

UNIVERSIDADE ESTADUAL DE CAMPINAS  
FACULDADE DE ENGENHARIA ELÉTRICA  
DEPARTAMENTO DE COMPUTAÇÃO E AUTOMAÇÃO INDUSTRIAL  
FEVEREIRO 1991

UM SISTEMA BASEADO EM CONHECIMENTO  
PARA CONFIGURAÇÃO E SUPERVISÃO DE  
ALGORITMOS DE CONTROLE ADAPTATIVO

Este exemplar corresponde à redação final da tese  
defendida por Rubén Mario Nazzetta

\_\_\_\_\_ e aprovada pela Comissão  
Julgadora em 28 / 02 / 1991

Wagner G. Amaral  
Orientador

por: Rubén Mario Nazzetta

Orientador: Prof. Wagner G. Amaral *assinado*

Tese apresentada à Faculdade de  
Engenharia Elétrica FEE - UNICAMP  
como parte dos requisitos exigidos  
para a obtenção do título de MESTRE  
EM ENGENHARIA

Campinas - 1991

Este trabalho contou com o apoio financeiro do Conselho Nacional de Desenvolvimento Científico e Tecnológico - **CNPq**.

Dedico à minha família  
pelo apoio e o carinho

## AGRADECIMENTOS

Ao prof. Wagner C. Amaral, pela orientação, o interesse e a compreensão.

Ao prof. Fernando Gomide, à Rosane Minghim e ao Mario Cabrera, quem tornaram possível este trabalho.

Aos grandes amigos Silvio, Mario e Sandra, pelo apoio, incentivo e alegrias.

Ao amigo Domenico, pela sua ajuda e encorajamento na etapa final deste trabalho.

A todos os que leram e releeram (incansavelmente) este trabalho para a sua correção.

A todos os amigos da FEE, e em particular aos do LCA.

AO BRASIL, pela oportunidade oferecida.

## RESUMO

O presente trabalho detalha o desenvolvimento de um sistema baseado em conhecimento para o auxílio nas tarefas de configuração e supervisão de algoritmos de modelagem e controle de processos. O sistema consta dos seguintes módulos: Módulo de controle onde encontram-se os algoritmos de controle, de três termos (PID), de Variância Mínima Generalizada (GMV) e o Controlador Preditivo Generalizado (GPC); Módulo de identificação com o algoritmo de matriz estendida; Módulo de detecção de não estacionariedades do processo, baseado no algoritmo do erro previsto; Máquina de inferência com encadeamento reverso, utilizada junto com a Base de Conhecimento de Configuração; Máquina de inferência com encadeamento direto, utilizada com a Base de Conhecimento de Supervisão; Base de Dados; e Interface Homem-Máquina.

A Base de Dados, acessível a ambas as máquinas de inferência é composta de fatos escritos em linguagem natural. As bases de conhecimento possuem a estrutura de regras de produção do tipo SE <condição> ENTÃO <conclusão>. A máquina de inferência de configuração obtém os valores dos objetos utilizados nos algoritmos de controle e identificação para satisfazer os requisitos de desempenho especificados. A máquina de inferência de supervisão verifica se os requisitos de desempenho do sistema em operação são satisfeitos. Quando o desempenho está fora da especificação, devido a perturbações no processo ou vizinhança, a máquina sugere alterações nos objetos para satisfazer os requisitos de identificação e controle.

# ÍNDICE

|                                                                      |               |
|----------------------------------------------------------------------|---------------|
| <b>CAPÍTULO 1: INTRODUÇÃO</b> .....                                  | <b>1</b>      |
| 1.1.-INTRODUÇÃO .....                                                | 2             |
| 1.2.-SISTEMAS ESPECIALISTAS, DEFINIÇÃO E CARACTERÍSTICAS ....        | 4             |
| 1.3.-REPRESENTAÇÃO DO CONHECIMENTO .....                             | 8             |
| 1.3.1.-SISTEMAS BASEADOS EM ESQUEMAS .....                           | 9             |
| 1.3.2.-SISTEMAS BASEADOS EM REGRAS DE PRODUÇÃO .....                 | 10            |
| 1.4.-SISTEMAS ESPECIALISTAS BASEADOS EM REGRAS DE PRODUÇÃO ..        | 11            |
| 1.4.1.-BASE DE DADOS .....                                           | 12            |
| 1.4.2.-BASE DE CONHECIMENTO .....                                    | 13            |
| 1.4.3.-MÁQUINA DE INFERÊNCIA .....                                   | 14            |
| 1.4.3.1.-ESTRATÉGIA DE CONTROLE POR<br>ENCADEAMENTO DIRETO .....     | 14            |
| 1.4.3.2.-ESTRATÉGIA DE CONTROLE POR<br>ENCADEAMENTO REVERSO .....    | 16            |
| 1.4.3.3.-ESTRATÉGIA DE CONTROLE POR<br>ENCADEAMENTO MISTO .....      | 19            |
| 1.4.4.-INTERFACE HOMEM-MÁQUINA .....                                 | 19            |
| 1.5.-RESUMO .....                                                    | 20            |
| <br><b>CAPÍTULO 2: CONTROLE ADAPTATIVO</b> .....                     | <br><b>21</b> |
| 2.1.-INTRODUÇÃO .....                                                | 22            |
| 2.2.-MODELO DO PROCESSO .....                                        | 23            |
| 2.3.-IDENTIFICAÇÃO DO PROCESSO .....                                 | 26            |
| 2.3.1.-INTRODUÇÃO .....                                              | 26            |
| 2.3.2.-ESTIMADOR DOS MÍNIMOS QUADRADOS<br>PONDERADOS RECURSIVO ..... | 27            |

