

ecai

Universidade Federal de Santa Catarina
Centro Tecnológico
Curso de Engenharia de Controle e
Automação Industrial

ufsc

Aplicação de Técnicas de Controle Adaptativo a Sistemas de Potência

*Monografia submetida à Universidade Federal de Santa Catarina
como requisito para a aprovação da disciplina:
EEL 5901: Projeto de Fim de Curso*

Nicole Sharon Orfali

Florianópolis, Julho de 1996

Aplicação de Técnicas de Controle Adaptativo a Sistemas de Potência

Nicole Sharon Orfali

Esta monografia foi julgada no contexto da disciplina
EEL 5901: Projeto de Fim de Curso
e aprovada na sua forma final pelo
Curso de Engenharia de Controle e Automação Industrial

Banca Examinadora:

Prof. Aguinaldo Silveira e Silva
Orientador do Curso

Prof. Augusto Humberto Bruciapaglia
Responsável pela disciplina e Coordenador do Curso

Prof. Antônio Augusto Rodrigues Coelho, Avaliador

Fábio Alonso da Silva, Debatedor

Gilberto da Cunha, Debatedor

Agradecimentos

Agradeço a Aguinaldo Silveira e Silva pela competência com que orientou este trabalho.

Ao Prof. Eraldo Silveira e Silva pela contribuição no desenvolvimento da comunicação serial, sem a qual não seria possível a consecução deste projeto.

A Alessandro Manzoni pelas inúmeras horas dispendidas em função da realização deste projeto, além de sua inteira disponibilidade para o esclarecimento de quaisquer dúvidas.

A minha família pelo amor e carinho compartilhados nesta longa jornada.

A meu noivo Wallace pelo eterno apoio, amor e compreensão.

Resumo

Este trabalho trata da aplicação de métodos de controle adaptativo auto-ajustável para síntese dos Estabilizadores de Sistemas de Potência, a fim de resolver o problema das oscilações eletromecânicas de sistemas elétricos de potência.

O método proposto é indireto, sendo primeiramente realizada a estimação paramétrica do modelo que representa o sistema. Nesta estimação paramétrica são usados algoritmos baseados no método dos mínimos quadrados recursivo, devido sua simplicidade e comprovada eficácia. Em seguida, é realizada a implementação do controlador, usando o modelo estimado, através da técnica de controle por alocação de pólos, sendo que um fator de deslocamento radial é usado para o deslocamento dos pólos, em direção ao centro do plano Z.

Um esquema de controle utilizando dois microcomputadores do tipo PC interligados é mostrado. Em um dos microcomputadores é colocado um simulador de sistemas de potência, que permite simular a dinâmica rápida e lenta deste tipo de sistema. No outro microcomputador, são implementadas as rotinas de identificação e controle. O controlador adaptativo usa como entrada o sinal proveniente do simulador e gera o sinal de controle.

Por fim os resultados obtidos com testes de simulação de dois sistemas de potência são apresentados. O bom desempenho obtido nestas simulações demonstra a potencialidade dos controladores adaptativos auto-ajustáveis.

Abstract

This work applies adaptive control methods to designing Power System Stabilizers, in order to damp electromechanical oscillations in power systems.

An indirect method is proposed in which the system parameters are initially estimated. The estimation method is the recursive least squares, which is a well-tested method. The control is then implemented, using the estimated model, through a pole placement technique, shifting the poles towards the Z-plane center by a shifting factor.

The control scheme uses two PC computers. A power system simulator, which can simulate power systems fast and slow dynamics runs in one of these computers. In the other computer the identification and control procedures are implemented. The adaptive controllers uses the power system simulator output and generates the control signal.

Simulation results are presented for two power systems and indicate the potential of adaptive control techniques for electrical energy systems.

Sumário

1. INTRODUÇÃO	2
2. SISTEMAS DE POTÊNCIA.....	5
2.1 INTRODUÇÃO	5
2.2 O PROBLEMA DE CONTROLE DE SISTEMAS DE POTÊNCIA.....	5
2.3 MODELAGEM DE SISTEMAS DE POTÊNCIA	8
2.3.1 <i>Componentes básicos de um sistema elétrico de potência</i>	11
2.3.1.1 Gerador Síncrono	11
2.3.1.2 Reguladores Automáticos	12
2.3.1.3 Cargas.....	12
2.3.1.4 Linhas de transmissão	12
2.4 CONCLUSÃO	13
3. CONTROLE ADAPTATIVO E APLICAÇÕES EM SISTEMAS DE POTÊNCIA.....	14
3.1 INTRODUÇÃO	14
3.2 IDENTIFICAÇÃO DE SISTEMAS.....	14
3.3 IDENTIFICAÇÃO RECURSIVA DOS PARÂMETROS	15
3.3.1 <i>O Método dos Mínimos Quadrados</i>	15
3.3.2 <i>Mínimos Quadrados Recursivo</i>	17
3.4 ESTRATÉGIAS DE CONTROLE	18
3.4.1 <i>Controlador por Alocação de Pólos Auto-Ajustável</i>	19
3.5 ALGUMAS CONSIDERAÇÕES PRÁTICAS	22
3.6 CONCLUSÃO	26
4. IMPLEMENTAÇÃO E SIMULAÇÃO DIGITAL	27
4.1 INTRODUÇÃO	27
4.2 ESQUEMA DE CONTROLE	27
4.3 IMPLEMENTAÇÃO	28
4.3.1 <i>Comunicação serial</i>	29
4.3.1.1 Portas de E/S.....	29
4.3.1.2 RS-232.....	30
4.3.1.3 Transferência/Recebimento de Dados	33
4.3.2 <i>Interface C-Matlab</i>	35
4.3.2.1 Arquivos MEX	36
4.3.2.2 Fluxo de informação C-Matlab/Matlab-C.....	36
4.4 CONCLUSÕES	37
5. RESULTADOS.....	38
5.1 INTRODUÇÃO	38
5.2 DESCRIÇÃO DA IMPLEMENTAÇÃO DO CONTROLE	38
5.2 SISTEMA DE DUAS BARRAS	39
5.2.1 <i>Implementação</i>	40
5.3 SISTEMA DE 9 BARRAS	43
5.3.1 <i>Implementação</i>	44
5.4 CONCLUSÕES	46
6. CONCLUSÕES GERAIS.....	47
REFERÊNCIAS BIBLIOGRÁFICAS	49
APÊNDICE	51

1. Introdução

Um sistema elétrico de potência, constantemente variando seu ponto de operação de modo a atender a demanda de energia, está sujeito a oscilações nos eixos dos rotores das máquinas geradoras. Tais oscilações, na faixa de 0,2 a 2,5 Hz são denominadas oscilações eletromecânicas e estão associadas a modos de oscilação ditos dominantes do sistema.

A ocorrência das oscilações eletromecânicas pouco amortecidas depende fundamentalmente da ação dos reguladores de tensão, principalmente os de ganhos altos e respostas rápidas.

De forma a melhorar, de maneira geral, a qualidade da energia entregue aos consumidores, faz-se uso de estabilizadores de sistemas de potência (ESP's) que são instalados junto às máquinas do sistema. Estes controladores devem ter seus parâmetros corretamente ajustados, de forma que o sistema de potência tenha o desempenho almejado. O projeto deste tipo de controlador é baseado em modelos lineares que representam o sistema em determinada condição típica de operação. Ocorre porém, que os Sistemas de Potência são fortemente não-lineares, sendo sua condição de operação variável durante seu funcionamento normal, o que pode contribuir para degradar o desempenho de seus controladores.

O rápido desenvolvimento da tecnologia de microcomputadores tornou possível o uso de controladores digitais em aplicações industriais em todos os níveis, inclusive em Sistemas Elétricos de Potência. Confiabilidade, capacidade de tomar decisões e flexibilidade nos algoritmos de controle são as maiores vantagens dos sistemas digitais de controle. Como o processamento de dados nos controladores digitais é feito de maneira direta e fácil, métodos de controle antes considerados demasiadamente complexos, entre eles o controle adaptativo, podem ser implementados a um custo razoavelmente baixo.

Um controlador adaptativo ajusta automaticamente os parâmetros do controlador em função do comportamento do processo, de modo a garantir as especificações de funcionamento. A estrutura adaptativa auto-ajustável é uma filosofia de projeto para monitorar variações no tempo e inserir estas mudanças no procedimento do projeto. O controle adaptativo pode ser usado como alternativa no tratamento de processo complexos como por exemplo: processos não-lineares, processos com atrasos distintos, processos com dinâmica assimétrica e processos com acoplamento entre variáveis. Devido a estas características um controlador adaptativo pode ser usado com sucesso na resolução do problema das oscilações eletromecânicas, em Sistemas de Potência.

O primeiro passo no projeto de um controlador é a obtenção de um modelo que descreva adequadamente o comportamento do processo. O método que consiste na determinação de um modelo matemático que represente os aspectos essenciais de um dado sistema, caracterizado por sinais de entrada e saída relacionadas por uma função de transferência é denominado *Identificação de Sistemas*.

O controle auto-ajustável necessita da contínua identificação dos parâmetros variáveis do modelo utilizado para o processo. A identificação em

tempo real, que segue a variação dos parâmetros identificados, chama-se identificação recursiva. Inúmeros métodos de identificação recursiva têm sido descritos na literatura, muitos deles bastante similares, com pequenas diferenças de acordo com a aplicação. O método de identificação recursiva mais utilizado é obtido a partir de um método de identificação *on-line*: o *método dos mínimos quadrados recursivo*. A preferência por este método em aplicações de controle adaptativo a Sistemas de Potência advém de sua simplicidade, tanto conceitual quanto a nível de implementação, e de sua comprovada eficácia.

Os controladores auto-ajustáveis são baseados na imposição do princípio da equivalência certa, ou seja, calculam a lei de controle supondo os parâmetros do processo conhecidos e em seguida os substitui pelos valores estimados. Na literatura são conhecidas várias técnicas de controle auto-ajustável. Entre elas podem ser citadas as técnicas de *variância mínima* e *variância mínima generalizada*, que calculam os parâmetros de controle de modo a minimizar uma função-custo; e a técnica de *alocação de pólos* que reposiciona os pólos de malha fechada para que o sistema tenha as propriedades especificadas. Em Sistemas de Potência, a técnica de controle por alocação de pólos é largamente utilizada. O reposicionamento dos pólos é feito através de um fator de redução, com valor entre 0 e 1. Este fator de redução faz com que haja um deslocamento dos pólos de malha fechada em direção à origem do plano Z, e conseqüentemente um maior amortecimento.

O objetivo deste projeto de fim de curso é o de desenvolver estabilizadores de sistemas de potência, usando técnicas de controle adaptativo auto-ajustável. A implementação prática deste projeto visa a observação de aspectos relevantes da implementação real.

Para a implementação do projeto, um esquema de controle foi desenvolvido. Foram utilizados dois microcomputadores do tipo PC, interligados entre si, comunicando-se através das portas seriais de seus processadores. Em um dos computadores foi colocado um simulador de sistemas elétricos de potência, que permite simular a dinâmica rápida e lenta de Sistemas de Potência. Este simulador foi desenvolvido em um trabalho de mestrado, usando programação orientada objeto, em linguagem C⁺⁺. No outro microcomputador foram implementadas as rotinas de identificação e controle. Este controlador adaptativo auto-ajustável é utilizado para controlar o sistema simulado usando como entrada o sinal proveniente do simulador.

As rotinas de identificação e controle foram desenvolvidas em linguagem Matlab, devido a facilidade oferecida por este software para a execução de cálculos matemáticos, entre eles, os cálculos matriciais, requeridos pelas rotinas de controle adaptativo. Como o Matlab não possui ferramentas para o acesso das portas seriais, por onde os computadores trocam as informações, a construção de módulos de envio e recepção de mensagem em linguagem C foi necessária, dado que esta linguagem permite este acesso. Uma interface entre as linguagens C e Matlab então tornou-se necessária.

O sistema de controle desenvolvido neste trabalho funciona da seguinte forma: em um intervalo de amostragem, o simulador envia o sinal através da comunicação serial, até o outro computador. O Matlab recebe este dado e

calcula o valor de controle, que é enviado ao simulador pela comunicação serial, no próximo intervalo de amostragem.

O objetivo primordial deste projeto fim de curso foi o de constatar e comprovar a eficácia e aplicabilidade dos controladores adaptativos auto-ajustáveis, principalmente quando são aplicados a Sistemas Elétricos de Potência.

Esta monografia é organizada da seguinte forma: no capítulo 2 são apresentados os componentes básicos de sistemas de potência e o problema das oscilações eletromecânicas é colocado. Alguns modelos para estes componentes são mostrados. No capítulo 3 um sumário da teoria de controle adaptativo auto-ajustável é apresentada. Os métodos de identificação recursiva e as técnicas de controle aplicados a sistemas de potência são descritos. Detalhes relevantes para a implementação de controladores adaptativos auto-ajustáveis são citados e discutidos. No Capítulo 4 a comunicação serial e a interface entre as linguagens C e Matlab são mostradas. Detalhes das implementações são colocados. No Capítulo 5 são mostrados e analisados os resultados de testes feitos com dois sistemas de potência distintos, usando controladores adaptativos. Os resultados mostram o bom desempenho dos controladores adaptativos auto-ajustáveis.

2. Sistemas de Potência

2.1 Introdução

Neste capítulo, os principais aspectos do problema de controle de Sistemas Elétricos de Potência são descritos. Os componentes utilizados nos sistemas de potência e modelos matemáticos no domínio do tempo para cada um destes componentes são apresentados de forma sucinta. Em seguida, é apresentada a modelagem do sistema de potência como um todo, que resulta em um sistema singular de equações não-lineares. A modelagem apresentada se adequa ao problema da estabilidade dinâmica dos sistemas de potência, tratado neste trabalho.

Os sistemas de potência são de alta ordem, complexos e fortemente não-lineares. Na prática industrial de sistemas de potência, o projeto de controladores é baseado em modelos simplificados e linearizados no domínio da frequência para o sistema. Modelando o sistema desta forma, o problema pode ser colocado no contexto de controle clássico.

2.2 O Problema de Controle de Sistemas de Potência

Diz-se que um sistema de potência encontra-se no estado normal de operação quando as seguintes condições são cumpridas:

- A demanda de todas as cargas alimentadas pelo sistema é satisfeita;
- A frequência é mantida constante em seu valor nominal (60 Hz).

Manter a frequência constante e igual ao seu valor nominal é importante por uma série de motivos, por exemplo, o desempenho da maioria dos motores de corrente alternada industriais é função da frequência. Porém, o motivo mais importante para manter a frequência igual ao seu valor nominal é o fato de que ela é um indicador de que o balanço de potência ativa está sendo adequadamente cumprido, isto é, a potência ativa fornecida pelos geradores do sistema é igual a potência ativa solicitada pelas cargas.

Quanto à tensão, pode-se igualmente dizer que o desempenho de vários componentes da carga é também fortemente ligado à tensão, além do fato de que a tensão, de maneira análoga à frequência para potência ativa, é um indicador do balanço de potência reativa.

São três os principais sistemas de controle que atuam sobre o gerador síncrono:

1. Controle primário de velocidade;
2. Controle automático de geração;
3. Controle de excitação.

O controle primário de velocidade, que é local, basicamente monitora a velocidade do eixo do conjunto turbina-gerador e controla o torque mecânico da turbina de modo a fazer com que a potência elétrica gerada pela unidade se adapte às variações de carga.

Como a atuação do controle primário normalmente resulta em desvios de frequência, é necessário que se conte com a atuação de um outro sistema de controle para restabelecer a frequência ao seu valor nominal. Este sistema é chamado Controle Automático da Geração que, no caso de sistemas interligados, tem ainda a incumbência de manter o intercâmbio de potência entre concessionárias vizinhas tão próximo quanto possível dos valores previamente programados.

Finalmente, os objetivos do controle de excitação, que é local, pode ser sumarizados como: a) manter a tensão terminal do gerador dentro das tolerâncias especificadas; b) regular fluxo de reativo entre máquinas; c) amortecer as oscilações do rotor da máquina quando da ocorrência de perturbações do sistema. Esta última função de regulador de tensão advém do fato de que a tensão do campo do gerador afeta significativamente o torque elétrico da máquina. O diagrama das malhas de controle de tensão e velocidade da máquina síncrona é apresentado na Figura 2.1 e o diagrama do controle de tensão e do estabilizador de sistemas de potência é mostrado na Figura 2.2.

