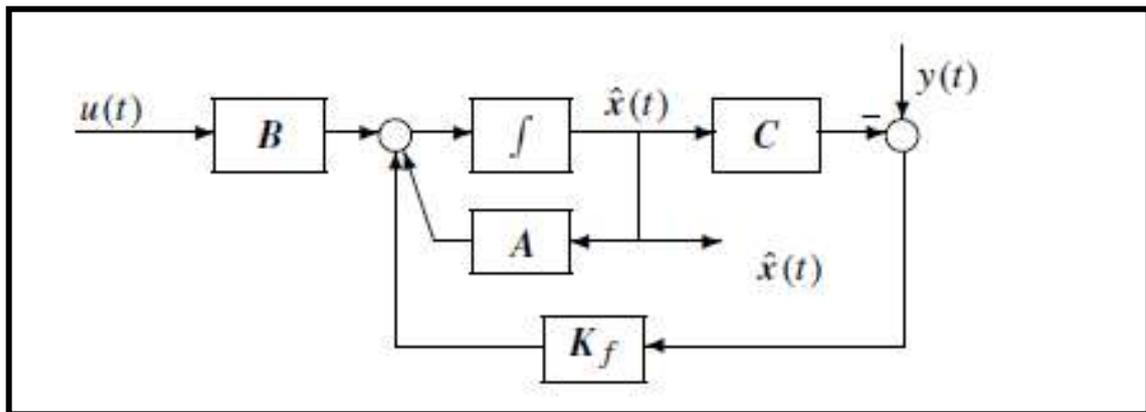


Figura 2- Diagrama do filtro de Kalman

Fonte: (XUE; CHEN; ATHERTON, 2007)

3.2.3 Princípio de separação para o projeto de LQG

Quando o sinal ótimo $\hat{x}(t)$ é obtido o diagrama de blocos do compensador LQG pode ser construído como mostrado na figura 3 com o controle ótimo $u^*(t)$ dado por:

$$u^*(t) = -K_c \hat{x}(t) \quad (3.18)$$

E a matriz de realimentação de estados ótimos K_c dada por

$$K_c = R^{-1} B^T P_c \quad (3.19)$$

Onde a matriz simétrica semipositiva-definida, P_c , satisfaz a seguinte EAR

$$A^T P_c + P_c A - P_c B R^{-1} B^T P_c + M^T Q M = 0 \quad (3.20)$$

Podemos observar que o problema de controle LQG, a estimação ótima e o problema de controle ótimo são resolvidos separadamente. Isso é conhecido como o princípio de separação, que significa que para projetar o controlador LQG, primeiro projetamos o estimador de estados e então utilizamos os estados estimados para projetar o controlador LQR, como se os estados fossem exatamente mensuráveis.

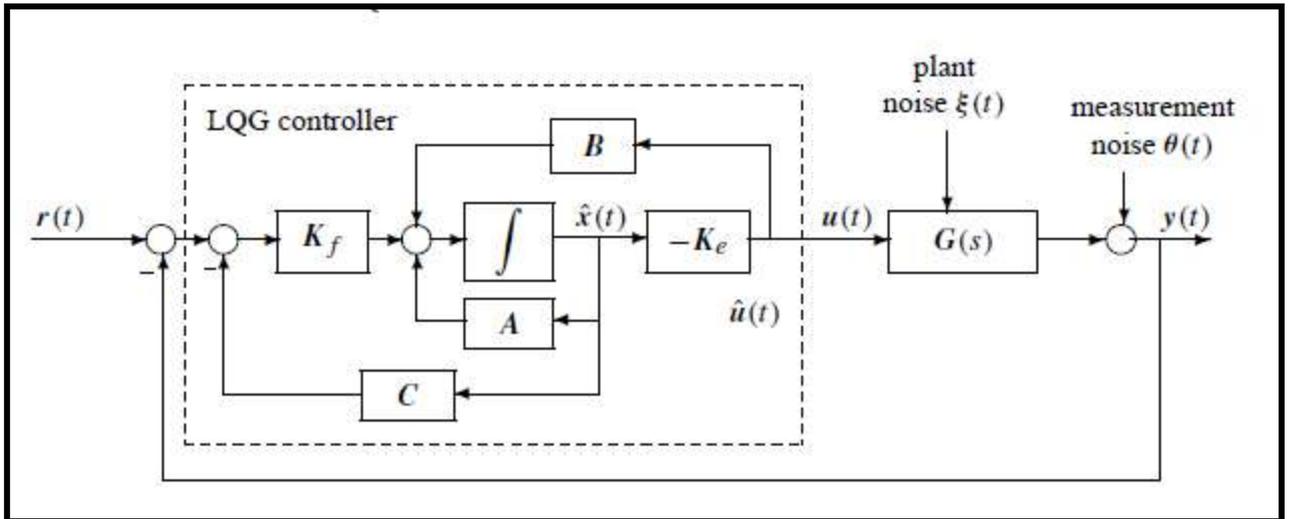


Figura 3- Estrutura do controlador LQG

Fonte: (XUE; CHEN; ATHERTON, 2007)

3.2.4 Controlador LQG

Para o modelo de planta representada em espaço de estados

$$\dot{x}(t) = Ax(t) + Bu(t) + \Gamma\xi(t) \quad (3.21)$$

$$y(t) = Cx(t) + Du(t) + \theta(t) \quad (3.22)$$

E critério de otimização

$$J \lim_{t_f \rightarrow \infty} E = \left\{ \int_0^{t_f} [x^T u^T] \begin{bmatrix} Q & N_c \\ N_c^T & R \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} dt \right\} \quad (3.23)$$

Onde N_c pode ser normalmente selecionado como uma matriz nula, a estrutura do controlador LQG com observador de estados é ilustrada na figura 4. Supõe-se que a matriz K_c e a matriz de ganho do filtro de Kalman K_f foram obtidas por meio do princípio de separação. Então, a equação dinâmica do filtro de Kalman pode ser escrita como

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + K_f(y - C\hat{x} - Du) \quad (3.24)$$

Então o controlador LQG pode ser compactamente formulado como segue

$$G_c(s) = \begin{bmatrix} A - K_f C - BK_c + K_f DK_c & K_f \\ K_c & 0 \end{bmatrix} \quad (3.25)$$

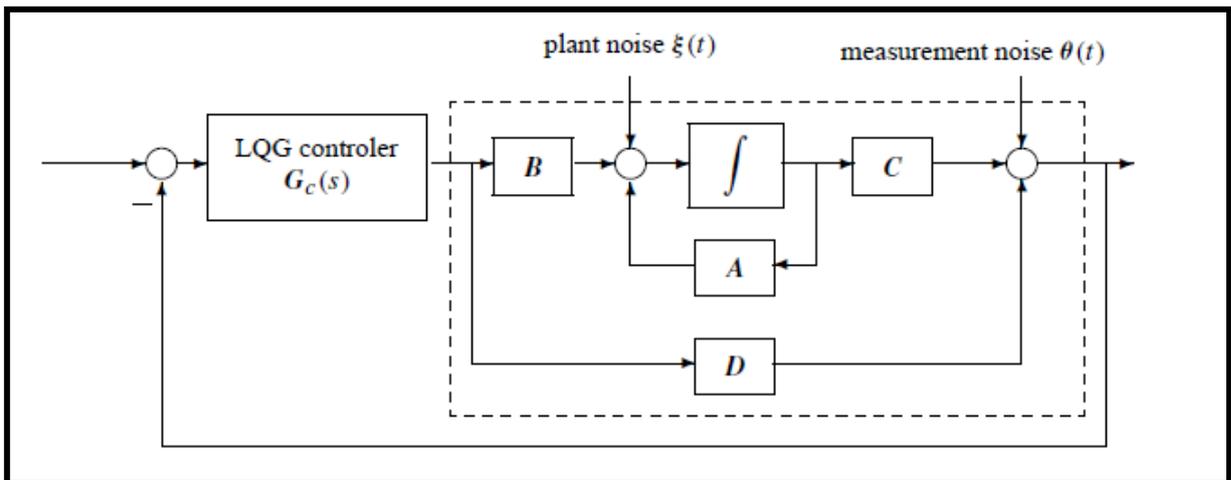


Figura 4: LQG com observador de estado

Fonte: (XUE; CHEN; ATHERTON, 2007)

3.3 Metodologia LQG/LTR

O Regulador Linear Quadrático-LQR e o Filtro de Kalman podem ser combinados para projetar um regulador dinâmico, o LQG, como visto na Seção 3.2. Sabe-se que o LQR possui ótimas propriedades de robustez (Lewis and Syrmos 1995), ou seja, margem de ganho infinita e margem de fase de 60 graus, desde que os métodos para obtenção do controlador K e do estimador L sejam duais.

No projeto LQG, que usa um estimador ótimo do estado, mas o processo de estimativa do estado destrói a Margem de robustez que é alcançada no caso LQR. No entanto, com o uso da técnica LTR as propriedades de robustez podem ser recuperadas. O procedimento para recuperar as mesmas é formado de dois passos: No primeiro, define-se a chamada Malha Objetivo (Target Feedback Loop) e, no segundo, através de um procedimento assintótico, recuperam-se as características de resposta em frequência dessa malha ajustando um determinado parâmetro (DOYLE AND STEIN 1981) (CRUZ 1996) (KWAKERNAAK 1969) (KWAKERNAAK AND SIVAN 1972).

A Recuperação do Ganho de Malha de Realimentação na Entrada considera constante o ganho do LQR e, em seguida, ajusta o ganho do Filtro de Kalman de tal forma que as propriedades do LQR sejam recuperadas. No entanto, é possível modificar o projeto do estimador de modo a recuperar as propriedades de robustez e estabilidade do LQR até certo ponto. Esse processo, chamado de Loop Transfer Recovery (LTR), é especialmente eficiente com sistemas de fase mínima. A introdução da técnica LTR no projeto LQG significa que o controlador pode ser projetado para alcançar a sensibilidade desejada $S(s)$ e funções complementares de

sensibilidade $T(s)$ em pontos críticos no sistema de realimentação, como na saída ou na entrada da planta. Formula se agora o problema LTR, supõe-se que todos os estados estão disponíveis e é possível medi-los, como no caso LQR. Com o controlador ótimo LQR, a função de transferência em malha aberta é simplesmente:

$$G_{LQSF}(s) = K_c(sI - A)^{-1}B \quad (3.26)$$

No entanto, controle LQG a malha aberta se torna:

$$G_{LQSF}(s) = K_c(sI - A + BK + LC)^{-1}LC(sI - A)^{-1}B \quad (3.27)$$

Seja $\Xi' = q\Xi$, pode ser mostrado que quando $q \rightarrow \infty$, a função de transferência em malha aberta do problema de controle LQG com Ξ' deverá se aproximar do LQR:

$$\lim_{q \rightarrow \infty} K_c(sI - A + BK + LC)^{-1}LC(sI - A)^{-1}B = K_c(sI - A)^{-1}B \quad (3.28)$$

Logo, o ponto chave no projeto do controlador LQG/LTR é selecionar um q grande o suficiente. Alternativamente, podemos primeiro solucionar o problema LQR de forma a se conseguir um vetor de ganho adequado e então usar a técnica LTR para fazer o sistema final com a função de transferência, incluindo o filtro de Kalman, se aproximar daquela do sistema LQR o máximo possível (FRANKLIN; POWELL; EMAMI-NAEINI, 2010). Consequentemente, temos o seguinte algoritmo de dois passos:

- a) Projetar um controlador LQR ótimo ajustando as matrizes de ponderação Q e R para que a função de transferência em malha aberta, $-K_c(sI - A)^{-1}B$, seja satisfatória.
- b) Fazer $\Gamma=B, W= W_0+qI, V=I$. Aumentar o valor de q de forma que a diferença retornada do sistema compensado se aproxime de $-K_c(j\omega I - A)^{-1}B$. Com o q selecionado, a equação de Ricatti do observador se torna:

$$\frac{P_f A^T}{q} + \frac{A P_f}{q} + \frac{P_f C^T V^{-1} C P_f}{q} + \frac{\Gamma W_0 \Gamma^T}{q} + \Gamma \Theta \Gamma^T = 0 \quad (3.29)$$

Onde q é referido como o coeficiente de ruído fictício. Quando o sistema original $C(sI - A)^{-1}B$ não possui zeros de transmissão no lado direito da planta s , o A matriz de ganho de filtro pode então ser avaliada a partir de:

$$K_f \rightarrow q^{1/2}BV^{-1/2} \quad qdo \quad q \rightarrow \infty \quad (3.30)$$

Na prática, q não deve ser muito grande, pois caso isto ocorra, irá introduzir truncagem numérica, erros que, por sua vez, podem afetar a robustez do sistema geral.

4 MODELAGEM INTELIGENTE VIA ALGORITMO GENÉTICO E REDE NEURAL

4.1 Modelagem Inteligente

O controle inteligente é baseado em simulação de processos biológicos, (Passino2004). Segundo (MEYSTEEL 1985), o controle Inteligente é um procedimento computacionalmente eficiente para atingir metas de um sistema dinâmico complexo de controle baseado em certos princípios de Inteligência Computacional. A inteligência computacional busca por meio de técnicas inspiradas na Natureza (redes neurais, computação evolucionária, lógica fuzzy, sistemas especialistas) o desenvolvimento de sistemas inteligentes que imitam aspectos do comportamento humano. No contexto deste trabalho, usa-se controle inteligente para estabelecer um conjunto de metodologias para o ajuste "inteligente" de ganhos de controladores LQR, LQG e LTR baseadas em modelos de otimização combinatória e técnicas de computação evolutiva, como algoritmos genéticos. Os modelos de otimização dedicam-se às seleções dos pares de matrizes de ponderações do estado e controle e matrizes de covariâncias da perturbação no estado e ruído de medida dos projetos LQR e LQG, respectivamente, que são consideradas como parâmetros para determinação das leis de controle do LQR e LQG, e à seleção de matrizes de ajuste para o ganho de recuperação da malha do projeto LQR.

4.2 Algoritmo Genético

Algoritmos Genéticos, (AGs), são métodos de otimização e busca inspirados nos mecanismos de evolução de populações de seres vivos. Foram introduzidos por John Holland (HOLLAND,1975) e popularizados por um dos seus alunos, David Goldberg (GOLDBERG, 1989). Estes algoritmos seguem o princípio da seleção natural e sobrevivência do mais apto, declarado em 1859 pelo naturalista e fisiologista inglês Charles Darwin em seu livro A Origem das Espécies. De acordo com Charles Darwin, “Quanto melhor um indivíduo se adaptar ao seu meio ambiente, maior será sua chance de sobreviver e gerar descendentes”.

Os AGs possuem uma larga aplicação em muitas áreas científicas, entre as quais podem ser citados problemas de otimização de soluções, aprendizado de máquina, desenvolvimento de estratégias e fórmulas matemáticas, análise de modelos econômicos, problemas de engenharia, diversas aplicações na Biologia como simulação de bactérias,

sistemas imunológicos, ecossistemas, descoberta de formato e propriedades de moléculas orgânicas (MITCHELL, 1997).

Os algoritmos simulam processos naturais de sobrevivência e reprodução das populações, essenciais em sua evolução. Na natureza, indivíduos de uma mesma população competem entre si, buscando principalmente a sobrevivência, seja através da busca de recursos como alimento, ou visando a reprodução. Os indivíduos mais aptos terão um maior número de descendentes, ao contrário dos indivíduos menos aptos. Os requisitos para a implementação de um AG são:

- Representações das possíveis soluções do problema no formato de um código genético;
- População inicial que contenha diversidade suficiente para permitir ao algoritmo combinar características e produzir novas soluções;
- Existência de um método para medir a qualidade de uma solução potencial;
- Um procedimento de combinação de soluções para gerar novos indivíduos na população;
- Um critério de escolha das soluções que permanecerão na população ou que serão retirados desta;
- Um procedimento para introduzir periodicamente alterações em algumas soluções da população. Desse modo mantém-se a diversidade da população e a possibilidade de se produzir soluções inovadoras para serem avaliadas pelo critério de seleção dos mais aptos.

A ideia básica de funcionamento dos algoritmos genéticos é a de tratar as possíveis soluções do problema como "indivíduos" de uma "população", que irá "evoluir" a cada iteração ou "geração". Para isso é necessário construir um modelo de evolução onde os indivíduos sejam soluções de um problema. A execução do algoritmo pode ser resumida nos seguintes passos:

- Inicialmente escolhe-se uma população inicial, normalmente formada por indivíduos criados aleatoriamente;
- Avalia-se toda a população de indivíduos segundo algum critério, determinado por uma função que avalia a qualidade do indivíduo (função de aptidão ou "fitness");

- Em seguida, através do operador de "seleção", escolhem-se os indivíduos de melhor valor (dado pela função de aptidão) como base para a criação de um novo conjunto de possíveis soluções, chamado de nova "geração";
- Esta nova geração é obtida aplicando-se sobre os indivíduos selecionados operações que misturem suas características (chamadas "genes"), através dos operadores de "cruzamento" ("crossover") e "mutação";
- Estes passos são repetidos até que uma solução aceitável seja encontrada, até que o número predeterminado de passos seja atingido ou até que o algoritmo não consiga mais melhorar a solução já encontrada, conforme fluxograma da figura 5

O primeiro passo de um Algoritmo Genético típico é a geração de uma população inicial de cromossomos, que é formada por um conjunto aleatório de cromossomos que representam possíveis soluções do problema a ser resolvido. Durante o processo evolutivo, esta população é avaliada e cada cromossomo recebe uma nota (denominada de aptidão no jargão da literatura de AGs), refletindo a qualidade da solução que ele representa. Em geral, os cromossomos mais aptos são selecionados e os menos aptos são descartados (Darwinismo). Os membros selecionados podem sofrer modificações em suas características fundamentais através dos operadores de *crossover* e mutação, gerando descendentes para a próxima geração.

• População inicial

A população inicial pode ser obtida escolhendo-se um valor para os parâmetros ou variáveis de cada cromossomo randomicamente entre o seu valor mínimo e valor máximo.

$$\text{Cromossomo}(i) = [p_1, p_2, p_3, \dots, P_N]$$

Onde P_j = valor randômico entre $P_{j \min}$ e $P_{j \max}$

Há um compromisso entre o tamanho da população inicial e número de gerações ou interações para se obter a convergência. De modo geral, o AG converge mais rapidamente, em termos de números de interações para a solução, quando a população inicial é maior. Outro critério que deve ser avaliado para se comparar o desempenho do AG é comparar o número de vezes que a função custo foi avaliada. E quanto maior a população, maior número de avaliações da função custo.

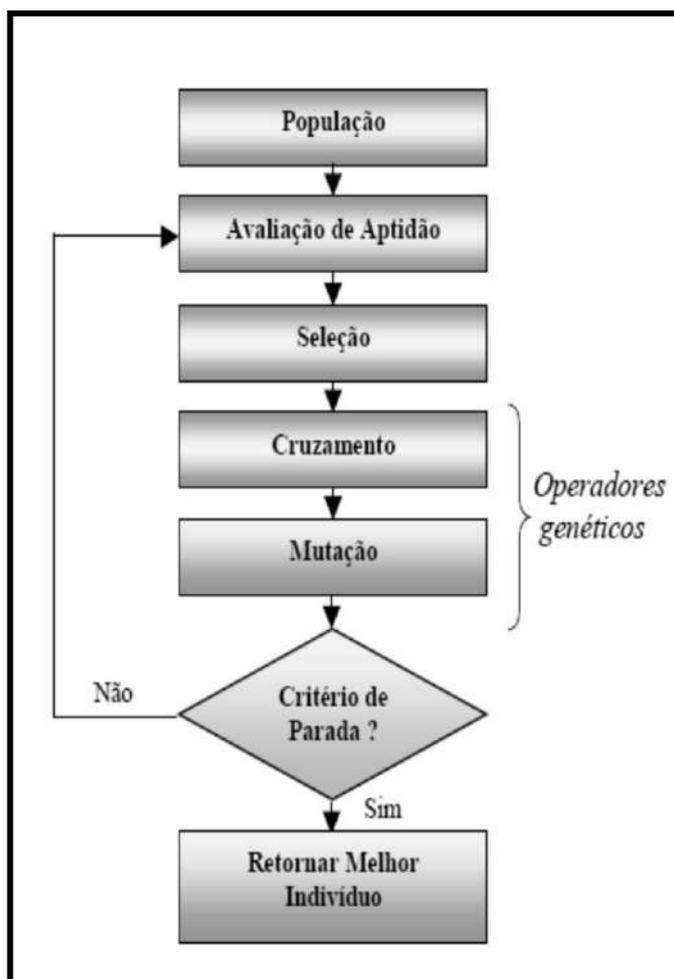


Figura 5: Estrutura básica de um Algoritmo Genético
Fonte: UFPA, [47]

• Seleção natural

A população inicial costuma ser maior do que a população nas gerações ou interações seguintes em alguns algoritmos genéticos. A escolha dos cromossomos que irão sobreviver, ou serão selecionados para serem aplicados os operadores de recombinação e mutação, deverão de alguma forma, valorizar aqueles cromossomos que apresentam melhor desempenho quando avaliados pela função custo do problema em questão. Assim sendo, este operador de seleção estará “imitando” a seleção natural e as espécies mais adaptados ao meio tendem a sobreviver. Logo, em um algoritmo de seleção clássico existe uma ordenação dos cromossomos de melhor para o pior, e normalizando assim, o valor de custo de cada cromossomo. E em seguida é realizada a escolha aleatória de um valor que representará a posição hierárquica do cromossomo sendo normalizado o seu custo.

• **O operador de recombinação (crossover)**

Este operador tem por objetivo combinar os parâmetros ou variáveis dos “pais”, gerando assim, cromossomos “filhos” que possam somar os pontos fortes de ambos produzindo, portanto, uma melhor solução para o problema em questão. Há diversos operadores de recombinação. O mais simples consiste em escolher um ou vários pontos aleatórios de corte no cromossomo, e permutar a partir destes pontos as variáveis dos cromossomos “pais” para gerar os “filhos”, de acordo com o esquema abaixo:

$$\begin{aligned} Pai_1 &= [p'_1, p'_2, p'_3, \dots, p_N] \rightarrow Filho_1 = [p_1, p_2, p'_3, \dots, p'_N] \\ Pai_2 &= [p_1, p_2, p_3, \dots, p_N] \rightarrow Filho_2 = [p'_1, p'_2, p_3, \dots, p_N] \end{aligned}$$

Neste operador nenhuma informação é introduzida nos cromossomos “filhos”. Uma outra possibilidade é realizar uma média ponderada entre os valores dos parâmetros. Tal método é chamado de “blending”, sendo dado por:

$$P_{NOVO} = \beta \cdot P_{pai1} + (1 - \beta) \cdot P_{pai2}$$

Onde β = fator de mistura ou parâmetro de ajuste podendo ser um valor randômico entre 0 e 1.

Neste operador nenhuma extrapolação além dos valores dos parâmetros existentes é feita.

Uma outra possibilidade é realizar a combinação linear entre os valores dos parâmetros. Tal método é chamado “crossover” linear. Neste, geram-se a partir dos “pais”, três cromossomos “filhos”, sendo um a média dos valores dos “pais”, e os outros dois valores, extrapolados, de acordo com as equações (4.1), (4.2) e (4.3).

$$P_{NOVO-1} = 0,5 \cdot P_{pai1} + 0,5 P_{pai2} \quad (4.1)$$

$$P_{NOVO-2} = 1,5 \cdot P_{pai1} + 0,5 P_{pai2} \quad (4.2)$$

$$P_{NOVO-3} = -0,5 \cdot P_{pai1} + 1,5 P_{pai2} \quad (4.3)$$

• **Operador de mutação**

Este operador tem por objetivo introduzir uma diversidade no cromossomo da população do AG, evitando assim, que o mesmo fique preso em mínimos locais. A implementação se dá de forma simples definindo uma probabilidade de mutação (PM) e definindo o número de parâmetros da população que sofre mutação (N_{PARMUT}).

$$(N_{PARMUT}) = PM.N \quad (4.4)$$

Onde N é o número de parâmetros da população.

Em seguida, escolhe-se de forma aleatória, dentro da população um parâmetro e aplica-se o operador de mutação dado por:

$$P_{J-NOVO} = \text{valor randômico entre } P_{J \min} \text{ e } P_{J \max} \quad (4.5)$$

Este procedimento será repetido até que se altere o número de parâmetros desejados em função da probabilidade de mutação Sistemas baseados em algoritmos genéticos (AG) 70

Um outro operador é o “não uniforme”, nele a perturbação permitida nos parâmetros vai diminuindo em função do número de gerações ou de iterações do AG. Aqui também existe uma probabilidade de mutação e um algoritmo que escolhe aleatoriamente qual o parâmetro do cromossomo (indivíduo) que vai sofrer mutação:

Cronossomo = $[p_1, p_2, p_3, \dots, p_N]$ e seja P_j o parâmetro que sofre mutação.

Então:

Onde \emptyset é um número aleatório igual a 0 ou 1, t é índice da interação ou geração atual do AG e a função Δ é:

$$\Delta[t, y] = y.R(1 - \left(\frac{t}{T}\right))^B \quad (4.6)$$

Onde, T é o número máximo de interações ou gerações do AG, R é um número aleatório entre 0 e 1, B é o fator de sintonia do operador. Portanto, a escolha do operador mais adequado depende do problema em questão.

4.3 Rede Neural

As redes neurais artificiais (RNAs) são sistemas de computação adaptativos inspirados nas características de processamento de informação encontradas nos neurônios reais e nas características de suas interconexões (S. HAYKIN, 1999).

O estudo de RNAs é baseado no comportamento das redes neurais biológicas. O cérebro humano possui uma enorme capacidade de processar informações, como por exemplo, o reconhecimento de fala e a segmentação de imagens. E, a partir desta análise, pesquisas são realizadas com a finalidade de reproduzir as reações do cérebro em máquinas (rede neural), tendo como resultado bastante promissor (apesar do pouco conhecimento do cérebro humano).

As RNAs são formadas por um conjunto de neurônios artificiais que interagem entre si, semelhantes ao funcionamento dos neurônios biológicos. Basicamente, são sistemas paralelos de computação e suas implementações, podem ser em hardware (realiza uma determinada tarefa a partir de componentes eletrônicos) ou em software (simulações por programação em um computador digital). As redes neurais têm a capacidade de coletar, utilizar e armazenar informações baseadas em experimentos (aprendizagem). O processo de aprendizagem é feito a partir de algoritmos de aprendizagem, onde os pesos sinápticos da rede são modificados de uma forma ordenada para alcançar o resultado desejado.

A partir da informação contida em pares de entradas e saídas, contidos em um banco de dados, denotados por $(\bar{p}_1 \bar{t}_1), (\bar{p}_2 \bar{t}_2), \dots, (\bar{p}_Q \bar{t}_Q)$ (as chamadas amostras de treinamento), as redes neurais devem “aprender” e generalizar o conhecimento contido nas amostras, de forma a responder de forma adequada quando entradas não presentes no conjunto de treinamento forem apresentadas à rede. Isto é possível devido ao fato de que a formulação é baseada numa representação (simplificada) de uma rede real de neurônios. Dessa forma, problemas relacionados a reconhecimento de padrões podem ser solucionados através das implementações e uso adequados desta técnica.

4.3.1 Neurônios Biológicos

Um neurônio (Figura 6) é composto por um corpo celular (ou soma), um axônio tubular e várias ramificações arbóreas conhecidas como dendritos. Os dendritos (zonas receptivas) formam uma malha de filamentos finíssimos ao redor do neurônio. Ao passo que o axônio (linhas de transmissão) consta de um tubo longo e fino que ao final se divide em ramos que terminam em pequenos bulbos que quase tocam os dendritos dos outros neurônios. O pequeno espaço entre o fim do bulbo e o dendrito é conhecido como sinapse, que tem como papel fundamental a memorização da informação.