|                                                                                   |    |
|-----------------------------------------------------------------------------------|----|
| 2.3.3.-ESTIMADOR DA MATRIZ ESTENDIDA RECURSIVO .....                              | 29 |
| 2.3.3.1.-RUÍDO COLORIDO E PERTURBAÇÕES NA CARGA ...                               | 30 |
| 2.4.-IDENTIFICAÇÃO DE SISTEMAS NÃO-ESTACIONÁRIOS .....                            | 31 |
| 2.4.1.-PARÂMETROS COM VARIAÇÕES LENTAS .....                                      | 34 |
| 2.4.2.-PARÂMETROS COM VARIAÇÕES RÁPIDAS .....                                     | 34 |
| 2.4.2.1.-DETEÇÃO DE NÃO-ESTACIONARIEDADES .....                                   | 34 |
| 2.4.2.2.-ATUAÇÃO SOBRE O GANHO DE CORREÇÃO .....                                  | 36 |
| 2.4.2.3.-CONSIDERAÇÕES NO ALGORITMO DE DETECÇÃO DE<br>NÃO-ESTACIONARIEDADES ..... | 37 |
| 2.5.-EXCITAÇÃO DO PROCESSO .....                                                  | 39 |
| 2.5.1.-DETEÇÃO DE EXCITAÇÃO NÃO-PERSISTENTE .....                                 | 39 |
| 2.6.-ALGORITMO DE CONTROLE PID .....                                              | 40 |
| 2.7.-CONTROLADOR DE VARIÂNCIA MÍNIMA GENERALIZADA .....                           | 42 |
| 2.7.1.-INTRODUÇÃO .....                                                           | 42 |
| 2.7.2.-ALGORITMO DE CONTROLE GMV .....                                            | 43 |
| 2.7.2.1.-PREVISÃO DA SAÍDA "d" PASSOS À FRENTE ....                               | 44 |
| 2.7.2.2.-MINIMIZAÇÃO DA FUNÇÃO DE CUSTO .....                                     | 46 |
| 2.7.2.3.-FUNÇÃO DE TRANSFERÊNCIA EM MALHA FECHADA .                               | 49 |
| 2.7.2.4.-ELIMINAÇÃO DO ERRO DE REGIME .....                                       | 50 |
| 2.7.2.5.-RUÍDO COLORIDO E PERTURBAÇÕES NA CARGA ...                               | 51 |
| 2.8.-CONTROLADOR PREDITIVO GENERALIZADO .....                                     | 52 |
| 2.8.1.-INTRODUÇÃO .....                                                           | 52 |
| 2.8.2.-PREVISÃO DA SAÍDA "j" PASSOS À FRENTE .....                                | 54 |
| 2.8.3.-DECOMPOSIÇÃO DA SAÍDA .....                                                | 57 |
| 2.8.4.-FUNÇÃO DE CUSTO GENERALIZADA .....                                         | 61 |
| 2.8.5.-FUNÇÃO DE TRANSFERÊNCIA EM MALHA FECHADA .....                             | 66 |
| 2.8.6.-RUÍDO COLORIDO E PERTURBAÇÕES NA CARGA .....                               | 70 |
| 2.9.-ALGORITMOS DE CONTROLE E IDENTIFICAÇÃO .....                                 | 71 |
| 2.10.-RESUMO .....                                                                | 73 |

## CAPÍTULO 3: SISTEMA ESPECIALISTA EM CONTROLE ..... 75

|                                                         |    |
|---------------------------------------------------------|----|
| 3.1.-INTRODUÇÃO .....                                   | 76 |
| 3.2.-REPRESENTAÇÃO DO CONHECIMENTO .....                | 77 |
| 3.2.1.-ESTRATÉGIAS DE MANIPULAÇÃO DO CONHECIMENTO ..... | 79 |
| 3.3.-ESCOLHA DA LINGUAGEM DE IMPLEMENTAÇÃO .....        | 81 |