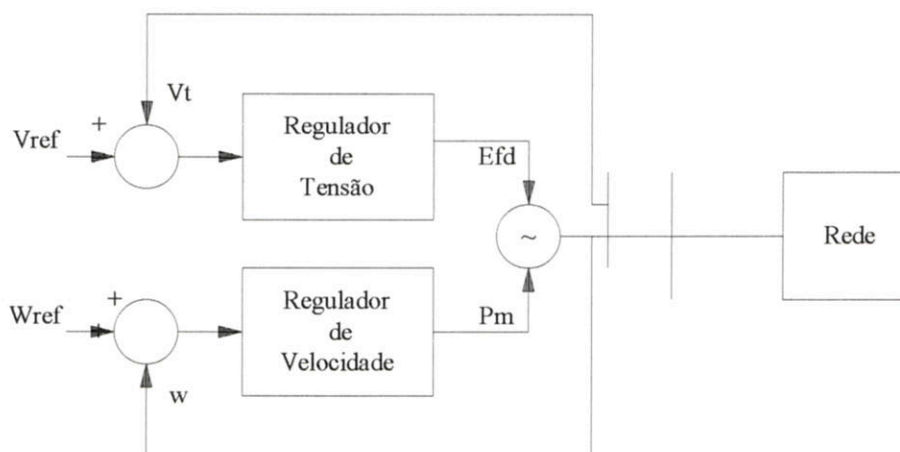


Figura 2.1- Malhas de Controle de Tensão e Velocidade da Máquina Síncrona.

A observação de um sistema multimáquinas qualquer, isento de grandes perturbações e operando na condição de regime permanente, nos mostra que este sistema sofre ao longo do tempo de pequenas alterações em torno do seu ponto de operação. Estas alterações decorrem das constantes entradas e saídas aleatórias de cargas e equipamentos do sistema.

Imediatamente após a ocorrência de cada uma destas pequenas perturbações, o desbalanço existente entre a carga e a geração total do sistema faz com que o fluxo magnético concatenado entre rotor e estator das

máquinas sofra rearranjos e induz nos enrolamentos acoplados ao rotor correntes cujo fluxo resultante procura opor-se a estas variações.

Esta constante oposição exercida pelo rotor às variações em seu ângulo de carga, encerra, de maneira geral, uma ação amortecedora da máquina em relação à rede, no intervalo de tempo em que o sistema elétrico migra de um ponto de operação ao outro ponto. Em função disto, sem a atuação de qualquer outro equipamento ativo, as respostas dos sistemas elétricos quando submetidos à pequenas perturbações tendem a ser perfeitamente estáveis.

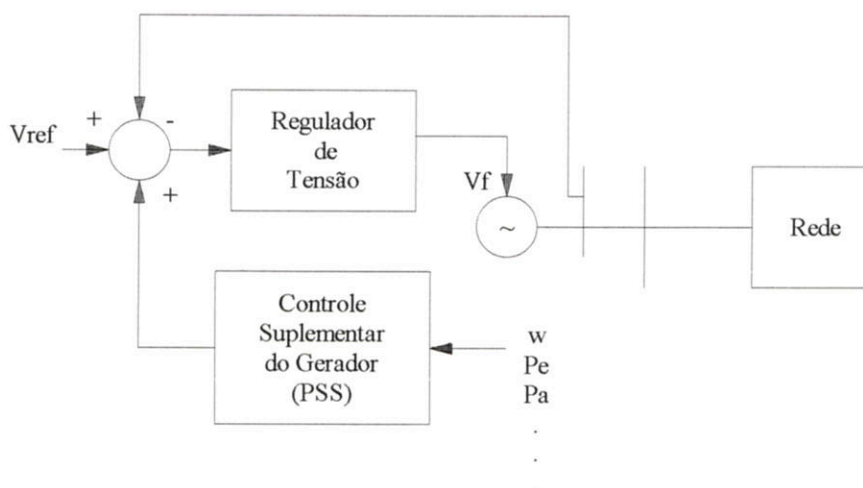


Figura 2.2 - Controle de Tensão e Estabilizador de Sistemas de Potência.

Em um sistema elétrico operando em regime permanente, a potência mecânica fornecida pelo eixo de cada grupo gerador é igual a potência elétrica exigida na saída de seu estator, uma vez que existe equilíbrio entre a carga e a geração total do sistema.

Face às pequenas perturbações resultantes da entrada/saída de cargas ou equipamentos do sistema, o desequilíbrio que passa a existir entre as potências mecânica e elétrica, em cada gerador, faz com que a potência acelerante em cada máquina seja diferente de zero. Supondo valores de potência acelerantes não nulos, as máquinas passam a acelerar, quando a potência mecânica torna-se maior que a elétrica, ou desacelerar, no caso inverso, segundo transitório normalmente oscilatório, que tende a estabilização caso sejam considerados apenas os comportamentos livres das máquinas.

Estes aspectos confirmam que o amortecimento intrínseco das máquinas, oferece uma importante contribuição à manutenção da estabilidade do sistema elétrico. Entretanto, a atuação dos reguladores de tensão das máquinas, quando suas malhas de controle de tensão são fechadas introduz elementos que podem afetar fortemente o efeito de reação de armadura e praticamente eliminam em alguns casos, o efeito benéfico do amortecimento natural da própria máquina. Tal influência ocorre principalmente no caso de reguladores de tensão rápidos e de ganhos altos.

Em função desta redução, torna-se comum em alguns sistemas com fracas interligações entre blocos de geração e carga, o surgimento de oscilações eletromecânicas pouco amortecidas (amortecimentos menores que 5%), que situam-se na faixa entre 0.5 e 2.0 Hz e que podem de uma maneira geral serem caracterizadas como oscilações de modo local (uma máquina oscilando contra o restante do sistema) ou oscilações interárea (grupos distintos de máquinas oscilando entre si). Estas oscilações, que não podem ser toleradas na operação diária, terminam por impor severos limites ao intercâmbio de potência entre regiões do sistema, resultando muitas vezes em uma subutilização dos recursos de geração e/ou transmissão.

Dada a impossibilidade de realização de rápidas alterações na malha do sistema (construção de novas linhas de transmissão, por exemplo), o recurso da instalação de sinais adicionais estabilizadores acoplados aos reguladores de tensão das máquinas vem sendo utilizado em larga escala atualmente.

A função básica de tais equipamentos é a de estabelecer, para uma determinada faixa de frequências pré-estabelecidas, sinais complementares à tensão de referência do regulador de tensão, originando torque elétrico com as variações de velocidade do rotor e aumentando assim o amortecimento de alguns modos de oscilação indesejados. A eliminação destes modos resulta em aumento da margem de estabilidade do sistema e permite, portanto, a prática de maiores intercâmbios de potência entre as regiões elétricas. A utilização dos sinais estabilizadores pode, assim, desprezar ou até mesmo substituir grandes investimentos no reforço de sistemas de transmissão.

Neste contexto, torna-se importante conhecer, para cada modo de oscilação quais máquinas são as mais indicadas para a instalação de um sinal estabilizador, de forma que maior proveito possa ser tirado dos mesmos, tanto no amortecimento do modo local quanto dos modos interárea.

Em uma concessionária típica de geração e transmissão de energia elétrica, os estudos de controle e estabilidade são importantes em diversos setores dentro da empresa. Na operação, por exemplo, é freqüente a realização de estudos de análise de estabilidade para os quais são importantes ajustes adequados dos diversos controladores do sistema. É portanto necessário que os engenheiros que atuam neste setor disponham de meios para projetar adequadamente os parâmetros dos diversos controladores, bem como detenham conhecimentos necessários para analisar os resultados de simulações dinâmicas do sistema e propor alterações em ajustes previamente estabelecidos.

2.3 Modelagem de Sistemas de Potência

Aborda-se nesta seção a modelagem matemática adotada para os equipamentos que compõem os sistemas elétricos de potência aqui estudados.

Em função da natureza do problema, são utilizados modelos linearizados, cujas equações organizadas de maneira adequada, originam as matrizes Jacobianas e de Estados, que serão por sua vez empregadas para representar os sistemas de potência estudados.

Analisar a estabilidade de um sistema de potência significa verificar se após a ocorrência de perturbações de qualquer ordem, as correntes, tensões e velocidades das máquinas se restabelecem dentro dos limites toleráveis.

O estudo da estabilidade do sistema quando tratado de forma genérica, requer equacionamento matemático sofisticado e envolve invariavelmente a resolução simultânea de um conjunto de equações diferenciais e algébricas, que descrevem o comportamento deste sistema ao longo do tempo, como segue:

$$\dot{x} = f(x, z)$$

$$0 = g(x, z)$$

onde:

\dot{x} é um vetor das derivadas das variáveis de estado;

x é um vetor contendo as variáveis de estado;

z é um vetor contendo as variáveis algébricas;

f e g são vetores de funções não-lineares.

O fato do sistema sofrer ao longo do tempo somente pequenas alterações em torno do seu ponto de operação, de maneira que cada conjunto de novos valores de suas variáveis característica esteja bastante próximo do anterior, permite que tais alterações sejam estudadas através de um modelo matemático linearizado. Esta simplificação, além de conduzir a uma significativa redução no esforço computacional envolvido na solução do problema, sem comprometimento dos resultados finais, possibilita também a utilização do grande ferramental de análise e síntese associados aos sistemas lineares.

Ressalta-se ainda, que a análise da estabilidade à pequenas perturbações de sistemas contendo grande número de geradores, torna-se praticamente inviável caso a mesma seja realizada utilizando-se um equacionamento não-linear.

Partindo-se então de um conjunto de equações como as que seguem, pode-se através da linearização em torno de um ponto de operação, obter:

$$\Delta \dot{x} = \left. \frac{\partial f}{\partial x} \right|_{(x^0, z^0)} \cdot \Delta x + \left. \frac{\partial f}{\partial z} \right|_{(x^0, z^0)} \cdot \Delta z$$

$$0 = \left. \frac{\partial g}{\partial x} \right|_{(x^0, z^0)} \cdot \Delta x + \left. \frac{\partial g}{\partial z} \right|_{(x^0, z^0)} \cdot \Delta z$$

Definindo:

$$J_A = \left. \frac{\partial f}{\partial(x)} \right|_{(x^0, z^0)}, \quad J_B = \left. \frac{\partial f}{\partial(z)} \right|_{(x^0, z^0)}$$

$$J_C = \left. \frac{\partial g}{\partial(x)} \right|_{(x^0, z^0)}, \quad J_D = \left. \frac{\partial g}{\partial(z)} \right|_{(x^0, z^0)}$$

Tem-se:

$$\begin{bmatrix} \dot{\Delta x} \\ 0 \end{bmatrix} = \begin{bmatrix} J_A & J_B \\ J_C & J_D \end{bmatrix} \cdot \begin{bmatrix} \Delta x \\ \Delta z \end{bmatrix}$$

onde as submatrizes J_A , J_B , J_C e J_D resultam da linearização das equações diferenciais e algébricas que representam a dinâmica dos elementos que integram os sistemas de potência. A matriz formada pelo conjunto destas submatrizes é a chamada Matriz Jacobiana.

A eliminação na matriz Jacobiana das linhas relativas às equações algébricas, conduz a obtenção da matriz de estados do sistema, que permite o estudo individualizado do comportamento de cada estado do sistema. Deste modo, a partir da equação tem-se:

$$\dot{\Delta x} = J_A \cdot \Delta x + J_B \cdot \Delta z$$

$$0 = J_C \cdot \Delta x + J_D \cdot \Delta z \Rightarrow -J_D^{-1} \cdot J_C \cdot \Delta x = \Delta z$$

substituindo, obtêm-se:

$$\dot{\Delta x} = (J_A - J_B \cdot J_D^{-1} \cdot J_C) \cdot \Delta z$$

onde a matriz formada por $(J_A - J_B \cdot J_D^{-1} \cdot J_C)$ é a matriz de estados A que contém de forma sintetizada as mesmas informações que a matriz Jacobiana a respeito da dinâmica do sistema que representa.

A composição da matriz de estados A juntamente com as matrizes de sinais de entrada (matriz B) e a de sinais de saída (matriz C), formam um sistema de equações diferenciais de primeira ordem, que descrevem o comportamento de um sistema linearizado e invariante no tempo, conforme segue:

2.3.1 Componentes básicos de um sistema elétrico de potência

Os componentes básicos utilizados para geração de energia elétrica são o gerador síncrono com seus controladores e a turbina que transmite a energia mecânica ao eixo do gerador. O regulador de tensão e o regulador de velocidade são os mais importantes para a análise da estabilidade dinâmica. A energia assim gerada é transmitida ao ponto de consumo pelas linhas de transmissão. As indústrias, residências e outros consumidores constituem as cargas do sistema. Estes são, portanto, os componentes de um sistema de potência que devem ser conhecidos e modelados matematicamente ao se fazer um estudo da estabilidade dinâmica.

2.3.1.1 Gerador Síncrono

A maior parte dos problemas de dinâmica dos sistemas de potência está associada à dinâmica do gerador síncrono. Isto faz com que sua modelagem seja a mais importante dentre aquelas dos componentes básicos de um sistema de potência. Seu modelo é também o mais complexo dentre eles.

O gerador síncrono transforma a energia mecânica de rotação de seu eixo em energia elétrica. A corrente contínua que percorre os enrolamentos do rotor induz tensão alternada nos enrolamentos fixos do estator segundo a lei de Faraday.

Cada usina do sistema reúne diversos geradores. Como os geradores de uma mesma usina são semelhantes, operam sob as mesmas condições e estão fortemente acoplados eletricamente, é natural que se comportem de maneira muito similar. Este fato realmente se verifica, o que permite a modelagem de todos os geradores de uma mesma usina como um único gerador equivalente. Na verdade, mesmo usinas eletricamente próximas, ou seja, conectadas por reatâncias relativamente pequenas, são usualmente agrupadas em um único gerador equivalente.

As equações que descrevem a máquina síncrona são:

$$\begin{aligned}\dot{\delta} &= \omega \\ \frac{2H}{\omega_r} \dot{\omega} &= P_m - P_e \\ \frac{dE_q}{dt} &= -\frac{E_q - (x_d - x'_d)I_d E_q}{T_d}\end{aligned}$$

onde a potência elétrica é dada por:

$$P_e = E_q I_q - (x_d - x'_d) I_d I_q$$

As variáveis I_d e I_q são eliminadas usando as equações de conexão da máquina à rede.

2.3.1.2 Reguladores Automáticos

Aos geradores síncronos estão associados reguladores automáticos de velocidade e tensão. A medida que os problemas de estabilidade tornam-se mais críticos a inclusão dos modelos destes reguladores se torna mais importante na simulação do comportamento do sistema de potência. Ambos os tipos de reguladores consistem de sensores, comparadores e amplificadores que devem ser levados em conta na modelagem.

Regulador de tensão

O regulador de tensão tem por objetivo reduzir as flutuações na tensão terminal do gerador. Isto ocorre através da atuação sobre a corrente de campo, ou seja, o regulador de tensão atua sobre o sistema de excitação do gerador. Vários tipos de reguladores de tensão associados a diferentes tipos de sistema de excitação são utilizados em sistemas de potência.

Regulador de velocidade

O regulador de velocidade atua sobre a potência mecânica fornecida ao rotor do gerador a fim de manter constante a velocidade do mesmo. A dinâmica associada a estes reguladores é de menor importância para a análise da estabilidade dinâmica de um sistema de potência em relação aos reguladores de tensão.

2.3.1.3 Cargas

Carga é o nome dado ao consumidor final da energia elétrica produzida e distribuída pelo sistema. A dinâmica das cargas em geral pode ser desprezada numa análise de estabilidade dinâmica. Neste trabalho as cargas são modeladas por impedâncias constantes.

2.3.1.4 Linhas de transmissão

A energia elétrica produzida pelas usinas é transmitida aos consumidores através de linhas de transmissão. A dinâmica associada a estas linhas é rápida e pode ser desprezada para o estudo de estabilidade dinâmica dos sistemas de potência. As linhas são modeladas por equações algébricas (fasoriais).

A representação do sistema como um todo é obtida a partir das equações oriundas da modelagem isolada dos componentes de um sistema descritas na seção anterior. Esta tarefa é realizada *conectando-se* as equações correspondentes a cada um dos componentes, ou seja, identificando as variáveis comuns a mais de um componente e representando esta interrelação em um único sistema de equações.

2.4 Conclusão

Neste capítulo foram apresentados os aspectos básicos do problema de controle de sistemas de potência. Os componentes básicos de sistemas de potência foram brevemente descritos. A modelagem de sistemas de potência para fins de análise de estabilidade dinâmica foi discutida. O modelo para o gerador síncrono é o aspecto mais crítico desta modelagem.