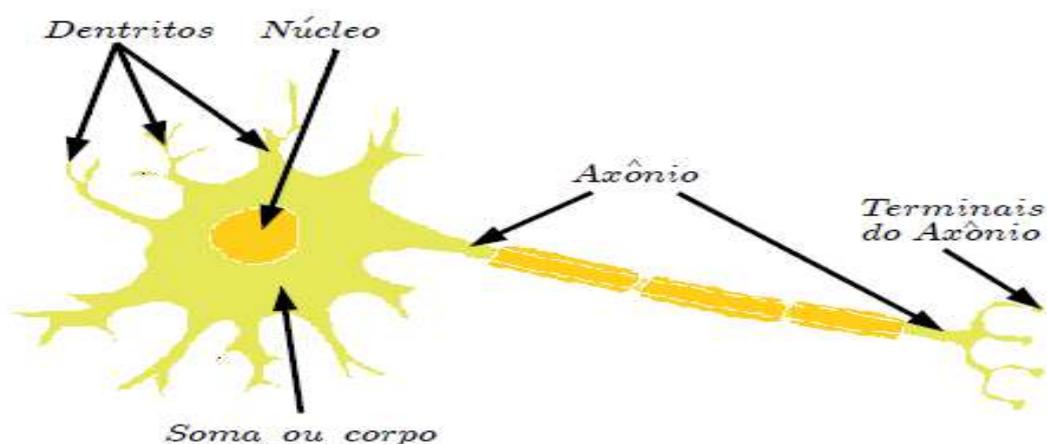


Figura 6-Modelo de um neurônio Biológico
Fonte: Matsunaga 2012

4.3.2 Modelo Artificial de um Neurônio

Basicamente, todos os tipos de redes neurais apresentam a mesma unidade de processamento: um neurônio artificial, que simula o comportamento do neurônio biológico. Esse neurônio artificial possui várias entradas, que correspondem às conexões sinápticas com outras unidades similares a ele, e uma saída, cujo valor depende diretamente da somatória ponderada de todas as saídas dos outros neurônios a esse conectado.

O modelo artificial de neurônio é mostrado na Figura 7, sendo uma generalização do modelo de McCulloch e Pitts. Esse modelo inclui um sinal adicional que bias favorece ou limita a possibilidade de ativação do neurônio. O processo sináptico é representado pelos pesos (w) que amplificam cada um dos sinais recebidos. A chamada função de ativação (f) modela a forma como o neurônio responde ao nível de excitação, limitando e definindo a saída da rede neural. A função de ativação pode ter diferentes representações. Os três tipos básicos de função de ativação são: limiar, linear e sigmóide. A escolha do tipo varia de acordo com o objetivo do projeto.

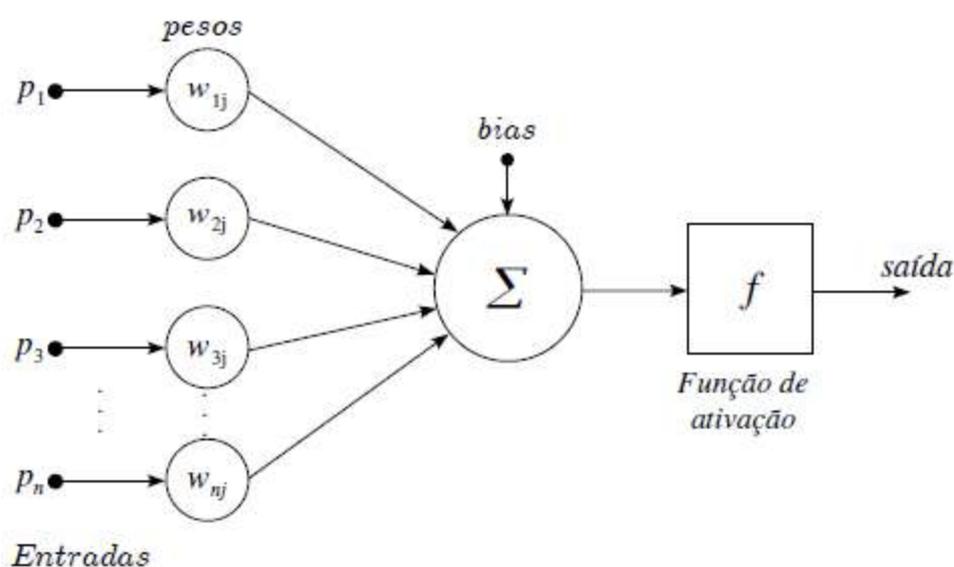


Figura 7- Modelo Artificial de Neurônio Biológico

Fonte: Matsunaga 2012

4.3.3 Arquitetura da Rede Neural

As disposições (arquitetura) de uma rede neural são de diversas formas, conforme a necessidade da aplicação, estando diretamente ligadas com o algoritmo de aprendizagem usado para treinar a rede. Basicamente, os itens que compõem a estrutura de uma rede neural são, portanto, sujeito a modificações, sendo os seguintes:

- Camadas intermediárias;
- Quantidade de neurônios;
- Função de transferência;
- Algoritmo de aprendizado

A maneira pela qual os neurônios de uma rede neural estão estruturados está intimamente ligada ao algoritmo de aprendizagem a ser usado para treinar a rede. Pode-se, portanto, falar de algoritmos (regras) de aprendizagem utilizados no projeto de redes neurais como sendo estruturados. Em geral, podemos classificar três tipos de arquiteturas de rede fundamentalmente diferentes, como descritos a seguir: *redes feedforward de uma única camada, redes feedforward de múltiplas camadas, e redes recorrentes.*

4.3.3.1 Rede Feedforward com Uma Única Camada

Este caso mais simples de rede em camadas consiste em uma camada de entrada e uma camada de saída. Geralmente os neurônios de entrada são lineares, ou seja, eles

simplesmente propagam o sinal de entrada para a próxima camada. São também denominados de neurônios sensoriais.

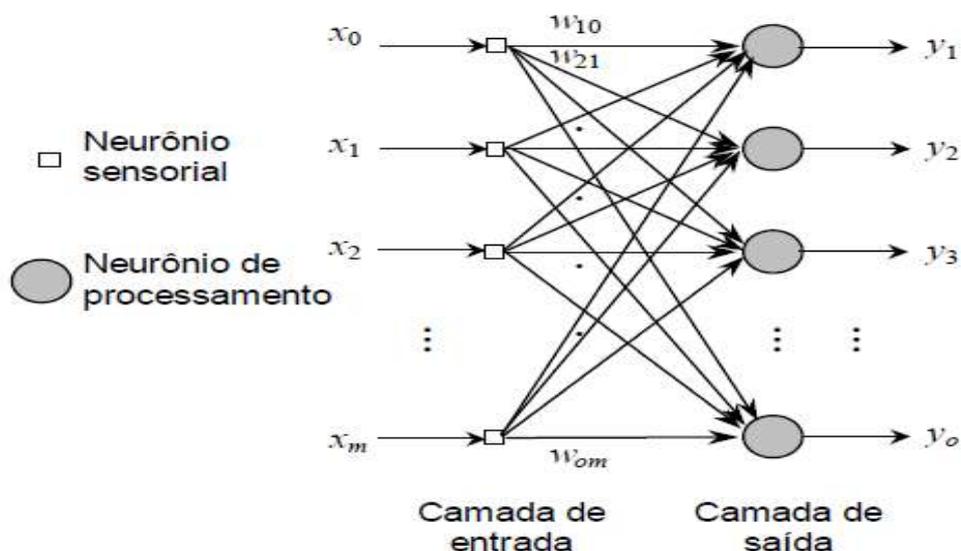


Figura 8- Rede denominada de feedforward

Fonte: Matsunaga 2012

Esta rede é denominada *feedforward* porque a propagação do sinal ocorre apenas da entrada para a saída, ou seja, é apenas no sentido positivo.

4.3.3.2 Rede Feedforward Múltiplas Camadas

As redes de múltiplas camadas possuem uma ou mais camadas intermediárias ou escondidas. Adicionando-se camadas intermediárias não-lineares é possível aumentar a capacidade de processamento de uma rede feedforward. A saída de cada camada intermediária é utilizada como entrada para a próxima camada. Em geral o algoritmo de treinamento para este tipo de rede envolve a retropropagação do erro entre a saída da rede e uma saída desejada conhecida. Seja W_k a matriz de pesos da camada k , contadas da esquerda para a direita, w_{ij}^k corresponde ao peso ligando o neurônio pós-sináptico i ao neurônio pré-sináptico j na camada k .

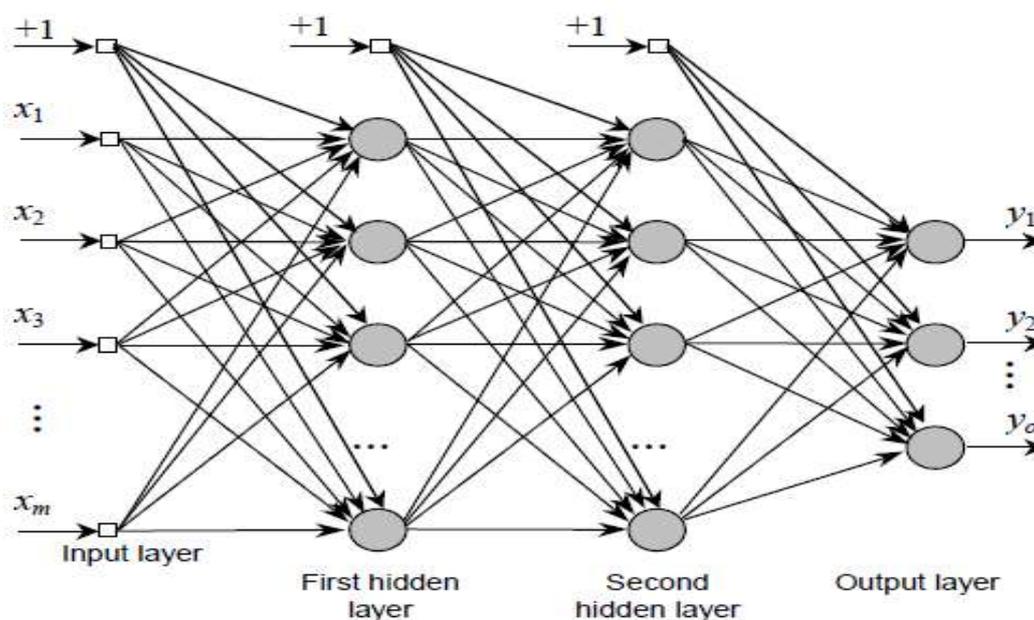


Figura 9- Rede denominada de feedforward múltiplas camadas
Fonte: Matsunaga 2012

4.3.3.3 Redes Recorrentes

O terceiro principal tipo de arquitetura de RNAs são as denominadas de redes recorrentes, pois elas possuem, pelo menos, um laço realimentando a saída de neurônios para outros neurônios da rede. A Realimentação as qualificam para processamento dinâmico de informações.

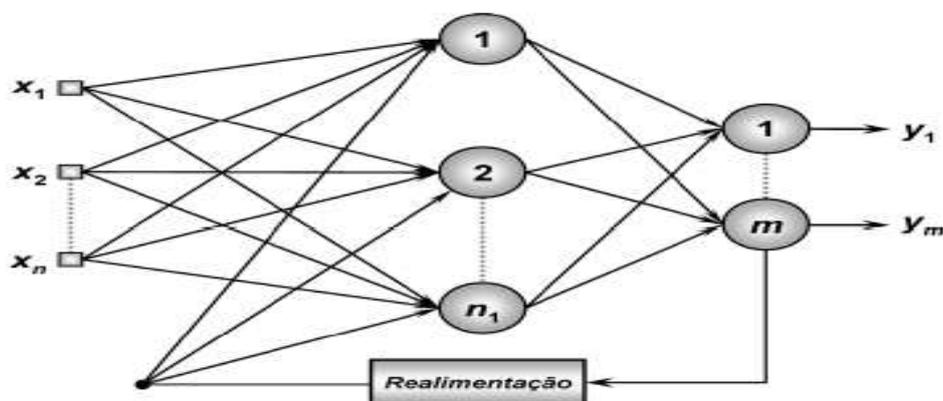


Figura 10- Modelo de rede Recorrente
Fonte: Matsunaga 2012

4.3.3.4 Rede Perceptron de Múltiplas camadas Recorrente

Foi a primeira rede neural que obteve sucesso prático. Inventado por Rosenblatt (em 1958). Esta rede estática consiste de um ou mais elementos de processamento (neurônios artificiais). A saída do perceptron será igual a 1 caso a somatória do produto de cada entrada pelo seu respectivo peso seja maior ou igual a zero. Caso contrário, a saída será igual a zero.

A Perceptron multicamada é uma rede estática que consiste de uma camada de entrada, uma camada de saída e uma ou mais de camadas ocultas (intermediárias) conectadas em feed-forward (cada camada se conecta à próxima camada, sem caminho de volta), isto é, a saída de um neurônio não tem dependência dos valores anteriores, pois os seus sinais se propagam num único sentido e suas saídas dependem somente dos sinais que chegam dos outros neurônios (não há laços nesse sistema). Na figura 11 é mostrado um exemplo de uma arquitetura gráfica do perceptron Multicamadas.

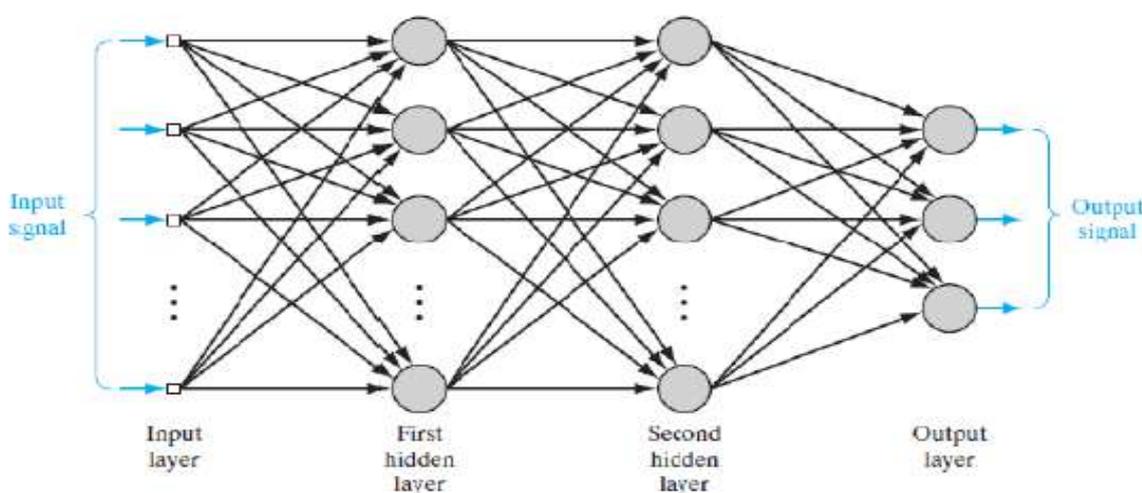


Figura 11- Arquitetura gráfica de um perceptron multicamada com duas camadas intermediárias

Fonte: Matsunaga 2012

4.3.4 Treinamento

O processo de treinamento de uma rede neural consiste basicamente em apresentar pares entrada-saída à rede, calcular as saídas da rede em função das entradas, calcular o erro entre a saída desejada e a saída calculada e alterar os valores sinápticos por algum tipo de algoritmo. À medida que os pares entrada-saída vão sendo apresentados e os pesos das sinapses atualizados, o erro médio quadrático (MSE) da saída da rede tende a diminuir.

Existem alguns tipos mais importantes de algoritmos, dentre os quais o mais utilizado e eficiente é o chamado de ‘Backpropagation’.

O método de ‘Backpropagation’ realiza, como o próprio nome diz, a retropropagação dos erros calculados das saídas em direção às entradas. O erro calculado na saída de um neurônio é multiplicado pela derivada da função de ativação daquele neurônio e propagado para a sua entrada. Este valor então é enviado para todos os neurônios da camada anterior pesado pelas respectivas sinapses.

O algoritmo de ‘Backpropagation’ utiliza o método do Gradiente Descendente para corrigir os valores sinápticos. Neste método, calculamos o gradiente da função objetivo a se minimizar e atualizamos o valor das sinapses somando-se a ele o valor do gradiente com o sinal invertido. O gradiente neste caso é a derivada parcial da função erro médio quadrático em relação à sinapse. Este processo faz com que o erro médio quadrático do problema “caminhe” sempre no sentido contrário de seu máximo crescimento.

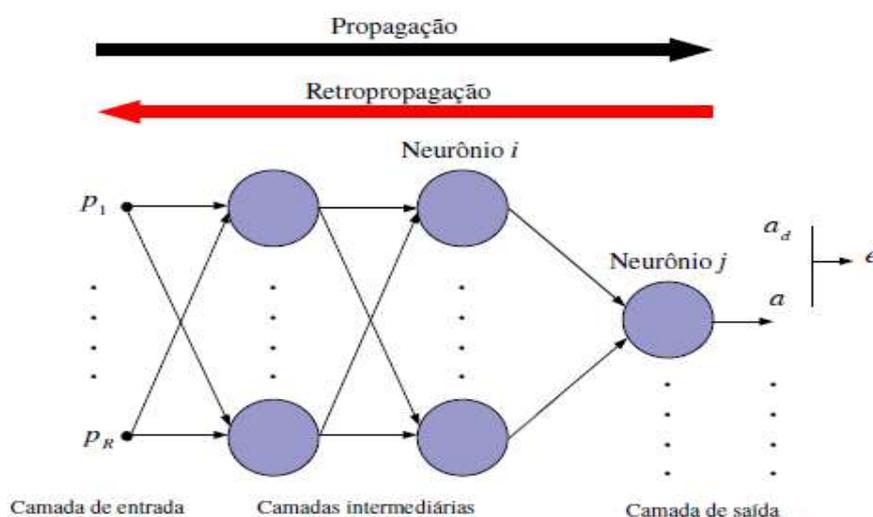


Figura 12- Ilustração do algoritmo Backpropagation
Fonte: Matsunaga 2012

5 REDE NEURAL APLICADA A CONTROLADORES ÓTIMOS

Redes neurais de feedforward e as redes neurais recorrentes são duas classes principais dos modelos de rede neural. Redes neurais Feedforward, como o perceptron multicamada popular (RUMELHART et al., 1986), geralmente são usados como modelos representacionais treinados usando um algoritmo de aprendizagem com base em um conjunto de amostras dados de entrada-saída. Está provado que o multi camada as redes neurais de feedforward são aproximadores universais (HORNIK et al., 1989). Também foi demonstrado que redes neurais treinadas com um número limitado de treinamentos. As amostras possuem uma boa capacidade de generalização. Em larga escala sistemas que contêm um grande número de variáveis e sistemas complexos onde poucos conhecimentos analíticos estão disponíveis são exemplos típicos de potencial rede neural aplicações (NARENDRA e PARTHASARATHY, 1990).

Redes neurais recorrentes, como as redes Hopfield (HOPFIELD, 1982, 1994), são geralmente utilizadas como modelos computacionais para resolver problemas computacionalmente intensivos. Exemplos típicos de aplicações de rede neural são problemas de otimização combinatória e grande escala ou problemas de computação em tempo real (HOPFIELD ETANK, 1985; TANK e HOPFIELD, 1986; BROCKETT, 1991).

As redes neurais são vantajosas em relação aos tradicionais abordagens para tais problemas porque a informação neural o processamento é inerentemente concorrente. Na literatura de controle, muita atenção foi dedicada, para reguladores lineares-quadráticos (LQR) e sistema de controle H infinito problemas de projetos, em grande parte como resultado do seu elegante problema de formulação, traçabilidade da solução e propriedades robustas com respeito a variações bastante grandes dos parâmetros do sistema. O problema de projetar um sistema de controle de feedback linear minimizando um índice de desempenho quadrático pode ser reduzido para o problema de obtenção de uma solução definitiva positiva de uma matriz Equação de Riccati. Uma variedade de algoritmos numéricos foram desenvolvidos para resolver a equação algébrica de Riccati (por exemplo, MAN, 1969; SALAMA e GOURISHANKAR, 1974; LAUB, 1979; GARDINER E LAUB, 1991; GHAVIMI et al., 1992; GARDINER, 1997; BUNSE-GERSTNER e FASSBENDER, 1997; IONESCU et al., 1997).

Veja uma visão abrangente em (WILLEMS et al., 1991). Embora os algoritmos paralelos possam computar as soluções para equações de Riccati mais rapidamente do que

algoritmos sequenciais, não houve relatório sobre neural soluções de rede para equações de Riccati. Devido à natureza distribuída em paralelo da computação neural, redes neurais podem ser um modelo de computação viável para sintetizar sistemas de controle linear em tempo real.

Nos últimos anos, redes neuronais favoráveis e recorrentes foram desenvolvidas para resolver uma grande variedade de matrizes problemas de álgebra (por exemplo, Wang e Mendel, 1991, 1992; Samardzija e Waterkind, 1991; Cichocki e Unbehauen, 1992a, 1992b; Luo e Zheng, 1992; Wang, 1992, 1993a, 1993b; Wang e Li, 1993; Wang e Wu, 1993). Em particular, redes neuronais recorrentes foram desenvolvidas para decomposição e factorização Cholesky (WANG e WU, 1993). Os resultados dessas investigações estabeleceram a base para a resolução das equações da matriz Riccati por uso de redes neurais em tempo real.

Este trabalho apresenta uma abordagem de rede neural para a solução equações algébricas de Riccati contínuas com base em uma rede neural recorrente multicamada. Ao resolver duas equações de matriz acoplada usando a rede neural recorrente, a abordagem da rede neural proposta é capaz de obter a solução definitiva simétrica e positiva para as equações algébricas de Riccati em um paralelo distribuído modal.

5.1 Controlador Inteligente LQR

O problema do *LQR* é formulado por uma estrutura de otimização não-linear, cujo foco consiste em minimizar um índice de desempenho quadrático. Este problema tem como gargalo, a seleção das matrizes de ponderação de estado e de controle, Q e R , respectivamente e resolver a equação algébrica de *Riccati* (*EAR*), de modo que possa alocar autovalores e autovetores especificados e que garanta a estabilidade e robustez do sistema de controle. São apresentadas as evoluções de duas abordagens de inteligência computacional para resolver o problema do *LQR*. Estas evoluções são resultados de pesquisas utilizando um *AG* na seleção das matrizes de ponderação Q e R e da solução da (*EAR*) por meio de uma (*RNR*), (WANG, 1998) e (IVANILDO, 2009)

Um dos grandes desafios na teoria de controle, utilizando o projeto do regulador linear quadrático é a seleção das matrizes de ponderação Q e R . Nesta pesquisa, utiliza-se uma abordagem de inteligência computacional para realizar a busca das mesmas, o *AG*.

Os parâmetros matriciais do projeto LQR são utilizados para realizar a sintonia do ganho ótimo do compensador. Apresenta-se o modelo matemático não-linear para exibir estas matrizes:

$$\begin{aligned} & \mathbf{Min}_{Q,R} \sum_{i=1}^n p_i s_i(Q,R) \\ & \text{s.a} \quad s_i(Q,R) \leq 1, \\ & i = 1, \dots, n \lambda_{i,esq.} \leq \lambda_{i,aloc.}(Q,R) \leq \lambda_{i,dir.}, \quad i = 1, \dots, n \end{aligned} \quad (5.1)$$

Sendo p_i os pesos das i -ésimas sensibilidades s_i / ε_i normalizadas com relação à i -ésima especificação de projeto ε_i . Este modelo aborda desde a representação genética das matrizes de ponderação, passando pelas operações cromossômicas, até chegar às avaliações de *fitness* (desempenho).

Percebe-se que a mesma constitui uma estrutura de otimização muito complexa, uma vez que envolve a solução do projeto LQR e o cálculo dos autovalores de malha fechada. O desempenho de cada cromossomo do indivíduo que compõe a população $QR_{ind.xg}$ é avaliado e cada indivíduo QR_z é pontuado. K_z é o ganho gerado por um indivíduo, A_z é a matriz de malha fechada para o ganho K_z , enquanto que λ_z, V_z, W_z constituem os autovalores e auto vetores à direita e à esquerda. Já S_z são as sensibilidades associadas com cada autovalor. As operações genéticas são representadas pelos modelos de seleção, *crossover* e mutação que podem ser consultados em (NETO, 2007).

Realizada a computação da auto-estrutura de malha fechada, a estratégia de pontuação de cada indivíduo é realizada da seguinte forma: Para cada autovalor, dentro da margem especificada na estrutura de otimização não linear pelo projetista, atribui-se valor um (01) ao indivíduo QR e um centésimo (0.01) para cada autovetor associado que possua sensibilidade normalizada, $S_z < 1$. O indivíduo que possuir maior pontuação será considerado o ótimo, e para o critério de empate, será considerado melhor, aquele que possuir menor soma das sensibilidades normalizadas.

5.2 Solução da Equação Algébrica de Ricatti via Rede Neuronal

Para resolver a EAR, usa-se problema de otimização combinatória, tendo como restrições as matrizes de ponderações. Esta estrutura é formulada para minimizar uma função

energia monotônica, não decrescente e limitada inferiormente, $\varepsilon(g, h)$, (WANG, 1998). As matrizes A , B , Q e R são apresentadas, sendo que seu objetivo é encontrar uma solução matricial simétrica e definida positiva, P . A seguir tem-se o problema de programação não-linear na forma vetorial

$$\min_{P,L} \sum_{i=1}^n \sum_{j=1}^n \left\{ e_{ij} [g_{ij}(P)] + e_{ij} [h_{ij}(P, L)] \right\} \quad (5.2)$$

$$G(P) = [g_{ij}(P)] = PSP - A^T P - PA - Q \quad (5.3)$$

$$H(P, L) = [h_{ij}(P, L)] = LL^T - P \quad (5.4)$$

$$\text{onde } g(P) = \sum_{k=1}^n \sum_{l=1}^n p_{ik} s_{kl} p_{lj} - \sum_{k=1}^n (a_{ki} p_{kj} + p_{ik} a_{kj}) - q_{ij}, \text{ e} \quad (5.5)$$

$e_{ij} : R \rightarrow R$ é uma função matricial convexa.

$$h_{ij}(P, L) = \sum_{k=1}^{\min(i,j)} l_{ik} l_{jk} - p_{ij} \quad (5.6)$$

Nota-se que e_{ij} é definida somente em função de g_{ij} ou h_{ij} .

As equações dinâmicas da rede neural Recorrente (RNR) para resolução da EAR na forma matricial para $i, j = 1, 2, \dots, n$ é dada por:

$$\frac{dp_{ij}(t)}{dt} = -n_p \left\{ \sum_{k=1}^n \sum_{l=1}^n [p_{ik}(t) s_{kl} u_{lj}(t) + u_{ik}(t) s_{kl} p_{lj}(t)] - \sum_{k=1}^n [a_{ki} u_{kj}(t) + u_{ik}(t) a_{kj}] - y_{ij}(t) \right\} \quad (5.7)$$

$$\frac{dz_{ij}(t)}{dt} = -n_z \sum_{k=j}^n y_{ik}(t) z_{kj}(t), \quad i \geq j \quad (5.8)$$

$$u_{ij}(t) = f_{ij} \left[\sum_{k=1}^n \sum_{l=1}^n p_{ik}(t) s_{kl} p_{lj}(t) - \sum_{k=1}^n [a_{ki} p_{kj}(t) + p_{ik}(t) a_{kj}] + q_{ij} \right] \quad (5.9)$$

$$y_{ij}(t) = f_{ij} \left[\sum_{k=1}^{\min\{i,j\}} z_{ik}(t) z_{jk}(t) - p_{ij}(t) \right] \quad (5.10)$$

Escrito na forma matricial,

$$\frac{dP(t)}{dt} = -\eta_p [P(t)SU(t) + U(t)SP(t) - AU(t) - U(t)A^T - Y(t)] \quad (5.11)$$

$$\frac{dZ(t)}{dt} = -\eta_z Y(t)Z(t) \quad (5.12)$$

$$U(t) = F[P(t)SP(t) - A^T P(t) - P(t)A + Q] \quad (5.13)$$

$$Y(t) = F[Z(t)Z^T(t) - P(t)] \quad (5.14)$$

Onde, $U(t) = [u_{ij}(t)]$, $P(t) = [p_{ij}(t)]$ e $Y(t) = [y_{ij}(t)]$

São matrizes quadradas dos estados de ativação de ordem $n \times n$, $Z(t) = [z_{ij}(t)]$, é uma matriz triangular inferior dos estados de ativação de ordem $n \times n$, e esta representa o fator de *Cholesky*. $F = [f_{ij}]$ é uma matriz não decrescente de funções de ativações; $P(t)$ é a solução da EAR. Os parâmetros positivos de ajuste da rede neuronal são os pesos η_p e η_z . Como F é simétrico, Z(t), U(t) e Y(t) também são simétricos. A simetria dessas matrizes pode ser utilizada na realização da rede neural para reduzir o número de neurônios das camadas substancialmente. A matriz S representa na Equação (5.13) a influência da matriz de ponderação R, sendo dada por $S = BR^{-1}B^T$, sendo que os ganhos do controlador ótimo são ajustados tendo como base as variações das matrizes Q e R.