|                                                                           |     |
|---------------------------------------------------------------------------|-----|
| 3.4.-ESTRUTURAS DE ARMAZENAMENTO DO CONHECIMENTO .....                    | 83  |
| 3.4.1.-ESTRUTURA DA BASE DE DADOS .....                                   | 84  |
| 3.4.2.-ESTRUTURA DA BASE DE CONHECIMENTO .....                            | 85  |
| 3.4.2.1.-TIPOS DE REGRAS DA BASE DE<br>CONHECIMENTO DE CONFIGURAÇÃO ..... | 87  |
| 3.4.2.2.-TIPOS DE REGRAS DA BASE<br>DE CONHECIMENTO DE SUPERVISÃO .....   | 89  |
| 3.5.-MANIPULAÇÃO DO CONHECIMENTO .....                                    | 90  |
| 3.5.1.-MÁQUINA DE INFERÊNCIA COM ENCADEAMENTO REVERSO ..                  | 91  |
| 3.5.2.-MÁQUINA DE INFERÊNCIA COM ENCADEAMENTO DIRETO ...                  | 93  |
| 3.6.-INTERFACE HOMEM-MÁQUINA .....                                        | 95  |
| 3.7.-RESUMO .....                                                         | 102 |

## CAPÍTULO 4: BASES DE CONHECIMENTO ..... 104

|                                                                              |     |
|------------------------------------------------------------------------------|-----|
| 4.1.-INTRODUÇÃO .....                                                        | 105 |
| 4.2.-CONTROLADOR PREDITIVO GENERALIZADO .....                                | 109 |
| 4.2.1.-HORIZONTES DE PREVISÃO DA SAÍDA E DE CONTROLE ...                     | 109 |
| 4.2.1.1.-PROCESSOS ESTÁVEIS EM MALHA ABERTA<br>E DE FASE MÍNIMA .....        | 111 |
| 4.2.1.2.-PROCESSOS INSTÁVEIS EM MALHA ABERTA<br>E DE FASE MÍNIMA .....       | 125 |
| 4.2.1.3.-PROCESSOS ESTÁVEIS EM MALHA ABERTA<br>E DE FASE NÃO MÍNIMA .....    | 126 |
| 4.2.2.-ESCOLHA DOS FILTROS DO CONTROLADOR .....                              | 137 |
| 4.2.2.1.-OS FILTROS $P(z^{-1})$ E $R(z^{-1})$ .....                          | 138 |
| 4.2.2.2.-ESCOLHA DO FILTRO DAS PERTURBAÇÕES $T(z^{-1})$ .                    | 141 |
| 4.2.3.-ESCOLHA DO FATOR DE PONDERAÇÃO DA EXCITAÇÃO ( $\lambda$ ) .           | 148 |
| 4.2.4.-REFERÊNCIA PRÉ-ESPECIFICADA .....                                     | 153 |
| 4.2.4.1.-PROCESSO DE FASE MÍNIMA .....                                       | 154 |
| 4.2.4.2.-PROCESSO DE FASE NÃO MÍNIMA .....                                   | 156 |
| 4.2.4.3.-INFLUÊNCIA DA REFERÊNCIA PRÉ-ESPECIFICADA<br>SOBRE OS FILTROS ..... | 163 |

|                                                             |     |
|-------------------------------------------------------------|-----|
| 4.3.-CONTROLADOR DE VARIÂNCIA MÍNIMA GENERALIZADA .....     | 165 |
| 4.3.1.-ESCOLHA DOS POLINÔMIOS DO CONTROLADOR .....          | 165 |
| 4.3.2.-ESCOLHA DO FILTRO DAS PERTURBAÇÕES $T(z^{-1})$ ..... | 172 |
| 4.4.-CONTROLADOR PID .....                                  | 174 |
| 4.5.-EXEMPLO DE APLICAÇÃO .....                             | 177 |
| 4.6.-CONCLUSÕES .....                                       | 185 |

## CAPÍTULO 5: CONCLUSÕES ..... 186

|                                                                                      |     |
|--------------------------------------------------------------------------------------|-----|
| 5.1.-SÍNTESE DO TRABALHO .....                                                       | 187 |
| 5.2.-CONCLUSÕES I: OS CONTROLADORES E<br>PROPOSTAS DE TRABALHOS FUTUROS .....        | 188 |
| 5.3.-CONCLUSÕES II: O SISTEMA ESPECIALISTA E<br>PROPOSTAS DE TRABALHOS FUTUROS ..... | 189 |
| 5.3.1.-REPRESENTAÇÃO DO CONHECIMENTO .....                                           | 189 |
| 5.3.2.-MANIPULAÇÃO DO CONHECIMENTO .....                                             | 190 |
| 5.3.3.-AQUISIÇÃO DO CONHECIMENTO .....                                               | 191 |
| 5.4.-CONCLUSÕES III: AS FERRAMENTAS UTILIZADAS .....                                 | 192 |

## APÊNDICE A ..... 193

|                                                                               |     |
|-------------------------------------------------------------------------------|-----|
| A.1.-SOLUÇÃO RECURSIVA DA IDENTIDADE POLINOMIAL<br>(EQUAÇÃO DIOFANTINA) ..... | 193 |
|-------------------------------------------------------------------------------|-----|