A descrição do sistema de potência por meio de um sistema singular de equações permite a simulação digital do comportamento do sistema. Este modelo é bastante complexo e dificilmente pode ser tratado por ferramentas de controle clássico. Modelos linearizados no domínio da frequência são preferidos para o projeto de estabilizadores de sistemas de potência. Este projeto é usualmente realizado de forma descentralizada, sendo cada máquina do sistema vista em um contexto máquina-barramento-infinito.

3. Controle Adaptativo e Aplicações em Sistemas de Potência

3.1 Introdução

O termo sistema adaptativo implica em um sistema capaz de levar em conta variações imprevisíveis, sendo estas variações internas ou externas a ele. Diversas publicações abordam o assunto, por exemplo [NAR80] E [HAR85].

Muitos sistemas têm características dinâmicas que variam com o tempo; um controlador fixo pode não ser capaz de lidar com estas variações. Um controlador adaptativo pode ser requerido. Deve-se salientar que esta não é a única alternativa para tratar sistemas que apresentam características dinâmicas variantes com as condições de operação. Porém, é uma alternativa de relativamente fácil compreensão e implementação [AST89].

O rápido desenvolvimento da tecnologia de microcomputadores tornou possível o uso de computadores digitais em controle industrial em todos os níveis. Confiabilidade, capacidade de tomar decisões e flexibilidade nos algoritmos de controle são as maiores vantagens dos sistemas digitais de controle. Como o processamento de dados nos controladores digitais é feito de maneira direta e fácil, métodos de controle antes considerados demasiadamente complexos, entre eles o controle adaptativo, podem ser implementados a um custo razoavelmente baixo.

Neste capítulo serão apresentados os conceitos básicos envolvidos na implementação de controladores adaptativos auto-ajustáveis, visando também a compreensão de seu uso em Sistemas de Potência.

3.2 Identificação de Sistemas

O primeiro passo no projeto de um controlador é a obtenção de um modelo que descreva adequadamente o comportamento do processo. As principais maneiras de obtenção deste modelo são através do conhecimento do funcionamento do processo, em conjunto com os princípios físicos básicos que regem seu comportamento, ou por experimentação direta através de testes aplicados ao processo. Frequentemente, é possível obter-se a partir dos conhecimentos existentes do sistema, um modelo em termos de parâmetros cujos valores podem ser estimados por experimentação. O método de construção de modelos e estimação dos melhores valores para os parâmetros desconhecidos é chamado de *Identificação de Sistemas*.

Ao se obter modelos para propósitos de controle, normalmente não se busca uma descrição complexa do processo, mas sim um modelo simples e adequado, que permita o projeto de um controlador de tal modo que o sistema como um todo funcione de maneira eficiente, conforme desejado.

Em controle adaptativo auto-ajustável, a técnica mais usada para obtenção de modelos é a estimação paramétrica pelos métodos dos mínimos quadrados ou suas variantes.

A Figura 3.1 ilustra um controlador adaptativo auto-ajustável.

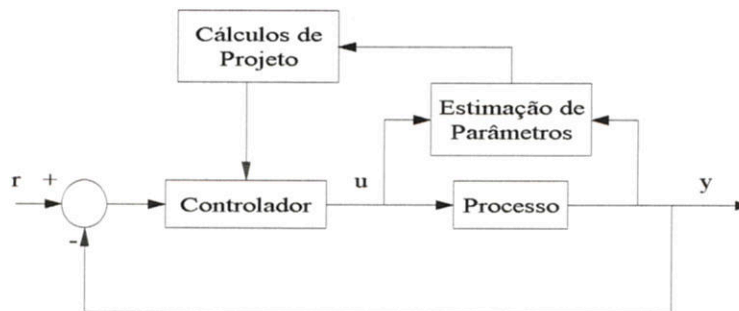


Figura 3.1- Controlador Adaptativo Auto-Ajustável

3.3 Identificação Recursiva dos Parâmetros

O controle auto-ajustável necessita da contínua identificação dos parâmetros variáveis do modelo utilizado para o processo. A identificação em tempo real, que segue a variação dos parâmetros identificados, chama-se identificação recursiva. Inúmeros métodos de identificação recursiva têm sido descritos na literatura, muitos deles bastante similares, com pequenas diferenças de acordo com a aplicação.

O método de identificação recursiva mais utilizado é obtido a partir de um método de identificação *off-line*: o método dos mínimos quadrados. A preferência por este método advém de sua simplicidade, tanto conceitual quanto a nível de implementação, e de sua comprovada eficácia.

3.3.1 O Método dos Mínimos Quadrados

O princípio do método dos mínimos quadrados foi formulado por Gauss e Legendre, no final do século XIII, para previsão de trajetórias de planetas e cometas, a partir de observações realizadas. Segundo este princípio, os parâmetros do modelo devem ser escolhidos de forma tal que a soma dos quadrados das diferenças entre os valores medidos e calculados, multiplicados por números que medem o grau de precisão, seja mínimo. No caso de identificação de sistemas dinâmicos lineares o método pode ser descrito matematicamente como segue.

Seja o sistema descrito pelo modelo de regressão abaixo:

$$y(t) = \phi'(t)\theta + \eta(t) \quad (3.1)$$

com $t \in \mathbb{N}$ e onde foram introduzidos o vetor de regressão $\varphi(t)$ e o vetor de parâmetros $\theta(t)$, definidos abaixo, e o erro de equação $\eta(t)$. O erro de equação é uma seqüência de variáveis aleatórias que representa o ruído injetado no sistema e o erro de modelagem. Este erro de modelagem é consequência da dinâmica não modelada do sistema físico e da linearização dos parâmetros.

$$\varphi(t) \triangleq [\varphi_1(t) \ \varphi_2(t) \ \cdots \ \varphi_n(t)]'$$

$$\theta(t) \triangleq [\theta_1(t) \ \theta_2(t) \ \cdots \ \theta_n(t)]'$$

Esta é a estrutura genérica tratada pelo método dos mínimos quadrados. No contexto de identificação de sistemas dinâmicos esta formulação em geral representa um modelo ARMAX¹:

$$A(q^{-1})y(t) = B(q^{-1})u(t) + \eta(t)$$

onde q é operador de avanço definido por:

$$q^i y(t) \triangleq y(t+i)$$

e:

$$A(q^{-1}) = 1 + a_1 q^{-1} + \cdots + a_{n_a} q^{-n_a}$$

$$B(q^{-1}) = b_0 + b_1 q^{-1} + \cdots + b_{n_b} q^{-n_b}$$

de forma que os vetores de regressão e de parâmetros são dados por:

$$\theta = [a_1 \ \cdots \ a_{n_a} \ b_0 \ \cdots \ b_{n_b}]'$$

$$\varphi(t) = [-y(t-1) \ \cdots \ -y(t-n_a) \ u(t-1) \ \cdots \ u(t-n_b-1)]'$$

As medidas $y(t)$ e os regressores $\varphi(t)$ são obtidos a cada instante de tempo a partir de um experimento.

O problema de identificação pela abordagem do mínimos quadrados consiste em determinar o conjunto de parâmetros que faz com que as saídas calculadas nos diversos instantes de tempo segundo o modelo dado pela

¹ Do inglês: **A**uto **R**egressive **M**oving **A**verage **E**Xogenous

equação (3.1) sejam tão próximas quanto possível das saídas observadas, ou seja, a estimativa de mínimos quadrados é o vetor θ que minimiza a função definida não negativa $V(\theta, t)$:

$$V(\theta, t) \triangleq \frac{1}{2} \|Y(t) - \Phi(t)\theta\|^2 = \sum_{i=1}^t (y(i) - \phi'(i)\theta)^2$$

onde:

$$Y(t) \triangleq [y(1) \ y(2) \ \cdots \ y(t)]'$$

$$\Phi(t) \triangleq \begin{bmatrix} \phi'(1) \\ \phi'(2) \\ \cdots \\ \phi'(t) \end{bmatrix} \quad (3.2)$$

Para um sistema linear nos parâmetros, como o descrito por (3.1) este problema de minimização admite solução analítica, apresentada pelo teorema a seguir.

Teorema 3.1 Seja a função $V(\theta, t)$ definida na equação (3.2). Se a matriz $(\phi'\phi)$ é não singular, então a função $V(\theta, t)$ tem um único mínimo em relação a θ e este mínimo é dado por [BAZ93]:

$$\min V = \frac{1}{2} Y'(I - \Phi(\Phi'\Phi)^{-1}\Phi')Y$$

onde I é a identidade. Além disto, este mínimo é obtido para $\theta = \hat{\theta}$ com $\hat{\theta}$ dado por:

$$\hat{\theta} = (\Phi'\Phi)^{-1}\Phi'Y \quad (3.4)$$

O último termo desta expressão é independente de θ , e portanto é irrelevante para a minimização. O primeiro termo é uma forma quadrática em $(\phi'\phi)$ que é semi-definida positiva. Portanto, este termo é sempre maior ou igual a zero e um valor de θ que o anule será um mínimo da função custo. Supondo que $(\phi'\phi)$ seja não-singular, este valor existe, é único e é dado por (3.4).

3.3.2 Mínimos Quadrados Recursivo

Em aplicações de identificação recursiva, a estimativa de θ deve ser calculada em tempo real, simultaneamente à obtenção dos dados, e

atualizada a cada instante de tempo, de acordo com os novos dados obtidos. A estimativa dada por (3.4) exige o conhecimento de todas as medidas do sistema desde $t = 0$. O cálculo de $\hat{\theta}$ em tempo real por meio desta equação é inviável computacionalmente, dado que a matriz ϕ é de ordem $n \times t$ e t cresce infinitamente com a evolução do sistema. Este problema é resolvido por meio de manipulações algébricas da expressão (3.4) que fornece uma expressão recursiva para $\hat{\theta}$, ou seja, uma expressão que fornece a estimativa atualizada $\hat{\theta}(t)$ em função da estimativa no instante anterior $\hat{\theta}(t-1)$. O algoritmo de identificação assim obtido é chamado de *mínimos quadrados recursivo*.

Teorema 3.2 Seja a função $V(\theta, t)$ dada pela equação (3.2). Se a matriz $\phi(t)$ tem posto completo para todo $t > 0$, então esta função tem um único mínimo com relação a θ para cada instante de tempo, o argumento deste mínimo é dado por $\hat{\theta}$ na equação (3.4), e os valores deste argumento obedecem a recursão [BAZ93]:

$$\hat{\theta}(t+1) = \hat{\theta}(t) + K(t+1)\varepsilon(t+1)$$

onde:

$$\varepsilon(t+1) = y(t+1) - \phi'(t+1)\hat{\theta}(t)$$

$$K(t+1) = \frac{P(t)\phi(t+1)}{1 + \phi'(t+1)P(t)\phi(t+1)}$$

$$P(t+1) = P(t) - \frac{P(t)\phi(t+1)\phi'(t+1)P(t)}{1 + \phi'(t+1)P(t)\phi(t+1)}$$

A matriz de covariância $P(t)$ deve ser inicializada com algum valor em $t=0$. Este valor deve ser alto, devido a falta de informações no início do processo, quando nenhum dado foi colhido. Com a evolução do tempo e a convergência das estimativas dos parâmetros, os elementos de $P(t)$ tendem a diminuir de magnitude. Isto será discutido mais tarde.

3.4 Estratégias de Controle

São muitas as estratégias de controle aplicadas em controle auto-ajustável, sendo que entre elas podem ser citadas a variância mínima, variância mínima generalizada e lineares quadráticos. A dificuldade de aplicação destes métodos reside no tratamento de sistemas de fase não-mínima e sua complexidade computacional limita a aplicação destes métodos a sistemas de potência.

A seguir será descrito o método de alocação de pólos. Esta técnica de controle é mais simples do ponto de vista computacional e tem sido amplamente utilizada em estabilizadores de sistemas de potência auto-

ajustáveis. Descrições mais detalhadas deste e dos outros métodos de controle podem ser encontradas em [AST89], [HUG86], [GOO84] e [HAR85]

3.4.1 Controlador por Alocação de Pólos Auto-Ajustável

Conforme citado anteriormente, os parâmetros do sistema são identificados em tempo real e os do controlador, ajustados também em tempo real, a partir da estimação de parâmetros, sendo que estes parâmetros são considerados os parâmetros reais do sistema. Os procedimentos de identificação e de controle são analisados e executados de forma separada. Desta forma, o procedimento de alocação de pólos será apresentado de maneira independente.

Seja o seguinte modelo determinístico para um processo, conforme apresentado na figura 3.2:

$$Ay(t) = q^{-k} Bu(t)$$

com o controlador na forma:

$$Fu(t) = G[r(t) - y(t)]$$

onde:

$$A = 1 + a_1 q^{-1} + \dots + a_{na} q^{-na}$$

$$B = b_0 + b_1 q^{-1} + \dots + b_{nb} q^{-nb}$$

$$F = 1 + f_1 q^{-1} + \dots + f_{nf} q^{-nf}$$

$$G = g_0 + g_1 q^{-1} + \dots + g_{ng} q^{-ng}$$

e supondo-se que os polinômios A e B são obtidos através de um método de estimação recursiva.

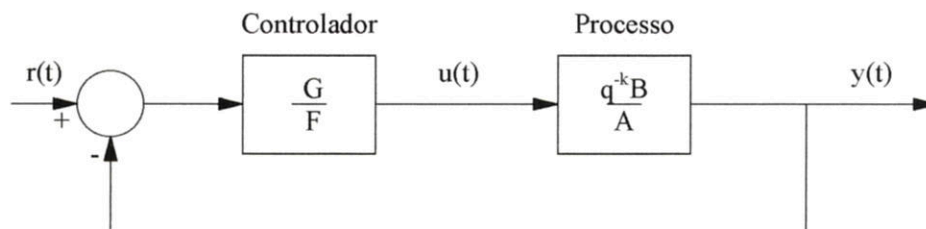


Figura 3.2- Controle Auto-ajustável por Alocação de Pólos.

A função transferência de malha fechada pode ser colocada como:

$$y(t) = q^{-k} \frac{BG}{AF + q^{-k}BG} r(t)$$

O controle por posicionamento de pólos sem especificação dos zeros desejados implica na escolha de uma equação característica desejada para o sistema, a fim de evitar problemas com processos de fase não mínima. A equação característica da função de transferência de malha fechada será:

$$AH + q^{-k}BG = T$$

onde T é o polinômio que contém os pólos desejados para o sistema.

Sendo A e B conhecidos por estimação e dado T, pode-se resolver a equação acima (denominada Equação Diofantina²), expressando-se o polinômio por:

$$T = 1 + t_1q^{-1} + \dots + t_nq^{-nt}$$

Fixando-se as ordens dos polinômios como [BAR95]:

$$n_g = n_a - 1$$

$$n_f = n_b + k - 1$$

$$n_t \leq n_a + n_b + k - 1$$

e comparando os coeficientes de igual potência em ambos os lados da equação, uma única solução para os coeficientes de F e G pode ser encontrada.

Esta solução pode ser representada em uma forma matricial. Para $k=1$ esta forma é apresentada a seguir.

² Em homenagem ao algebrista grego Diophantus, do século IV, que estudou equações similares no campo dos números inteiros.

$$\begin{bmatrix} 1 & 0 & \dots & 0 & b_0 & 0 & \dots & 0 \\ a_1 & 1 & \dots & \vdots & b_1 & b_0 & \dots & \vdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & a_1 & \dots & \vdots & \vdots & b_1 & \dots & \vdots \\ \vdots & \vdots & \ddots & 0 & \vdots & \vdots & \dots & 0 \\ a_{na} & \vdots & \dots & 1 & b_{nb} & \vdots & \dots & b_0 \\ 0 & a_{na} & \dots & a_1 & 0 & b_{nb} & \dots & b_1 \\ \vdots & 0 & \dots & \vdots & \vdots & 0 & \dots & \vdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & a_{na} & 0 & 0 & \dots & b_{nb} \end{bmatrix} \begin{bmatrix} f_1 \\ \vdots \\ f_{nf} \\ g_0 \\ \vdots \\ g_{ng} \end{bmatrix} = \begin{bmatrix} t_1 - a_1 \\ t_2 - a_2 \\ \vdots \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Um sistema linear gerado pela equação acima para a solução dos coeficientes do controlador deve ser resolvido em cada intervalo de amostragem, usando-se os coeficientes estimados da planta do processo e os coeficientes do polinômio dos pólos desejados.