5.3 Formulação do problema

Em vista do fato de que a equação algébrica de Riccati tem soluções múltiplas, uma restrição adicional deve ser incluída para obter uma definição simétrica e positiva única P. Uma vez que qualquer matriz real, simétrica e positiva tem um fator Cholesky, uma restrição de simetria e positiva. A definição de P pode ser adicionada exigindo:

$$LL^T = P \quad (5.15)$$

Onde $L \in \mathfrak{R}^{n \times n}$ é uma matriz triangular inferior com uma diagonal positiva (fator Cholesky). Ao definir matrizes de blocos \tilde{P} e Λ a equação de matriz Riccati pode ser descrita como:

$$\tilde{P}\Lambda\tilde{P}^T = Q + I \quad (5.16)$$

Onde I é a matriz identidade,

$$\tilde{P} \triangleq (P \ I), \Lambda \triangleq \begin{pmatrix} S & -A \\ -A^T & I \end{pmatrix} \quad (5.17)$$

Como Q é simétrica e definida positiva (semidefinida), $Q + I$ é sempre simétrica e definida positiva. Observe que A também é simétrica. Portanto, o problema da síntese de um LQR pode ser reformulado como o problema de resolver as equações 5.15 e 5.16, pois elas têm o mesmo formato.

5.3.1 Arquitetura da rede

O modelo recursivo da RNAR é baseado na rede neural desenvolvida por (Wang, 1998). A estrutura de otimização é dada por um conjunto de equações diferenciais e funções de ativação F linear saturada que constituem a arquitetura da rede neural na forma matricial.

A Figura 13 ilustra a arquitetura da rede neuronal para resolver a EAR, sendo formada por quatro camadas conectadas bidirecionalmente. A camada de saída $P(t)$ é a solução da EAR, $Z(t)$ representa o fator de *Cholesky* de P .

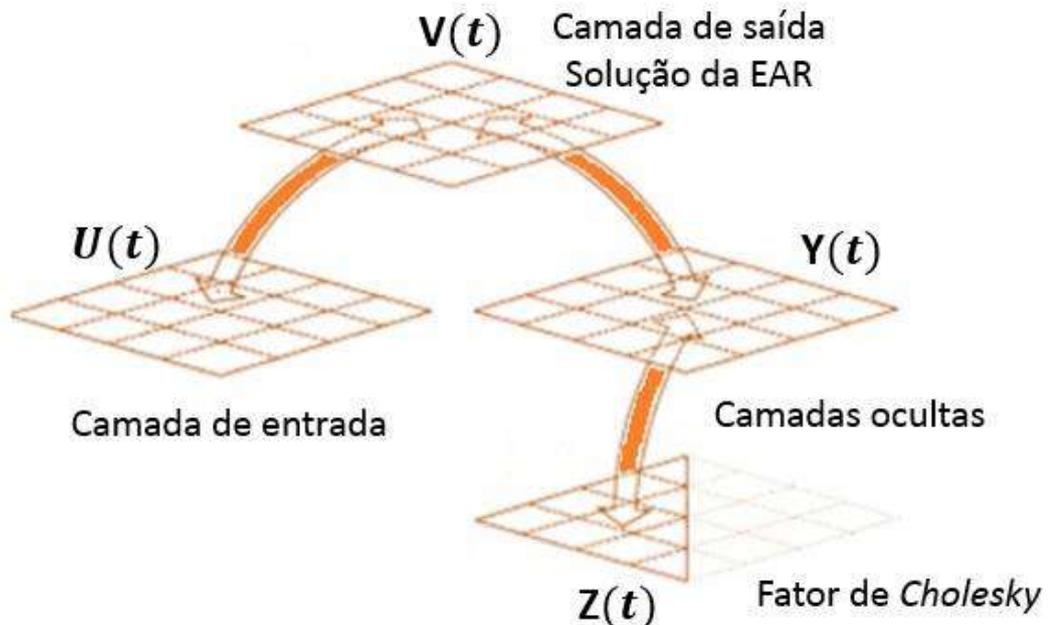


Figura 13- Arquitetura do RNAR para a EAR

Fonte: Wang e Wub 1998

5.3.2 Dinâmica do Diagrama de Blocos

A Figura 14 ilustra o bloco diagrama da rede neural recorrente multicamada para solução as equações algébricas de Riccati. O papel das constantes de escala (η_z, η_v) é dimensionar o taxa de convergência. Especificamente, o maior (η_z, η_v) é o mais rápido a convergência. Portanto, a convergência do processo de computação neural pode ser acelerada selecionando um par (η_z, η_v) com valores suficientemente grandes. Além disso, para garantir que o estado estacionário de $V(t)$ seja Definido positivo, requer que $Z(t)$ convirta mais rápido do que $V(t)$ definindo (η_z, η_v) . Como a taxa de convergência da rede neuronal recorrente aumenta à medida que as constantes de escala (η_z, η_v) aumentam, a convergência da rede neural recorrente. A rede pode ser acelerada selecionando suficientemente grandes constantes de escala. Conforme mostrado nas Eqs. (5.14) e (5.15), a taxa de convergência da rede neural recorrente multicamada é em grande parte magnitudes dos autovalores de $U(t)$ e $Y(t)$, bem como os parâmetros (η_z, η_v) . Quanto maiores forem as magnitudes dos autovalores e quanto maior o valor de (η_z, η_v) , mais rápido a convergência. Devido à natureza paralela do neural computação e relação de taxa de convergência e autovalores e parâmetros de projeto, a taxa de convergência da rede neural normalmente não é uma função crescente da dimensão de um sistema de controle.

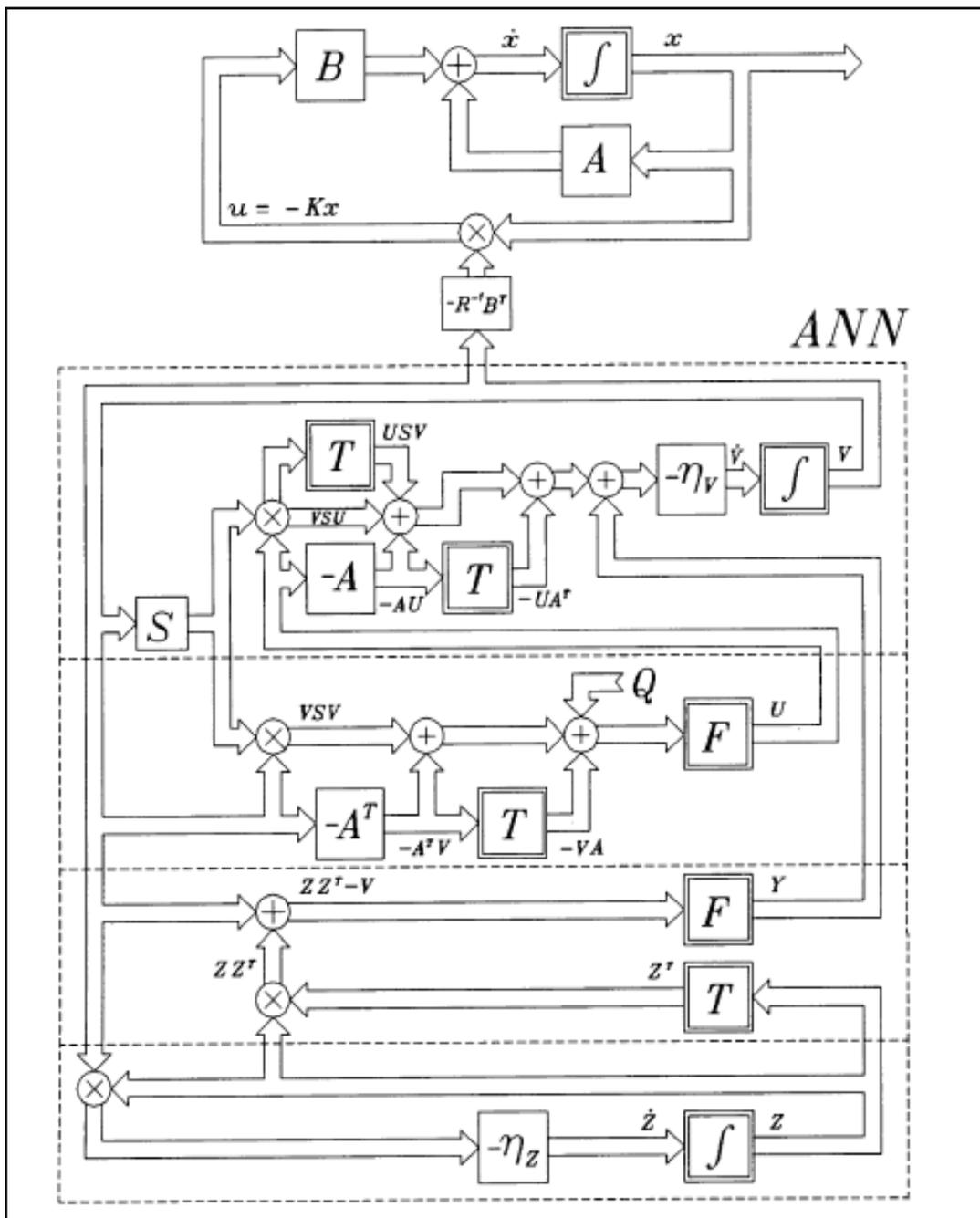


Figura 14- Síntese do diagrama de bloco do processo dinâmico para o Regulador Linear Quadrático

Fonte: Wang e Wub 1998

5.3.3 Análise de Estabilidade e Solvabilidade

A rede recorrente proposta é praticamente um sistema dinâmico não-linear contínuo. A estabilidade dos sistemas dinâmicos não-lineares pode ser analisada usando o método direto de Lyapunov. Se cada função de ativação é contínua, pelo menos diferenciável por partes, monótona não decrescente

$$\left(\text{isto é, } \frac{df_{ij}(\xi)}{d\xi} \geq 0 \text{ para } -\infty < \xi < +\infty \right), \text{ e } f_{ij}(\xi) = 0$$

se $\xi = 0$ ($i, j = 1, 2, \dots, n$), então a rede neural recorrente é considerada assintoticamente estável

(isto é, $\forall P(0), Z(0), \exists Z, P$ de modo que o $\lim_{t \rightarrow \infty} Z(t) = \bar{Z}$ e $\lim_{t \rightarrow \infty} P(t) = \bar{P}$).

A solvabilidade consiste em verificar se o problema de controle automático admite ou não solução. A análise de estabilidade indica que a transição de estado de ativação forma um fluxo de gradiente que minimiza E. O seguinte teorema fornece uma condição necessária e suficiente para que o fluxo de gradiente converge para a solução final positiva da equação algébrica de Riccati, P, de praticamente qualquer estado inicial. Suponha que cada função de ativação é contínua, pelo menos diferenciável por partes, monótona não decrescente

$$\left(\text{isto é, } \frac{df_{ij}(\xi)}{d\xi} \geq 0 \text{ para } -\infty < \xi < +\infty \right), \text{ e } f_{ij}(\xi) = 0 \text{ se } \xi = 0.$$

A Matriz de estado estacionário da rede neural recorrente é sempre a solução simétrica e definida positiva para a equação de Riccati de qualquer estado inicial simétrico P(0) e quaisquer estados iniciais não nulos Z(0) (isto é, $\forall P(0) = P(0)^T, \forall Z(0) \neq 0, \bar{P} = P$) se somente se o posto $\bar{P}S - A = n$ e todos os elementos diagonais em Z são nulos (ou seja, $\forall i, \bar{z}_{ii} \neq 0$ para $i = 1, 2, \dots, n$).

5.4 Filtro de Kalman para estimação dos estados

Neste capítulo, formula-se o Filtro de Kalman, FK, de forma a alocar os autovalores e autovetores no problema de estimação de estado estocástica. Esta nova formulação permite a utilização de técnicas de computação evolutiva para determinar um observador estocástico para estimar os estados com uma dinâmica desejada, onde não se tem acesso para mensurá-los, a partir do *Filtro de Kalman*. Fazendo-se uso das restrições de autoestrutura, restrições de autovalores e autovetores, pode-se formular o problema de estimação de estado estocástica, que é encontrar $L = \Sigma(\Xi, \Theta)C^T\Theta^{-1}$ do FK, de modo que sejam satisfeitas as restrições. Este problema pode ser formulado, como um problema de otimização permitindo a determinação de um estimador estocástico $L(\Xi, \Theta)$ por meio de técnicas de busca aleatória direcionada.

Os primeiros pesquisadores a perceber o problema de alocação de polos e sua importância foram J. Bertram em 1959 e o matemático húngaro Rudolf Emil Kalman. Os trabalhos de Kalman certamente constituem contribuições muito importantes e também precursoras para a concepção e a solução do problema de alocação de autovalores otimizados ou não.

O Filtro de Kalman é um método de estimação estocástico utilizado para obter estimativas ótimas das variáveis de estados de um sistema dinâmico de modo que o erro seja minimizado estatisticamente. Um AG para a busca das matrizes de covariâncias Ξ da perturbação do estado e Θ do ruído de medida para o *Filtro de Kalman* é desenvolvido. Para tanto, tem-se interesse em realizar um algoritmo numérico, que possa ser utilizado em qualquer problema de estimação de estado estocástico, objetivando-se a determinação de um ganho $L(\Xi, \Theta)$ de forma a alocar autovalores e autovetores.

No projeto do FK são considerados ruídos do processo e da medida, ambos gaussianos de média zero, variância conhecida e não correlacionados entre si, os quais serão utilizados para se obter a estimativa do estado. Considere o sistema dinâmico representado por:

$$\dot{x}(t) = Ax(t) + G\xi(t) \quad (5.18)$$

Em que $x(t) \in \mathbb{R}^m$ é um processo estocástico chamado de ruído no estado (ou Processo), que se admite ser branco, gaussiano e tal que:

$$E[\xi(t)] = 0 \quad (5.19)$$

$$E[\xi(t)\xi(t + \tau)^T] = \Xi\delta(t - \tau) \quad (5.20)$$

$\Xi = \Xi^T > 0$ é a matriz de covariância do ruído no estado ou densidade espectral do ruído do estado, que indica a potência do ruído e $\delta(t - \tau)$ é o **Delta de Dirac** que traduz se o valor do ξ no instante t não está relacionado com seu valor em outro instante $t \neq 1$. Admite-se que os sensores meçam não o estado diretamente, mas apenas variáveis de saída sujeitos à ação de um ruído de medida aditivo, isto é,

$$y(t) = Cx(t) + v(t) \quad y(t) \in \mathbb{R}^m \quad (5.21)$$

$v(t)$ é suposto branco, gaussiano, independente de $\xi(t)$ e tal que

$$E[v(t)] = 0 \quad (5.22)$$

$$E[v(t)v(t + \tau)^T] = \Theta\delta(t - \tau) \quad (5.23)$$

$$E[v(t)v(t + \tau)]^T = 0, \forall t, \tau \quad (5.24)$$

Onde $\Theta = \Theta^T > 0$ é a matriz de covariância (ou densidade espectral) do ruído de medida que indica a potência do ruído. O par (A, C) dada equação de estado é suposto observável e o sistema dinâmico descrito pode ser representado na forma de diagrama de blocos, conforme mostra a figura 15

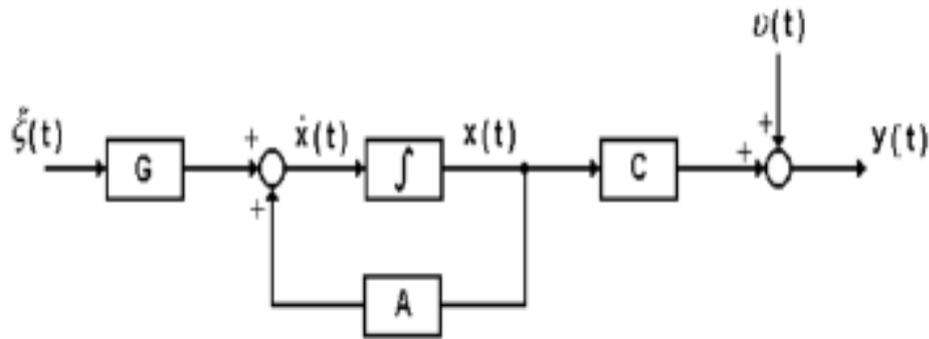


Figura 15- Diagrama de blocos do sistema Dinâmico Estocástico

Fonte: Xue, Shen e Atherton, 2007

O problema a ser resolvido consiste em obter uma estimativa $\hat{x}(t)$ do estado $x(t)$ a partir da observação da saída $\{y(\tau), \tau \leq t\}$.

O filtro de Kalman (FK) é um sistema dinâmico, com a estrutura representada na figura 16, em que a matriz de ganho L é dada por:

$$L = \sum C^T \Theta^{-1} \quad (5.25)$$

Onde Σ é a única solução simétrica definida positiva da equação de Riccati.

$$A \Sigma + \Sigma A^T + L \Theta L^T - \sum C^T \Theta^{-1} C \Sigma = 0 \quad (5.26)$$

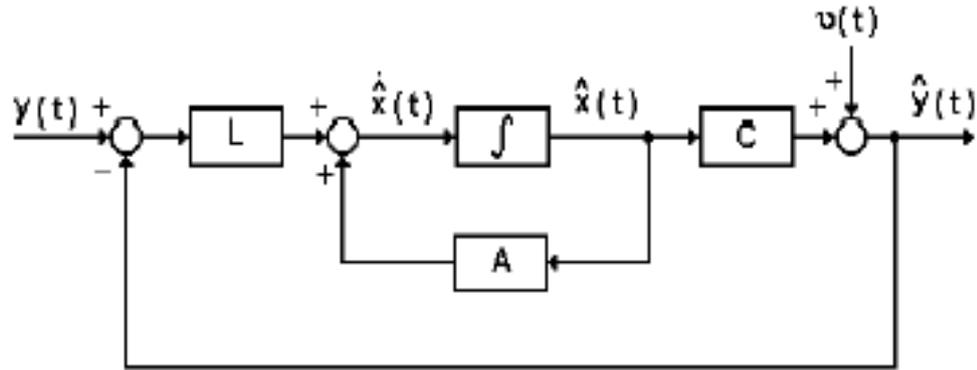


Figura 16- Diagrama em Blocos do filtro de Kalman

Fonte: Xue, Shen e Atherton, 2007

A estimativa encontrada pelo FK é ótima no sentido de que a variância do erro da estimativa $\tilde{x}(t) = x(t) - \hat{x}(t)$ é mínima, isto é:

$$\min \sum_{i=1}^n E\{[x_i(t) - \hat{x}_i(t)]^2\} \quad (5.27)$$

Sendo assim, a dinâmica do FK é dada por:

$$\dot{\hat{x}}(t) = A\hat{x}(t) + L[y(t) - C\hat{x}(t)] \quad (5.28)$$

Filtro de Kalman é tal que:

$$R_e[\lambda_i(A - LC)] < 0 \quad (i = 1, 2, 3, \dots, n) \quad (5.29)$$

O que significa que a malha objetivo é nominalmente estável.

5.4.1 Dualidade entre a Estimação e o Controle

O Filtro de Kalman e o LQR são duais (Blackmore and Bitmead 1995), o que significa que é possível obter relações análogas nos dois casos simplesmente fazendo-se a correspondência entre os parâmetros conforme tabela 1

Tabela 1 - Parâmetros duais do LQR e FK

| LQR | FK |
|-----|-------------|
| A | A^T |
| B | C^T |
| Q | $L \Xi L^T$ |
| R | Θ |
| K | Σ |
| G | H^T |

Foi verificado, então, que uma estimativa ótima do estado pode ser obtida, a partir de medidas afetadas por ruído de combinações lineares das variáveis de estado, usando um filtro. Esse filtro é constituído de um modelo do sistema com um sinal de realimentação proporcional da diferença entre a medida real e esperada.

5.5 Controlador LQG/LTR

O problema *LQG/LTR* com recuperação da malha de realimentação, é formulado como um problema de otimização que é aplicado na recuperação da malha pela entrada. As metodologias do projeto *LQG/LTR* apresentadas em (Doyle, 1979) fornecem importantes resultados teóricos, mas a abordagem é desprovida de modelos, algoritmos e procedimentos eficientes, no sentido do método de busca, ajuste de ganho de malha e sua recuperação, salvo métodos de tentativas e erros.

O projeto do Regulador Linear Quadrático (*LQR*) e o Filtro de *Kalman* (*FK*) formam o Controle Linear Quadrático Gaussiano, *LQG*, contribuindo para viabilizar as implementações em tempo real de compensadores ótimos. A força do *LQG* consiste na disponibilização de todos os estados para a realimentação ótima. A desvantagem do *LQG*, as propriedades de robustez do *LQR* são perdidas pela inserção do observador de estado. Este problema é contornado com a metodologia de projeto *LQG/LTR* ou *LTR* que recupera a robustez garantida do *LQR*. Sob o ponto de vista do objetivo de controle, estas metodologias podem ser utilizadas para alocar a Auto-estrutura em Sistemas *MIMO*. Este ponto de vista considera a fusão das metodologias *LQR* e *LQG* do controle moderno, e da metodologia *LTR* de controle robusto, (CRUZ 1996).

A Recuperação do Ganho de Malha de Realimentação na Entrada considera constante o ganho do *LQR* e, em seguida, ajusta o ganho do Filtro de Kalman de tal forma que as propriedades do *LQR* sejam recuperadas. É importante destacar que a fixação do ganho do controlador, assim como das matrizes de covariâncias Ξ da perturbação no estado e do ruído de medida iniciais, os quais serão utilizados para recuperação das propriedades de robustez, são produzidos pelos algoritmos genéticos.

Na figura 17 Considere-se apenas o problema do regulador sem estimação de estado, como ilustrado. Admitindo-se $\Phi(s) = (sI - A)^{-1}$ a função de transferência da planta é:

$$G(s) = \Phi(s)B \quad (5.29)$$

$$\Phi(s) = (sI - A)^{-1} \quad (5.30)$$

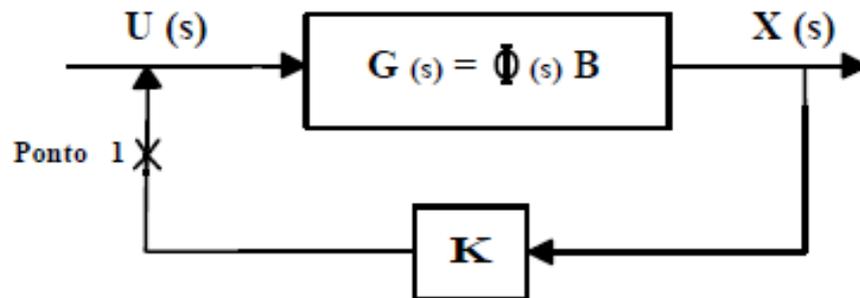


Figura 17- Esquema para realimentação de estados

Fonte: Cruz, 1996 EDUSP

Abrindo-se a malha na entrada (Ponto 1), define-se o ganho de malha aberta do LQR como sendo:

$$L_{LQR}(s) = K\phi B \quad (5.31)$$

Nesta pesquisa, um procedimento de projeto para ajustar os ganhos dos observadores é apresentado para recuperar a malha na entrada. O ajuste do ganho é baseado em modificações automática no parâmetro ν , tomando como referência as matrizes de covariâncias determinadas pelo FK , para determinar o ganho ótimo do observador de estado e recuperar as propriedades de robustez, conforme a estrutura de otimização formulada.

5.5.1 Modelo de ajuste de ganho LTR

O problema LQG/LTR para recuperação das propriedades de robustez da malha da função de transferência é formulado como um problema de otimização. O índice de desempenho é a melhor escolha que minimiza o erro da frequência da decomposição em valores da estrutura de controle dos projetos LQR e LQG . Destacam-se como objetivos a determinação do controlador $K_{LQR/LTR}$, a seleção das matrizes de ponderação Q e R do problema LQR , e a determinação do ganho do observador $L_{LQR/LTR}$. Considerando a recuperação pela entrada, o problema de LQG associado com a seleção das matrizes de covariâncias.

O uso de procedimentos por tentativa e erro para o projeto *LTR* justifica o desenvolvimento de um algoritmo que fornece um método ótimo para determinar o ganho de recuperação de malha.

O procedimento para a recuperação do ganho de malha do *LTR* na entrada mostra que escolhas apropriadas das matrizes de covariâncias é função de um parâmetro v_i , que pode aproximar a malha de controle *LTR* da malha de controle *LQR*. A abordagem de recuperação das propriedades de robustez do projeto de pesquisa, utilizando algoritmo genético e redes neurais foi motivada no artigo de (SERGI, 2012).

5.5.2 Modelo de Busca *LTR*

A seleção das matrizes (Q, R) e (Ξ, Θ) são as duas alternativas para recuperação de malha do projeto *LQR*. Na recuperação pela entrada, a busca é direcionada para as matrizes. As variações das matrizes de ponderações têm como referência as matrizes (Q_0, R_0) e a lei para esta variação é

$$Q_i = v_i^2 Q_0 C C^T, \quad R = v_i^2 R_0 \quad (5.32)$$

que é função de um escalar v_i . Na recuperação pela entrada, a busca é direcionada para o par de matrizes (Ξ, Θ) , sendo que as matrizes de covariâncias têm como referência as matrizes de covariâncias (Ξ_0, Θ_0) e uma lei que estabelece suas variações $(\Xi_i = v_i^2 \Xi_0 + B B^T, \Theta = v_i^2 \Theta_0)$ como função de um escalar v_i .

O problema é modelado como uma estrutura de otimização combinatória, permitindo a determinação do ganho do controlador $K = R^{-1} B^T P_{LQR}(Q, R)$ ou ganhos do observador de estado $L_{\Xi, \Theta}$ que recupera o ganho de malha L_{LQR} por meio do projeto $L_{LQG/LTR}$. A recuperação da malha do projeto *LQR* é realizada no domínio da frequência. Uma vez o sistema é aberto no Ponto 1, Figura 17, a função de transferência de malha aberta é dada por:

$$L(s)_{LTR}^{input} = F(s)G(s) = K\Phi, LC\Phi B, \quad (5.33)$$

onde $L(s)_{LTR}^{input}$ é a função de transferência na entrada.

O problema é formulado de maneira que se determine os ganhos $L_{\Xi, \Theta}$ do observador de estado o qual recupera a malha da função de transferência $L(s)^{LQR}$, suportado por uma estrutura de otimização combinatória,

$$\min \sum_{L_{\Xi, \Theta}} \left(\sigma_i^{L_{LQR}} - \sigma_i^{K\Phi_r LC\Phi B} \right) \quad (5.34)$$

$$s.a \quad |L_{\Xi, \Theta}| \leq \varepsilon$$

$$\lambda_{i, esq.} \leq \lambda_{calc.} \leq \lambda_{i, dir.}$$

$$S_i \leq \varepsilon, i = 1, \dots, n.$$

onde $|L_{\Xi, \Theta}| \leq \varepsilon$ são restrições operacionais do problema.