## APÊNDICE B ..... 194

|                                                                                                  |     |
|--------------------------------------------------------------------------------------------------|-----|
| B.1.-INCORPORAÇÃO DO FILTRO $T(z^{-1})$ NO CONTROLADOR GMV .....                                 | 194 |
| B.2.-FUNÇÃO DE TRANSFERÊNCIA EM MALHA FECHADA<br>DO CONTROLADOR GMV COM FILTRO $T(z^{-1})$ ..... | 196 |

## REFERÊNCIAS BIBLIOGRÁFICAS ..... 197

## **CAPÍTULO**

### **I**

## **INTRODUÇÃO**

## 11-INTRODUÇÃO

A utilização de dispositivos de controle clássicos (como por exemplo PID) permite o funcionamento, ainda hoje, da grande maioria dos processos industriais. No entanto estes controladores possuem parâmetros constantes, os quais devem ser "sintonizados" para o correto funcionamento do sistema. Se as características do processo mudam, então o controlador deve ser resintonizado. O processo de resintonização é normalmente realizado em forma manual em processos com variações esporádicas, que não afetam o desempenho do sistema. Entretanto em processos com muitas variações esta estratégia não pode ser aplicada. Surge assim a idéia dos controladores *Adaptativos*.

Na maioria dos controladores adaptativos realiza-se uma estimação dos parâmetros do modelo do processo, e a partir destes parâmetros calculam-se os parâmetros do controlador e a lei de controle otimizando um critério de desempenho especificado. As tarefas a serem realizadas em um controle adaptativo são:

- i).-Estimação dos parâmetros do processo;
- ii).-Cálculo dos parâmetros do controlador;
- iii).-Cálculo do sinal de controle.

A primeira das tarefas recebe o nome genérico de *Identificação*, e as duas últimas de *Controle*.

A utilização por muitos anos dos controladores PID permitiu armazenar uma quantidade de conhecimento suficiente para uma boa sintonização dos ganhos deste controlador. Os ajustes são realizados por operadores com pouco conhecimento em controle, mas que com a prática tem conseguido um grande conhecimento dos processos e as suas relações com os parâmetros do PID. Por outro lado, a existência de métodos de ajuste empíricos simples (Regras Ziegler-Nichols [KUC88]), tornam desnecessária a presença de uma pessoa com conhecimento específico em PID. Entretanto com a rápida expansão dos controladores adaptativos e a falta de métodos

empíricos simples para o ajuste dos parâmetros destes controladores, torna-se necessária a existência de um "Tutor" com conhecimento específico do controlador/identificador para o desempenho eficiente do controle. Em muitas empresas a existência deste tutor pode ser inviável, mas a utilização de *Sistemas Especialistas* permite a "existência do conhecimento do tutor" sem custos elevados.

Este trabalho é dedicado à construção de uma ferramenta baseada em conhecimento para simplificar as tarefas de *Configuração* e *Supervisão* dos algoritmos de identificação e controle.

O presente capítulo é dedicado à apresentação dos *Sistemas Especialistas* em forma genérica.

Na seção 1.2 apresenta-se a estrutura de um sistema especialista.

A representação do conhecimento dentro de um sistema "Inteligente" é descrita na seção 1.3.

Finalmente, na seção 1.4 são descritos os sistemas especialistas baseados em regras de produção e na seção 1.5. é apresentado o resumo do capítulo.

No capítulo II são apresentados os algoritmos de identificação e controle implementados. Para a identificação recorre-se ao método da Matriz Estendida, e os algoritmos de controle são: i) o clássico PID, ii) o controlador de variância mínima generalizado (GMV) com alocação de polos, e iii) o controlador preditivo generalizado (GPC).

O sistema especialista implementado é descrito no capítulo III. A aquisição do conhecimento e a geração das bases de conhecimento, são apresentadas no capítulo IV. Finalmente no capítulo V são apresentadas as conclusões obtidas durante o

desenvolvimento do presente trabalho.

Para a apresentação dos algoritmos optou-se por utilizar uma linguagem de pseudocódigo. Este pseudocódigo possui uma estrutura tipo PASCAL que permite uma melhor estruturação dos algoritmos. A seguinte simbologia é utilizada para a descrição das estruturas implementadas: os símbolos que pertencem à linguagem do pseudocódigo são escritos em *ITALICO*; os valores respondidos pelo sistema são escritos em **negrito**; os procedimentos implementados são escritos em **ORATOR**. As variáveis são representadas tendo a primeira letra do nome ou dos nomes em Maiúscula, estando os nomes de uma mesma variável unidos por hífen, a menos que o nome da variável seja muito significativo (por exemplo:  $y(t)$ ).