Alguns problemas devem ser colocados:

- O polinômio T precisa ser cuidadosamente escolhido para que se obtenha um controlador adequado;
- Não se pode super-dimensionar a ordem do processo, pois isto induz o aparecimento de raízes iguais nos polinômios A e B na estimação do processo. Como estes termos podem não ter sido considerados na escolha de T, pode haver problemas da resolução da equação de Diofantina[ALL90];
- O esquema pode exigir muito tempo para a obtenção do controle, pois é necessário resolver um sistema linear cuja ordem depende do número de parâmetros a serem determinados, além da estimação recursiva. Portanto, em processos que exigem intervalos de amostragem curtos (processos rápidos), pode haver problemas de tempo na implementação dos algoritmos.

Para auxiliar a escolha do polinômio T, sugere-se usar um polinômio com raízes iguais aos pólos de malha aberta (raízes do polinômio A) deslocados radialmente em direção a origem do plano Z, melhorando assim a estabilidade do sistema. Este método pode ser denominado posicionamento de pólos por deslocamento radial uniforme, e tem sido usado em diversas aplicações em sistemas de potência [SIL92].

Os pólos de malha aberta são deslocados em direção ao centro do círculo unitário, através de um fator de deslocamento α , com valor entre 0 e 1, escolhido conforme a aplicação.

A Figura 3.3 ilustra esta técnica.

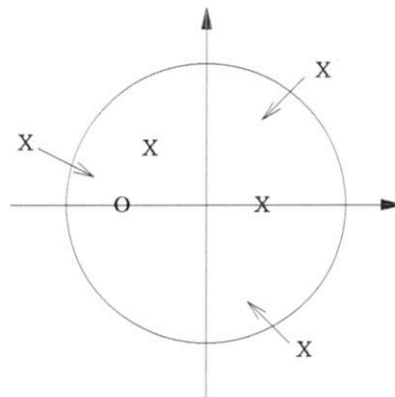


Figura 3.3 - Deslocamento radial dos pólos de malha aberta em direção ao círculo unitário.

3.5 Algumas Considerações Práticas

Os controladores adaptativos auto-ajustáveis não são métodos completamente automáticos para determinação da lei de controle. Estes controladores devem ser adequadamente ajustados. Algumas considerações de sua implementação serão brevemente discutidas nesta seção. Maiores detalhes podem ser encontrados em [WEL82].

Inicialização :

No início ($t=0$), valores para $\theta(0)$ e $P(0)$ devem ser atribuídos. Normalmente, uma boa *suposição* dos valores dos parâmetros é obtida através de um conhecimento prévio do sistema. Estes valores podem ser usados para $\theta(0)$, com $P(0)=mI$, e m sendo um valor pequeno (por exemplo, 10), indicando assim confiabilidade nas estimativas.

Quando não há um conhecimento prévio do sistema, θ deve ser inicializado com valores pequenos e m com um valor alto, como 10^3 ou mais.

Alternativamente, a estimação pode ser congelada até que dados suficientes sejam obtidos e estas estimativas usadas para fazer a recursão.

Excitação

Para que o sistema possa ser identificado, é necessário que o sistema seja permanentemente excitado, a fim de evitar que a matriz de medidas ϕ tenha linhas em comum, e conseqüentemente, seja singular. Desta forma, o vetor de parâmetros θ pode ser obtido.

Em aplicações de sistemas de potência, o sinal utilizado para excitação do sistema para a identificação é um sinal do tipo PRBS (Pseudo-Random Binary Sequence). Este sinal é usado pois excita as faixas de freqüência relativas aos modos eletromecânicos, os quais são o objetivo deste trabalho.

A geração do sinal PRBS para fins de identificação é coberta por muitas referências, entre elas [DAV70] e [BAR95].

Um PRBS é um sinal que pode assumir dois valores apenas, $+b$ ou $-b$. Estes valores podem mudar apenas em intervalos discretos de tempo Δt e a mudança de tempo ocorre de maneira determinística pseudo-aleatória. A seqüência gerada é periódica com período $N\Delta t$ (onde N é um inteiro), mas o período pode ser suficientemente longo para que a seqüência possa ser considerada aleatória para dada aplicação. A seqüência mais comumente usada é chamada de *seqüência de comprimento máximo*, cujo comprimento é $N=2^n - 1$ (onde n é o número de bits no *shift register* usado para geração da seqüência). Neste projeto, a seqüência é implementada com $n=6$ e $\Delta t=T$, onde T é o período de amostragem e para sistemas de potência o valor considerado adequado é de 100ms.

A seqüência considerada é gerada pelo *shift register* apresentado na Figura 3.3, com o estado inicial proposto pela referência [BAR95]. Como a seqüência binária gerada tem comprimento 63 ($N=2^6-1$). Após a ter sido feita a associação '1'= $+b$ e '0'= $-b$, a quantia $b/63$ deve ser subtraído do sinal gerado. O valor de b deve ser o máximo possível, porém sem perturbar o sinal em demasia.

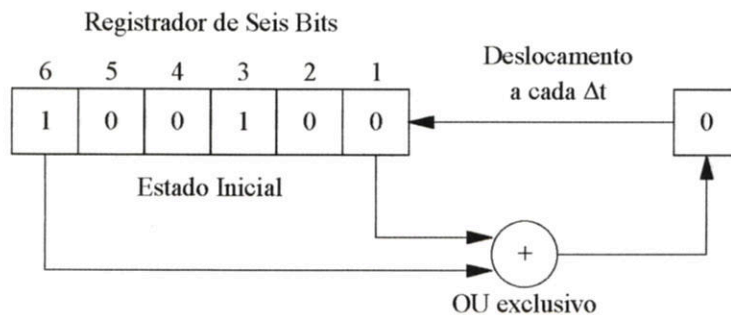


Figura 3.4 - *Shift Register* de seis bits para geração de um sinal PRBS, com estado inicial indicado.

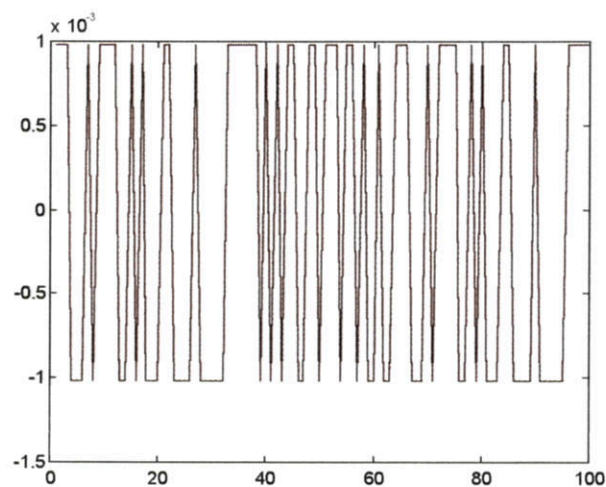


Figura 3.5 - Sinal PRBS gerado pelo *Shift Register* apresentado na Figura 3.3.

Parâmetros Variantes no Tempo

O rastreamento de parâmetros variantes no tempo é um importante problema na identificação de sistemas. Independente da aplicação dada ao estimador, é necessário que ele seja eficaz no rastreamento das mudanças no sistema identificado. Quando o número de iterações aumenta, os parâmetros estimados podem convergir. Quando as estimativas dos parâmetros convergem, os elementos do vetor de ganhos K tornam-se pequenos e, caso ocorram variações no sistema, embora o erro de estimação seja muito grande, a estimativa dos parâmetros não será ajustada. Esta convergência é normalmente refletida pela diminuição dos elementos da matriz de covariância $P(t)$. Esta convergência é desejável para o caso de sistemas invariantes no tempo. Entretanto, quando o sistema a ser identificado é um sistema variante no tempo, é necessário evitar que os elementos da matriz $P(t)$ tornem-se “muitos pequenos” para que possa ocorrer correção nos estimadores $\hat{\theta}(t)$.

Na prática, procura-se um compromisso entre a capacidade de adaptação e a convergência no algoritmo de estimação.

A seguir serão descritos alguns métodos de controle da magnitude dos elementos de $P(t)$.

a) Atualização de $P(t)$

Os elementos de $P(t)$ são aumentados através da adição de uma matriz diagonal constante semi-definida positiva $Q(t)$. Então, a nova matriz de covariância é dada por:

$$P(t) = P(t) + Q(t)$$

a.1) Random Walk

A cada iteração soma-se a matriz $Q(t)$, cujos elementos descrevem a suposta razão de variação dos parâmetros do sistema.

É usual fazer $Q(t)$ uma matriz diagonal por simplicidade. Se somente certos parâmetros mudam, então $Q(t)$ pode ser selecionada com zeros em todas as posições da diagonal, exceto aqueles que correspondem aos parâmetros variantes.

$$Q(t) = \frac{\text{tr } P(t)}{n}$$

onde:

tr é o traço³ da matriz;

n é o número de parâmetros a serem estimados.

³ Somatória dos elementos da diagonal principal da matriz.

a.2) Covariance Resetting

Dado um traço mínimo tr_0 , nas iterações em que $tr[P(t)] < tr_0$, faz-se uma reinicialização de $P(t)$ somando-se a matriz $Q(t)$, cujos elementos dão um ganho abrupto ao estimador.

O erro de estimação $[y(t) - \hat{y}(t)]$ pode também ser observado para reinicialização de $P(t)$ somente se o valor absoluto do erro de estimação excede um limite especificado pelo usuário.

b) Fator de Esquecimento

Como já foi dito anteriormente, deve-se evitar que os termos de $P(t)$ tornem-se pequenos. Uma outra maneira de evitar esta diminuição é o uso do fator de esquecimento. Este método elimina progressivamente a influência das medidas anteriores na atualização de $P(t)$.

Neste caso, a atualização de $P(t)$ fica sendo:

$$P(t) = \frac{1}{\lambda} \left[P(t-1) - \frac{P(t)\phi(t+1)\phi'(t+1)P(t)}{\lambda + \phi'(t+1)P(t)\phi(t+1)} \right]$$

onde o fator de esquecimento λ é um número ligeiramente menor que 1, geralmente na faixa de $0.97 \leq \lambda \leq 1$.

Com o fator de esquecimento λ incluído, os elementos de $P(t)$ não decrescem a zero quando $t \rightarrow \infty$ e desde modo, o algoritmo pode ser usado em sistemas variantes no tempo. Contudo, se nenhuma nova informação entrar no estimador por um longo período de tempo, a contínua divisão por λ pode tornar os elementos de $P(t)$ muito grandes, alcançando os limites numéricos do computador. Isto é conhecido na literatura como *explosão da matriz de covariância* [SEB86]. Para evitar este problema, usa-se o fator de esquecimento variável [FOR81], cuja idéia básica é manter λ próximo a um, quando os parâmetros não apresentam variações e, assim que alguma variação for detectada (por exemplo, observando-se o erro de estimação), diminui-se λ .

Ordem do Modelo :

A ordem do modelo pode ser escolhida baseada, principalmente, no conhecimento do projetista sobre o processo considerado. A ordem do modelo pode ser escolhida como sendo a menor ordem possível, porém grande o suficiente para representar as características dinâmicas importantes do processo. Modelos de ordem alta geram controladores de ordem alta, também. Isto faz com o tempo de identificação seja grande, causa transitórios de má identificação e são mais difíceis de ajustar do que os controladores de baixa

ordem. Entretanto, modelos de baixa ordem não levam em consideração modos rápidos de oscilação, o controle gerado às vezes pode excitar estes modos, causando problemas não esperados. Portanto, um compromisso deve ser conseguido.

Limites no Sinal de Controle:

Limites no sinal de controle deve ser imposto pelo algoritmo de controle auto-ajustável, de modo que a ação de controle esteja dentro dos limites de saturação do sistema, a fim de evitar a excitação da faixa não-linear e a possível ocorrência de ciclos limite ou outros efeitos.

Período de Amostragem :

O período de amostragem é geralmente escolhido baseado nos modos dominantes do sistema. Uma regra prática sugere uma frequência de amostragem entre 6 e 10 vezes mais rápida que o modo de dinâmica mais importante do sistema. O fato de que deve-se sempre levar em conta que o período de amostragem deve ser suficiente para executar os cálculos do algoritmo empregado.

Muitos fatores do controle auto-ajustável são interrelacionados e alguns ajustes devem ser feitos *on-line*.

3.6 Conclusão

Neste capítulo a teoria de controle auto-ajustável foi discutida. O controle adaptativo auto-ajustável consiste em basicamente duas etapas: identificação e controle. O sistema é identificado em tempo real e os parâmetros do controlador são calculados a partir dos parâmetros resultantes desta identificação, como se estes parâmetros fossem os valores verdadeiros. Diversos algoritmos de identificação e controle podem ser encontrados na literatura. Foi dada ênfase à identificação através do método dos mínimos quadrados recursivo e à estratégia de controle por alocação de pólos, devido sua simplicidade e eficácia e também por sua larga utilização em controle auto-ajustável de Sistemas de Potência.

Os controladores adaptativos auto-ajustáveis não são métodos completamente automáticos para determinação da lei de controle. Estes controladores devem ser adequadamente ajustados. Detalhes relevantes da implementação prática de controladores auto-ajustáveis foram apresentados.

4. Implementação e Simulação Digital

4.1 Introdução

O objetivo deste trabalho de fim de curso é a implementação de técnicas de controle digital, especialmente técnicas de controle adaptativo, a Sistemas de Potência, a fim de resolver o problema das oscilações eletromecânicas, já descrito anteriormente.

Neste capítulo são descritos os procedimentos utilizados para a implementação de uma estrutura de controle, que envolve dois microcomputadores e um simulador de sistemas de potência. Nas próximas seções também será descrito detalhadamente o desenvolvimento das rotinas de comunicação serial e da interface das linguagens C e Matlab, necessários para a implementação deste projeto.

4.2 Esquema de Controle

Um esquema de controle foi proposto para a aplicação de técnicas de controle adaptativo auto-ajustável a Sistemas de Potência.

Para a implementação prática deste esquema de controle foi necessário o uso de dois microcomputadores do tipo PC, comunicando-se de forma serial. Em um dos microcomputadores foi colocado um simulador digital, o qual permite simular um sistema elétrico de potência. O controlador adaptativo auto-ajustável foi implementado no outro computador. Isto é ilustrado na Figura 4.1.

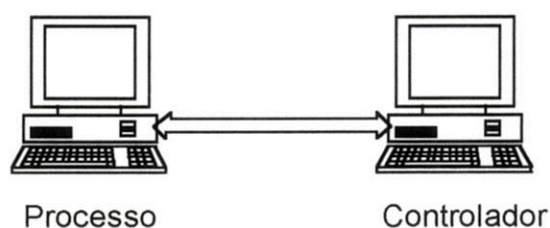


Figura 4.1 - Comunicação serial entre os dois microcomputadores

O desenvolvimento de rotinas que possibilitassem a passagem de dados entre dois microcomputadores tornou-se necessário. Esta comunicação é indispensável pois os dois computadores (um simulando o processo e outro o controlador) trocam dados entre si, durante a simulação.

Os algoritmos de identificação e controle adaptativo foram desenvolvidos em linguagem Matlab. Isto se deve a capacidade deste software

manipular cálculos matemáticos, entre eles matriciais, requeridos pelas rotinas de identificação e controle auto-ajustável. Em virtude da utilização do Matlab, tornou-se necessário o desenvolvimento de uma interface entre este software e a linguagem C, dado que a comunicação entre os computadores é realizada através de rotinas em linguagem C, devido ao fato de que esta linguagem permite o acesso à porta de comunicação serial dos processadores. Estes algoritmos de identificação e controle auto-ajustável foram instalados no computador II.

No computador I foi instalado um simulador de um Sistema Elétrico de Potência. Este simulador permite simular a dinâmica rápida e lenta de Sistemas de Potência, incluindo a modelagem detalhada de geradores síncronos, reguladores de tensão e velocidade, estabilizadores de sistemas de potência (ESP's), turbinas térmicas e hidráulicas, caldeiras, reatores nucleares, dispositivos FACTS⁴, cargas estáticas e dinâmicas e controle automático de geração. Para a implementação deste simulador foi utilizada programação orientada a objeto (OOP), através da linguagem C++. Maiores detalhes acerca do simulador podem ser obtidos na referência [MAN96].

O controlador adaptativo auto-ajustável é utilizado para controlar o sistema simulado, utilizando como entrada o sinal proveniente do simulador.

Este esquema de controle funciona da seguinte forma: o simulador de sistemas de potência roda de maneira independente, no computador I. A cada período de amostragem, o simulador envia o sinal de entrada para o estabilizador, no computador II, transferindo assim a função de estabilizador de sistemas de potência (ESP) para o controlador auto-ajustável. Este dado é transferido pelo módulo de comunicação serial, em linguagem C. Através da interface C-Matlab a rotina de controle recebe o valor enviado. Esta rotina então faz os cálculos de identificação e controle a partir do dado atual, armazenando o valor de controle calculado para que este seja enviado ao simulador no próximo período de amostragem. Na Figura 4.2 é apresentada esta seqüência de operações.