A solução da recuperação do projeto *LTR* na entrada é dada pela seleção do par de matrizes de covariâncias, que é realizada pelo modelo de busca $LTR_{\Xi, \Theta}$, sendo que essas matrizes constituem os parâmetros matriciais da equação algébrica de *Riccati*, $A\Sigma + \Sigma A^T + G(\nu^2 \Xi_0 + BB^T)G^T - \Sigma C^T (\nu^2 \Theta_0)^{-1} C \Sigma = 0$ sendo ν é o parâmetro escalar usado para ajustar as matrizes de covariâncias. A solução da equação algébrica de Riccati é utilizado para computar os ganhos do observador

$$L_{LTR} = \Sigma C^T (\nu^2 \Theta_0)^{-1} \quad (5.35)$$

onde L_{LTR} são os ganhos do observador do *Filtro de Kalman*.

5.5.3 Desempenho *AG-LQG/LTR*

O desempenho do projeto do *AG-LQG/LTR* é verificado, comparando-se os valores singulares do projeto *LQR* e do projeto *LTR*. Analisa-se os efeitos de variações de parâmetros, tais como semente do gerador pseudo-aleatório, condições iniciais, probabilidade de mutação e tamanho da população no desempenho do *AG-LQG/LTR*, bem como os efeitos de variações na semente do gerador pseudo-aleatório no comportamento da população inicial. Verifica-se também o desempenho da população de cada geração em relação às variações nas matrizes de covariâncias iniciais, a influência da probabilidade de mutação na diversidade genética da população e quantidade de indivíduos mutados.

Comparando-se os valores singulares do projeto *LQR* e do projeto *LTR*. A implementação do algoritmo considera uma população composta por quatorze indivíduos e um ciclo de busca de sete gerações. O parâmetro de operação de *crossover* tem como valores 0.1 e 0.05. A operação de mutação tem probabilidade de ocorrência $probmut = 5\%$.

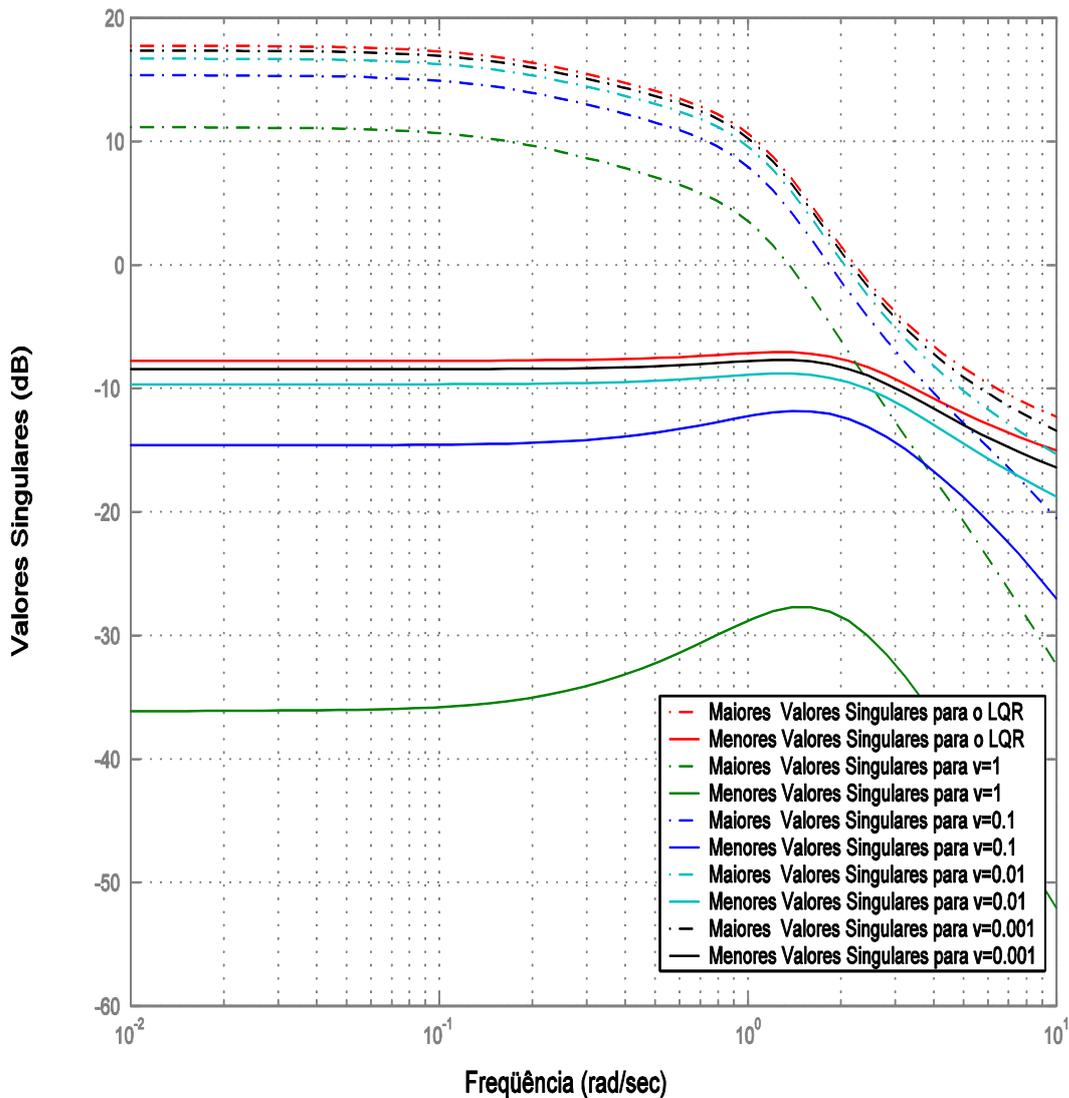


Figura 18- Valores singulares da recuperação

A Figura 18 ilustra os valores singulares da recuperação das propriedades de robustez do projeto *LQR* via projeto *LTR*, usando algoritmo genético. Verificou-se que quando $\nu \rightarrow 0$, as propriedades de robustez perdidas com a estimação, são completamente recuperadas.

Segundo (LEWIS, 1995), os limites para garantir as propriedades de robustez são dadas em termos dos valores singulares mínimos, sendo grandes nas baixas frequências (para desempenho da robustez) e os valores singulares máximos são pequenos nas altas frequências (para estabilidade de robustez).

6 MODELO HIERÁRQUICO PARA SINTONIA

A proposta de um Modelo Hierárquico de Algoritmos Genéticos para o projeto de Controladores LQR , LQG e LTR com modelos de otimização combinatória que é resolvido por computação evolutiva para ajustes dos ganhos das malhas. Na referência (Fonseca Neto 2000), o problema de AAE é modelado em estruturas de otimização para selecionar as matrizes Q e R , este modelo foi estendido por (Ferreira 2003) para a alocação de auto estrutura no projeto LQG . O modelo Hierárquico atua no sentido de garantir às propriedades de robustez dos projetos LQR e LQG e para garantir o melhor desempenho robusto dos sistemas de controle baseado nestas metodologias. Além de apresentar o Modelo Hierárquico de Algoritmos para recuperação da malha, apresenta-se o modelo de um algoritmo genético para Recuperação de malha utilizando metodologia de Projeto LTR para recuperar as propriedades de robustez do LQR . Esta metodologia, por meio de um procedimento assintótico, recupera as características de resposta em frequência dessa malha ajustando um determinado parâmetro, normalmente por tentativa e erro. Para contornar este problema, propõe-se um ajuste inteligente para as matrizes de covariância para Recuperação do Ganho da Malha na Entrada ou por meio de variações nas ponderações Q e R do LQR para Recuperação na Saída.

O AG Hierárquico consiste de uma estratégia de busca das matrizes de ponderações e covariâncias a fim de alocar autoestrutura sem sistemas dinâmicos multivariáveis.

No *Modelo Hierárquico para Sintonia LQR , LQG e LTR* , apresentam-se os modelos de busca que são utilizados para sintonizar os ganhos das malhas LQR , LQG e LTR . Em todos os modelos de sintonia mostra-se que a hierarquia coordena a modelagem inteligente para o ajuste de ganho, atuando em uma segunda camada de otimização, diferente da otimização de controle ótimo e robusto.

O Modelo Hierárquico para Sintonia atua na Recuperação da Malha em Controladores LQG/LTR . Estes modelos estão fundamentados em otimização combinatória para sintonizar as malhas dos Sistemas de Controle LQR , LQG e LTR . Estes modelos são dedicados a seleção de pares de matrizes (Q,R) e (Ξ, Θ) dos índices de desempenho dos projetos LQR e LQG , respectivamente, e a seleção das matrizes de ajuste para o ganho de recuperação de malha do projeto LTR . As estruturas de otimização combinatória permitem o desenvolvimento de métodos baseados em algoritmos genéticos para realizar busca destas matrizes nas instâncias de projetos LQR , LQG e LTR .

6.1 Modelo de Sintonia do Ganho LQR

O Modelo para Sintonia dos Ganhos LQR enfoca a seleção das matrizes de ponderação $(Q;R)$. O problema é modelado como um mapeamento do espaço de matrizes $((Q;R)\mathbb{R})$ em um controlador $(K_{Q,R}, \mathbb{R})$ que aloca uma determinada autoestrutura. Um modelo de otimização combinatória é construído por Fonseca, (FONSECA NETO 2000), para resolver o problema de Alocação de Autoestruturas em Sistemas Dinâmicos Multivariáveis. O método proposto por Fonseca Neto é uma alternativa para alocar a Auto-estrutura através do LQR , superando as buscas por *tentativa e erro* do projetista para determinar as matrizes de ponderação.

6.2 Modelo de Sintonia do Ganho LQG

O Modelo para Sintonia dos Ganhos LQG enfoca a seleção das matrizes de covariâncias (Ξ, Θ) . Um método para o problema de AAE de Estimador de Estado Estocástico via FK e AG estão publicados em (FERREIRA, 2003), o projeto do filtro de *Kalman* leva em consideração a alocação da Auto-estrutura no problema de estimação de estado estocástico. Esta formulação permite a utilização de técnicas de Computação Evolutiva para determinar um observador estocástico.

6.3 Modelo de Sintonia do Ganho LTR

O problema LQG/LTR com recuperação de malha de realimentação é formulado como um problema de otimização que pode ser aplicado tanto para as recuperações das malhas de saída ou de entrada. Os objetivos são: a) encontrar as matrizes de ponderação Q e R do problema LQR , determinando o controlador $KLQR=LTR$ com recuperação pela saída, b) encontrar as matrizes de covariância do problema LQG , determinando o ganho do observador $LQG=LTR$ com recuperação pela entrada. A utilização de procedimentos por tentativa e erro para o projeto (LTR) justifica o desenvolvimento de um algoritmo que fornece o ganho da recuperação de malha. O ganho do controlador K , fixado no procedimento, é determinado por meio da metodologia desenvolvida por (FONSECA NETO, 2000) e (FONSECA NETO 2003).

O procedimento para recuperação do ganho de malha de realimentação (LTR) na entrada, (FERREIRA e FONSECA *et al.* 2003), mostra devidamente a escolha das matrizes

de covariâncias. Esta escolha é função de variações de um parâmetro v_i que aproxima a malha de controle *LTR* para a malha de controle *LQR*. Porém, a metodologia aplicada faz uso do método de tentativa e erro para determinação do parâmetro v_i .

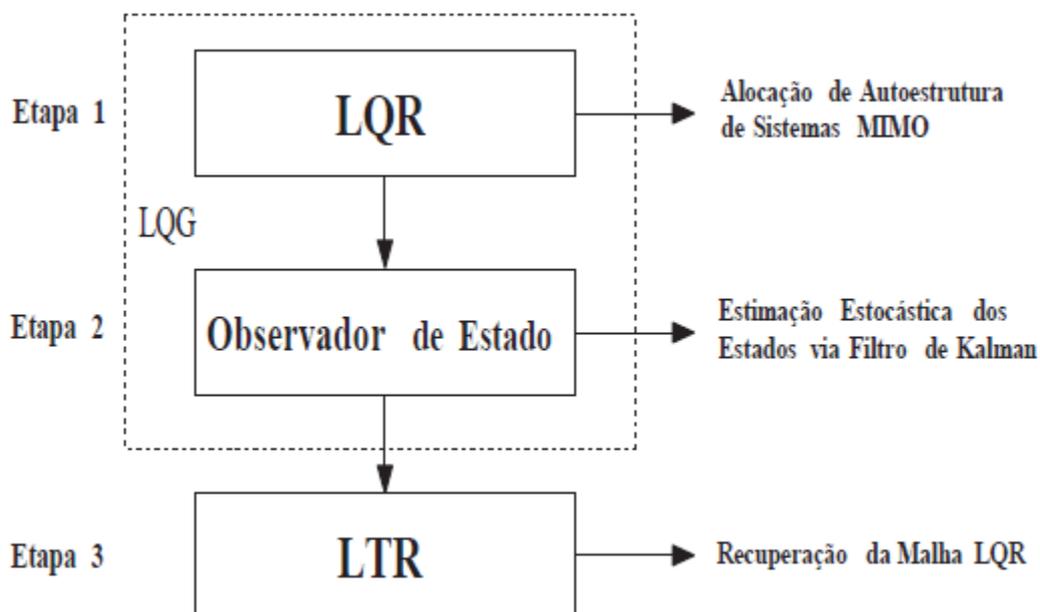


Figura 17: Estrutura hierárquica de algoritmos para recuperação da malha LQR
Fonte: RÊGO, Patrícia, 2007

7 SISTEMA DE CONTROLE ROBUSTO

Neste Capítulo apresenta-se a aplicação de uma metodologia de controle robusto, conhecida como LQG/LTR (do Inglês, *Linear Quadratic Gaussian/Loop Transfer Recovery*).

Os sistemas físicos e o ambiente externo no qual eles operam não podem ser modelados precisamente, podem mudar de uma maneira imprevisível e estar sujeitos a perturbações significantes. O projeto de sistemas de controle na presença de incertezas significantes requer que o projetista almeje um sistema robusto (*DORF*, 2012). O modelo do processo sempre será uma representação inexata do sistema físico real devido:

- Variações nos parâmetros;
- Dinâmica não modelada;
- Atrasos de transporte não modelados;
- Alterações no ponto de equilíbrio;
- Ruído do sensor;
- Entradas de perturbação imprevistas.

O objetivo do projeto de sistemas robustos é a garantia de desempenho do sistema apesar das inexatidões do modelo e variações. Um sistema é robusto quando ele apresenta variações no desempenho devido a variações ou inexatidões do modelo. A robustez para sistemas de controle é desejada à medida que se espera uma boa operação do controlador em caso de variações paramétricas e possíveis distúrbios aplicados ao sistema.

O projetista procura obter um sistema que apresente um desempenho adequado sobre uma grande faixa de parâmetros incertos. Um sistema é dito como robusto quando é durável, resistente e resiliente.

Um sistema de controle é robusto quando:

- 1º) Ele tem sensibilidades pequenas;
- 2º) Ele é estável sobre a faixa de variação de parâmetros;
- 3º) O desempenho continua a atender às especificações na presença de um conjunto de variações nos parâmetros do sistema.

7.1 Sensibilidade do sistema

Então Robustez é a baixa sensibilidade aos efeitos que não são considerados nas fases de análise e de projeto.

Perturbações, ruído de medida e dinâmica não modelada. O sistema deve ser capaz de resistir a estes efeitos que foram desprezados quando executar as tarefas para as quais ele foi projetado. Para pequenas perturbações paramétricas, pode-se usar, como uma medida de robustez, as sensibilidades diferenciais. A sensibilidade do sistema é definida como:

$$S_{\alpha}^T = \frac{\partial T/T}{\partial \alpha/\alpha} \quad (7.1),$$

- Em que α é o parâmetro e T é a função de transferência do sistema. A sensibilidade da raiz é definida como:

$$S_{\alpha}^{r_i} = \frac{\partial r_i}{\partial \alpha/\alpha} \quad (7.2)$$

- Quando os zeros de $T(s)$ são independentes do parâmetro α , mostrou-se que:

$$S_{\alpha}^T = - \sum_{i=1}^n S_{\alpha}^{r_i} \cdot \frac{1}{s + r_i}, \quad (7.3)$$

para sistema de n -ésima ordem.

7.2 Matriz de sensibilidade S e matriz de sensibilidade complementar T :

A matriz de sensibilidade é definida como a função de transferência entre d_i e u_p para o caso em que a malha KG é obtida abrindo-se a entrada da planta e como a função de transferência entre d e y para o caso em que a malha GK é obtida abrindo-se a saída da planta, conforme figura 19.

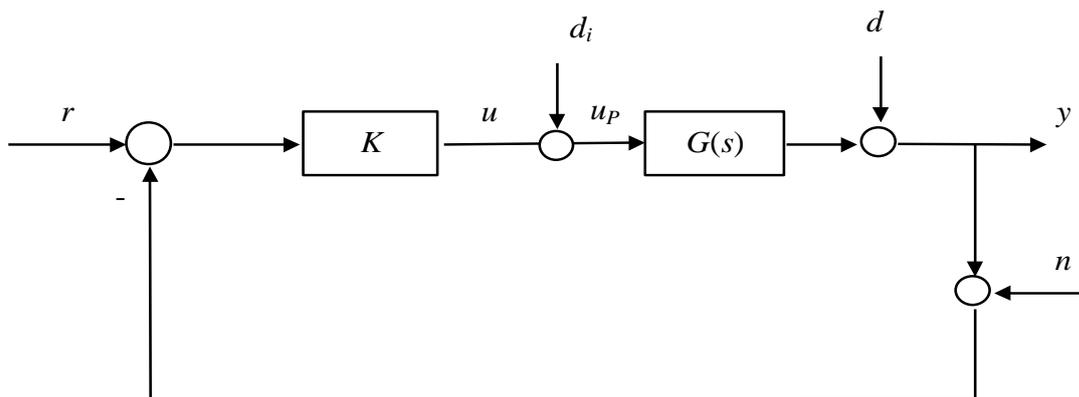


Figura 19- Diagrama de blocos do sistema com perturbações

Fonte: Cruz, 1998, EDUSP

r : entradas de referência;
 d_i : perturbações entrada planta;
 d : perturbações saída da planta;
 n : ruído de sensores.

Seja o sistema de realimentação padrão no domínio da frequência mostrado na Figura 18 A planta é $G(s)$ e $K(s)$ é o compensador retroalimentado, podendo ser projetada utilizando técnicas do controle moderno. A saída da planta é $z(t) \in \mathbb{R}^q$, o controle de entrada da planta é $u(t) \in \mathbb{R}^m$ e a entrada de referência é $r(t) \in \mathbb{R}^q$.

Pode se observar que rastreamento perfeito pode não ser alcançado a menos que o número m das entradas de controle $u(t)$ seja maior ou igual ao número q das saídas de desempenho $z(t)$. Portanto, assume-se que $m = q$ para que a planta $G(s)$ e o compensador $k(s)$ sejam quadráticos. Isso é apenas uma consequência do projeto prático, e não uma restrição sobre os tipos de plantas que podem ser consideradas.

Observa-se que nessa Figura 18 foram adicionados alguns itens para caracterizar as incertezas. O sinal $d(t)$ representa uma perturbação que atua direto sobre o sistema, típico no que aparece no controle clássico. Isso poderia representar, por exemplo, rajadas de vento. O ruído ou erros de medição do sensor são representados por $n(t)$. Ambos os sinais são geralmente vetores de dimensão q . Tipicamente, as perturbações ocorrem em baixas frequências, abaixo de algum ω_d , enquanto o ruído de medida $n(t)$ tem seu efeito predominante em altas frequências, acima de algum valor ω_n .

O erro de rastreamento é dado por:

$$e(t) = r(t) - z(t). \quad (7.4)$$

Devido a presença de ruído na medida, o sinal $s(t)$ pode ser representado pela equação:

$$s(t) = r(t) - z(t) - n(t) = e(t) - n(t). \quad (7.5)$$

Na saída da planta:

$$z(t) = G(s)K(s)s(t) + d(t). \quad (7.6)$$

Analisando o sistema no domínio de frequência para verificação dos efeitos das incertezas sobre o desempenho do sistema e aplicando a transformada de Laplace nas equações (7.4), (7.5) e (7.6) têm-se:

$$E(s) = R(s) - Z(s) \quad (7.7)$$

$$S(s) = R(s) - Z(s) - N(s) \quad (7.8)$$

$$Z(s) = G(s)K(s)S(s) + D(s) \quad (7.9)$$

Substituindo a equação (7.8) na equação (7.9) têm-se:

$$Z(s) = G(s)K(s)[R(s) - Z(s) - N(s)] + D(s) \quad (7.10)$$

$$Z(s) + G(s)K(s)Z(s) = G(s)K(s)R(s) - G(s)K(s)N(s) + D(s). \quad (7.11)$$

Por simplicidade, denota-se $G(s)K(s)$ por GK , e substituindo na equação (7.10) (7.11) têm-se:

$$Z(s) + GKZ(s) = GKR(s) - GKN(s) + D(s)$$

$$Z(s)[I + GK] = GK[R(s) - N(s)] + D(s)$$

$$Z(s) = [I + GK]^{-1} GK[R(s) - N(s)] + [I + GK]^{-1} D(s). \quad (7.12)$$

Substituindo a equação (7.12) na equação (7.7) têm-se:

$$E(s) = R(s) - [(I + GK)^{-1} GK(R(s) - N(s)) + (I + GK)^{-1} D(s)] \quad (7.13)$$

$$E(s) = R(s) - (I + GK)^{-1} GKR(s) + (I + GK)^{-1} GKN(s) - (I + GK)^{-1} GKD(s) \quad (7.14)$$

Logo o erro $E(s)$ em função das entradas $R(s)$, $D(s)$ e $N(s)$ pode ser representado pela seguinte equação:

$$E(s) = [I - (I + GK)^{-1} GK] R(s) + (I + GK)^{-1} GKN(s) - (I + GK)^{-1} GKD(s). \quad (7.15)$$

De acordo com o lema da inversão de matrizes [17] a equação (3.28) é dada por:

$$E(s) = (I + GK)^{-1} GK(R(s) - D(s)) + (I + GK)^{-1} GKN(s). \quad (7.16)$$

Sendo GK quadrada e inversível, então pode-se escrever a identidade como:

$$\begin{aligned} (I + GK)^{-1} GK &= [(GK)^{-1}(I + GK)^{-1}]^{-1} = [(GK)^{-1} + I]^{-1} \\ &= [(GK)^{-1}(I + GK)]^{-1} = GK[(I + GK)]^{-1}. \end{aligned} \quad (7.17)$$

Após realizar as devidas substituições, as equações que representam respectivamente a saída $Z(s)$ e o erro $E(s)$ são dadas por:

$$Z(s) = GK(I + GK)^{-1}(R(s) - N(s)) + GK(I + GK)^{-1} D(s) \quad (7.18)$$

$$E(s) = (I + GK)^{-1}(R(s) - D(s)) + GK(I + GK)^{-1} N(s). \quad (7.19)$$

Portanto, para o sistema da Figura 19, o ganho de malha é dado por $G(s)K(s)$. As equações que representam a função sensibilidade e sensibilidade complementar são respectivamente dadas por:

$$S(s) = (I + GK)^{-1} \quad (7.20)$$

$$T(s) = GK(I + GK)^{-1} = (I + GK)^{-1} GK \quad (7.21)$$

desde que

$$S(s) + T(s) = (I + GK)(I + GK)^{-1}. \quad (7.22)$$

Note-se que o inverso da função sensibilidade é dado por:

$$L(s) = I + GK. \quad (7.23)$$

A saída e o erro do sistema podem agora ser expresso em termos da função sensibilidade e sensibilidade complementar representadas respectivamente pelas equações:

$$Z(s) = T(s)(R(s) - N(s)) + S(s)D(s) \quad (7.24)$$

$$E(s) = S(s)(R(s) - D(s)) + T(s)N(s). \quad (7.25)$$

Para garantir pequenos erros de rastreamento, deve-se ter $S(j\omega)$ pequena nas frequências ω onde a entrada de referência $r(t)$ e a perturbação $d(t)$ são grandes. Produzindo assim, uma boa rejeição de perturbação. Por outro lado, para uma rejeição satisfatória do ruído do sensor, deve-se ter $T(j\omega)$ pequena nas frequências ω onde $n(t)$ é grande. Infelizmente $S(j\omega)$ e $T(j\omega)$ não podem ser simultaneamente pequenos em qualquer frequência ω .

7.2.1 Rejeição de perturbações

As seguintes condições para rejeição de perturbações podem ser estabelecidas particularmente em baixas frequências onde os sinais d e d_i são normalmente significativos.

1. Rejeição a perturbações d e d_i na saída da planta y : pequenos valores para máximos de S e SG , com

2.

$$S = \frac{1}{(1+GK)} \quad (7.26)$$

2. Rejeição a perturbações d e d_i na entrada da planta u_p : pequenos valores para máximos de S e SK .

3. Observa-se que:

$$GK \gg 1 \Leftrightarrow SG \approx 1/K \quad (7.27)$$

$$GK \gg 1 \Leftrightarrow KS \approx 1/G \quad (7.28)$$

Rejeição de perturbação d e d_i na saída y :

$$GK \gg 1, \quad e \\ K \gg 1$$

Rejeição de perturbação d e d_i na entrada u_p

$$GK \gg 1, \quad e \\ G \gg 1$$

7.2.2 Rejeição de erro de medida

Pode-se observar agora que existe um conflito entre rejeição de perturbações d e d_i e rejeição de erro de medida (tipicamente grande em alta frequência)

$$y = T(r - n) + SPd_i + Sd_i \\ \approx r - n \text{ para } |KG| \gg 1 \quad (7.29)$$

Esta situação é piorada fora da faixa passante da planta: $|KG| \gg 1$ e $|G| \ll 1$ quando u torna-se inaceitável, podendo provocar saturação do atuador, desde que:

$$\begin{aligned} u &= KS(r - n - d) - Td_i = SK(r - n - d) - Td_i \\ &\approx G^{-1}(r - n - d) - d_i \end{aligned} \quad (7.30)$$

Amplificam-se, portanto as perturbações e erros de medida uma vez que nesta faixa de frequência $|G| \ll 1$. Pelo mesmo motivo não se pode fazer $|K|$ muito grande onde $|GK| \ll 1$, pois,

$$u = KS(r - n - d) - Td_i = K(r - n - d) \quad (7.31)$$

7.3 Análise de Robustez

Considere o sistema em malha fechada mostrado na figura 19 os objetivos do sistema incluem manter um pequeno erro de rastreamento para uma entrada $R(s)$ e conservar a saída $Y(s)$ pequena para uma perturbação $T_p(s)$.

A função sensibilidade é:

$$S(s) = [1 + G_c(s)G(s)]^{-1} \quad (7.32)$$

A função sensibilidade complementar é:

$$C(s) = \frac{G_c(s)G(s)}{1 + G_c(s)G(s)} \quad (7.33)$$

Tem se também a relação

$$S(s) + C(s) = 1 \quad (7.34)$$

Para sistemas fisicamente realizáveis o ganho em malha aberta $L(s) = G_c(s)G(s)$ deve ser pequena para altas frequências. Isto significa que $S(j\omega)$ tende a 1 em altas frequências.

7.3.1 Perturbação aditiva

$$G_a(s) = G(s) + A(s) \quad (7.35)$$

Em que $G(s)$ é o processo nominal e $A(s)$ é a perturbação que é limitada em magnitude. Admita que $G_a(s)$ e $G(s)$ possuem o mesmo número de polos no semiplano direito do plano s . então a estabilidade do sistema não mudará se

$$|A(j\omega)| < |1 + G(j\omega)| \text{ para todo } \omega$$

Isto garante a estabilidade mas não o desempenho dinâmico

7.3.2 Perturbação multiplicativa

$$G_m(s) = G(s)[1 + M(s)] \quad (7.36)$$

A perturbação é limitada em magnitude, e admite-se novamente que $G_m(s)$ e $G(s)$ possuem o mesmo número de polos no semiplano direito do plano s .

Então a estabilidade do sistema não mudará se:

$$|M(j\omega)| < \left| 1 + \frac{1}{G(j\omega)} \right| \quad (7.37)$$

Para todo ω . A equação é chamada de o critério de estabilidade robusta.