## 12.-SISTEMAS ESPECIALISTAS, DEFINIÇÃO E CARACTERÍSTICAS

Na área de sistemas de computação, a *inteligência artificial*, e mais especificamente os *sistemas especialistas*, têm provado ser uma ferramenta poderosa em inúmeras aplicações, como por exemplo, em prospecção geológica o PROSPECTOR [DUD78], e em diagnóstico médico o MYCIN [SH076].

A seguir explica-se a estrutura dos Sistemas Especialistas, ressaltando as diferenças com as ferramentas clássicas (baseadas em algoritmos), e as suas limitações.

Os Sistemas Especialistas são programas de computador que utilizam explicitamente o conhecimento de uma pessoa "especialista" na área de aplicação para resolver problemas dessa área. A resolução deve ser transparente para o usuário.

Do parágrafo anterior surge a estrutura (partes constituintes) dos sistemas especialistas, como é detalhado a seguir.

**Os Sistemas Especialistas são programas de computador que utilizam explicitamente o conhecimento de uma pessoa "especialista" na área de aplicação**

Os sistemas especialistas são um subgrupo dos denominados sistemas baseados em conhecimento, cuja característica principal é que o conhecimento sobre o problema a ser resolvido, e a forma para obter esse conhecimento não formam uma só estrutura, (isto é, um *algoritmo*). Estas duas estruturas, estão explicitamente separadas no que se denominam *Base de conhecimento* e *Máquina de inferência*.

Na base de conhecimento armazena-se o conhecimento sobre uma determinada área, sendo que o conhecimento é fornecido por um perito da citada área. Este é um problema crucial dos sistemas especialistas, já que eles podem ser no máximo tão bons quanto a pessoa que fornece o conhecimento sendo, em geral, o seu desempenho prejudicado pela má transferência do conhecimento. Esta tarefa, denominada *aquisição de conhecimento*, é a principal dificuldade no desenvolvimento destes sistemas, já que ela é complexa e de características artesanais, devido a carência de metodologias apropriadas [CAR88].

A máquina de inferência é um dispositivo de propósito geral que permite resolver problemas através de uma interação com a base de conhecimento e o conjunto de fatos do problema em questão. Esta máquina segue uma *linha de raciocínio* até a obtenção da solução final. A forma que a máquina de inferência procura a informação na base de conhecimento depende em parte das restrições próprias da área de aplicação e em parte de decisões de projeto.

**para resolver problemas dessa área**

Os resultados (soluções) parciais armazenados junto com os dados provenientes de outras fontes (dados também pertencentes ao problema atual a se resolver) formam a *Base de Dados*. A estrutura

desta base de dados depende em grande parte da área de aplicação e da eficiência (velocidade) desejada para o sistema. Quando os dados são de tipos similares é possível utilizar estruturas de matrizes ou ainda de vetores. Entretanto a medida que eles diferem em tipo deve-se adotar estruturas mais complexas que permitam maior generalidade.

**A resolução deve ser transparente para o usuário.**

Quando um usuário não perito observa um especialista trabalhando para a resolução de algum problema, ele pode ter várias expectativas. As mais úteis para ele são : ficar convicto que a solução achada é válida, saber se existem outras soluções, e por último aprender para poder imitar o especialista. Todas estas possibilidades são consideradas num sistema especialista, permitindo que ele tenha *Transparência* ao resolver um determinado problema. Para tanto deve-se dispor de uma *Interface Homem-Máquina* que permita responder a questões do tipo *Como* ou *Por que?*, e também concordar ou discordar com as soluções achadas.

A descrição fornecida nos parágrafos anteriores definem um sistema especialista com a estrutura ilustrada na figura 1.1.