4.3 Implementação

Nesta seção são mostrados os detalhes de implementação da comunicação serial entre os dois microcomputadores. São apresentados detalhes do desenvolvimento das rotinas de envio e recebimento de mensagem, implementadas em linguagem C.

São colocados também nesta seção, alguns detalhes importantes para o desenvolvimento da interface entre as linguagens C e Matlab.

⁴ FACTS - Flexible AC Transmission System.

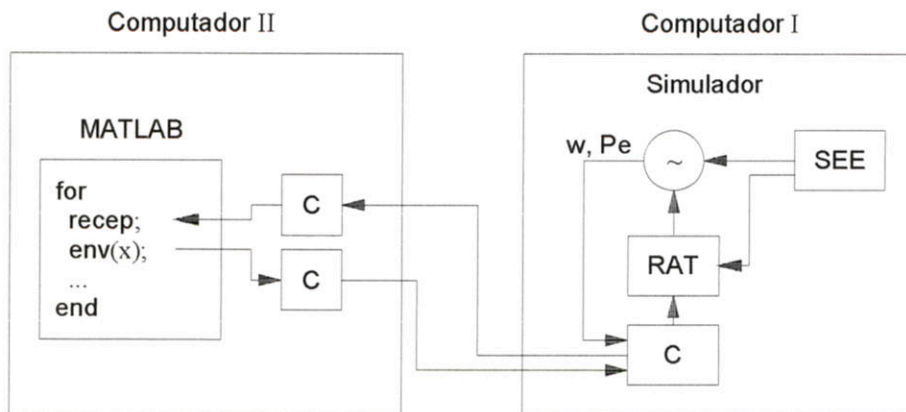


Figura 4.2 - Esquema de controle proposto

4.3.1 Comunicação serial

Rotinas para a comunicação serial foram desenvolvidas para que os dois microcomputadores pudessem trocar dados entre si durante a simulação. A seguir, são descritos alguns detalhes relevantes para o desenvolvimento dos módulos de recepção e envio de mensagem.

4.3.1.1 Portas de E/S

A família de microprocessadores 8086 comunica-se com muitas partes do computador através do uso das portas de entrada e saída (E/S). As portas de entrada e saída são vias através das quais a informação passa enquanto viaja de um dispositivo para o outro, como um teclado em uma impressora.

Toda porta é identificada por um número de porta de 16 bits, que pode variar na faixa de 00H a FFFFH (65.535). A CPU identifica uma porta qualquer pelo seu número.

Assim como quando está acessando a memória, a CPU usa as barras de dados e endereços como canais de comunicação com as portas. Para acessar uma porta, a CPU primeiro envia um sinal ao barramento do sistema, que avisa a todos os dispositivos de entrada e saída que o endereço no barramento é o de uma porta. O dispositivo com o endereço de porta enviado responde à CPU.

O número de porta endereça um local na memória que é associado ao dispositivo de E/S, mas não faz parte da memória principal. Em outras palavras, um número de porta de E/S não é o mesmo que um endereço de memória. Para o acesso das portas, as instruções reservadas são `IN` e `OUT` (assembly). Muitas linguagens de programação de alto nível fornecem funções que acessam as portas de E/S. As funções do C são `inp` e `outp`, e serão descritas mais tarde.

A referência [NOR91] apresenta descrições mais detalhadas sobre as portas de E/S e seus respectivos números de porta.

4.3.1.2 RS-232

Esta seção discute os serviços da porta assíncrona de comunicações seriais RS-232.

A porta de E/S RS-232 conecta eletricamente o processador ao mundo exterior. Pode ser usada para interconectar muitos tipos de equipamentos de processamento de informação fora do computador. Ela também pode ser ligada diretamente a outra interface RS-232 através de um cabo, desde que a distância seja pequena. Caso a ligação seja feita a longas distâncias, faz-se necessária a utilização de um outro equipamento chamado *modem* (modulador/demodulador), que converte sinais digitais em analógicos e sinais analógicos em digitais.

Quatro serviços de comunicação são comuns a todos os tipos de processadores. Estes serviços são chamados com a interrupção 14H, sendo numeradas de 00H a 03H. Alguns tipos de processador oferecem ainda dois serviços adicionais. A Figura 4.3 descreve estes serviços.

Serviço	Descrição
00H	Inicializa porta serial
01H	Emite um caractere
02H	Recebe um caractere
03H	Obtém estado da porta serial
04H	Inicializa porta serial estendida
05H	Controla porta de comunicação estendida

Figura 4.3 - Serviços da porta serial RS-232 disponíveis por meio da interrupção 14H.

A descrição dos serviços oferecidos pela RS-232 é feita a seguir:

Serviço 00H: Inicializa a porta serial

O serviço 00H define vários parâmetros da rs232 e inicializa a porta serial. Ela define quatro parâmetros: a taxa de transferência, a paridade, o número de bits de parada e o tamanho do caractere. Os parâmetros são combinados em um código de 8 bits, que é colocado no registrador AL, conforme mostrado na Figura 4.4. Os valores de bit para cada código são mostrados na Figura 4.5. Quando o serviço termina, o estado da porta de comunicação é informado.

Bit								Uso
7	6	5	4	3	2	1	0	
X	X	X	-	-	-	-	-	Código de taxa de transferência
-	-	-	X	X	-	-	-	Código de paridade
-	-	-	-	-	X	-	-	Código de bit de parada
-	-	-	-	-	-	X	X	Código de tamanho do caractere

Figura 4.4 - A ordem de bit para os parâmetros da porta serial passados no registrador AL para o serviço 00H.

Taxa de Transferência					Paridade			
Bit					Bit			
7	6	5	Valor	Bits por seg.	4	3	Valor	Significado
0	0	0	0	110	0	0	0	Nenhuma
0	0	1	1	150	0	1	1	Paridade ímpar
0	1	0	2	300	1	0	2	Nenhuma
0	1	1	3	600	1	1	3	Paridade par
1	0	0	4	1200				
1	0	1	5	2400				
1	1	0	6	4800				
1	1	1	7	9600				

Bits de Parada			Tamanho do Caractere			
Bit			Bit			
2	Valor	Significado	1	0	Valor	Significado
0	0	Um	0	0	0	Não usado
1	1	Dois	0	1	1	Não usado
			1	0	2	7 bits
			1	1	3	8 bits

Figura 4.5 - Os valores de bits para os quatro parâmetros de porta serial no serviço 00H.

Serviço 01H : Emite um caractere

O serviço 01H transmite um caractere pela porta serial especificada em DX. Quando o serviço 01H for chamado, deve-se colocar o caractere a ser transmitido em AL. Quando o serviço 01H retornar, ele informará o estado da porta de comunicação. Com AH=00H tem-se o serviço completado com sucesso. Caso contrário, o bit 7 de AH indica que houve um erro, e os outros bits de AH informam o tipo de erro. Estes bits são mostrados na discussão do serviço 03H, o serviço de estado.

Há uma irregularidade na informação de erro fornecida por este serviço: como o bit 7 informa que ocorreu um erro, ele não está disponível para indicar um erro de tempo de resposta (como sugerem os detalhes no serviço 03H). Consequentemente, quando este serviço ou o serviço 02H informar um erro, a

forma mais simples e confiável de checar a natureza do erro é usar a informação completa de estado dada pelo serviço 03H, ao invés de confiar no código de estado incompleto retornado com o erro nos serviços 01H e 02H.

Serviço 02H : Recebe um caractere

O serviço 02H recebe um caractere pela linha de comunicações e o retorna no registrador AL. O serviço espera por um caractere ou qualquer sinal que indique o termino do serviço, como um decurso-de-tempo. AH informa o sucesso ou falha do serviço no bit 7, como explicado na discussão do serviço 01H.

Serviço 03H : Obtém estado da porta serial

O serviço 03H retorna o estado completo da porta serial no registrador AX. Os 16 bits de estado em AX são divididos em dois grupos: AH informa a linha de estado (que também é informado quando ocorrem erros com os serviços 01H e 02H), e AL informa o estado do modem, quando for aplicável a Figura 4.6 contém as codificações dos bits de estado. Alguns códigos informam erros, e outros simplesmente informam uma condição.

Bit	7	6	5	4	3	2	1	0	significado*	Bit	7	6	5	4	3	2	1	0	significado*
Registrador AH (estado da linha)										Registrador AL (estado de modem)									
1	erro de decurso de tempo	1	detecção de sinal de linha recebido
.	1	reg. de transferência vazio	.	1	indicador de chamada
.	.	1	reg. de manutenção vazio	.	.	1	grupo de dados pronto
.	.	.	1	erro de detecção do break	.	.	.	1	limpa para enviar
.	.	.	.	1	erro de enquadramento	1	delta de sinal de linha
.	1	.	.	.	erro de paridade	1	.	.	.	detecção de subida de pulso
.	1	.	.	erro de transposição	1	.	.	delta grupo de dados pronto
.	1	.	dado pronto	1	.	delta limpa para enviar

* Quando for 1.

Figura 4.6 - A codificação de bits dos bytes de estado retornados no registrador AX pelo serviço 03H.

Serviço 04H : Inicializa porta serial estendida

O serviço 04H está disponível apenas em alguns tipos de processadores. Este serviço expande as capacidades do serviço 00H, fornecendo suporte às portas seriais melhoradas destes processadores. Comparando o serviço 04H com o 00H, observa-se que os quatro parâmetros de inicialização da porta serial passados em AL no serviço 00H são separados em quatro registradores no serviço 04H (Figura 4.7). Além disto, o serviço 04H

retorna um estado de modem e linha no registrador AX, exatamente como no serviço 03H. Como o serviço 04H tem estas capacidades expandidas, geralmente deve-se usá-lo no lugar do serviço 00H para a inicialização da porta serial destes processadores.

Break (reg. AL)		Paridade (reg. BH)		Tamanho de palavra (reg. CH)	
Valor	Significado	Valor	Significado	Valor	Significado
00H	sem break	00H	nenhuma	00H	5 bits
01H	break	01H	ímpar	01H	6 bits
		02H	par	02H	7 bits
		03H	paridade fixa ímpar	03H	8 bits
		04H	paridade fixa par		

Taxa de transferência (reg. CL)				Bits de parada (reg. BL)	
Valor	Significado	Valor	Significado	Valor	Significado
00H	110 baud	05H	2400 baud	00H	1
01H	150 baud	06H	4800 baud	01H	2 */1,5**
02H	300 baud	07H	9600 baud		
03H	600 baud	08H	19200 baud		
04H	1200 baud				

* Para tamanhos de palavra igual a seis, sete ou oito.

** Para tamanhos de palavra igual a cinco.

Figura 4.7 - Valores de registrador para a inicialização de porta serial com a interrupção 14H, no serviço 04H.

Serviço 05H : Controla porta de comunicações estendida

O serviço 05H está disponível apenas em alguns tipos de processadores. Este serviço permite que estes processadores possam ler ou escrever no registrador de controle do modem de uma porta de comunicações seriais especificada. Ao chamar o serviço 05H com AL=00H, o serviço 05H retorna com o registrador BL contendo o valor no registrador de controle de modem da porta serial especificada. Ao chamar o serviço 05H com AL=01H, o processador copia o valor passado no registrador BL no registrador de controle do modem para a porta especificada. Em ambos os casos, o serviço 05H retorna o estado do modem e estado de linha nos registradores AL e AH, como faz o serviço 03H.

4.3.1.3 Transferência/Recebimento de Dados

Os módulos de transferência e recebimento de mensagens foram implementado em linguagem C, pois esta permite o acesso da porta serial RS-232, através dos comandos *inp* e *outp*. Estes comandos são descritos a seguir.

Comando inp

A função *inp* lê um byte da porta do hardware 80X86. Um, dois ou quatro bytes podem ser lidos de cada porta, dependendo do hardware.

O valor retornado é o byte que foi lido.

Sinopse

```
#include <conio.h>
unsigned int inp ( int port);
```

Comando outp

A função *outp* escreve um byte na porta do hardware 80X86. Um, dois ou quatro bytes podem ser escritos em cada porta, dependendo do hardware.

O valor retornado é o valor transmitido.

Sinopse

```
#include <conio.h>
unsigned int outp ( int porte, int value);
```

Rotinas de envio e recepção de mensagem

A cada período de amostragem, dois valores de 4 bytes são passados de um computador ao outro. Os computadores *sabem* que devem receber 8 bytes de informação. Como as funções da linguagem C utilizadas para a comunicação serial foram *inp* e *outp*, foi necessário que houvesse o *empacotamento* e *desempacotamento* destes valores, pois estas funções tratam apenas um byte de cada vez.

A informação é passada byte a byte, pelo computador emissor. O computador receptor recebe um byte a cada vez e aguarda até que oito bytes sejam recebidos. Quando isto acontece, a informação é desempacotada. Isto quer dizer que os primeiros quatro bytes são considerados como a primeira informação e os outro quatro bytes, como a segunda informação.

As rotinas de empacotamento e desempacotamento são mostradas abaixo. O programa completo é apresentado no Apêndice.

```
double desempacota_double ()
{
double *auxiliar;
auxiliar = (double *)& mensagem [ auxiliar_tam_msg ];
auxiliar_tam_msg += sizeof ( double );
return ( *auxiliar );
}
```

```
double recebe ()
{
double temp, valor;
// --- Prepara a recepcao ---
int i;
```

```

for ( i = 0; i++; i < tamanho_mensagem ) mensagem [ i ] = 0;
auxiliar_tam_msg = 0;
// --- Espera ate receber toda a mensagem ---
while (1) {
    mensagem [ auxiliar_tam_msg++ ] = recebe_car ();
    if ( auxiliar_tam_msg == tamanho_mensagem ) break;
};
auxiliar_tam_msg = 0;
temp = desempacota_double ();
valor = desempacota_double ();
return ( valor );
}

void empacota_double ( double *ptr_dupla_precisao )
{
*((double *)& mensagem [ tam_msg_envio ] ) = *ptr_dupla_precisao;
tam_msg_envio += sizeof ( double );
}

void transmite ( double temp, double valor )
{
int i;
tam_msg_envio = 0;
empacota_double ( &temp );
empacota_double ( &valor );
for ( i = 0; i < tam_msg_envio; i++ ) {
    tw = timewait;
    while ( tw-- );
    envia_car ( mensagem [ i ] );
};
}

```

Durante o envio e a recepção de mensagem alguns problemas podem ocorrer. Caso um byte de mensagem seja enviado sem que o anterior tenha sido lido, haverá perda da informação anterior. Isto é conhecido como *problema de superposição*. Como o computador receptor espera por oito bytes e um deles foi perdido, os dois computadores ficam em estado de espera infinitamente. O sistema fica então bloqueado.

4.3.2 Interface C-Matlab

O desenvolvimento desta interface faz-se necessário pois, como foi colocado anteriormente em detalhes, o acesso às portas de comunicação serial é feito através da linguagem C. A figura 4.8 ilustra esta interface. Nas próximas seções serão mostrados alguns detalhes desta implementação.

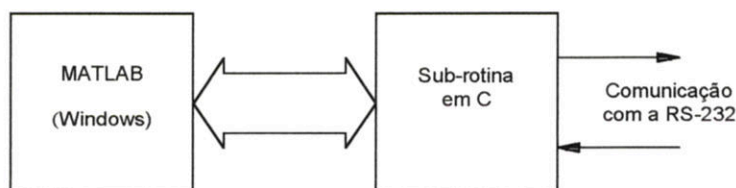


Figura 4.8 - Interface entre C e Matlab.

4.3.2.1 Arquivos MEX

Rotinas construídas em linguagem de programação C ou Fortran podem ser chamadas a partir do Matlab, como se fossem funções do tipo M (funções construídas em linguagem Matlab). Estes programas são chamados de arquivos MEX, sendo que algumas das vantagens de seu uso podem ser citadas:

- Grande número de códigos desenvolvidos em C e Fortran já existe. Descarta-se então a necessidade de reescrevê-los como arquivos M;
- Códigos que não são executados de forma suficientemente rápida podem ser recodificados em C ou Fortran, para maior eficiência;
- Placas A/D e D/A ou outro software pode ser acessado diretamente para aquisição de dados ou para aplicações de controle.

Se uma função qualquer é chamada de dentro do Matlab, o interpretador procura no *Matlabpath*, parando em cada diretório a procura de arquivos .mex ou .m. Caso os dois tipos de arquivo forem encontrados, o arquivo .mex tem precedência.