Essa forma de perturbação é frequentemente usada porque ela satisfaz as propriedades intuitivas de:

- 1º- Ser pequena em baixas frequências nas quais o modelo nominal do processo é usualmente bem conhecido;
- 2º- Ser grande em altas frequências onde o modelo nominal é sempre inexato.

7.4 Barreira de robustez

Os requisitos de acompanhamento do sinal de referência, rejeição de perturbações e insensibilidade a variações na planta ($g_N(j\omega)k(j\omega)$ "grande") são conflitantes com a rejeição do erro de medida (que requer $g_N(j\omega)k(j\omega)$ "pequena"). Assim o diagrama de Bode do ganho da função de transferência de malha deve estar situado entre as barreiras de baixa e alta frequência, conforme representado na figura 19

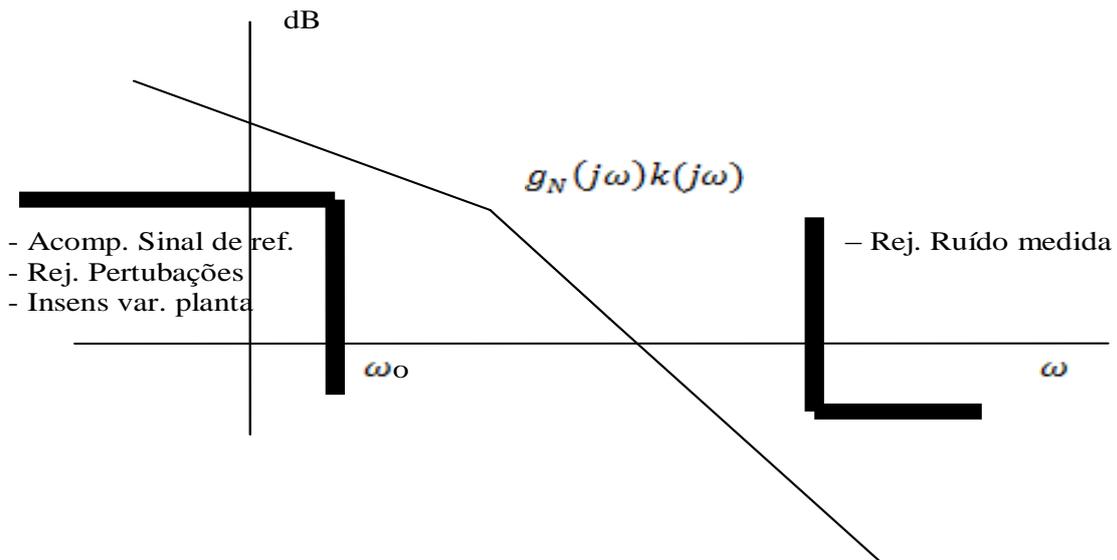


Figura 19- Barreiras de robustez de baixa e alta frequências

Fonte: Cruz, 1998, EDUSP

7.5 Modelo de redução e estabilidade Robusta

Em algumas situações, temos um modelo de alta ordem que é inconveniente usar para projeto do controlador. Exemplos ocorrem no controle do motor e no controle da nave espacial e etc. Em tais situações, é possível calcular um modelo de ordem reduzida do sistema, que pode então ser usado para o projeto do controlador. Aqui mostra-se uma técnica conveniente para redução de modelos bem como a estabilidade de robustez vinculada $m(\omega)$. A técnica descrita aqui é de Athans et al. (1986).

7.5.1 Redução do Modelo por Expansão de Fração Parcial

Suponha que a planta real seja descrita por:

$$\dot{x} = Ax + Bu \quad (7.38)$$

$$z = Hx \quad (7.39)$$

Com $x \in R^n$. Se A com Autovalores λ_i , Autovetores a direita u_i e Autovetores a esquerda v_i então temos que:

$$Au_i = \lambda_i u_i, \quad v_i^T A = \lambda_i v_i^T \quad (7.40)$$

Então a função de transferência,

$$G'(s) = H(sI - A)^{-1}B \quad (7.41)$$

Deve ser escrita em expansão de fração parcial

$$G'(s) = \sum_{i=1}^n \frac{R_i}{s - \lambda_i} \quad (7.42)$$

Com resíduos das matrizes dado por:

$$R_i = H u_i v_i^T B \quad (7.43)$$

Se o valor de n for grande, pode ser desejável encontrar uma aproximação de ordem reduzida para 7.38 para o qual um compensador simplificado $K(s)$ na Fig. 19 pode ser projetado. Então, se a aproximação é boa, o compensador $K(s)$ deve funcionar bem quando usado em uma planta real $G'(s)$.

Para encontrar uma aproximação $G(s)$ de ordem reduzida para a planta, podemos proceder da seguinte forma:

Decida qual dos autovalores λ_i em (7.42) deve ser retido em $G(s)$. Isso pode ser feito usando julgamento de engenharia, ao omitir modos de alta freqüência, ao omitir termos em (7.42) que têm pequenos resíduos, e assim por diante.

Deixe que os valores reais retidos em $G(s)$ sejam $\lambda_1, \lambda_2, \dots, \lambda_r$.

$$\text{Defina a matriz } Q = \text{diag}\{Q_i\} \quad (7.44)$$

Onde Q é uma matriz $r \times r$ e os blocos Q_i são definidos por:

$$Q_i = \left\{ \begin{array}{ll} 1, & \text{Para cada autovalor real retido} \\ \begin{bmatrix} \frac{1}{2} & -\frac{j}{2} \\ \frac{1}{2} & \frac{j}{2} \end{bmatrix}, & \text{Para cada par complexo retido} \end{array} \right\} \quad (7.45)$$

As matrizes são Calculadas por:

$$V \equiv Q^{-1} \begin{bmatrix} v_i^T \\ \vdots \\ v_r^T \end{bmatrix} \quad (7.46)$$

$$U \equiv [u_i^T \dots u_r^T] Q \quad (7.47)$$

Em termos dessas construções, o sistema de ordem reduzida não passa de uma projeção em um espaço de dimensão r com estado definido por:

$$w = Vx \quad (7.48)$$

As matrizes do sistema de ordem reduzida aproximadas do sistema

$$\dot{w} = Fw + Gu \quad z = Jw + Du \quad (7.49)$$

$$F = VAU \quad G = VB \quad J = HU \quad (7.50)$$

Com a matriz de alimentação direta dada em termos dos resíduos dos autovalores negligenciados por:

$$D = \sum_{i=r+1}^n -\frac{R_i}{\lambda_i} \quad (7.51)$$

A motivação para selecionar essa matriz D é a seguinte:

A função de transferência,

$$G(s) = J(sI - F)^{-1}G + D \quad (7.52)$$

A redução do sistema 7.50 é dada por:

$$G(s) = \sum_{i=1}^r \frac{R_i}{s - \lambda_i} + \sum_{i=r+1}^n -\frac{R_i}{\lambda_i} \quad (7.53)$$

Avaliando $G(j\omega)$ e $G'(j\omega)$ em $\omega = 0$, é visto que eles são iguais. Portanto, os erros de modelagem induzidos pela tomada de $G(s)$ ao invés do $G'(s)$ ocorrem em maior frequências. Na verdade, eles dependem das frequências dos autovalores negligenciados da equação de estado.

Para determinar o $M(s)$ que é induzido pela redução da ordem, observe que:

$$G' = (I + M)G \quad (7.54)$$

Então,

$$M = (G' - G)G^{-1} \quad (7.55)$$

ou,

$$M(s) = \left[\sum_{i=r+1}^n -\frac{R_i}{\lambda_i} \frac{s}{s - \lambda_i} \right] G^{-1}(s) \quad (7.56)$$

Em seguida, o limite de robustez de alta frequência é dado em termos de:

$$M(j\omega) = \bar{\sigma}(M(j\omega)) \quad (7.57)$$

Observe que $M(j\omega)$ tende a zero quando ω se torna pequeno, refletindo a nossa certeza perfeita da planta real.

8 APLICAÇÃO NEURO-GENÉTICA NA PLANTA DEXTEROUS HAND MASTER – DHM (MÃO MESTRE)

8.1 Função de transferência e equações de estado

O sistema DHM tem como objetivo realizar a leitura do ângulo das articulações da mão humana através do dispositivo mestre. Desta forma os sinais gerados nos sensores são enviados para dispositivos de controle que irão executar os movimentos no dispositivo escravo de modo que seja idêntico ao movimento da mão do operador.

O objetivo do controle é fazer com que os ângulos das articulações da mão robótica escravo sejam iguais aos da mão do operador. O diagrama de blocos do sistema DHM é mostrado na figura 20.

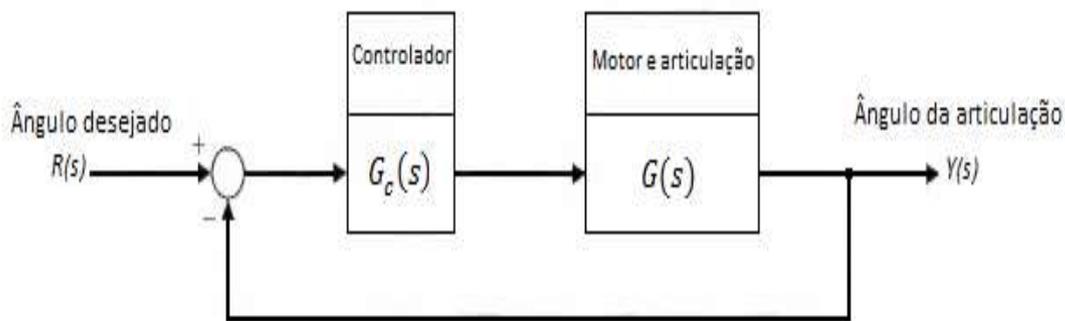


Figura 20- Diagrama de blocos do sistema DHM

Fonte: Adaptado de (DORF; BISHOP, 2010)

Onde o modelo matemático $G(s)$, ou seja, a função de transferência da articulação do DHM é dado por:

$$G(s) = \frac{1}{s(s+5)(s+10)} \quad (8.1)$$

Na forma de matriz vetorial, a equação de estado, $\dot{x} = Ax + Bu$ torna-se para o projeto de controle robusto LQG/LTR, como visto nos capítulos anteriores, é necessário que representação do sistema seja feita em espaços de estados. Então a equação de estados é dada por:

Na forma de matriz vetorial, a equação de estado torna-se: $\dot{x} = Ax + Bu$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -15 & -50 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} u(t) \quad (8.2)$$

E a equação de saída:

$$y = [0 \quad 0 \quad 1] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + [0]u(t) \quad (8.3)$$

Com a representação do sistema em espaço de estados, antes de partirmos para o projeto da malha alvo LQR, precisamos verificar se o sistema é controlável e observável, como mostrado no anexo B. Verifica-se que o sistema é controlável através da matriz de controlabilidade, que é dada por:

$$\begin{bmatrix} 1 & -15 & 175 \\ 1 & 0 & -15 \\ 0 & 1 & 0 \end{bmatrix}$$

Onde n é a ordem do sistema, então temos para o nosso problema que $n=3$, logo, a matriz de controlabilidade resultante possui $\text{ranque} = 3$, logo o nosso sistema é controlável.

Det = 1

Verifica-se agora se o sistema é observável, temos então a matriz de observabilidade dada por:

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

A matriz de observabilidade resultante possui $\text{ranque} = n = 3$, logo o sistema é, também, observável. Det = -1

8.2 Desempenho da população inicial

O perfil da população inicial pode ser definido pelos valores da função objetivo apresentados na Figura 21. Percebe-se claramente uma diversidade genética, sendo baixa entre o vigésimo quinto e o vigésimo sétimo indivíduo e relativamente satisfatória para os outros indivíduos. Essas conclusões a respeito do processo de inicialização da população inicial, implicam em uma população satisfatória no processo de busca do indivíduo QR ótimo.

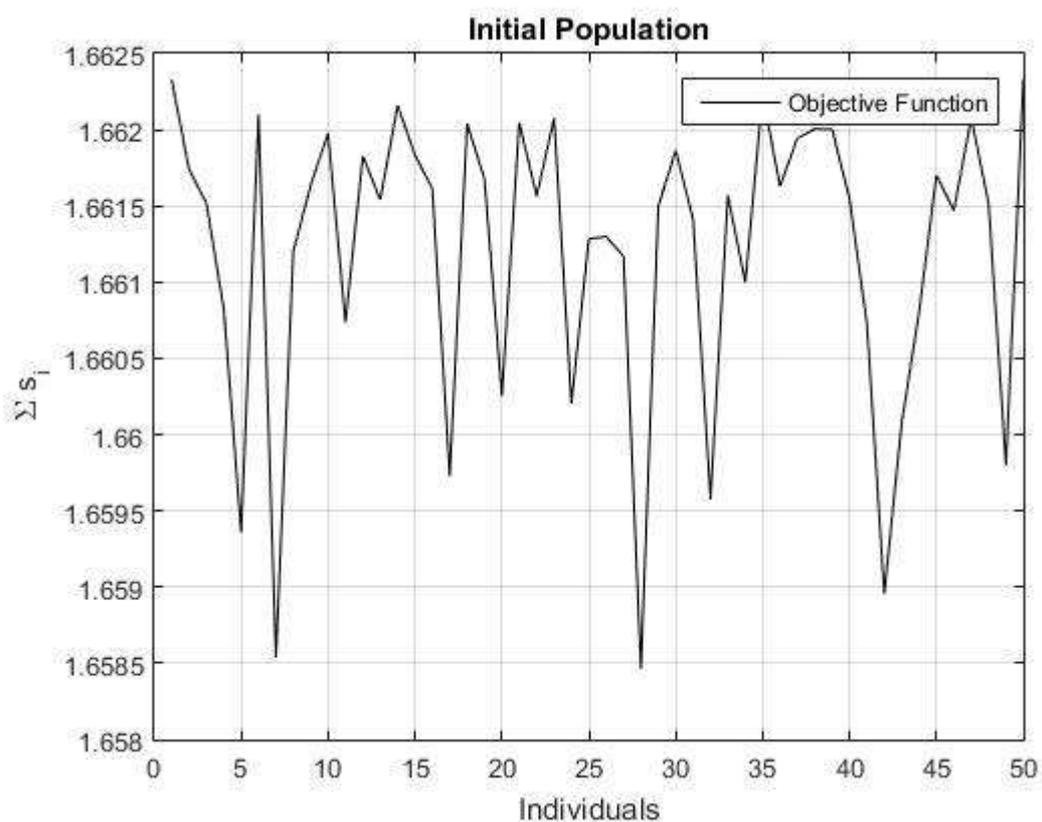


Figura 21- Sensibilidade total de cada indivíduo

Fonte: Autor, 2017

Nas figuras 22, 23 e 24 ilustram as sensibilidades da população inicial S_1 , S_2 e S_3 , referentes a 50 indivíduos do processo de geração inicial e a sensibilidade total a cada autovetor alocado no intervalo do algoritmo

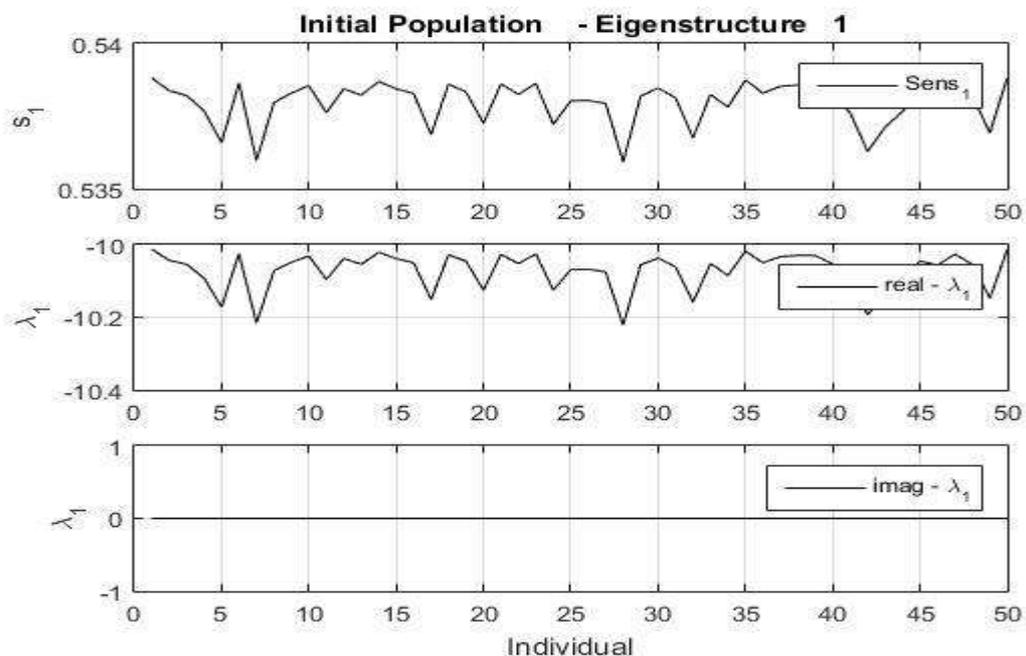


Figura 22- Sensibilidade 1 da população inicial

Fonte Autor, 2017

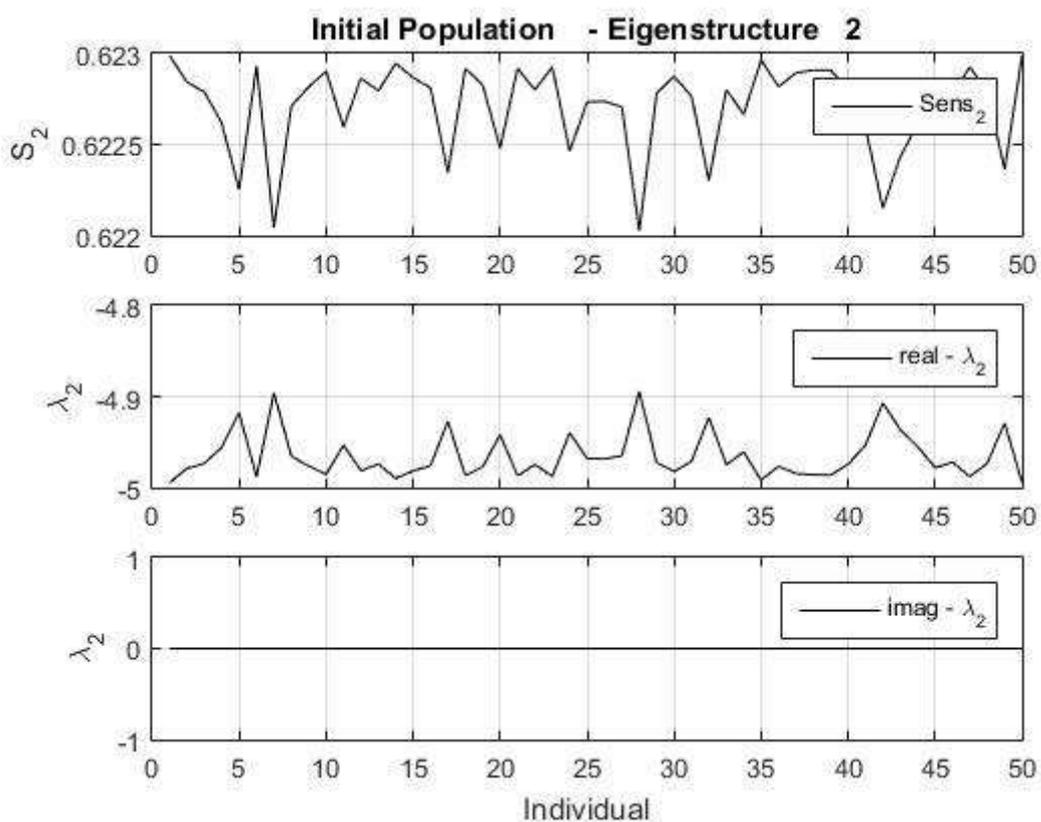


Figura 23- Sensibilidade 2 da população inicial
 Fonte Autor, 2017

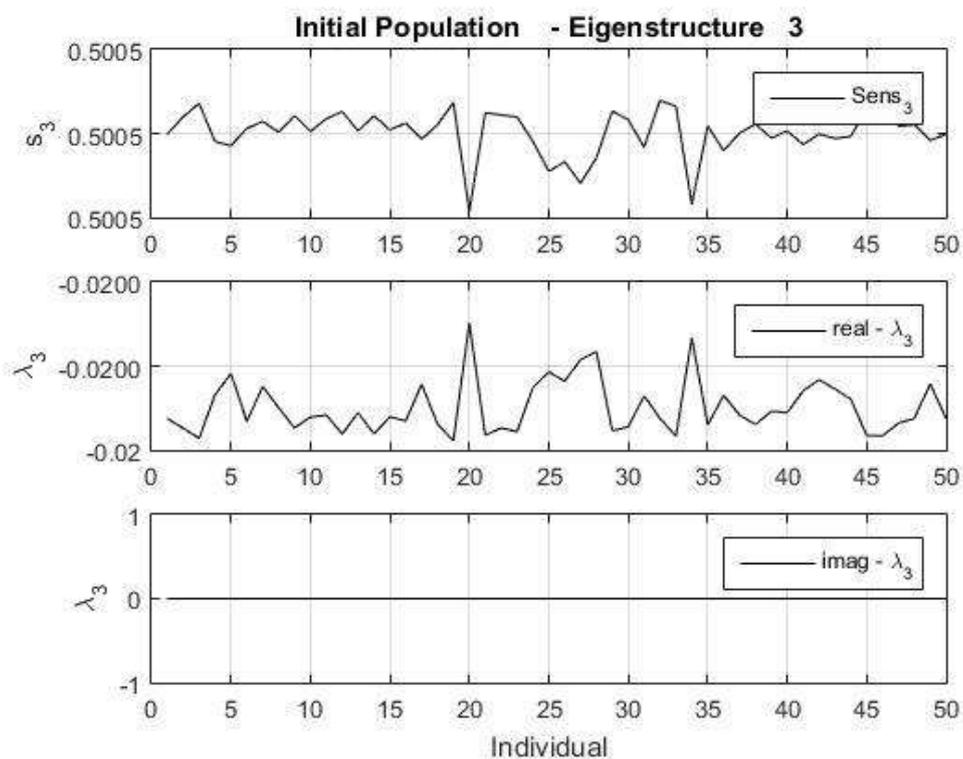


Figura 24- Sensibilidade 3 da população inicial
 Fonte Autor, 2017

8.3 Desempenho da população final

Os valores da função objetivo para a população final, em comparação com a média da função objetivo para cada indivíduo da população, durante o processo de busca, são apresentados na Figura 25. Os valores médios de cada indivíduo da população final apresentaram melhorias praticamente em todos os indivíduos.

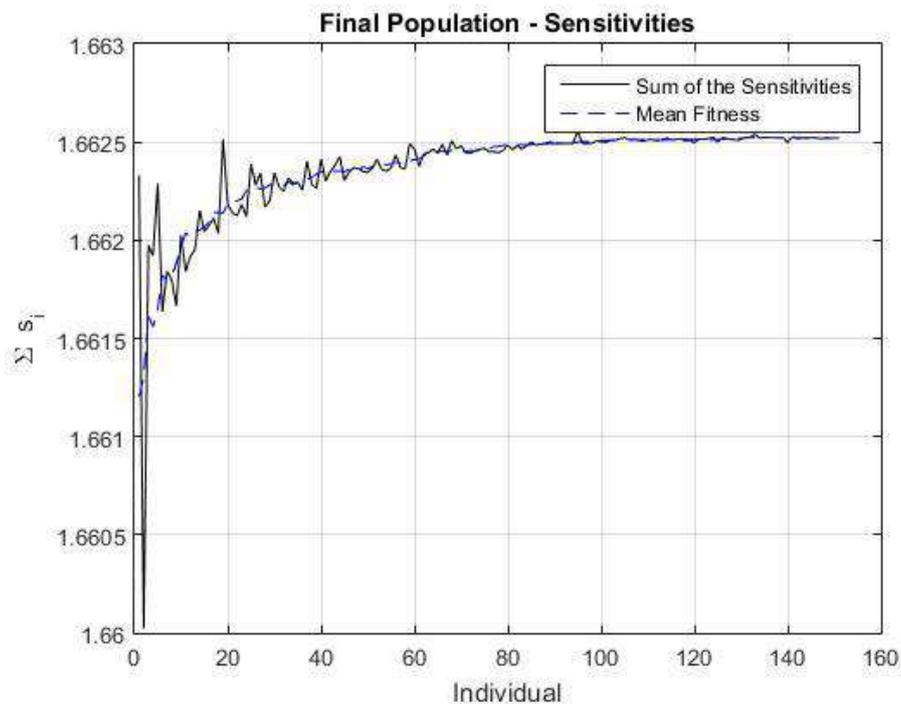


Figura 25- Evolução do processo - função objetivo média e melhor função objetivo das gerações

Fonte Autor, 2017

A busca realizada pelo AG encontrou cinco indivíduos viáveis que estão presentes na Tabela 2. Conseqüentemente, como esperado, os resultados computacionais para resposta ao sinal impulso mostraram uma ligeira diferença entre todas as soluções viáveis. Entre estes, destaca-se o indivíduo sexagésimo que gerou as matrizes de ponderação Q e R ideais

$$Q_{LQR} = \begin{bmatrix} 0.92513 & 2.5352 & 2.5111 \\ 2.5352 & 2.4907 & 4.1903 \\ 2.5111 & 4.1903 & 33.780 \end{bmatrix} \quad R_{LQR} = [33.780] \quad (8.4)$$

Tabela 2- Pop. final AG-sensibilidades e função objetivo dos indivíduos viáveis.

| Indivíduos | | s_1 | s_2 | s_3 | $\sum s_i$ |
|------------|----|-------|-------|-------|------------|
| 1 | 20 | 0,975 | 0,815 | 0,694 | 2,4840 |
| 3 | 13 | 0,919 | 0,831 | 0,677 | 1,7500 |
| 5 | 17 | 0,967 | 0,845 | 0,684 | 2,4960 |
| 7 | 14 | 0,996 | 0,883 | 0,686 | 2,5650 |
| 9 | 48 | 0,986 | 0,986 | 0,683 | 2,6550 |

Os testes para avaliação entre os indivíduos estão presentes na Tabela 3. Nesta, todos os indivíduos viáveis estão próximos das restrições de sensibilidade e da função objetivo. Além disso, a condição de positividade das matrizes são mostradas em termos dos traços e autovalores. As matrizes de ponderação ótimas Q e R do melhor indivíduo factível, os traços e os autovalores, confirmam a positividade das matrizes de ponderação. Essa positividade pode ser usada para orientar o processo de busca do AG, de modo que evite a formação indesejável de indivíduos.

Tabela 3- Pop. final AG-sensibilidades e função objetivo dos indivíduos viáveis.

| Nº | Matriz Q | | Matriz R | |
|----|-------------|-----------|-------------|-----------|
| | λ_Q | $Traço_Q$ | λ_R | $Traço_R$ |
| 1 | 17.466 | 137.288 | 10.285 | 26.793 |
| 2 | 27.654 | | 16.508 | |
| 3 | 37.660 | | | |
| 4 | 54.508 | | | |

O processo de busca do AG com a configuração de seus parâmetros produziu soluções muito próximas uma das outras, ou seja, o conjunto solução matricial QR apresentada pelo AG, tem apenas pequenas discrepâncias. As sensibilidades da população final e a média da função de fitness de cada geração é apresentada na Figura 26.