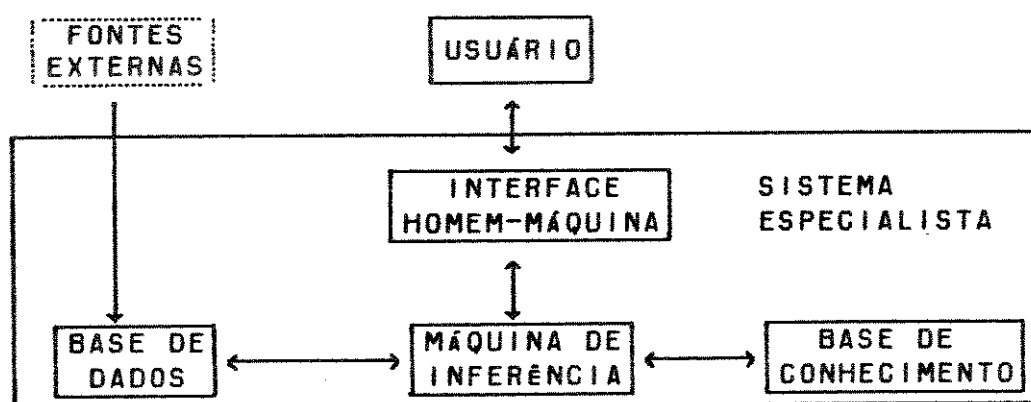


Fig 1.1 Estrutura de um Sistema Especialista

Uma das características atrativas dos sistemas especialistas, é a *flexibilidade*, entendendo-se por flexibilidade a facilidade de correção, alteração e adição. No projeto de software estas tarefas, realizadas na fase de "manutenção aperfeiçoativa", [PRE87, BOR87] consome mais de 50% das atividades de manutenção, sendo que estas consomem no total entre 50% e 60% do esforço gasto para o desenvolvimento do sistema. Estas tarefas em geral introduzem erros no sistema, que podem ser de muito difícil detecção. Em outras ocasiões a modificação necessária do código são tão grandes que seria preferível trocar o algoritmo, isto em sistemas tradicionais. Entretanto nos sistemas baseados em conhecimento, a estrutura da base de conhecimento, geralmente com características modulares, permite que a realização de tarefas que dependem somente do conhecimento do sistema, e não da forma de manipulação deste, sejam feitas de forma simples e com maior controle sobre os efeitos por elas causados.

Uma característica que somente os sistemas especialistas possuem e que também facilita o seu desenvolvimento é a *transparência*. Como principais vantagens pode-se citar [ARR88]:

- i).-A detecção de erros e a sua correção é mais simples que em sistemas tradicionais sendo seu desenvolvimento mais rápido.
- ii).-Os efeitos de mudanças no sistema são fáceis de prever e testar.
- iii).-O usuário tem maior possibilidade de acreditar que a solução é correta, já que ele pode seguir passo a passo a sua obtenção.
- iv).-Permite que o usuário seja treinado sem a ajuda de uma pessoa com conhecimentos específicos na área.

### 13.-REPRESENTAÇÃO DO CONHECIMENTO

A principal particularidade que distingue um Sistema Especialista de sistemas convencionais é como o conhecimento sobre o problema é estruturado. Nas ferramentas convencionais este conhecimento está incluído no algoritmo, de tal forma que o seu funcionamento é praticamente rígido, isto é, caminhos alternativos à solução ocorrem somente quando existe uma parte do algoritmo dedicado a este fim. Esta condição não é inconveniente para determinadas aplicações, mas não é adequada em situações onde as condições externas alteram-se, e é necessário simular as respostas humanas aos estímulos provenientes do exterior. Nestas circunstâncias precisa-se de sistemas que reajam às modificações externas.

Desta forma, nos Sistemas Especialistas os dados dirigem as operações, enquanto que nos sistemas convencionais, é o programa (como algoritmo) que dirige os dados [HER88]. Assim, nos Sistemas Especialistas, o conhecimento sobre o domínio de aplicação é separado do processo que atua sobre ele (Máquina de Inferência), requerendo uma forma de representação adequada para converter problemas complexos em problemas simples [HER88].

Na literatura de Inteligência Artificial têm sido proposto diferentes formalismos para a representação do conhecimento, sendo os mais comumente citados [HER88,SOU89,SIG90]:

- i).-*Linguagens funcionais*, baseadas em cálculo funcional (e.g., *Lambda Cálculo* quando se utiliza LISP como linguagem de programação).
- ii).-*Linguagens de predicados*, baseados em cálculo de predicados de primeira ordem (e.g., PROLOG com seus recursos básicos).

iii).-*Redes semânticas*. Sua estrutura básica é constituída de *grafos*, isto é um conjunto de nós e ligações entre estes [HER88].

iv).-*Sistemas baseados em Esquemas (Frames)*. Estrutura também fundamentada em grafos, e organizada hierarquicamente [HERR88,SIG90].

v).-*Sistemas baseados em Regras de Produção*, cuja estrutura é de um conjunto de sentenças implicacionais (Regras) da forma "SE\_ENTÃO\_".

Destes formalismos, os *sistemas baseados em Esquemas*, e os *sistemas baseados em Regras de Produção*, têm sido os mais utilizados na construção de Sistemas Especialistas [REDB4, SAN86, GAR88], e serão explicados com mais detalhes nas seções seguintes.

#### 1.3.1.-SISTEMAS BASEADOS EM ESQUEMAS

No campo da Inteligência Artificial, o termo *Esquema* (Frame), refere-se a uma forma especial de representar conceitos e situações frequentes da vida diária.

Minsky [MYN75] sugeriu que todo processo intelectual é guiado por estruturas de expectativas de conhecimento. Ele propôs que quando uma pessoa encontra-se diante de uma nova situação (ou muda o ponto de vista do problema considerado), é selecionado na memória uma estrutura (Esquema) que representa com maior exatidão a nova situação.

Um Esquema é uma estrutura que deve ser adaptada à realidade. Assim os Esquemas podem ser identificados como estruturas que representam situações estereotipadas [HER88].