Arquivos mex são subrotinas dinamicamente lincáveis, escritas em C ou Fortran, que são lincadas com um módulo de interface especial.

4.3.2.2 Fluxo de informação C-Matlab/Matlab-C.

Em qualquer tipo de interface entre programas, linguagens ou ferramentas diferentes ocorrem erros devido a passagem/retorno de parâmetros. Para evitar este tipo de erro, o Matlab define uma interface própria para o envio/recebimento de parâmetros. Esta interface própria deve ser seguida por qualquer módulo que tente realizar troca de dados como Matlab.

```
void mexFunction( int nlhs, Matrix *plhs[], int nrhs, Matrix *prhs[] )
{
    double *vr;

    // Testa o numero de argumentos
    if ( nrhs != 0 ) {
        mexErrMsgTxt( "ERRO : rotina nao utiliza argumentos" );
        return;
    } else if ( nlhs > 1 ) {
        mexErrMsgTxt( "ERRO : argumentos de saida diferente de 1" );
        return;
    }
    // Create a matrix for the return argument
    plhs[0] = mxCreateFull ( 1, 1, REAL );

    // Assign pointers to the various parameters
    vr = mxGetPr ( plhs[0] );

    inicia_uart ();
    vr [0] = recebe ();
    return;
}
```

4.4 Conclusões

Neste capítulo foram descritos os procedimentos utilizados para a implementação de uma estrutura de controle, que envolve dois microcomputadores e um simulador de sistemas de potência.

A descrição do esquema de controle proposto foi apresentada com detalhes. A necessidade do desenvolvimento de uma comunicação serial e da interface entre a linguagem C e Matlab para a implementação deste projeto foi colocada.

Detalhes da implementação desta comunicação serial e da interface da linguagem C e Matlab foram apresentados.

5. Resultados

5.1 Introdução

Este capítulo apresenta os resultados e a análise de alguns testes realizados usando o esquema de controle apresentado no Capítulo 4.

Foram realizados testes com dois sistemas de potência distintos. O primeiro sistema é constituído por duas barras e duas máquinas, sendo que uma das máquinas representa uma barra-infinita. O outro sistema constitui-se de nove barras e três máquinas. Detalhes dos sistemas utilizados são dados nas seções seguintes e no Apêndice.

O objetivo primordial dos testes realizados foi o de constatar e comprovar a eficácia e aplicabilidade dos controladores adaptativos auto-ajustáveis a Sistemas Elétricos de Potência.

5.2 Descrição da Implementação do Controle

Foram utilizados dois microcomputadores do tipo PC interligados, trocando dados através de suas portas seriais. Em um dos computadores foram implementadas as rotinas de identificação e controle, em linguagem Matlab. Estas rotinas são apresentadas no Apêndice. No outro computador foi instalado um simulador de sistemas de potência. Detalhes da implementação deste esquema de controle foram mostrados no Capítulo 4.

O controlador adaptativo auto-ajustável foi utilizado para controlar o sistema simulado usando como entrada o sinal proveniente deste simulador. Isto procede-se da seguinte maneira: em um intervalo de amostragem, o simulador envia o sinal através da comunicação serial, até o outro computador. O Matlab recebe este dado e calcula o valor de controle, que é enviado ao simulador pela comunicação serial, no próximo intervalo de amostragem.

Para identificação dos sistemas foi utilizado o método dos mínimos quadrados recursivo, com modelos de segunda e de terceira ordem, a fim de comparar seus resultados. O método de *covariance resetting* foi usado, para que a matriz de covariância conservasse seu poder de identificação, no caso de qualquer mudança no sistema.

Para a excitação do sistema foi utilizado um sinal PRBS, conforme apresentado no Capítulo 3. O sinal foi implementado em uma rotina em linguagem Matlab, apresentada no Apêndice.

Para a identificação do modelo de segunda ordem, o sistema é representado no plano Z pela seguinte função de transferência:

$$H(z^{-1}) = \frac{(b_0 + b_1 z^{-1})z^{-1}}{1 + a_1 z^{-1} + a_2 z^{-2}}$$

onde parâmetros a serem rastreados pelo identificador são b_0 , b_1 , a_1 e a_2 .

Para a identificação do modelo de terceira ordem, o sistema é representado no plano Z pela função de transferência abaixo:

$$H(z^{-1}) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3}}$$

Neste caso, os parâmetros a serem rastreados pelo identificador são b_0 , b_1 , b_2 , a_1 , a_2 e a_3 .

O método de controle utilizado no controlador adaptativo auto-ajustável é o posicionamento de pólos, através de um fator de redução α , que desloca os pólos em direção à origem do plano Z, conforme apresentado no Capítulo 3.

Os resultados obtidos para os dois sistemas são apresentados nas próximas seções.

5.2 Sistema de Duas Barras

Este sistema é do tipo máquina-barra-infinita, sendo que a barra-infinita é representada por uma máquina de modelo clássico, com alta inércia e baixa impedância.

O diagrama deste sistema é mostrado na Figura 5.1.



Figura 5.1a - Sistema de duas barras.

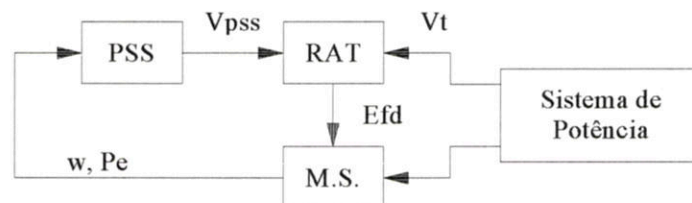


Figura 5.1b - Diagrama de blocos do Gerador I e seus controladores.

A unidade geradora I, apresentada em detalhes na Figura 5.1b, é formada pela máquina síncrona (MS), um estabilizador de sistemas de potência (PSS), que no caso é um controlador adaptativo auto-ajustável e um

regulador automático de tensão (RAT). Este regulador é representado por um ganho e uma constante de tempo. A máquina I está ligada ao sistema de potência, que é constituído pela linha de transmissão e pela máquina II (barramento-infinito). Maiores detalhes sobre este sistema são encontrados no Apêndice.

5.2.1 Implementação

Para identificação deste sistema foi utilizado o método dos mínimos quadrados recursivo, com modelos de segunda e de terceira ordem, a fim de comparar seus resultados. Foi utilizada uma matriz de covariância inicial de 10^9 , na diagonal principal. O método de *covariance resetting* foi usado, cada vez que o traço da matriz de covariância baixava para menos de 10^3 , para que esta conservasse sua capacidade de identificação, caso houvesse qualquer mudança no sistema. Os parâmetros do vetor *teta* foram inicializados em 10^{-3} , para que houvesse uma convergência mais rápida para os parâmetros verdadeiros. A simulação foi realizada durante 10 segundos, com o período de amostragem de 0,1 segundo.

Para a excitação do sistema foi utilizado um sinal PRBS, com $b=0.001$.

Os resultados para a identificação do modelo de segunda ordem para o sistema de potência são apresentados na Figura 5.2 e os resultados para a identificação do modelo de terceira ordem para o sistema de potência são apresentados na Figura 5.3.

Através da Figuras 5.2 e 5.3 pode-se observar a boa convergência dos parâmetros, para ambos os modelos, em um tempo relativamente curto. Melhores resultados ainda poderiam ser obtidos, através de ajustes no identificador.

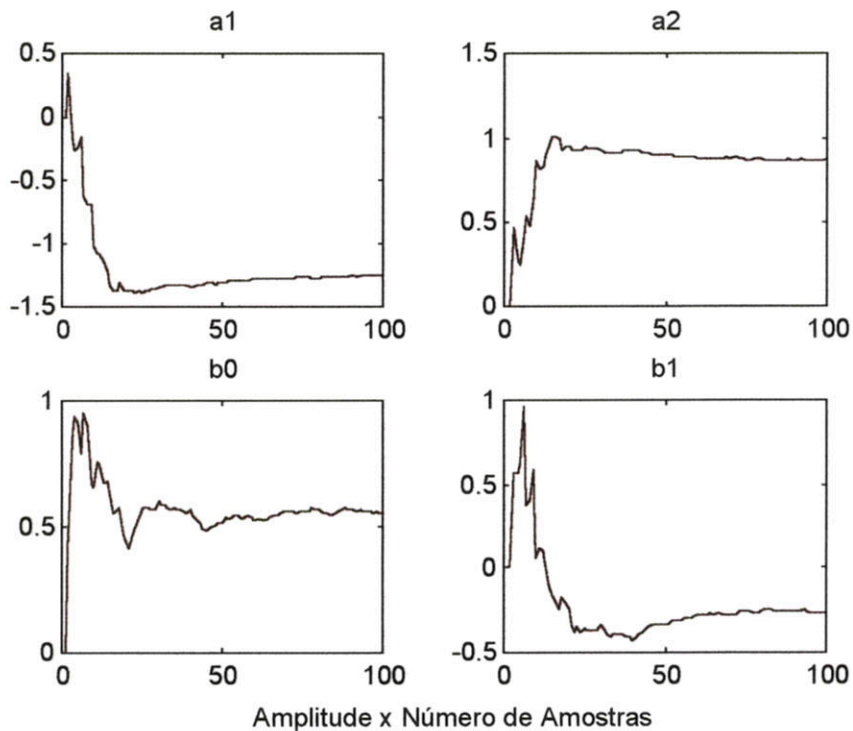


Figura 5.2 - Identificação de Segunda Ordem para o Sistema de duas Barras

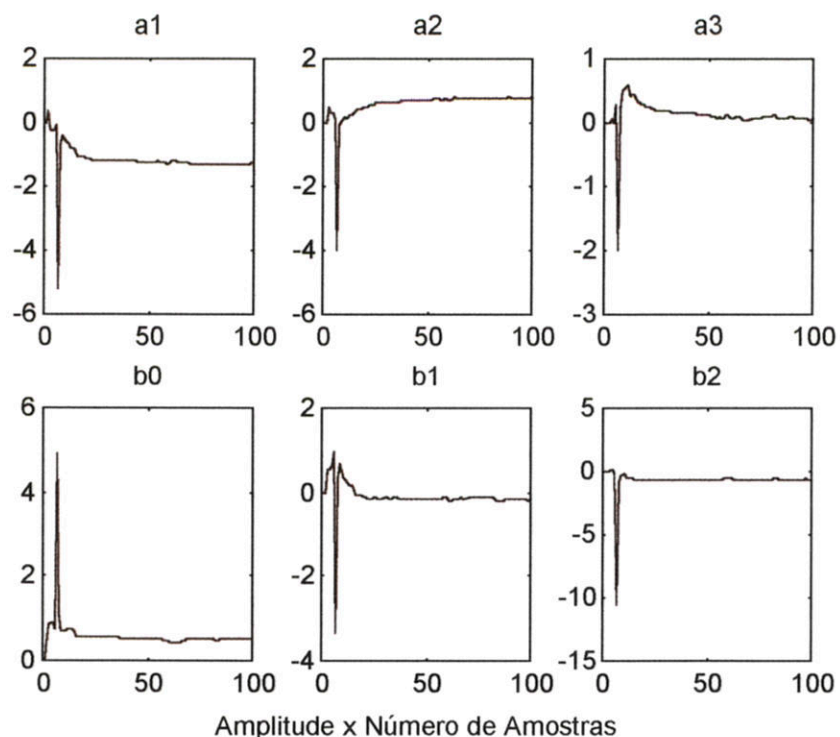


Figura 5.3 - Identificação de Terceira Ordem para o Sistema de duas Barras.

A técnica de controle empregada é a Alocação de Pólos, com fator de redução $\alpha=0.9$. Há um atraso de um período de amostragem entre o cálculo do valor de controle e da sua aplicação no sistema.

Para testar a eficiência do controlador adaptativo, foi aplicado um curto-circuito de duração de um período de amostragem, no tempo de simulação $t=5$ segundos. Este curto-circuito agiu como uma perturbação sobre o sistema.

O comportamento do sistema sob ação dos controladores de segunda e terceira ordem pode ser analisado nas Figuras 5.4 e 5.5, respectivamente. As curvas representam o ângulo da máquina síncrona. A linha pontilhada representa a resposta do sistema em malha aberta, sob as mesmas condições.

Pela análise das curvas apresentadas nas Figuras 5.4 e 5.5 pode-se notar o bom desempenho dos controladores adaptativos de segunda e terceira ordem. Ambos os controladores são eficazes no amortecimento das oscilações, após a aplicação da perturbação.

Para o caso com o controlador de segunda ordem, as oscilações são amortecidas rapidamente e o sistema volta a operar nas mesmas condições anteriores ao curto-circuito. A resposta para o sistema, quando um controlador de terceira ordem do sistema também é satisfatória, porém um pouco mais oscilatória, quando comparada com a resposta do sistema com controlador de segunda ordem. Ajustes no controlador adaptativo podem melhorar a resposta de maneira significativa. Tais ajustes não foram realizados devido a limitação de tempo.

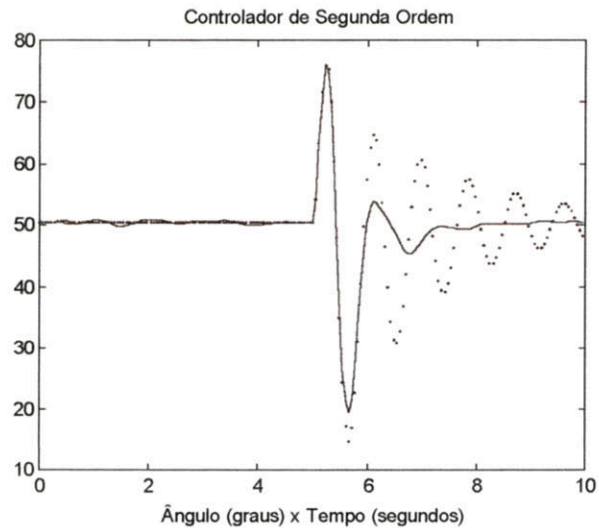


Figura 5.4 - Comportamento do Sistema com Controlador de Segunda Ordem.

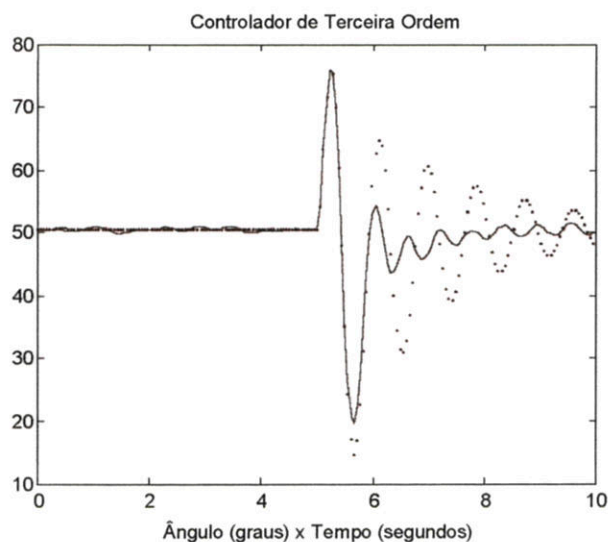


Figura 5.5 - Comportamento do Sistema com o Controlador de Terceira Ordem.

É importante observar que a perturbação não retirou o sistema do seu ponto de operação. Esta mudança quando o sistema sofre uma perturbação é perfeitamente possível. Um controlador convencional provavelmente teria uma performance pobre, já que são projetados para um determinado ponto de operação.

Para analisar o desempenho de um controlador adaptativo auto-ajustável para esta condição, foi realizado um novo teste. A abertura de $\frac{1}{4}$ de linha de transmissão foi realizada aos cinco segundos de simulação. Para esta simulação foi utilizado o controlador de terceira ordem. A resposta do sistema é apresentada na Figura 5.6.

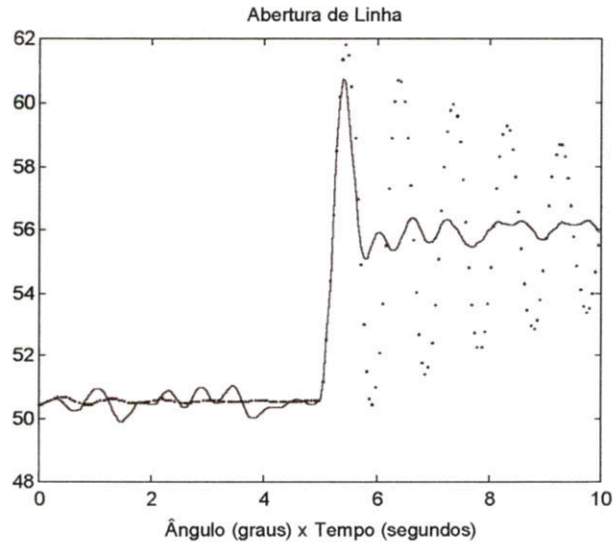


Figura 5.6 - Resposta do Sistema com Controlador de Terceira ordem, com abertura de Linha aos 5 segundos.