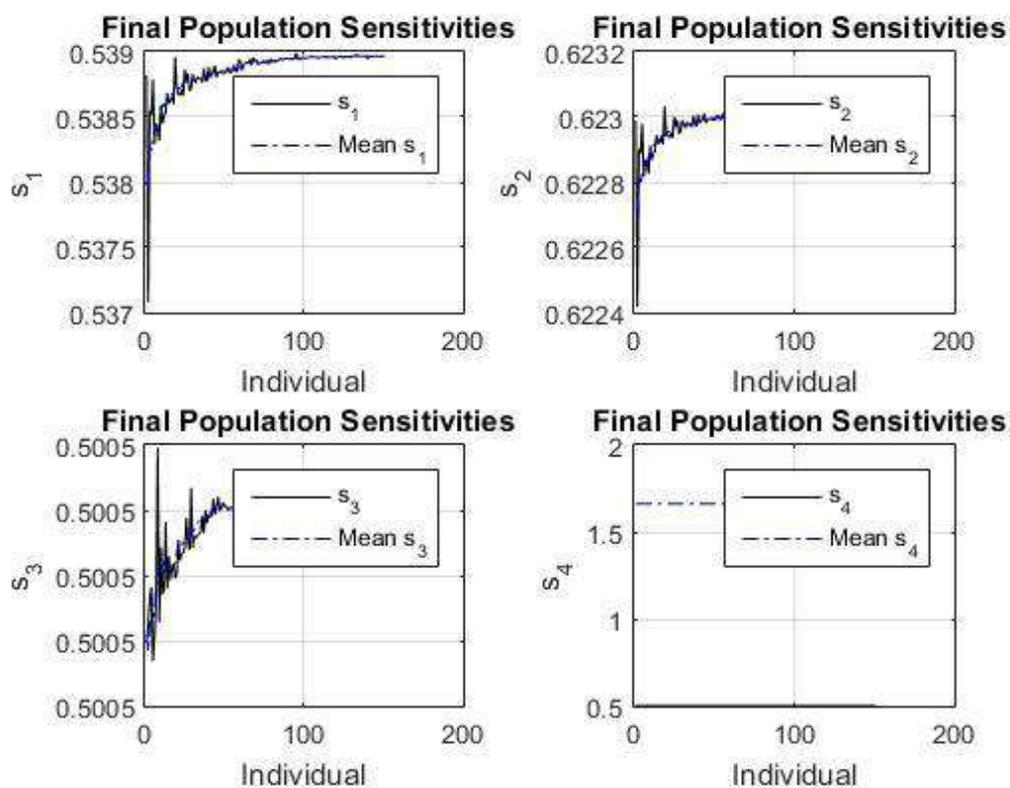


Figura 26- Fitness normalizada de cada geração e de cada indivíduo
 Fonte: Autor, 2017

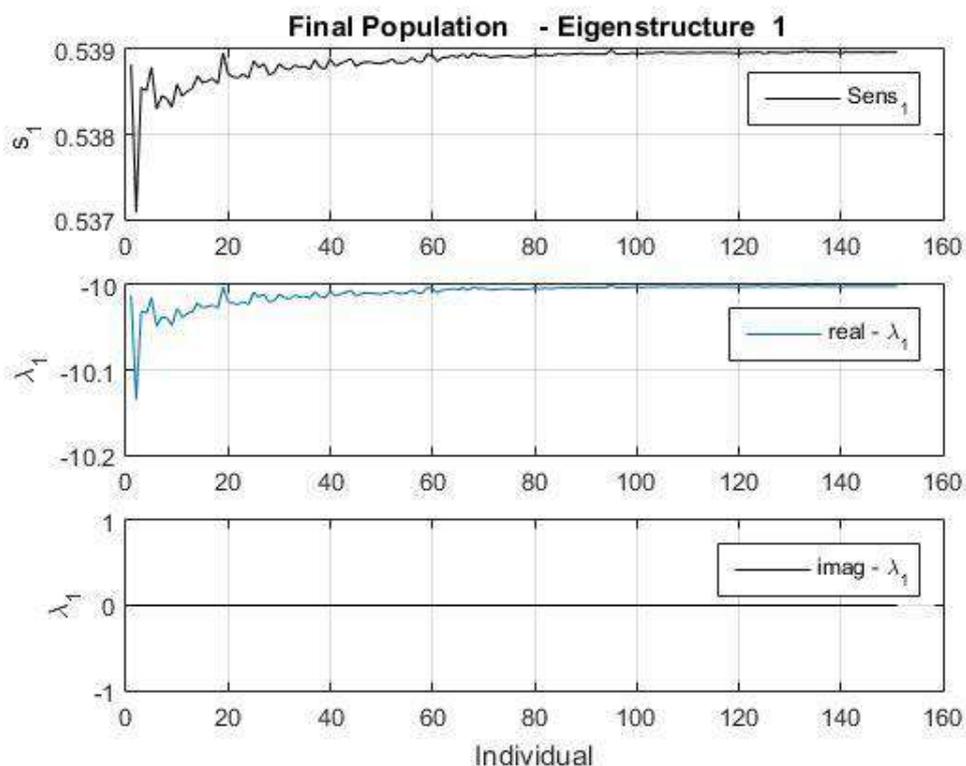


Figura 27-Sensibilidade1 da Pop.final de cada geração e de cada indivíduo
 Fonte: Auto, 2017

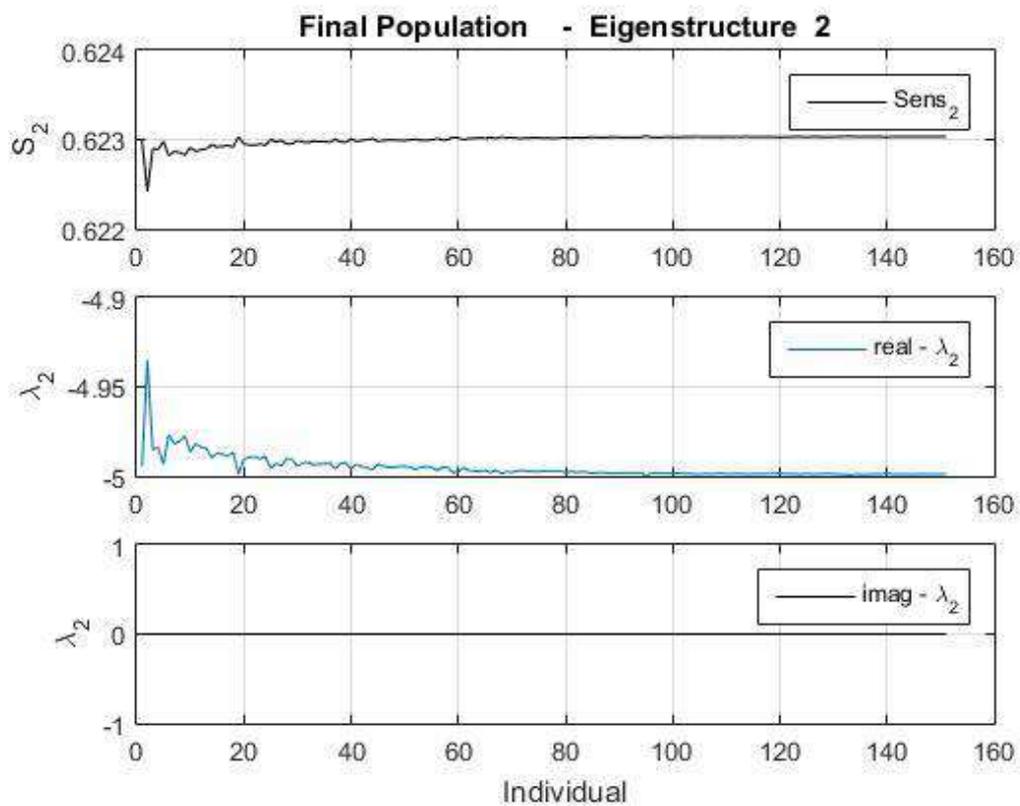


Figura 28-Sensibilidade 2 da Pop.final de cada geração e de cada indivíduo
 Fonte: Autor, 2017

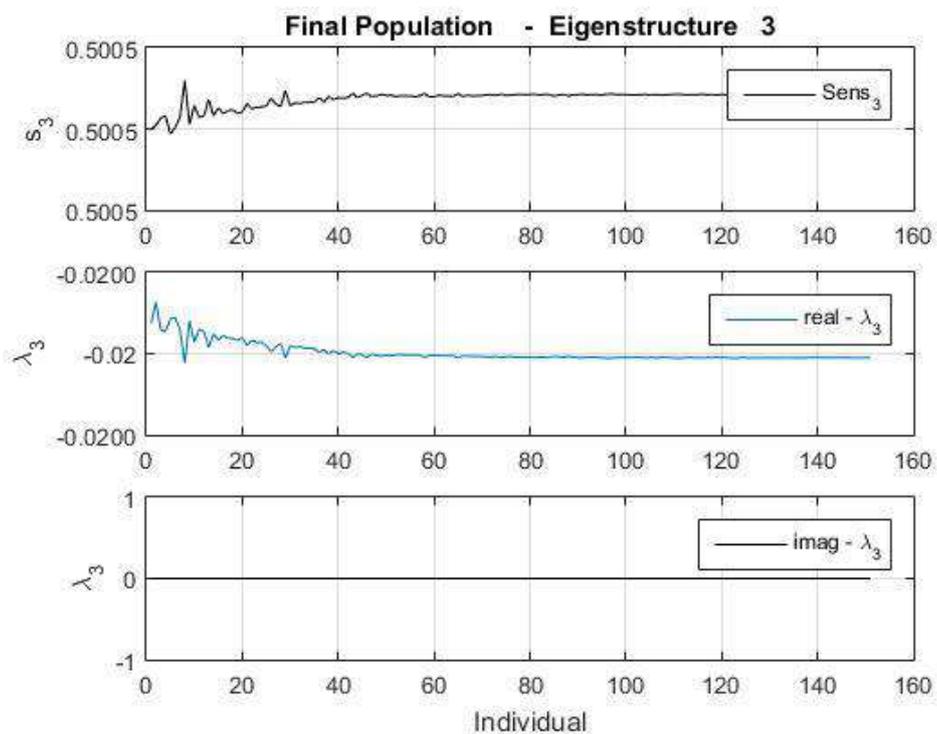


Figura 29- Sensibilidade 3 da Pop.final de cada geração e de cada indivíduo
 Fonte: Autor, 2017

8.4 Solução da EAR via Rede Neural Recorrente Não Supervisionada

Os métodos de ajuste de convergência e análise da *RNAR* para resolver a equação de *Riccati*. Os parâmetros de ajustes, η_v, η_z do modelo neuronal, são suportados pela norma do infinito e superfícies da função energia da solução da equação de *Riccati*. A norma do infinito e a superfície da função energia, como funções dos parâmetros (pesos) de dimensionamento η_v, η_z , são avaliadas para definir uma configuração para melhorar a estabilidade de convergência e características de solvabilidade da *RNAR*. Na Figura 30 e 31 é apresentada a superfície da norma do infinito da matriz de entrada U , sendo que os intervalos de modificação para os parâmetros η_v, η_z , são respectivamente, [1000 10000] e [1 10]. Os pesos escolhidos dos parâmetros η_v e η_z são 1000 e 1, respectivamente.

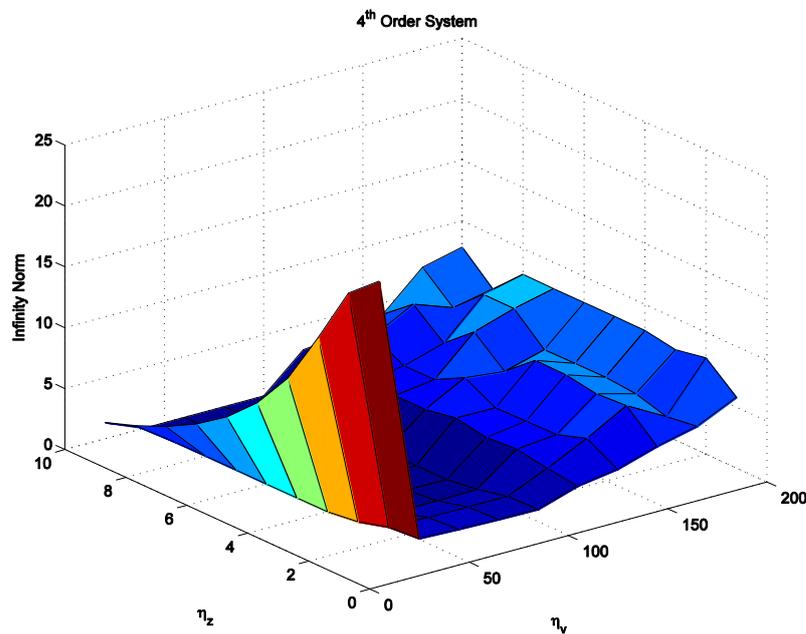


Figura 30- Superfície da função de energia para $\eta_v = 1000-10000$ e $\eta_z = 1-10$.

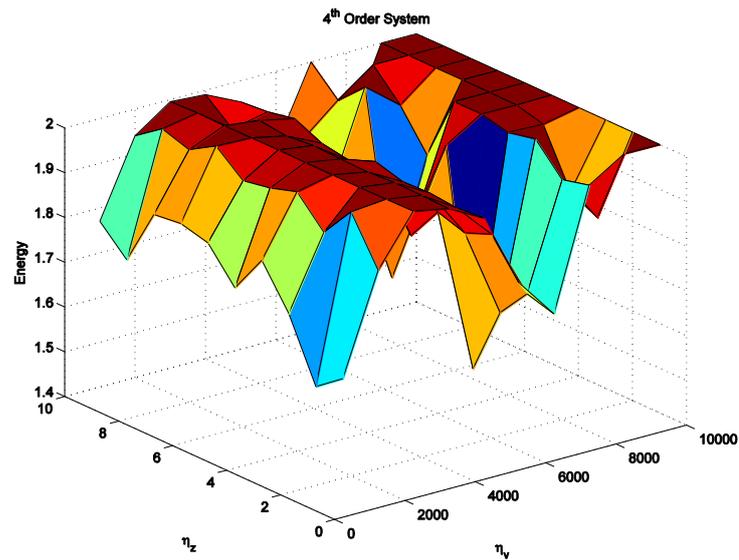


Figura 31- Superfície norma do infinito para $\eta_v=1000-10000$ e $\eta_z=1-10$.

A solução neuronal da equação de *Riccati*, fundamentado no modelo neuronal que resolve a estrutura de otimização que minimiza a função energia, camada de saída da RNAR, V , é dada por

$$P_{LQR} = \begin{bmatrix} 0.7 & 10.1 & 33.8 \\ 10.1 & 150.4 & 504.9 \\ 33.8 & 504.9 & 1694.9 \end{bmatrix} \quad (8.5)$$

8.5 Análise do filtro de Kalman

Um AG para a busca das matrizes de covariâncias Ξ da perturbação do estado e Θ do ruído de medida para o *Filtro de Kalman* é desenvolvido. Para tanto, tem-se interesse em realizar um algoritmo numérico, que possa ser utilizado em qualquer problema de estimação de estado estocástico, objetivando-se a determinação de um ganho $L(\Xi, \Theta)$ de forma a alocar autovalores e autovetores. Utiliza-se o mesmo procedimento para o projeto *LQR*.

No modelo em estudo, a Figura 32, exibe os resultados da Filtragem de Kalman. Nela verifica-se as trajetórias dos estados estimados. Nota-se que, apesar da presença de ruídos, a ação reguladora é detectada em todos os estados. Para uma análise mais precisa, os observadores de estado são projetados para reconstruir todas as variáveis de estado, mas na prática as variáveis de estado podem ser precisamente medidas e não necessitam ser estimadas a Figura 32 ilustra o erro de estimação para o estado 1 e 3. Observou-se que o Estado 1

apresenta o menor erro de estimação, enquanto que o maior erro de estimação foi verificado para o Estado 3.

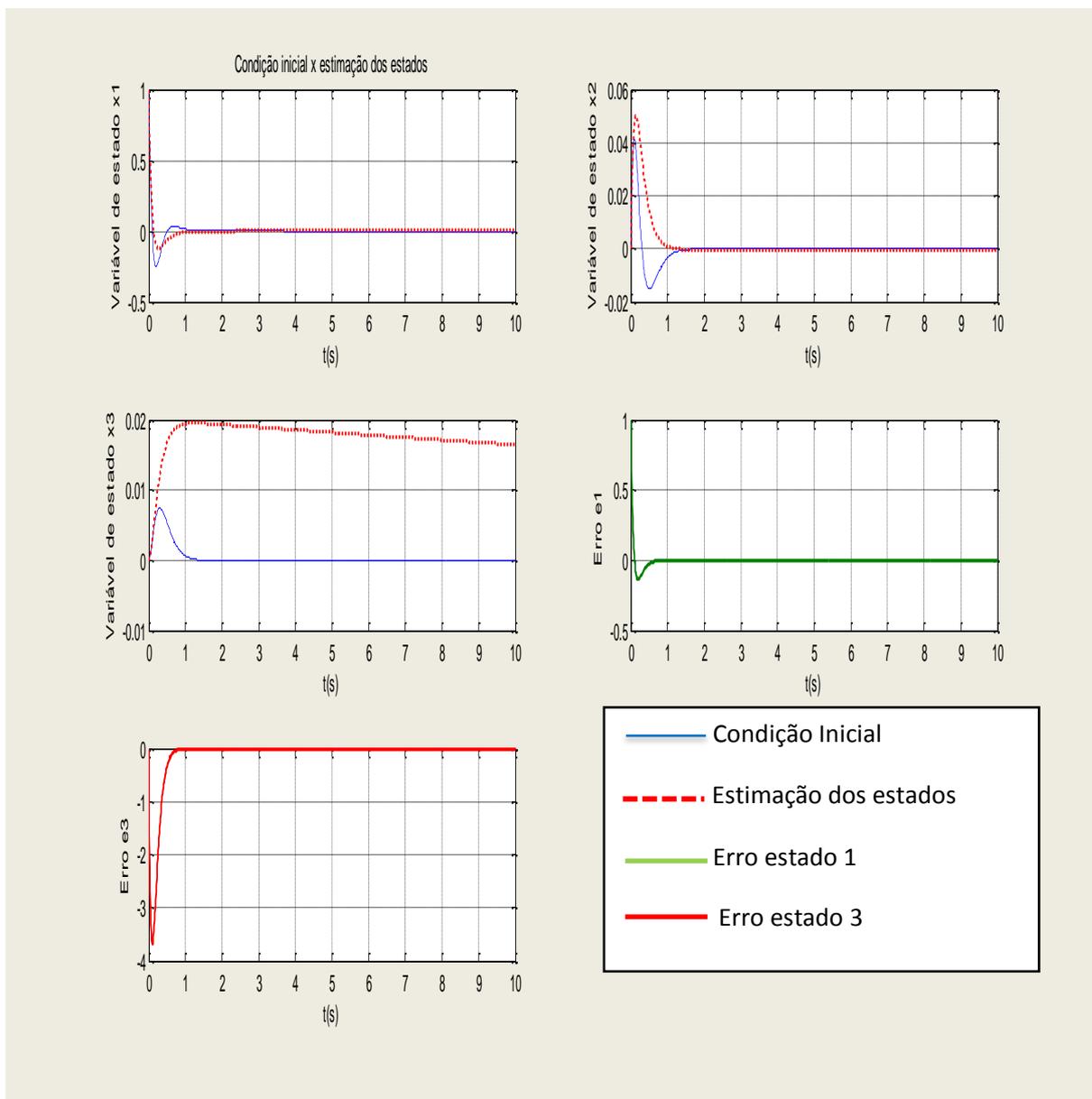


Figura 32- Condição inicial x Estimação dos estados e os erros

Fonte: Autor, 2017

8.6 Recuperação das propriedades de robustez

A figura 33 mostra a resposta em frequência da função de transferência de malha para vários valores de q ($q = 1, 10, 100$), junto com a resposta em frequência da função de transferência de LQR ideal. Como pode ser visto a partir desta figura, o ganho da malha tende a se aproximar do LQR quando o valor de q aumenta tendendo para infinito.

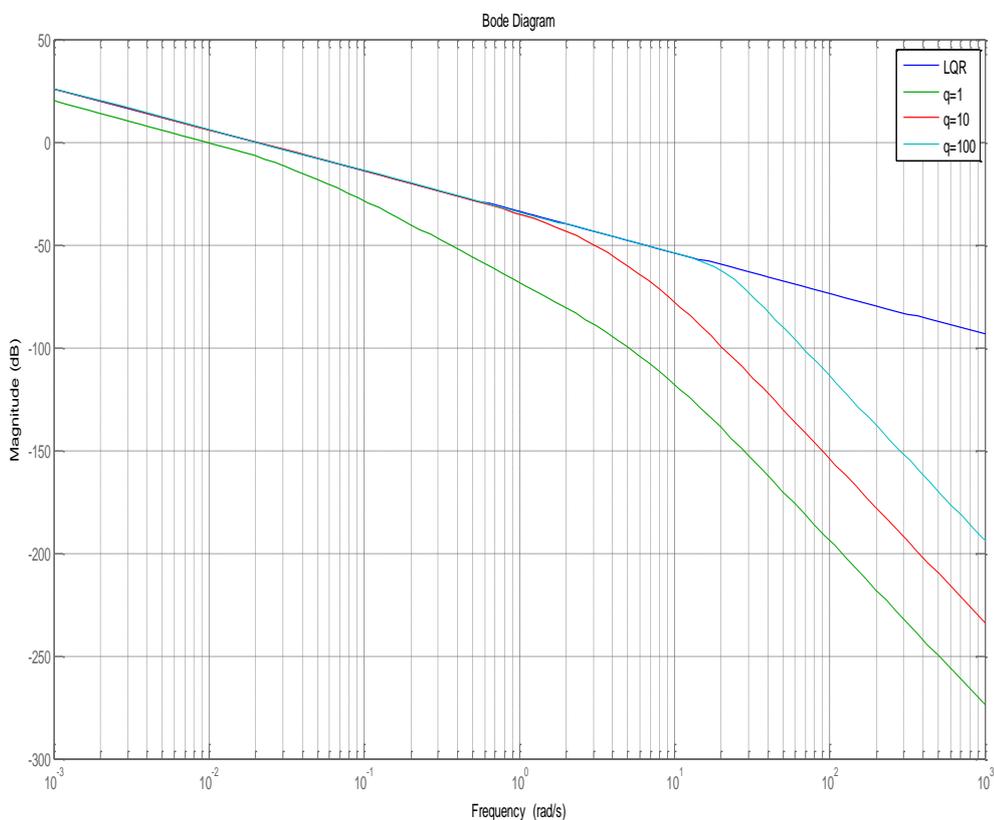


Figura 33-Gráfico de recuperação de malha

Fonte: Autor, 2017

A figura 34 mostra o diagrama de Bode referente ao ganho logarítmico e a fase, percebe-se claramente que as propriedades de robustez do projeto do controlador LQR está de fato garantida, isto é, margem de ganho infinita e margem de fase aproximadamente 60° .

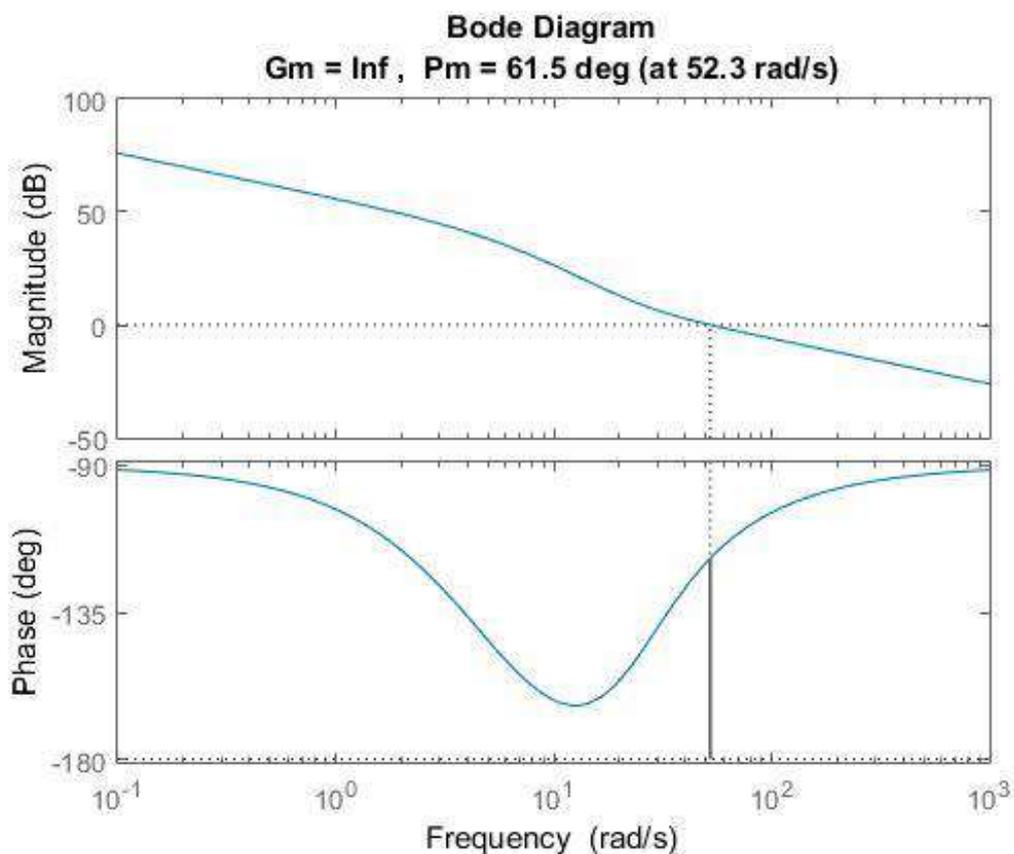


Figura 34- Diagrama de Bode do Ganho de malha

Fonte: Autor, 2017

8.7 Barreira de robustez

Na figura 35, o diagrama de Bode do ganho da função de transferência de malha estar situado entre as barreiras de baixa e alta frequência. Portanto, à esquerda observa-se as barreiras de desempenho associada às especificações de acompanhamento do sinal de referência, rejeição de perturbações e insensibilidade a variações na planta; à direita, tem-se a barreira correspondente à especificação de rejeição do erro de medida.

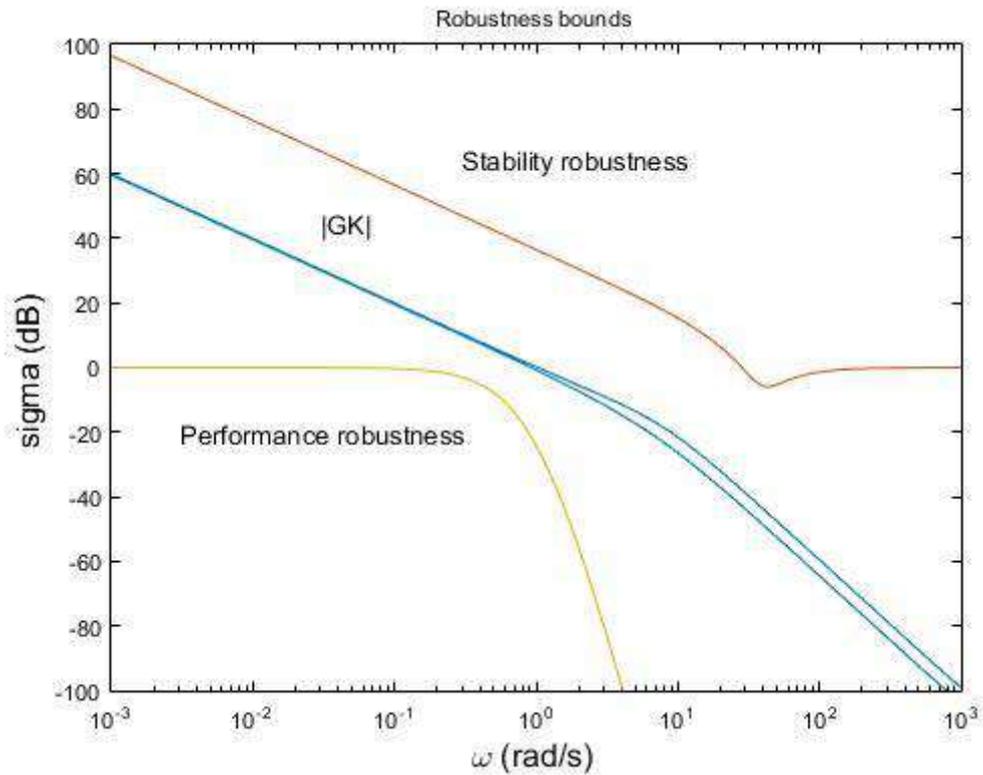


Figura 35- Barreira de desempenho nominal

Fonte: Autor, 2017

8.8 Estabilidade robusta e redução por frações parciais

Para ilustrar o procedimento de redução de modelo e mostrar também como se calcula o limite superior $m(\omega)$ na modelagem de alta frequência dos erros induzidos.