A estrutura dos esquemas é composta de campos (*slots*), que são as unidades elementares de armazenamento de conhecimento. Cada campo pode ser preenchido com: valores, procedimentos, outros

campos, e até mesmo outros esquemas ou regras de produção. As estruturas baseadas em esquemas formam uma rede, onde os nós representam conceitos (descritos pelos campos) e os arcos indicam o relacionamento entre estes conceitos. A rede é organizada em forma hierárquica, onde os nós dos níveis inferiores herdam as propriedades (tais como valores, procedimentos, etc) dos níveis superiores.

Pelas características próprias dos Esquemas, eles são amplamente utilizados em aplicações onde os dados (forma e conteúdo) são de vital importância para a resolução do problema como por exemplo, reconhecimento de imagem ou processamento da voz.

As vantagens da utilização de Esquemas são [FAV88]:

- i).-O conhecimento específico à solução do problema é agrupado de forma que o acesso e a manipulação desse conhecimento torna-se mais eficaz;
- ii).-A hierarquização permite a estruturação do conhecimento como é comumente encontrado entre especialistas, isto é parte-se do geral para o específico;
- iii).-Devido a possibilidade de agrupamento do conhecimento, a estruturação da base de conhecimento surge como uma consequência natural na utilização dos Esquemas como método de representação do conhecimento.

### 1.3.2.- SISTEMAS BASEADOS EM REGRAS DE PRODUÇÃO

"Baseados em regras" são os sistemas em que o conhecimento do especialista é armazenado numa estrutura de regras denominadas de *produção*. Estas regras são sentenças implicacionais do tipo "SE-ENTÃO-".

A representação do conhecimento através de regras de produção é um modo natural para se descrever processos onde o meio muda rapidamente e/ou de forma aleatória [WAT86, HER88]. Em um sistema baseado em regras de produção, estas são comparadas com os fatos ou conhecimento sobre a situação atual existente na base de dados, para determinar sua validade e em função da sua validade, atuar sobre o meio.

Este paradigma facilita a aquisição e representação do conhecimento de um especialista, já que é comum a utilização de regras do tipo "SE\_ENTÃO\_" para a resolução de problemas. Esta representação é adequada para formalizar ordens e estratégias, especialmente quando o conhecimento do domínio vem de associações empíricas obtidas como resultado de anos de experiência resolvendo problemas nesse domínio.

Uma vantagem fornecida pela representação por regras de produção é que a *linha de raciocínio* até a solução pode ser analisada, isto é, o sistema explica como obteve a solução. Esta característica denomina-se *transparência*.

Ao contrário da representação por esquemas, nos sistemas baseados em regras de produção, existe uma clara separação entre o conhecimento do domínio e o conhecimento sobre o problema atual (dados). Assim para construir um sistema baseado em conhecimento por regras de produção deve-se projetar a *Base de conhecimento* (do domínio) e a *Base de Dados*.

#### 14.-SISTEMAS ESPECIALISTAS BASEADOS EM REGRAS DE PRODUÇÃO

Devido às características próprias das tarefas a serem realizadas em controle e identificação de processos, adota-se a representação do conhecimentos por regras de produção como paradigma para nosso sistema. Esta representação favorece a aquisição do conhecimento, já que as tarefas de identificação e controle podem ser indicadas como um conjunto de regras que

determinam as ações a serem executadas. Esta representação também permite simplificar a máquina de inferência e a interface homem-máquina, facilitando a implementação do Sistema Especialista.

#### 1.4.1.-BASE DE DADOS

O domínio do problema a ser tratado pelo sistema influi sobre a representação do conhecimento adotada e na estrutura de dados. Estes dados devem representar o conhecimento sobre o problema particular tratado até o instante presente, isto é, a base de dados deve conter todos os fatos provenientes da interface e os dados inferidos na base de conhecimento na busca de uma solução.

Uma forma completa de descrição de um fato é através da tripla <atributo-objeto-valor>, onde se relaciona ao atributo de um objeto determinado o valor através de um predicado. A seguir mostra-se um fato descrito pela tripla anteriormente citada:

| atributo | objeto      | predicado | valor    |
|----------|-------------|-----------|----------|
| ↓        | ↓           | ↓         | ↓        |
| O atraso | do processo | não é     | variável |

Esta forma de descrição de fatos não é a única possível. Uma outra forma normalmente utilizada é a denominada de par <atributo-valor>, onde o objeto passa a ser parte do atributo, e o fato toma a seguinte forma:

<atributo> predicado <valor>

Com esta representação o exemplo dado anteriormente pode ser armazenado na base de conhecimento como:

| atributo             | predicado | valor    |
|----------------------|-----------|----------|
| ↓                    | ↓         | ↓        |
| O atraso do processo | não é     | variável |