Como pode ser observado na resposta do sistema, o ponto de operação do sistema foi bruscamente modificado, após a abertura da linha de transmissão. O controlador adaptativo amorteceu as oscilações de maneira satisfatória, mesmo com esta nova condição de operação. Novamente aqui poderiam ter sido feitos mais ajustes para que as oscilações fossem amortecidas de maneira ainda mais eficiente. Pela mesma razão anteriormente citada, isto não foi feito.

5.3 Sistema de 9 Barras

Este sistema possui nove barras, nove linhas de transmissão, três cargas e três máquinas síncronas. As máquinas estão ligadas nas barras 1, 2 e 3. As cargas são representadas por impedâncias constantes e estão ligadas nas barras 5, 6 e 8. O diagrama do sistema é mostrado na Figura 5.8.

Este é um sistema largamente utilizado para testes em sistemas de energia elétrica. Os dados são encontrados na referência [AND77].

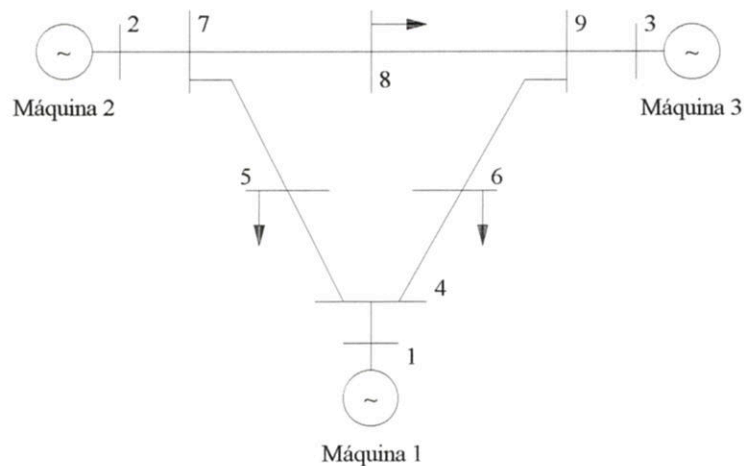


Figura 5.8 - Sistema de Nove Barras.

5.3.1 Implementação

Para a identificação deste sistema foi utilizado o método dos mínimos quadrados recursivo, sob as mesmas condições estabelecidas para o sistema de duas barras, apresentadas na seção anterior.

Para este sistema foi utilizado apenas o identificador de terceira ordem. Devido ao porte deste sistema de nove barras, sua dinâmica não seria bem representada por um modelo de segunda ordem.

Os resultados para a identificação do sistema são mostrados na Figura 5.9.

A identificação dos parâmetros é bastante razoável, sendo que a convergência destes parâmetros é dado em um curto intervalo de tempo.

A técnica de controle utilizada para este sistema foi a alocação de pólos com deslocamento radial. O fator de deslocamento utilizado para este caso foi $\alpha=0.6$, devido ao grande porte do sistema.

Uma simulação de 10 segundos foi realizada. O período de amostragem utilizado é de 0.1 segundo. O controlador foi colocado na máquina 2, sendo que a máquina 1 foi usada como referência. As respostas das máquinas 2 e 3, sob a ação de um curto circuito aplicado aos 5 segundos de simulação, são apresentadas na Figura 5.10. As curvas apresentadas representam o ângulo das máquinas, sendo que a linha pontilhada representa o sistema de nove barras em malha aberta.

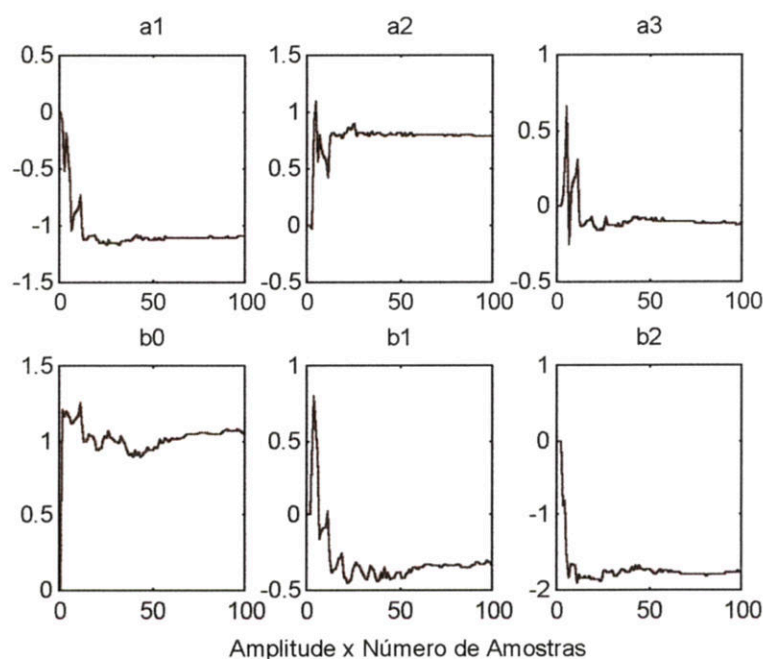


Figura 5.9 - Identificação para o sistema de nove barras.

Pela análise das Figura 5.10a e 5.10b, pode-se notar que houve amortecimento das oscilações, porém não com a mesma intensidade que no caso do sistema de duas barras, apresentado na seção anterior. Isto deve-se ao fato de que na prática é utilizado um estabilizador (neste caso, o controlador adaptativo auto-ajustável) para cada máquina. Por razões de tempo, não foi possível implementar uma rotina de controle que permitisse o cálculo de parâmetros de vários controladores.

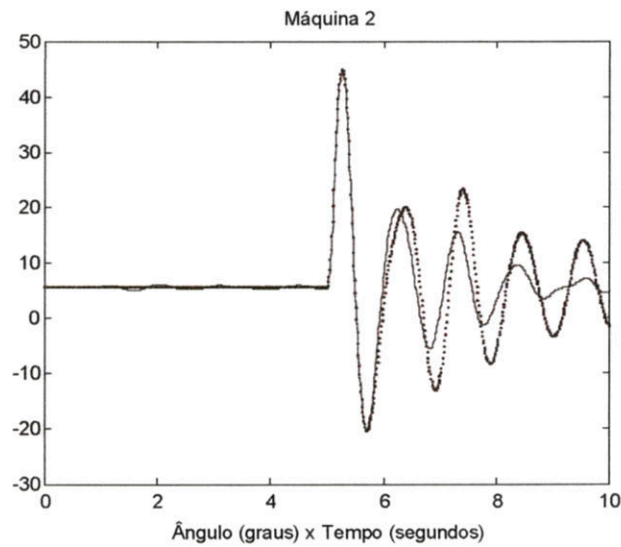


Figura 5.10a - Resposta da Máquina 2.

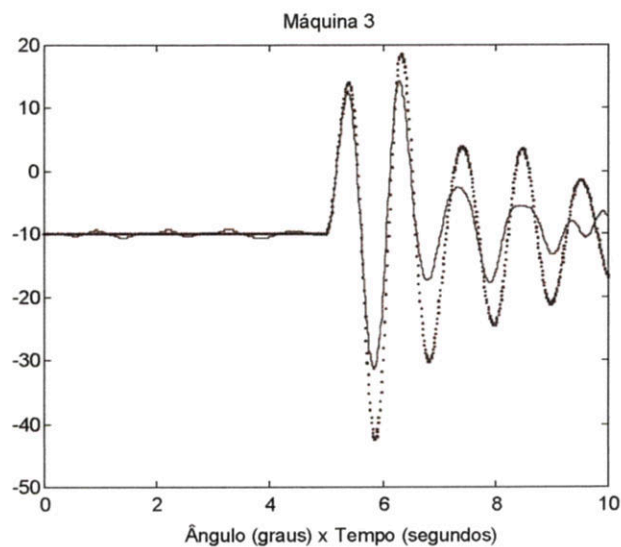


Figura 5.10b - Resposta da Máquina 3.

5.4 Conclusões

Neste capítulo foram mostrados e analisados os resultados obtidos para dois sistemas de potência diferentes. O primeiro sistema é um sistema de duas barras, com duas máquinas, sendo que uma destas máquinas representa uma barra-infinita. O outro sistema analisado é um sistema de nove barras, largamente utilizado para testes em sistemas de energia elétrica.

O método de identificação utilizado para os dois sistemas foi o método dos mínimos quadrados recursivo, sendo que para o sistema de duas barras foram feitos testes com identificação de modelos de segunda e terceira ordem e para o sistema de nove barras apenas foi utilizado o modelo de terceira ordem, devido ao porte deste sistema. Todos os resultados foram satisfatórios, com rápida convergência dos parâmetros identificados.

A técnica de controle empregada para os dois casos foi a alocação de pólos com deslocamento de pólos, já que na prática esta é a técnica utilizada. O fator α foi escolhido em função da aplicação.

Para o sistema de duas barras, testes com controladores de segunda e terceira ordem são apresentados. Ambos amorteceram as oscilações causadas pela aplicação de uma perturbação, um curto-circuito com duração de um período de amostragem. Para comprovar a eficiência do controle auto-ajustável foi aplicada uma perturbação que retirou o sistema de seu ponto de operação. O controlador adaptativo teve um ótimo desempenho, amortecendo as oscilações causadas, onde um controlador tradicional teria uma performance pobre, já que são projetados para um ponto de operação específico.

Para o sistema de nove barras, os testes foram realizados apenas com um controlador adaptativo auto-ajustável de terceira ordem, pois a dinâmica do sistema não seria bem representada por um modelo de segunda ordem, devido ao seu porte. Um curto-circuito foi aplicado para que o desempenho do controlador, sob a ação de uma perturbação, fosse analisado. O controlador amorteceu as oscilações, porém não obteve a mesma eficiência que no caso anterior. Isto deve-se ao fato de que na prática é utilizado um controlador adaptativo auto-ajustável para cada máquina. Neste trabalho não foi possível a implementação de uma rotina de controle capaz de calcular os parâmetros do controle para várias máquinas simultaneamente.

Alguns ajustes em vários parâmetros do cálculo da identificação e do controle poderiam ter sido feitos, a fim de melhorar o desempenho do sistema como um todo. Isto não pode ser feito por limitação de tempo.

6. Conclusões Gerais

Este trabalho de fim de curso tem como objetivo a implementação de um esquema de controle, sendo que o processo em questão é um Sistema de Potência. Os estabilizadores de sistemas de potência foram desenvolvidos através de técnicas de controle adaptativo auto-ajustável. A implementação prática deste projeto visa a observação de aspectos relevantes da implementação real, como a necessidade de desenvolvimento de uma comunicação serial entre controlador-processo e uma interface entre a linguagem C e o Matlab, para que esta comunicação serial pudesse ser realizada.

O modelo do sistema de potência foi apresentado no Capítulo 2, sendo formulado em termos dos diversos componentes do sistema de potência. O problema das oscilações eletromecânicas foi apresentado.

A teoria de controle adaptativo auto-ajustável é mostrada no Capítulo 3. Os métodos de identificação recursiva e as técnicas de controle aplicados a sistemas de potência são citados e discutidos, juntamente com os detalhes relevantes para a implementação destes controladores adaptativos auto-ajustáveis. No Capítulo 4 o esquema de controle proposto é descrito. A necessidade do desenvolvimento da comunicação serial entre computadores e a interface entre as linguagens C e Matlab é mostrada. Detalhes das implementações são colocados. No Capítulo 5 são mostrados e analisados os resultados de testes feitos com dois sistemas de potência distintos, usando controladores adaptativos auto-ajustáveis como estabilizadores de sistemas de potência.

Os testes realizados com os Sistemas de Potência mostram que os controladores adaptativos auto-ajustáveis apresentam bom desempenho. Estes controladores não necessitam ser projetados, pois se adaptam ao sistema. Os parâmetros do modelo que representa o sistema são corretamente identificados em um tempo relativamente baixo. As oscilações ocasionadas por ação de perturbações são bem amortecidas, em um intervalo de tempo aceitável. O controlador reage bem quando uma perturbação é aplicada, retirando o sistema de seu ponto de operação, fato este comum em sistemas elétricos de potência. Um controlador tradicional dificilmente teria um bom desempenho sob esta condição, dado que são projetados para funcionamento em determinado ponto de operação previamente especificado, devido à não-linearidade destes sistemas.

O esquema de controle proposto por este trabalho dificilmente poderia ser aplicado na prática, pois por razão de simplicidade os métodos de identificação e controle são implementados em Matlab. Esta linguagem é interpretada, e portanto muito lenta para fins industriais. Uma alternativa para a aplicação prática deste tipo de controlador é a utilização de uma linguagem mais rápida e compilada, como por exemplo, a Linguagem C. Com isto, o tempo gasto na interface entre linguagens seria eliminado. Atualmente, existe um compilador que transforma código em Matlab em códigos em C. Isto poderia ser utilizado como uma alternativa válida.

A substituição do simulador de sistemas de potência por um modelo analógico deste tipo de sistema é uma sugestão para projetos futuros. Através desta substituição, outros problemas ocorridos na prática poderiam ser analisados, entre eles a conversão de sinais analógicos para sinais digitais e sinais digitais para analógicos, a escolha do período de amostragem, etc.

Referências Bibliográficas

- [ALL90] A.Y.Allidina, B.K. Khosrowshahi e F.M. Hughes. *Implicit Pole-Assignment Self-tuning Algorithm*. Int. J. Control, v.51, 1, p.51-70, 1990.
- [AND77] P.M. Anderson e A.A. Fouad. *Power System control and stability*. The Iowa State University Press, 1977.
- [AST89] K.J. Aström e B. Wittenmark. *Adaptive Control*. Addison Wesley Publishing Company, Series in Eletrical Engineering: Control Engineering, 1989
- [BAR95] J. A. L. Barreiros. *Métodos de Controle Adaptativo Auto-Ajustável Aplicados à Síntese de Estabilizadores de Sistemas de Potência*. Tese de Doutorado, UFSC, 1995.
- [BAZ93] A.S.Bazanella. *Controle Auto-Ajustável de Sistemas de Potências*. Dissertação de Mestrado, UFSC, 1993.
- [DAV70] W.D.T. Davies. *System Identification for Self-Adaptive Control*. John Wiley & Sons Ltda, 1970.
- [FOR81] .T.R. Fortescue, L.S.Kershenbaum e B.E.Ydstie. *Implementation of Self-tuning Regulators with Variable Forgetting Factors*. Automatica, v.17,6,p.831-835, 1981.
- [GOO84] G.C. Goodwin e K.S. Sin. *Adaptive Filtering, Prediction and Control*. Prentice Hall, 1984.
- [HAR85] C.J. Harris e S.A.Billings. *Self-tuning and Adaptive Control: Theory and Application*. Peter Peregrinus, 1985.
- [HUG86] F.M. Huges. *Self-Tuning and Adaptive Control - A Review of some Basic Techniques*. Trans. Inst. M.C., v. 8, n. 2, p.100-110, 1986.
- [MAN96] A. Manzoni. *Desenvolvimento de um módulo dinâmico de um simulador para o ensino e treinamento de sistemas de potência usando programação orientada a objeto*. Dissertação de Mestrado, UFSC, 1996.
- [NAR80] K.S. Narendra e R.V.Monopoli. *Applications of Adaptive Control*. New York, Academic Press, 1980.
- [NOR91] P. Norton e R.Wilton. *Novo guia para programadores do IBM PC&PS/2*. Ed. Campus: Rio de Janeiro, 1991.
- [SEB86] D.E. Seborg, T.F.Edgart e S.L.Shah. *Adaptive Control Strategies for Process Control: A Survey*. AIChE Journal 32,6,p.881-913, 1986.

[SIL92] A.S. Silva e J.A.L. Barreiros. *Application of Adaptive Controllers to a Multimachine Power Systems*. LATICON'92 Conferencia Latino-Americana del IEEE, Santiago, Chile, p.9-13, 1992.

[WEL82] P.E. Wellstead e P.Zanker. *Techniques on Self-tuning*. Technical Report n.432, Control System Centre, UMIST, July, 1982.