Para tornar mais fácil de ver o que está ocorrendo, toma-se uma forma de Jordan para o sistema.

$$J = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -10 & 0 \\ 0 & 0 & -5 \end{bmatrix} \quad (8.6)$$

Logo o sistema fica:

$$\dot{x} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -10 & 0 \\ 0 & 0 & -5 \end{bmatrix} x + \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} u \quad (8.7)$$

$$z = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} x \quad (8.8)$$

Os autovetores são dados por $u_i = e_i v_i = e_i$ $i = 1,2,3$, com e_i a matriz identidade 3×3 . Assim, a função de transferência é dada pela expansão da fração parcial, com redução de ordem do sistema com polos em $\lambda = 0$ e $\lambda = -10$ definindo Q , V e U .

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad V = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad U = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \quad (8.9)$$

Na figura 35 e 36 temos o gráfico dos valores singulares da planta real e da planta reduzida por aproximação, pois como a planta do DHM tem uma ordem baixa. Praticamente a planta reduzida acompanha a planta real nas baixas frequências e nas altas não é possível controlar

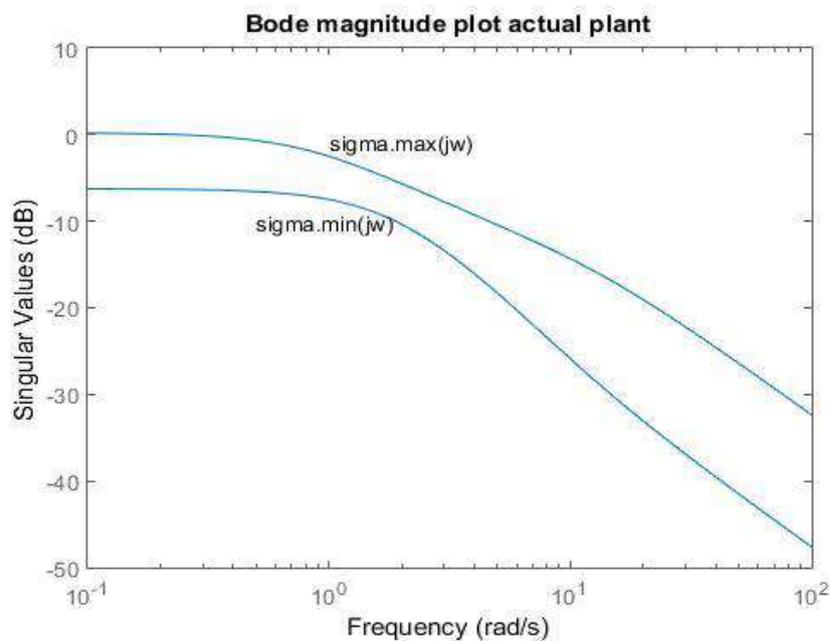


Figura 35- Gráfico de Magnitude de bode dos valores singulares planta atual
Fonte: Autor, 2017

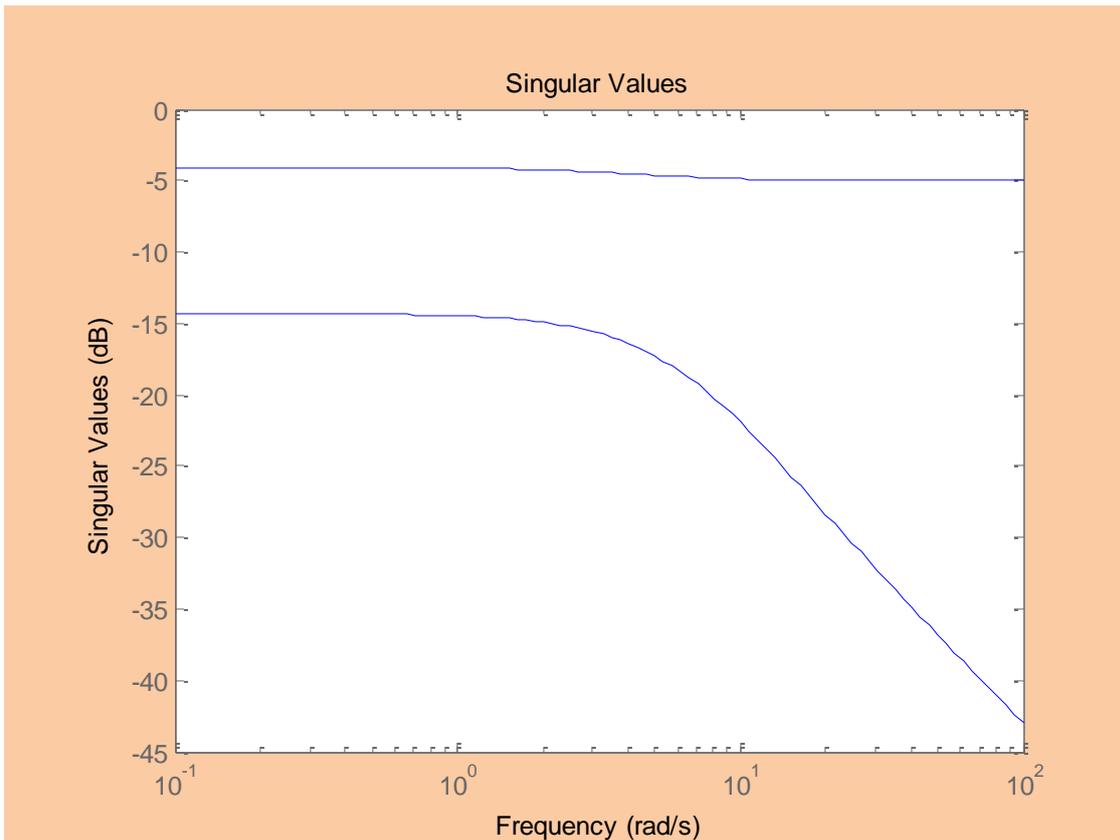


Figura 36- Gráfico de Magnitude de Bode dos valores singulares Aproximação de ordem reduzida

Fonte: Autor, 2017

Na figura 37 e 38 temos o gráfico de alta frequência do erro multiplicativo e do inverso do erro multiplicativo real, O erro multiplicativo é dado pelo limite de alta frequência no ganho de loop $GK(j\omega)$. Observe que os erros de modelagem se tornam apreciáveis (isto é, de magnitude um) a uma frequência de 9,0 rads / seg. Acima desta frequência, devemos garantir que a restrição na magnitude do ganho de loop para garantir a estabilidade de robustez. Esta será uma restrição a qualquer compensador $K(s)$ projetado usando a ordem reduzida.

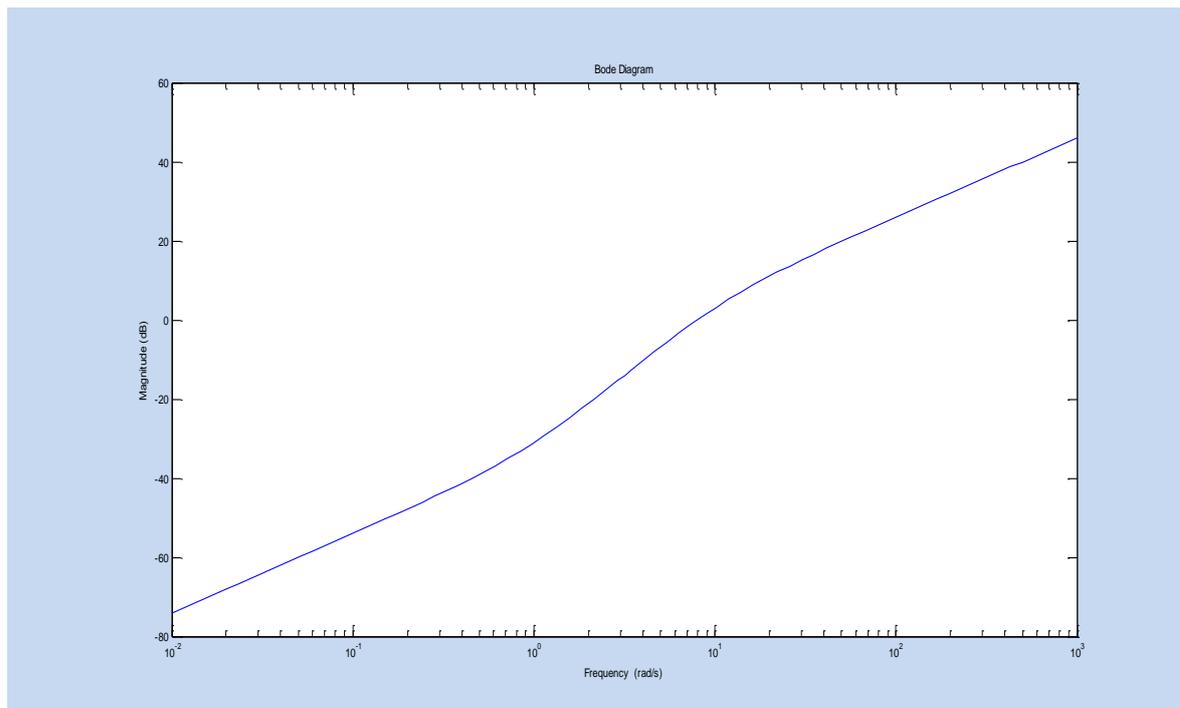


Figura 37- Estabilidade Robusta em alta frequência do erro multiplicativo $m(w)$

Fonte: Autor, 2017

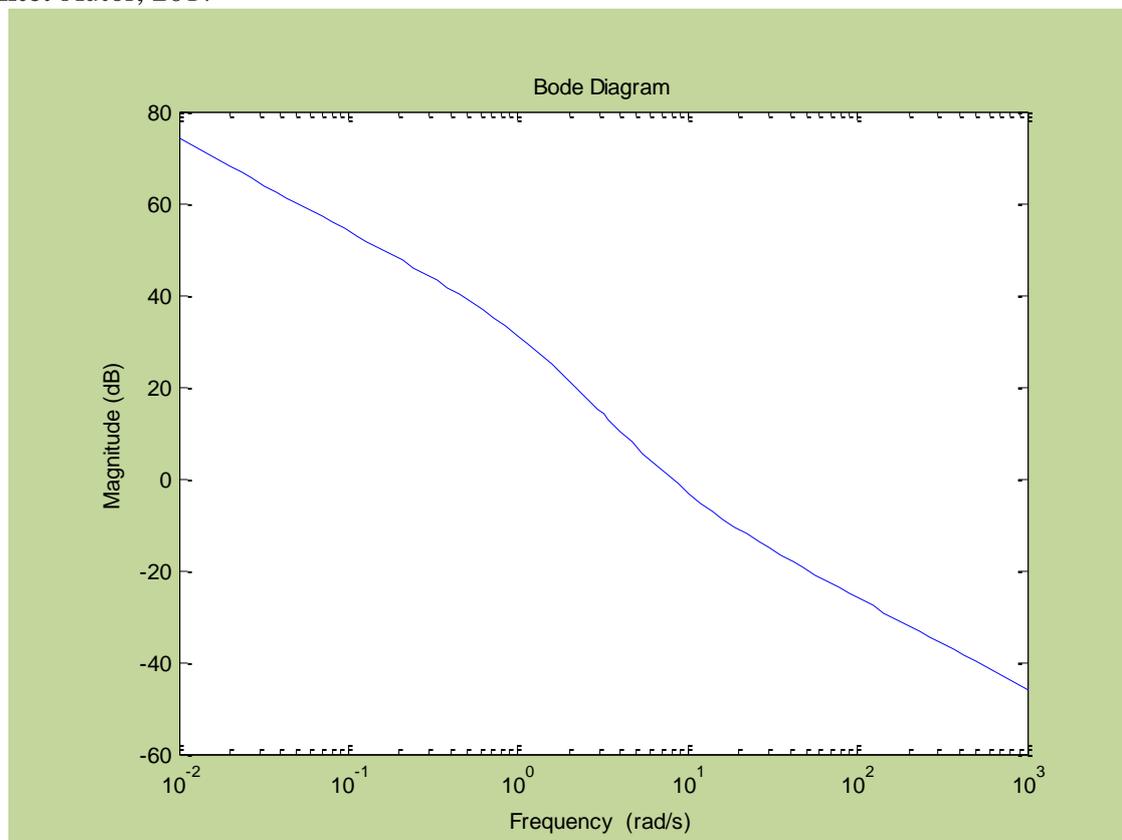


Figura 38- Estabilidade Robusta em alta frequência com inverso do erro multiplicativo $1/m(w)$

Fonte: Autor, 2017

9 CONCLUSÃO

Apresentou-se nesse trabalho, um modelo hierárquico de algoritmos genéticos para recuperação da malha de transferência na entrada, como uma boa solução para ajuste de ganhos de controladores e observadores no espaço de estados. O desenvolvimento dos controladores LQR , LQG e LQG/LTR e métodos de projeto de observadores baseado em modelos de algoritmos genéticos e redes neurais artificiais, são utilizados para ajustar os ganhos de malha. Baseado na implementação o modelo do controlador, observador e o método de ajuste proposto, poderá ser visto como uma alternativa para aplicações práticas em tempo real, utilizando técnicas de inteligência computacional.

Na estimação de estados, utilizou-se o LQR e o Filtro de *Kalman*, como metodologias de projeto. Os resultados obtidos neste trabalho foram satisfatórios, recuperou-se as propriedades de robustez perdidas pela estimação das variáveis de estado, utilizando como ferramenta o projeto LQG/LTR e o modelo de redução da ordem por frações parciais, usando Jordan, para o erro multiplicativo nas baixas e altas frequências. A barreira de robustez na figura 34 mostra que o ganho de malha aberta ficou entre as barreiras de baixa e alta frequência acompanhando o sinal de referência, rejeição de perturbações e insensibilidade a variações na planta e a rejeição do erro de medida na alta frequência. O desenvolvimento dos modelos culminou em um procedimento para o projeto inteligente de sintonia de controladores LQR , LQG e LQG/LTR que se baseia em uma hierarquia composta por algoritmos genéticos.

9.1 Trabalhos futuros

Como perspectivas futuras propõem-se, portanto:

- O estudo da complexidade do algoritmo aplicado no DHM e proposta de otimização;
- Verificar o desempenho da recuperação de malha na planta do DHM;
- Implementar um Projeto de Controladores H infinito algoritmo genético a nível de comparação aos resultados encontrados neste trabalho;
- Realizar RFTM genético hierárquico para DHM de forma tal que realize a recuperação pela saída.

REFERÊNCIAS

- [1] ATHANS, Michael and Falb, L. Peter. Optimal control 'an introduction to the theory and its applications', 2007.
- [2] ABREU, Ivanildo and Fonseca Neto; Neural-genetic state space controllers synthesis based on linear quadratic regulator design for eigenstructure assignment; Systems, man, and Cybernetics, part B; Cybernetics, IEEE Transaction on, 2010.
- [3] ABREU, Ivanildo.; FONSECA, N. Controle Inteligente LQR Neuro-Genético para Alocação de Autoestrutura em Sistemas Dinâmicos Multivariáveis: Tese de Doutorado, 2008. Cruz,
- [4] BARR, A.; FEIGENBAUM, E. The Handbook of Artificial Intelligence, volume 1. William Kaufmann, Inc., 1981.
- [5] CRUZ, José Jaime da (1996). Controle Robusto Multivariável. EDUSP1996.
- [6] CRUZ, José Jaime da. Contribuição ao estudo da robustez da estabilidade de reguladores não lineares multivariáveis. Tese(Doutorado)- INPE, São José dos Campos-SP, 1988.
- [7] CASTRUCCI, Plínio de Lauro. Controle automático. Rio de Janeiro: LTC, 2011.
- [8] CHEN, Chi-Tsong. Linear system theory and design. New York: Oxford University press, Inc, 1999.
- [9] CHIN, Cheng Siong. Computer – Aided Control Systems Design. Taylor & Francis Group, LLC, 2013.
- [10] CHIU, D.K; LEE, S. Design and experimentation of a jump impact controller. IEEE, 1997.
- [11] CAMPOS, Mario Massa de; SAITO, Kaku. Sistema inteligentes em controle e automação de processos. Rio de Janeiro: Ciência Moderna, 2004.
- [12] DATTA, Biswa Nath. Numerical methods for linear control systems. St. Louis: Elsevier Science & Technology books, 2003.
- [13] DORF, Richard C. Sistemas de controle modernos. Rio de Janeiro: LTC, 2014.
- [14] DOYLE, J. C. and G. Stein (1979). Robustness with Observers. *IEEE Transactions on Automatic Control* 24(4), 607(611).
- [15] GASPARYAN, Oleg. Linear and nonlinear multivariable feedback control: A classical approach. England: John Wiley Inc, 2008.

- [16] GOLDBERG, David Edward. Genetic algorithms in search, optimization, and machine learning. Inc, USA: Addison-Wesley Publishing, 1989.
- [17] GREFENSTETTE, J.J. Optimization of control parameters for genetic algorithms. IEEE transactions on systems, man, and cybernetics v.16(1) p. 122-128, 1980.
- [18] HAUPT, R.L.; HAUPT, S.E. The continuous parameter genetic algorithm, practical genetic algorithm. USA: Jon Wiley & Sons, Inc., 1988.
- [19] HECHT-NIELSEN, R. Neurocomputing. Addison-Wesley publishing company, 1990.
- [20] KAILATH, Thomas. Linear systems. First.ed. Inc.. Englewood Cliffs, N.J:Prentice-Hall, 1980.
- [21] KALMAN, R.E. Contribution to the theory of optimal control. 1960a. Bol.5.p. 102-119.
- [22] KALMAN, R.E. On the general theory of control. In: PROCEEDINGS OF FIRST IFAC CONGRESS. Butter Worth, London. 1960b.
- [23] KAWAKERNAK, H.; SIVAN, R. Linear Optimal Control Systems. New York: Wiley Interscience, 1972.
- [24] LAUB, A. J. A schur method for solving algebraic riccati equations. IEEE Transactions on Automatic Control. 1979. v.24(6) p. 913-921.
- [25] LEWIS, F.L.; VRABIE, D.L.; SYRMOUS, V.L. Optimal control. USA: John Wiley & Sons Inc., 2012.
- [26] LEWIS, Frank L. and Vassilis L. Syrmos (1995). *Optimal Control*. John Wiley and Sons, Inc.. USA.
- [27] MAYRO. The origins of feedback control. Cambridge, Mass: MIT Press, 1970.
- [28] MICHALEWICZ, Z.; J.B. Krawczyk, M.; C.Z Janikow. Genetic algorithms and optimal control.. In: PROCEEDING OF THE 29th CONFERENCE ON DECISION AND CONTROL, 1990.
- [28] MULLER, B.; REINHARDT, J. Neural networks: An introduction. New York: Springer-Verlag Berlin Heidelberg, 1995.
- [29] NETO, João V. and ABREU, Ivanildo ; A genetic algorithm convergence and models for eigenstructure assignment via linear quadratic regulator, Revista IEEE América Latina, 2007.
- [30] OGATA, Katsuhiko. Engenharia de controle moderno. São Paulo: Pearson Prentice Hall, 2010.

- [31] RÊGO, Patricia Helena Moraes. Convergência de Algoritmo Genético Hierárquico para Recuperação da Malha LQR por Controladores LQG/LTR: Dissertação de Mestrado, 2007.
- [32] SERGI, F. Design and characterization of a compact rotary series elastic actuator for knee assistance during overground walking. Roma, 2012.
- [33] SILVA, Fábio Nogueira da. Métodos Neurais para a solução da Equação Algébrica de Riccati e o LQR: Dissertação de Mestrado, 2008.
- [34] SIVANADAN, S.N.; DEEPA, S.N. Introduction to genetic algorithms. Índia: Springer-Verlag Berlin Heidelberg, 2008
- [35] SKOGESTAD, S.; POSTLETHWAITE, I. Multivariable feedback control: Analysis and design, v. 2. Wiley New York, 2007.
- [36] XUE, Dingyu; CHEN, YangQuan; ATHERTON, Derek P. Linear feedback control: Analysis and design with Matlab. Philadelphia: Siam, 2007.
- [37] WANG, Jun and WUB and GUANG; A multilayer recurrent neural network for solving continuous-time algebraic Riccati equations, 1998.
- [38] MEDEIROS, José Charles, Um Controlador Robusto LQR/LQG e filtro de Kalman em um manipulador Robótico via modelo genético neural, dissertação de mestrado 2017.
- [39] M. de LACERDA, Estefane G. e F de CARVALHO, André Carlos P. L. Introdução a algoritmos Genético.
- [40] A. OLIVEIRA, Vilma. O Filtro de Kalman e recuperação da malha de transferência USP São Carlos, Junho 2002.
- [41] MATSUNAGA, Victória Yukie. Curso de Redes Neurais utilizando Matlab. Belém-Pa, 2012.
- [42] COELHO, Leandro Oliveira. Recuperação das propriedades de Robustez de um manipulador Robótico Teleoperado via projeto LTR. Monografia de graduação, 2015.
- [43] BRITO FILHO, Joaquim Gomes. Controle Robusto LQG/LTR com Recuperação do Ganho da malha de Transferência. Dissertação de Mestrado 2006.
- [45] IA006 – Prof. Leandro N. de Castro & Fernando J. Von Zuben DCA/FEEC/Unicamp
- [44] LIMA JR. Herman, GESUALDI, Aline da Rocha, PORTES DE ALBUQUERQUE, Marcelo, PORTES DE ALBUQUERQUE Márcio, ESQUEF Israel Andrade. Aplicação de Redes Neurais no reconhecimento de letras em placas de veículos automotores brasileiros. Apostila, 2000.
- [45] POZO, Aurora, CAVALHEIRO Andrea de Fatima, ISHIDA, Celso, SPINOSA Eduardo, RODRIGUES, Ernesto Malta. COMPUTAÇÃO EVOLUTIVA. Grupo de Pesquisas em Computação Evolutiva UFPA, Departamento de Informática.

APÊNDICES

APÊNDICE A - CONTROLE ÓTIMO

No controle ótimo as técnicas de otimização são de extrema importância, pois, as mesmas são usadas para se determinar as condições necessárias ou suficientes que possam garantir a existência de uma solução ótima para o controle, se ela existir.

A compreensão do comportamento das dinâmicas dos sistemas é uma das etapas fundamentais, pois, é necessário modelar o processo ou fenômeno de modo a se construir a função objetivo que será otimizada e quais as possíveis restrições relacionadas ao problema. Uma vez construída a função objetivo e compreendido o comportamento da mesma, pode-se então definir quais critérios e métodos seriam mais adequados para minimização ou maximização da função em questão. Técnicas como o gradiente descendente, programação dinâmica, métodos mais recentes como redes neurais e algoritmos genéticos têm sido usados com grande sucesso no processo de minimização ou maximização dessas funções dependendo do objetivo que se deseja com o processo de otimização.

Para se realizar o controle é necessário o conhecimento das leis que regem o sistema, deste modo uma das etapas seria modelagem matemática, pois, um modelo matemático que descreva a dinâmica do sistema torna possível a aplicação do controle de modo mais eficiente.

Nas metodologias de projeto de controle, existe uma série de figuras de méritos que representam uma série de questões desejáveis e que nos dão mais informação a cerca do processo de controle e resposta dinâmica do sistema. Algumas dessas figuras de mérito como: tempo de subida, tempo de resposta, tempo de acomodação, ganho de margem e ganho de fase e largura de banda, são ferramentas que auxiliam no processo de controle e fornecem várias informações sobre o comportamento do processo e como o controle está atuando.

Métodos clássicos são usados e foram formulados com o propósito de solucionar problemas de controle de sistemas lineares com uma entrada e uma saída. Para o controle de sistemas mais complexos, não lineares e com múltiplas entradas e saídas as metodologias clássicas não são adequadas. A teoria de controle ótimo tem sido usada para o controle sistemas mais complexos lineares, não-lineares e multivariáveis. A indústria bélica e aeroespacial teve grande contribuição nas últimas décadas.

A aplicação da técnica de controle ótimo necessita de uma clara descrição dos índices de desempenho a serem alcançados pelo sistema, de um modelo matemático bem descrito que é representado frequentemente por equações diferenciais ordinárias, da compreensão e determinação das restrições do sistema e do projeto de controle completo. Na

prática, a dinâmica do processo de controle ótimo é frequentemente um processo iterativo, em que o índice de desempenho é alterado a cada observação atual e avaliado e então atua na planta do sistema de modo a restabelecer ou mesmo manter a estabilidade do sistema, para um estado de operação próximo do especificado pelo projetista.

Os problemas de otimização apresentados em cálculo visam a minimização ou maximização de uma função no espaço vetorial de dimensão finita. Para o problema de controle ótimo, as variáveis independentes de interesse são as trajetórias de controle estado, sendo estas funções do tempo, portanto o problema de encontrar uma trajetória e controle que minimize um índice de desempenho, implica em encontrar o mínimo de um funcional, ou seja, uma função no espaço de funções, o que é um problema do cálculo de variações.

O caso mais clássico para um índice de desempenho quadrático instantâneo, assume a forma de uma integral no tempo, em que do um sistema linear variante no tempo relacionadas a seguir:

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0 \quad (\text{A.1})$$

$$y(t) = Cx(t) \quad (\text{A.2})$$

sendo $x(t) \in \mathbb{R}^n$ o vetor de estado do sistema, $u(t) \in \mathbb{R}^m$ o vetor controle, $A \in \mathbb{R}^{n \times n}$ são matrizes associadas ao vetor de estado $x(t)$ e ao vetor de entrada $u(t)$ respectivamente, x_0 representa o vetor com o estado inicial com $m \leq n$. A forma do índice de desempenho quadrático pode ser expressa da seguinte forma:

$$J = h(x(t_f), t_f) + \int_{t_0}^{t_f} g(x(t), u(t), t) dt \quad (\text{A.3})$$

Sendo t_0 e t_f , o tempo inicial e final, respectivamente. A função h é usada no problema de horizonte finito associado ao ajuste da ponderação do estado no tempo final e a função g calcula o valor instantâneo no índice de desempenho para um sinal de controle específico e particular estado.

Como os problemas de controle ótimo possuem a relação apresentada pela Eq. (A.1) que é vista como uma restrição para o problema de otimização, deste modo, para que possamos lidar com a relação do estado adiciona-se uma variável $p(t)$ representado o coestado. O coestado assume papel semelhante aos multiplicadores de *Lagrange* no cálculo de

funções, ou seja, converte o problema de otimização restrita em um de otimização irrestrita. Deste modo temos que o índice de desempenho agora será representado do seguinte modo:

$$J = h(x(t_f), t_f) + \int_{t_0}^{t_f} g(x(t), u(t), t) + p(t)^T [f(x(t), u(t), t) - \dot{x}(t)] dt$$

(A.4)

Solucionar o índice de desempenho Eq. (A.4), em relação as variações independentes de $x(t), p(t), u(t)$, é semelhante a minimiza o índice de desempenho original, Eq. (3.3) em relação a $u(t)$. Para extremizar o índice de desempenho aumentado, Eq. (A.4), basta aplicar as condições de otimalidade:

$$\dot{x}^*(t) = f(x^*(t), u^*(t), t) \quad (\text{A.5})$$

$$\dot{p}^*(t) = \left[\frac{\partial f}{\partial x}(x^*(t), u^*(t), t) \right]^T p^*(t) - \frac{\partial g}{\partial x}(x^*(t), u^*(t), t)$$

(A.6)

e

$$\frac{\partial g}{\partial u}(x^*(t), u^*(t), t) + \left[\frac{\partial f}{\partial u}(x^*(t), u^*(t), t) \right]^T p^*(t) = 0$$

(A.7)

Sendo que *, representa os valores ótimos. Uma equação adicional fornece os valores de contorno para as equações diferenciais, (Kirk 1970). Em geral, a equação que fornece os valores iniciais para a equação de estado Eq. (A.5) e as condições finais para a equação do coestado Eq. (A.6). Solucionar o problema de controle ótimo se resume a solucionar um problema de contorno em torno de dois valores limites.