Apêndice

Rotina de Envio

```

#include <math.h>
#include "mex.h"

#include <conio.h>
#include <stdio.h>
#include <stdlib.h>

#define TAM_MAX_MSG 50

int PORTA=0x2f8;

long int timewait = 10000, tw;

unsigned char tamanho_mensagem = 2*sizeof(double),
             auxiliar_tam_msg = 0,
             tam_msg_envio,
             mensagem [ TAM_MAX_MSG ]; /* buffer de recepcao e transmissao */

/* ===== */
/*      testa o estado do registro de transmissao      */
/* ===== */

unsigned int espera_TX ()
{ return ( (unsigned int)(inp(PORTA+5) & 0x20) ); }

/* ===== */
/*      testa status da maquina de transmissao      */
/* ===== */

unsigned int espera_MTX ()
{ return ( (unsigned int)(inp(PORTA+5) & 0x40) ); }

/* ===== */
/*      envia um caracter para PORTA      */
/* ===== */

void envia_car ( char caracter )
{
    while ( espera_TX() == 0x00 );
    outp ( PORTA, caracter );
    while ( espera_TX() == 0x00 ); /* espera acabar de transmitir */
    while ( espera_MTX() == 0x00 );
}

/* ===== */
/*      inic. porta p/caracter de 8 bits, velocidade de 9600 bps ...      */
/* ===== */

void inicia_uart ()
{
    outp ( PORTA+3, 0x80); /* prepara porta para a programacao de velocidade */
    outp ( PORTA, 0x0c); /* programa porta para 9600 bps */
}

```



```

outp ( PORTA+1, 0x00); /* programa porta para 9600 bps */
outp ( PORTA+3, 0x07); /* programa caracter para 8 bits, 1 stop bits etc */
}

/*=====
==*/
/*===== ROTINAS PARA TRATAMENTO DE ENVIO A UM NIVEL MEDIO
=====*/
/*=====
==*/

void empacota_double ( double *ptr_dupla_precisao )
{
*((double *)& mensagem [ tam_msg_envio ] ) = *ptr_dupla_precisao;
tam_msg_envio += sizeof ( double );
}

void transmite ( double temp, double valor )
{
int i;
tam_msg_envio = 0;
empacota_double ( &temp );
empacota_double ( &valor );
for ( i = 0; i < tam_msg_envio; i++ ) {
tw = timewait;
while ( tw-- );
envia_car ( mensagem [ i ] );
};
}

/*=====
==*/
/*===== INTERFACE ENTRE O C++ E O Matlab =====*/
/*=====
==*/

void mexFunction( int nlhs, Matrix *plhs[], int nrhs, Matrix *prhs[] )
{
double *t,*v;

// Testa o numero de argumentos
if ( nrhs != 2 ) {
mexErrMsgTxt( "ERRO : argumentos de entrada diferente de 2" );
return;
} else if ( nlhs > 1 ) {
mexErrMsgTxt( "ERRO : rotina nao retorna argumentos" );
return;
}

// Assign pointers to the various parameters
t = mxGetPr ( prhs[0] );
v = mxGetPr ( prhs[1] );

inicia_uart ();
transmite ( t[0], v[0] );
return;
}

```

Rotina de Recepção

```

#include <math.h>
#include "mex.h"

#include <conio.h>
#include <stdio.h>
#include <stdlib.h>

#define TAM_MAX_MSG 50

int PORTA=0x2f8;

unsigned char tamanho_mensagem = 2*sizeof(double),
             auxiliar_tam_msg = 0,
             tam_msg_envio,
             mensagem [ TAM_MAX_MSG ]; /* buffer de recepcao e transmissao */

/* ===== */
/*   permite testar o status do registro de recepcao   */
/* ===== */

espera_RX ()
{ return ( inp(PORTA+5) & 0x01 ); }

/* ===== */
/* retorna um caracter previamente testado com espera_RX */
/* ===== */

char recebe_car ()
{ while ( espera_RX()==0x00 ); return ( inp(PORTA) ); }

/* ===== */
/* inic. porta p/caracter de 8 bits, velocidade de 9600 bps ... */
/* ===== */

void inicia_uart ()
{
  outp ( PORTA+3, 0x80); /* prepara porta para a programacao de velocidade */
  outp ( PORTA, 0x0c); /* programa porta para 9600 bps */
  outp ( PORTA+1, 0x00); /* programa porta para 9600 bps */
  outp ( PORTA+3, 0x07); /* programa caracter para 8 bits, 1 stop bits etc */
}

/*=====
==*/
/*===== ROTINAS PARA TRATAMENTO DE RECEPCAO A NIVEL INTEMEDIARIO
=====*/
/*=====
==*/

double desempacota_double ()
{
  double *auxiliar;
  auxiliar = (double *)& mensagem [ auxiliar_tam_msg ];
  auxiliar_tam_msg += sizeof ( double );
  return ( *auxiliar );
}

double recebe ()
{

```

```

double temp, valor;
// --- Prepara a recepcao ---
int i;
for ( i = 0; i++; i < tamanho_mensagem ) mensagem [ i ] = 0;
auxiliar_tam_msg = 0;
// --- Espera ate receber toda a mensagem ---
while (1) {
    mensagem [ auxiliar_tam_msg++ ] = recebe_car ();
    if ( auxiliar_tam_msg == tamanho_mensagem ) break;
};
auxiliar_tam_msg = 0;
temp = desempacota_double ();
valor = desempacota_double ();
return ( valor );
}

/*=====
==*/
/*===== INTERFACE ENTRE O C++ E O Matlab =====*/
/*=====
==*/

void mexFunction( int nlhs, Matrix *plhs[], int nrhs, Matrix *prhs[] )
{
    double *vr;

    // Testa o numero de argumentos
    if ( nrhs != 0 ) {
        mexErrMsgTxt( "ERRO : rotina nao utiliza argumentos" );
        return;
    } else if ( nlhs > 1 ) {
        mexErrMsgTxt( "ERRO : argumentos de saida diferente de 1" );
        return;
    }

    // Create a matrix for the return argument
    plhs[0] = mxCreateFull ( 1, 1, REAL );

    // Assign pointers to the various parameters
    vr = mxGetPr ( plhs[0] );

    inicia_uart ();
    vr [0] = recebe ();
    return;
}

```

Interface com Matlab

```

#include <conio.h>
#include <stdio.h>
#include <stdlib.h>

#define TAM_MAX_MSG 50

int PORTA=0x2f8;

long int timewait = 10000, tw;

unsigned char tamanho_mensagem = 2*sizeof(double),
             auxiliar_tam_msg = 0,
             tam_msg_envio,
             mensagem [ TAM_MAX_MSG ]; /* buffer de recepcao e transmissao */

/* ===== */
/*   testa o estado do registro de transmissao   */
/* ===== */

unsigned int espera_TX ()
{ return ( (unsigned int)(inp(PORTA+5) & 0x20) ); }

/* ===== */
/*   testa status da maquina de transmissao   */
/* ===== */

unsigned int espera_MTX ()
{ return ( (unsigned int)(inp(PORTA+5) & 0x40) ); }

/* ===== */
/*   envia um caracter para PORTA   */
/* ===== */

void envia_car ( char caracter )
{
    while ( espera_TX() == 0x00 );
    outp ( PORTA, caracter );
    while ( espera_TX() == 0x00 ); /* espera acabar de transmitir */
    while ( espera_MTX() == 0x00 );
}

/* ===== */
/*   permite testar o status do registro de recepcao   */
/* ===== */

espera_RX ()
{ return ( inp(PORTA+5) & 0x01 ); }

/* ===== */
/* retorna um caracter previamente testado com espera_RX */
/* ===== */

char recebe_car ()
{ while ( espera_RX()==0x00 ); return ( inp(PORTA) ); }

/* ===== */
/* inic. porta p/caracter de 8 bits, velocidade de 9600 bps ... */
/* ===== */

```

```

void inicia_uart ()
{
  outp ( PORTA+3, 0x80); /* prepara porta para a programacao de velocidade */
  outp ( PORTA, 0x0c); /* programa porta para 9600 bps */
  outp ( PORTA+1, 0x00); /* programa porta para 9600 bps */
  outp ( PORTA+3, 0x07); /* programa caracter para 8 bits, 1 stop bits etc */
}

/*=====
==*/
/*===== ROTINAS PARA TRATAMENTO DE RECEPCAO A NIVEL INTEMEDIARIO
=====*/
/*=====
==*/

double desempacota_double ()
{
  double *auxiliar;
  auxiliar = (double *)& mensagem [ auxiliar_tam_msg ];
  auxiliar_tam_msg += sizeof ( double );
  return ( *auxiliar );
}

double recebe ()
{
  double temp, valor;
  // --- Prepara a recepcao ---
  int i;
  for ( i = 0; i++; i < tamanho_mensagem ) mensagem [ i ] = 0;
  auxiliar_tam_msg = 0;
  // --- Espera ate receber toda a mensagem ---
  while (1) {
    mensagem [ auxiliar_tam_msg++ ] = recebe_car ();
    if ( auxiliar_tam_msg == tamanho_mensagem ) break;
  };
  auxiliar_tam_msg = 0;
  temp = desempacota_double ();
  valor = desempacota_double ();
  return ( valor );
}

/*=====
==*/
/*===== ROTINAS PARA TRATAMENTO DE ENVIO A UM NIVEL MEDIO
=====*/
/*=====
==*/

void empacota_double ( double *ptr_dupla_precisao )
{
  *((double *)& mensagem [ tam_msg_envio ] ) = *ptr_dupla_precisao;
  tam_msg_envio += sizeof ( double );
}

void transmite ( double temp, double valor )
{
  int i;
  tam_msg_envio = 0;
  empacota_double ( &temp );
  empacota_double ( &valor );
  for ( i = 0; i < tam_msg_envio; i++ ) {
    tw = timewait;
  }
}

```

```
while ( tw-- );  
envia_car ( mensagem [ i ] );  
};  
}
```

Rotinas de identificação e controle

```

% Nicole Sharon Orfali (LABSPOT)
%
% Estimador de Mínimos Quadrados Recursivo Estendido
%
% Controle por Alocação de Polos
%
% Sem Atualização de P(t)
%
% Segunda Ordem com Identificação de b0, b1, a1, a2 (sem Ident. do Erro)

% VALORES MAIS RECENTES OCUPAM AS ÚLTIMAS POSIÇÕES DOS VETORES

clear
clc

tempo_sim = 15;
interv_amostr=0.1;
num_leit_ident=10;

num_leit_tot= tempo_sim / interv_amostr;

%Condições Iniciais

y=[0 0 0];
u=[0 0 0];
yc=[0 0 0];
e=0 ;
teta=[.001 .001 .001 .001];
v=[1 0 0 1 0 0];
pinic=10^9;
p=pinic*eye(4);
me=0.0;
mr=0.0;
sinal=0.0;

% Identificação e Cálculo do valor de Controle

for i=1:num_leit_tot
    % Interface com o simulador de SEE
    mr =ccomrec;
    [sinal,v]=prbs2(v);
    [ i ]
    ss = sinal-me;
    ccomenv(0,ss);

    % Eliminação dos Valores mais antigos
    yp =y (2:3);
    up =u (2:3);
    ycp=yc(2:3);
    y =[yp mr];
    u =[up ss];
    yc=[ycp me];

    % Identificação
    ym =[-y(2) -y(1) u(2) u(1)];
    yhat =ym*teta';
    e =y(3)-yhat;
    ganho =(p*ym')/(1+ym*p*ym');
    teta =teta+(ganho)*e;

```

```

p = p-ganho*(1+ym*p*ym)*(ganho');

if (i < 50)
    a1 =teta(1,1);
    a2 =teta(1,2);
    b0 =teta(1,3);
    b1 =teta(1,4);
end
if (i > 65)
    a1 =teta(1,1);
    a2 =teta(1,2);
    b0 =teta(1,3);
    b1 =teta(1,4);
end

% Guarda os valores para plotagem
aux1(i)=a1;
aux2(i)=a2;
aux3(i)=b0;
aux4(i)=b1;
aux5(i)=e;
xx(i)=trace(p);

% Cálculo do controle
if ( i > num_leit_ident )
    r =roots([1 a1 a2]);    % Cálculo do Polinômio T desejado
    r =0.9*r;
    T =poly(r);
    t1 =T(2); t2=T(3);

    Ax=[ 1 b0 0 ;
         a1 b1 b0 ;
         a2 0 b1 ];

    bx=[t1-a1; t2-a2; 0];

    x=Ax\bx;
    f1=x(1);
    g0=x(2); g1=x(3);

    f1i(i)=f1; g0i(i)=g0; g1i(i)=g1; % Armazena parâmetros do controle

    me =-f1*yc(2)+g0*y(3)+g1*y(2);

    if (me > 0.1)            % Limitador [-0.1, 0.1]
        me=0.1;
    end
    if (me < -0.1)
        me=-0.1;
    end
end
end
end

```



```

% Nicole Sharon Orfali (LABSPOT)
%
% Estimador de Mínimos Quadrados Recursivo Estendido
%
% Controle por Alocação de Polos
%
% Sem Atualização de P(t)
%
% Terceira ordem com Identificação de b0,b1,b2,a1,a2,a3 (sem Ident. do Erro)

% VALORES MAIS RECENTES OCUPAM AS ÚLTIMAS POSIÇÕES DOS VETORES

clear
clc

tempo_sim = 10;
interv_amostr=0.1;
num_leit_ident=10;

num_leit_tot= tempo_sim / interv_amostr;

%Condições Iniciais

y=[0 0 0 0];
u=[0 0 0 0];
yc=[0 0 0 0];
e=[0] ;
teta=[.001 .001 .001 .001 .001 .001];
v=[1 0 0 1 0 0];
pinic=10^9;
p=pinic*eye(6);
me=0.0;
mr=0.0;
sinal=0.0;

% Identificação e Cálculo do valor de Controle

for i=1:num_leit_tot

    % Interface com o simulador SEE
    mr =ccomrec;
    [sinal,v]=prbs2(v);
    i
    ss= sinal-me;
    ccomenv(0,ss);

    % Eliminação dos Valores mais antigos
    yp=y(2:4);
    up=u(2:4);
    ycp=yc(2:4);
    y= [yp mr];
    u= [up ss];
    yc=[ycp me];

    % Identificação
    ym =[-y(3) -y(2) -y(1) u(3) u(2) u(1) ];
    yhat =ym*teta';
    e =y(4)-yhat;
    ganho =(p*ym')/(1+ym*p*ym');
    teta =teta+(ganho)*e;
    p =p-ganho*(1+ym*p*ym')*(ganho');

```

```

% Parâmetros Identificados

if (i < 50)
    a1 =teta(1,1);
    a2 =teta(1,2);
    a3 =teta(1,3);
    b0 =teta(1,4);
    b1 =teta(1,5);
    b2 =teta(1,6);
end
if (i > 65)
    p=(10^6)*eye(6);
    a1 =teta(1,1);
    a2 =teta(1,2);
    a3 =teta(1,3);
    b0 =teta(1,4);
    b1 =teta(1,5);
    b2 =teta(1,6);
end

% Guarda valores para a plotagem
aux1(i)=a1;
aux2(i)=a2;
aux3(i)=a3;
aux4(i)=b0;
aux5(i)=b1;
aux6(i)=b2;

% Cálculo do Controle
if (i>num_leit_ident)
    r =roots([1 a1 a2 a3]);    % Cálculo do Polinômio T desejado
    r =0.7*r;
    T =poly(r);
    t1 =T(2); t2=T(3); t3=T(4);

    Ax=[ 1 0  b0 0 0 ;
         a1 1  b1 b0 0 ;
         a2 a1  b2 b1 b0 ;
         a3 a2  0  b2 b1 ;
         0 a3  0  0  b2];

    bx=[t1-a1; t2-a2; t3-a3; 0; 0];

    x=Ax\bx;
    f1=x(1); f2=x(2);
    g0=x(3); g1=x(4); g2=x(5);

    me =-f1*yc(3)-f2*yc(2)+g0*y(4)+g1*y(3)+g2*y(2);

    % Limitador
    if (me > 0.1)
        me=0.1;
    end
    if (me < -0.1)
        me=-0.1;
    end
end
end
end

```

PRBS

```
% Nicole Sharon Orfali (LABSPOT)
%
% Geração de um Sinal PRBS

function [sinal,v]=prbs2(v);

% Condições Iniciais
sinal=[];
b=0.001;

% Geração do Sinal

x=v(1);
y=v(6);
e=xor(x,y);
vl=v(2:6);
v=[vl e];

if e==0
    z=-b-(b*0.0158730158);
else
    z=b-(b*0.0158730158);
end

sinal=[z];
```