As equações diferenciais, Eq. (A.5) e Eq. (A.6) são geralmente equações de segunda ordem ou de ordem maior e em sua maioria equações não lineares. Soluções analíticas para equações diferenciais deste tipo geralmente não estão disponíveis. Técnicas de integração numérica como o método de *Runge - Kutta* não são aplicáveis, por não contornar o problema de descontinuidade nos limites das condições de contorno.

Métodos tentativa e erro podem ser mais demorados de retornar uma solução satisfatória ou até mesmo podem não retornar uma solução viável, porque a equação do coestado é o oposto da adjunta da equação de estado, ou seja, a estabilidade da equação de estado implica em instabilidade na equação do coestado.

APÊNDICE B- CONTROLABILIDADE E OBSERVABILIDADE

As condições de controlabilidade e observabilidade podem governar a existência de uma solução completa para o projeto de sistemas de controle, pois a solução para um problema pode não existir se o sistema for considerado não controlável. Embora a maior parte dos sistemas físicos sejam controláveis e observáveis os modelos matemáticos correspondentes podem não possuir essa propriedade. Então é necessário saber as condições sob as quais o sistema é controlável e observável.

Segundo Dorf e Bishop (2010), um sistema é completamente controlável se existir um controle irrestrito $u(t)$ que pode transferir qualquer estado inicial $x(t_0)$ para qualquer outra localização desejada $x(t)$ em um tempo finito, $t_0 \leq t \leq T$.

Técnicas avançadas de projetos de variáveis de estado podem lidar com situações em que o sistema não é completamente controlável, porém os estados (ou as combinações lineares dos mesmos) que não podem ser controlados são inerentemente estáveis. Estes sistemas são classificados como estabilizáveis. Se um sistema é completamente controlável, é também estabilizável.

Considere o sistema de tempo contínuo:

$$\dot{x} = Ax + Bu \quad (\text{B.1})$$

Onde

x = vetor de estados (vetor- n)

u = sinal de controle (escalar)

A = matriz $n \times n$

B = matriz $n \times 1$

B.1 CONTROLABILIDADE

O sistema descrito pela equação B.1 é dito controlável em $t = t_0$ se for possível construir um sinal de controle irrestrito que irá transferir um estado inicial para qualquer estado final em um intervalo finito de tempo $t_0 \leq t \leq t_1$. Se todos os estados são controláveis, então o sistema é dito completamente controlável. Devemos derivar a condição para controlabilidade completa de estados. Sem perda de generalidade, assumiremos que o estado final é a origem do espaço de estado e que o tempo inicial é zero ou $t_0 = 0$.

A solução da B.1 é

$$x(t) = e^{At}x(0) + \int_0^t e^{A(t-r)}Bu(\tau)d\tau \quad (\text{B.2})$$

Aplicando a definição de controlabilidade completa de estados dada, temos

$$x(t_1) = 0 = e^{At_1}x(0) + \int_0^{t_1} e^{A(t_1-\tau)} Bu(\tau) d\tau$$

Ou

$$x(0) = - \int_0^{t_1} e^{-A\tau} Bu(\tau) d\tau \quad (\text{B.3})$$

$e^{-A\tau}$ pode ser escrito como

$$e^{-A\tau} = \sum_{k=0}^{n-1} \alpha_k(\tau) A^k$$

(B.4)

Substituindo a equação B.3 na equação B.4 temos

$$x(0) = - \sum_{k=0}^{n-1} A^k B \int_0^{t_1} \alpha_k(\tau) u(\tau) d\tau \quad (\text{B.5})$$

Seja

$$\int_0^{t_1} \alpha_k(\tau) u(\tau) d\tau = \beta_k$$

Então a equação B.5 se torna

$$\begin{aligned} x(0) &= - \sum_{k=0}^{n-1} A^k B \beta_k \\ &= - [B \quad AB \quad \dots \quad A^{n-1}B] \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{n-1} \end{bmatrix} \end{aligned} \quad (\text{B.6})$$

Se o sistema é completamente controlável, então, dado qualquer estado inicial $x(0)$, a equação B.6 deve ser satisfeita. A partir dessa análise, podemos afirmar a condição para a controlabilidade completa de estados como segue:

O sistema dado é completamente controlável se e somente se os vetores $B, AB, \dots, A^{n-1}B$ forem linearmente independentes ou a matriz $n \times n$

$$[B \quad A \quad \dots \quad A^{n-1}B]$$

for de ranque n . A matriz $[B \quad A \quad \dots \quad A^{n-1}B]$ é comumente chamada de matriz de controabilidade.

No projeto prático de um sistema de controle, podemos querer controlar a saída em vez de controlar os estados do sistema. A controlabilidade completa dos estados não é nem necessária e nem suficiente para controlar a saída do sistema. Por essa razão é desejável definir separadamente a controlabilidade completa da saída.

Considere o sistema descrito por

$$\dot{x} = Ax + Bu \quad (\text{B.7})$$

$$y = Cx + Du \quad (\text{B.8})$$

Onde

x = vetor de estados (vetor n)

u = vetor de controle (vetor r)

y = vetor de saída (vetor m)

A = matriz $n \times n$

B = matriz $m \times n$

C = matriz $m \times n$

D = matriz $m \times r$

O sistema descrito pelas equações é dito completamente controlável se for possível construir um vetor irrestrito $u(t)$ que irá transferir qualquer saída inicial $y(t_0)$ para qualquer saída final $y(t_1)$ em um intervalo finito de tempo $t_0 \leq t \leq t_1$.

Pode ser provado que a condição para controlabilidade completa da saída é como segue:

O sistema é completamente controlável se e somente se a matriz $m \times (n + 1)r$ For de ranque m .

$$[C \quad CAB \quad CA^2B \quad \dots \quad CA^{n-1}B \quad D]$$

Note que a presença do termo Du na equação B.8 sempre ajuda a estabelecer a controlabilidade da saída.

B.2. Observabilidade

Nessa seção discutiremos a observabilidade de sistemas lineares. Considere o sistema descrito nas seguintes equações:

$$\dot{x} = A \quad (\text{B.9})$$

$$y = Cx \quad (\text{B.10})$$

Onde

x = vetor de estados (vetor n)

y = vetor de saída (vetor m)

A = matriz $n \times m$

C = matriz $m \times n$

O sistema é dito ser completamente observável se todo estado $x(t_0)$ pode ser determinado a partir da observação de $y(t)$ em um intervalo de tempo finito $t_0 \leq t \leq t_1$.

O sistema é, então, completamente observável se toda transição de estado eventualmente afeta cada elemento do vetor de saída.

O conceito de observabilidade é útil na resolução do problema de reconstrução de variáveis de estado que não podem ser medidas a partir de variáveis que podem ser medidas na menor duração de tempo possível. Nesta seção tratamos apenas sistemas lineares e invariantes no tempo. Logo, sem perda de generalidade, podemos assumir que $t_0 = 0$.

O conceito de observabilidade é muito importante porque, na prática, a dificuldade encontrada com controle por realimentação de estados é que algumas variáveis de estado não estão acessíveis para medições diretas, com o resultado de que se torna necessário estimar as variáveis de estado que não podem ser medidas a fim de construir os sinais de controle (OGATA, 2010).

Então

$$x(t) = e^{At}x(0) + C \int_0^t e^{A(t-\tau)} Bu(\tau) d\tau \quad (\text{B.11})$$

E $y(t)$ é

$$y(t) = Ce^{At}x(0) + C \int_0^t e^{A(t-\tau)} Bu(\tau) d\tau + Du \quad (\text{B.12})$$

Considerando que as matrizes A, B, C e D são conhecidas e $u(t)$ também é conhecido, os últimos dois termos no lado direito dessa última equação são quantidades conhecidas. Conseqüentemente, elas podem ser subtraídas do valor observado de $y(t)$. Assim, para investigar a condição necessária e suficiente para a completa observabilidade, é suficiente considerar o sistema descrito pelas equações B.11 e B.12.

O vetor de saída $y(t)$ é

$$y(t) = Ce^{At}x(0)$$

Nós temos

$$e^{At} = \sum_{k=0}^{n-1} \alpha_k(t) A^k$$

Onde n é o grau do polinômio característico. Assim obtemos

$$y(t) = \sum_{k=0}^{n-1} \alpha_k(t) CA^k x(0)$$

Ou

$$y(t) = \alpha_0(t)Cx(0) + \alpha_1(t)CAx(0) + \dots + \alpha_{n-1}(t)CA^{n-1}x(0) \quad (\text{B.13})$$

Se o sistema é completamente observável, então, dada a saída $y(t)$ em um intervalo de tempo $0 \leq t \leq t_1$, $x(0)$ é unicamente determinado pela equação B.13. Pode ser mostrado que isso requer o ranque da matriz $nm \times n$

$$\begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

seja n .

A partir dessa análise, podemos definir a condição para a completa observabilidade como segue:

O sistema descrito pelas equações é completamente observável se e somente se a matriz for de ranque n ou tiver n vetores coluna linearmente independentes. Essa matriz é chamada de matriz de observabilidade.

$n \times nm$

$$[C * \quad A * C * \quad \dots \quad (A *)^{n-1} C *]$$

APÊNDICE C - DECOMPOSIÇÃO DE VALOR SINGULAR

A decomposição do valor singular é dado por:

Qualquer matriz A complexa $l \times m$ pode ser composta em uma decomposição de valor singular

$$A = U\Sigma V^H V^H = V^{-1} \text{ (transposta conjugada complexa de } V)$$

Onde a matriz U $l \times l$ e a matriz V $m \times m$ são matrizes unitárias e a matriz Σ $l \times m$ contém uma matriz diagonal Σ_1 com valores singulares reais, não negativos, σ_i , dispostos em ordem decrescente como em:

$$\Sigma = \begin{bmatrix} \Sigma_1 \\ 0 \end{bmatrix}; l \geq m \quad (\text{C.1})$$

$$\text{ou } \Sigma = [\Sigma_1 \quad 0]; l \leq m \quad (\text{C.2})$$

Onde,

$$\Sigma_1 \text{diag}\{\sigma_1, \sigma_2, \dots, \sigma_k\}; \quad k = \min\{l, m\} \quad (\text{C.3})$$

$$\text{e } \bar{\sigma} \triangleq \sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_k \triangleq \underline{\sigma}. \quad (\text{C.4})$$

As matrizes unitárias U e V formam bases ortonormais de um espaço coluna (saída) e linha (entrada) de A . Aqui os vetores colunas de V denotado v_i chamado à direita ou vetores singulares de entrada e os vetores colunas de U , denotado u_i , chamado à esquerda ou valores de saída.

Note que esta decomposição é não única desde que $A = U'\Sigma V'^H$, onde $U' = VS^{-1}$ e $S = \text{diag}\{e^{j\theta_i}\}$ θ_i é um número real qualquer, também é um SVD de A . Portanto, os valores singulares θ_i são únicos.

Matriz unitária: Uma matriz complexa U é unitária se

$$U^H = U^{-1} \quad (\text{C.5})$$

Todos os autovalores da matriz unitária tem valor absoluto igual a 1 e todos os seus valores singulares são, portanto igual a 1.

Os valores singulares são raízes quadradas dos $k = \min(l, m)$ que são os maiores autovalores tanto AA^H e $A^H A$ têm-se:

$$\sigma_i(A) = \sqrt{\lambda_i(A^H A)} = \sqrt{\lambda_i(AA^H)}. \quad (\text{C.6})$$

Além disso. As colunas de U e V são autovetores unitários de AA^H e $A^H A$, respectivamente. Calculando (C.6) escreve-se:

$$AA^H = (U\Sigma V^H)(U\Sigma V^H)^H = (U\Sigma V^H)(V\Sigma^H U^H) = U\Sigma\Sigma^H U^H \quad (\text{C.7})$$

E equivalentemente desde que U seja unitário e satisfaça $U^H = U^{-1}$.

$$(AA^H)^U = U\Sigma\Sigma^H, \quad (\text{C.8})$$

U é uma matriz de autovetores de AA^H e $\{\sigma_i^2\}$ são seus autovalores. Similarmente, V é uma matriz de autovalores de AA^H .

Portanto, no projeto de sistemas modernos multivariáveis, a decomposição dos valores singulares é de suma importância e apresenta algumas vantagens sobre a decomposição de autovalores para analisar ganhos e Algumas dessas vantagens são:

- 1 -Os valores singulares dão melhores informações sobre os ganhos da planta;
- 2- As direções da planta obtidas a partir da Decomposição do Valor Singular (SVD) são ortogonais;
- 3 A SVD também se aplica diretamente à planta não quadrática.

C.1 Valores singulares para desempenho

O valor máximo singular é muito útil em termos de desempenho e robustez no domínio da frequência. O ponto importante é que em baixas frequências é requerido que os valores singulares mínimos sejam grandes, enquanto que em altas frequências os valores singulares máximas sejam pequenas.

Para um sistema de controle realimentado de um grau de liberdade, a resposta de frequência em malha fechada, r , para o erro de controle $e = y - r$ é $e = Sr$. Para o sistema SISO $S(j\omega)$ é avaliado como uma função da frequência que fornece informações úteis sobre a efetividade do controle de realimentação, porque para senoídes $|e(\omega)| / |r(\omega)| = |S(j\omega)|$. Para sistemas MIMO, uma generalização útil, resulta se considerar a razão $\|e(\omega)\|_2 / \|r(\omega)\|_2$ onde $\|\cdot\|_2$ é um vetor de norma-2. Esta razão depende da direção de $r(\omega)$ e

$$\underline{\sigma}(S(j\omega)) \leq \frac{\|e(\omega)\|_2}{\|r(\omega)\|_2} \leq \sigma(S(j\omega)). \quad (\text{C.9})$$

Em termos de desempenho, é razoável exigir que a razão $\|e(\omega)\|_2 / \|r(\omega)\|_2$ permaneça pequena para qualquer direção de $r(\omega)$, incluindo “pior caso” da direção que dá razão como $\bar{\sigma}(S(j\omega))$. Deixar em cada frequência $\frac{1}{|\omega_p(j\omega)|}$ representar a magnitude máxima, permitida de $\|e\|_2 / \|r\|_2$. Isso resulta no seguinte requisito de desempenho: $\bar{\sigma}(S(j\omega)) \leq 1 / |\omega_p(j\omega)|, \forall \omega \Leftrightarrow \bar{\sigma}(\omega_p S) \leq 1, \quad \forall \omega \Leftrightarrow \|\omega_p S\|_\infty \leq 1$ (C.10)

Onde a norma H_∞ é definida como um pico de máximo valor singular de frequência de resposta $\|M(s)\|_\infty \triangleq \max \bar{\sigma}(M(j\omega))$. (C.11)

Tipicamente, eles são pequenos para baixar frequências e a realimentação é eficaz, e se aproxima de 1 em altas frequências porque qualquer sistema real é estritamente

adequado. O valor máximo singular, $\sigma(S(j\omega))$, tem geralmente um pico maior do que 1 em frequência cruzada. Este pico é indesejável, mas é inevitável para sistemas reais.

APÊNDICE D -DESEMPENHO E ESTABILIDADE ROBUSTA

Neste apêndice discute-se o desempenho e estabilidade robusta de Sistemas Dinâmicos Multivariáveis. A análise de Robustez considera perturbações externas, ruídos da medida, variações na planta e erro entrada/saída para o desempenho robusto.

D.1 Modelo do sistema Dinâmico

As relações entre as funções de transferência do compensador, planta, sinais e manipulações algébricas via lema da inversa fornece,

Equação de saída:

$$Y(s) = C(s)[R(s) - D(s)] + S(s)D(s) \quad (D.1)$$

Equação do erro:

$$E(s) = S(s)[R(s) - D(s)] + C(s)D(s) \quad (D.2)$$

sendo,

1. $S(s) \rightarrow$ sensibilidade

$$S(s) = \frac{1}{I+G(s)K(s)} \quad (D.3)$$

2. $C(s) \rightarrow$ sensibilidade complementar

$$C(s) = \frac{G(s)K(s)}{I+G(s)K(s)} \quad (D.4)$$

A análise de robustez considera a equação de saída Y e a equação do erro E , normas e valores singulares de sinais e funções de transferência são utilizados para avaliar o desempenho e a estabilidade robusta do sistema.

D.2 Robustez - Desempenho e Estabilidade

Nesta seção apresentam-se as condições que devem ser satisfeitas para que o controlador possa ser chamado de robusto. A primeira seção considera o desempenho robusto, explora-se quatro tipos de situações em que o controlador deve satisfazer a fim de garantir o ponto de operação especificado pelo projetista. A segunda seção analisa sob o ponto de vista de garantir estabilidade na ocorrência de dinâmicas não modeladas do sistema real.

D.2.1 Desempenho Robusto

O desempenho robusto é analisado para as seguintes situações: acompanhamento do sinal de referência, rejeição a distúrbio, rejeição erros de medidas e insensibilidade a

variações na planta. A robustez é avaliada em termos dos valores singulares das matrizes de ganho de malha GK e sensibilidades,

a - Acompanhamento do sinal de referência

- Os sinais $R(s)$ são em baixa frequência. O conjunto das frequências de $R(s)$,

$$\Omega_r = \{\omega \in R | \omega \leq \omega_r\}$$

- A partir das equações (B.1) e (B.2), considerar a contribuição $R(s)$ em $Y(s)$ e $E(s)$,

Saída:

$$Y(s) = C(s)R(s) \quad (D.5)$$

Erro:

$$E(s) = S^0(s)R(s) \quad (D.6)$$

A fim de facilitar uma primeira análise a saída e o erro são expressos em termos das sensibilidades,

Saída:

$$Y(s) = \frac{G(s)K(s)}{1+G(s)K(s)}R(s) \quad (D.7)$$

Erro:

$$E(s) = \frac{1}{1+G(s)K(s)}R(s) \quad (D.8)$$

Em relação a equação de saída é necessário que

$$GK \gg I$$

Para garantir que o controlador K imponha o sinal de referência na saída Y. Em termos de valores singulares significa impor

$$\sigma_m[GK] \gg 1$$

para garantir um bom acompanhamento do sinal de referência.

A condição de robustez para o erro é garantida para frequências limitadas superiormente, isto é, dado por um supremo, da equação (D.8),

$$\alpha_r(\omega) \ll 1$$

Logo,

$$\frac{\|E(s)\|}{\|R(s)\|} \ll \alpha_r(\omega) \quad (D.9)$$

]

ou

$$\frac{\|R(s)\|}{\|E(s)\|} \gg \frac{1}{\alpha_r(\omega)} \quad (\text{D.10})$$

Em termos de valores singulares,

$$\sigma_m[I + G(s)K(s)] \geq \frac{1}{\alpha_r(\omega)} \geq 1 \quad (\text{D.11})$$

A condição de robustez para o acompanhamento do sinal de referência que leva em conta os sinais de saída e erro,

$$\sigma_m[G(s)K(s)] \geq \frac{1}{\alpha_r(\omega)} \quad (\text{D.12})$$

b - Perturbações externas

- Os sinais $D(s)$ são em baixas frequências. O conjunto das frequências de $D(s)$,

$$\Omega_d = \{\omega \in R | \omega \leq \omega_d\}$$

- A partir das equações (B.1) e (B.2), considerar a contribuição $D(s)$ em $Y(s)$ e $E(s)$,

Saída:

$$Y(s) = S(s)D(s) \quad (\text{D.13})$$

Erro:

$$E(s) = C(s)D(s) \quad (\text{D.14})$$

De maneira similar ao acompanhamento do sinal de referência. A condição de robustez para que o controlador garanta a rejeição para perturbações externas,

$$\sigma_m[G(s)K(s)] \geq \frac{1}{\alpha_d(\omega)} \quad (\text{D.15})$$

O acompanhamento do sinal de referência leva em conta os sinais de saída e de erro.

c - Insensibilidade a variações na planta

- Os sinais são em baixas frequências. O conjunto das frequências de $\delta G(s)$,

$$\Omega_\delta = \{\omega \in R | \omega \leq \omega_\delta\}$$

- A partir da equação (B.1) considerar a contribuição $R(s)$ em $Y(s)$,

Saída:

$$Y(s) = C(s)R(s) \quad (\text{D.16})$$

Ainda,

$$Y(s) = \frac{G(s)K(s)}{I+G(s)K(s)} R(s) \quad (D.17)$$

Os efeitos das variações da planta $G(s)$ na saída $Y(s)$ são modelados através da equação:

$$\delta Y(s) = \frac{\delta G(s)G(s)^{-1}}{I+G(s)K(s)} Y(s) \quad (D.18)$$

A idéia consiste em determinar uma expressão que indique uma estratégia para assegurar desempenho robusto para variações na planta. Manipulando a equação (D.15),

$$\frac{\delta Y(s)}{Y(s)} = \frac{\delta G(s)G(s)^{-1}}{I+G(s)K(s)} \quad (D.19)$$

$$\frac{\frac{\delta Y(s)}{Y(s)}}{\delta G(s)G(s)^{-1}} = \frac{1}{I+G(s)K(s)} \quad (D.20)$$

A equação é expressa em termos das normas das variações na saída e na planta sendo limitada pelo seu maior valor singular,

$$\frac{\frac{\|\delta Y(s)\|}{\|Y(s)\|}}{\|\delta G(s)G(s)^{-1}\|} \leq \frac{1}{\sigma_M[I+G(s)K(s)]} \quad (D.21)$$

Considerando que o conjunto de frequências deve satisfazer a restrição (B.18) e deve possuir um limitante que restrinja a propagação de $\delta G(s)$ em $Y(s)$,

$$\alpha_\delta(\omega) \ll 1$$

A desigualdade (B.18),

$$\frac{\frac{\|\delta Y(s)\|}{\|Y(s)\|}}{\|\delta G(s)G(s)^{-1}\|} \leq \alpha_\delta(\omega) \quad (D.22)$$

As desigualdades (B.18) e (B.19) podem ser manipuladas de forma que se tenha uma relação que permite verificar a robustez em questão,

$$\sigma_m[I + G(s)K(s)] \geq \frac{1}{\alpha_\delta(\omega)} \gg 1 \quad (D.23)$$

Uma aproximação,

$$\sigma_m[G(s)K(s)] \geq \frac{1}{\alpha_\delta(\omega)} \gg 1 \quad (D.24)$$

d - Erro de Medida

- Instrumentos de medição introduzem erros nas altas-frequências. O conjunto das frequências de $N(s)$,

$$\Omega_N = \{\omega \in R | \omega \geq \omega_N\}$$

- A partir da equação (B.1), considerar a contribuição $N(s)$ em $Y(s)$,

Saída:

$$Y(s) = -C(s)N(s) \quad (\text{D.25})$$

Ainda,

$$Y(s) = -\frac{G(s)K(s)}{I+G(s)K(s)}N(s) \quad (\text{D.26})$$

A condição de que deve ser satisfeita para garantir robustez a rejeição ao ruído,

$$G(s)K(s) \ll I \quad (\text{D.27})$$

Então, a matriz $G(s)K(s)$ é pequena se o maior valor singular é pequeno,

$$\sigma_M[G(s)K(s)] \ll 1 \quad (\text{D.28})$$

De maneira similar ao acompanhamento do sinal de referência. A condição de robustez para que o controlador garanta a rejeição para perturbações externas,

$$\sigma_M[G(s)K(s)] \geq \frac{1}{\alpha_d(\omega)} \quad (\text{D.29})$$

APÊNDICE E – RECUPERAÇÃO DE MALHA DE TRANSFERÊNCIA-RFTM

A introdução de um estimador em uma malha de controle por realimentação de estados pode afetar adversamente as propriedades de estabilidade robusta do sistema [isto é, as propriedades de margem de fase (PM) e de margem de ganho (GM) podem se tornar arbitrariamente pobres, como mostrado pelo famoso exemplo de Doyle (Doyle, 1978)]. No entanto, é possível modificar o projeto do estimador de forma a tentar “recuperar” as propriedades da estabilidade robusta LQR em certa medida. Esse processo é chamado de recuperação de função de transferência de malha (RFTM) e é especialmente eficaz para sistemas de fase mínima. Para realizar a recuperação, alguns dos polos do estimador são colocados nos zeros da planta (ou próximo deles) e os polos restantes são movidos para o SPE (suficientemente longe). A ideia por trás da RFTM é reprojeter o estimador de forma a moldar as propriedades do ganho de malha, aproximando-as das propriedades do LQR.

O uso da RFTM significa que os controladores por realimentação podem ser projetados para alcançar as funções desejadas de sensibilidade $[S(s)]$ e de sensibilidade complementar $[T(s)]$ nos pontos críticos (pontos de quebra da malha) do sistema realimentado (por exemplo, em uma entrada ou em uma saída da planta). Claro, há um preço a ser pago por esta melhoria em termos da estabilidade robusta. O novo sistema de controle projetado pode ter piores propriedades de sensibilidade ao ruído no sensor. Intuitivamente, pode se pensar em fazer (alguns) os polos dos estimadores arbitrariamente rápidos para que o ganho de malha seja aproximadamente o ganho do LQR. De forma alternativa, essencialmente pode-se pensar em “inverter” a função de transferência da planta de modo que todos os polos no SPE da planta sejam cancelados pelo compensador dinâmico para obter a forma da malha desejada. Obviamente há custos, e o projetista precisa ter o cuidado de fazer a escolha correta para o problema dado, dependendo das especificações do sistema de controle.

Hoje, a técnica RFTM é bem conhecida, e procedimentos específicos para o projeto prático foram identificados (Athans, 1986; Stein e Athans, 1987; Sabieri et al. 1993). OS mesmos procedimentos também podem ser aplicados a sistemas de fase não mínima, mas não há garantia sobre extensão da possível recuperação.

A técnica RFTM pode ser vista como um procedimento sistemático para estudar os custos de projeto do compensador linear quadrático (Doyle e Stein, 1981).

Sendo w e v processos Gaussianos brancos não correlacionados e com média zero e ruído no sensor com matrizes de covariância $R_w \geq 0$ e $R_v \geq 0$. O projeto do estimador fornece

$$\begin{aligned}\dot{\hat{x}} &= F\hat{x} + Gu + L(y - \hat{y}) \\ \hat{y} &= H\hat{x}\end{aligned}$$

Resultando no compensador dinâmico usual

$$D_c(s) = -K(sI - F + GK + LH)^{-1}L$$

Agora, vamos tratar os parâmetros de ruído, R_w e R_v , como projetos das “arrestas” no projeto do compensador dinâmico. Sem perda de generalidade, vamos escolher $R_w = TT^T$ e $R_v = 1$. Para a RFTM, assumamos que $T = qG$, onde q é um parâmetro de projeto escalar. O projeto do estimador, então, é baseado no projeto específico dos parâmetros R_w e R_v . Pode-se demonstrar que, para um sistema de fase mínima, à medida que q se torna grande (Doyle e Stein, 1979),

$$\lim_{q \rightarrow \infty} D_c(s)G(s) = K(sI - F)^{-1}G$$

Esta é precisamente a razão da recuperação plena de função de transferência de malha não ser possível para um sistema de fase não mínima. Este comportamento limitador pode ser explicado usando o lugar das raízes simétrico. Quando $q \rightarrow \infty$, alguns dos polos do estimador se aproximam dos zeros de

$$G_e(s) = H(sI - F)^{-1}T$$

E os restantes tendem ao infinito. Na prática, o procedimento de projeto RFTM ainda pode ser aplicado a uma planta de fase não mínima. O grau de recuperação vai depender da localização específica dos zeros de fase não mínima. Uma recuperação suficiente pode ser possível em muitas frequências se os zeros no SPD estiverem localizados fora da largura de banda em malha fechada especificada. Limites no desempenho de sistemas realimentados em razão dos zeros no SPD são discutidos em Freudenberg e Looze (1985).