#### 1.4.2.-BASE DE CONHECIMENTO

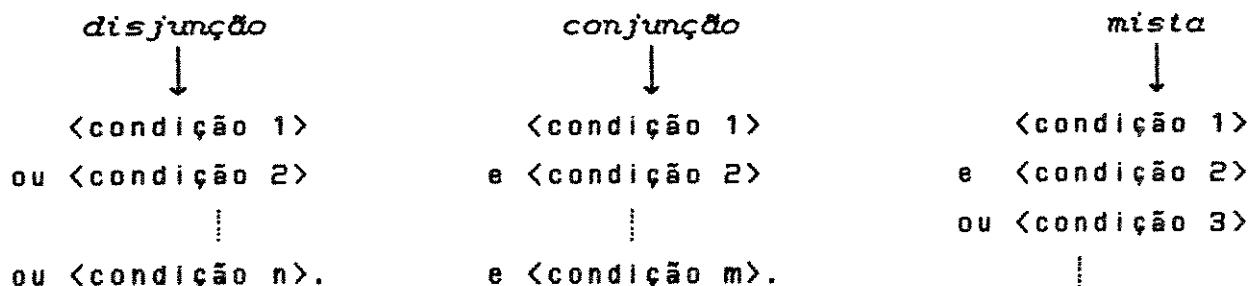
Uma base de conhecimento descrita como regras de produção é uma coleção de asserções em forma implicacional do tipo

Se <Condição> Então <Conclusão>.

ou do tipo

Se <Condição> Então <Ação>.

Seu significado é : se <Condição> é válida então também é válida a <Conclusão> ou se executa a <Ação>. A estrutura de <Condição> pode ser de uma conjunção, disjunção, ou mista , isto é, conjunção de disjunções como se ilustra a seguir.



Na base de conhecimento as regras possuem uma ordem de hierarquia com relação ao conhecimento representado na base, o que obriga a máquina de inferência a buscar inicialmente as regras de maior hierarquia, as que possuem conhecimento mais genérico. A seguir são aplicadas as regras de hierarquia inferior, as que possuem conhecimento mais específico sobre o problema que está procurando-se resolver.

A utilização de uma hierarquia na base de conhecimento permite a resolução de conflitos entre as regras aplicáveis a uma base de dados. Esta hierarquia pode ser dada explicitamente como um índice [RED84], por mecanismos que considerem a inovação introduzida pelos dados, ou por outros tipos de mecanismos. Um método comumente utilizado é o de priorização pela posição da

regra dentro da base de conhecimento (por exemplo, as primeiras regras em um nível hierárquico mais alto).

#### 1.4.3.-MÁQUINA DE INFERÊNCIA

Realizar uma inferência em um sistema baseado em regras de produção é aplicar uma regra de produção a uma base de dados para obter um novo estado desta [NIL82]. Esta operação é repetida até que a base de dados obtida corresponda ou contenha o objetivo desejado.

Para a realização de inferências necessita-se de um mecanismo de controle conhecido como *estratégia de controle*, que deve possuir as seguintes características [HER88]:

- i).-*Provocar alteração*: A aplicação das regras deve provocar mudanças na base de dados: caso contrário o problema não seria resolvido.
- ii).-*Ser sistemática*, isto é, para um estado determinado do sistema, gerar somente uma vez a Base de Dados.
- iii).-*Ser eficiente*, isto é, evitar o problema da explosão combinatória.

A forma que a máquina de inferência seleciona as regras a serem aplicadas à base de dados, isto é, as *estratégias de controle*, são classificadas como: *encadeamento Direto*, *encadeamento reverso*, e *encadeamento misto*, que são descritas a seguir.

##### 1.4.3.1.- ESTRATÉGIA DE CONTROLE POR ENCADEAMENTO DIRETO

Nesta estratégia de controle a máquina de inferência examina os antecedentes de todas as regras da base de conhecimento, e seleciona uma (se houver) destas regras que tenha os seus antecedentes (parte <Condição>) satisfeitos pelos fatos da base de

dados. Se a conclusão desta regra coincide com a solução do problema, termina-se a inferência. Se a conclusão não é a solução procurada, esta regra é então "disparada", isto é, a parte <Conclusão> ou <Ação>, é incluída ou executada sobre os dados, o que modifica a base de dados. Quando existe mais de uma regra com os seus antecedentes satisfeitos (o conjunto formado por estas regras é denominado *conjunto de discórdia* [NILB2]), necessita-se de um mecanismo para a resolução de conflitos: os métodos mais comumente utilizados são o de priorização ou a hierarquização das regras.

Uma vez disparada uma regra do conjunto de discórdia, podem ser aplicadas duas estratégias para a seleção da próxima regra a ser aplicada. Uma estratégia é escolher a regra seguinte do conjunto de discórdia (em prioridade) e acioná-la, e assim sucessivamente até esgotar esse conjunto. A outra estratégia é analisar novamente a base de dados, modificada pela aplicação da primeira regra, para construir um novo conjunto de discórdia, e assim continuar a inferência até a solução.

Independentemente da estratégia aplicada para a escolha da regra seguinte, o processo de inferência continua até que seja obtida a solução do problema (a conclusão de uma regra é o fato procurado (Meta)); ou até que não se tenha mais regras a serem aplicadas à base de dados, isto é, não existam mais regras com os seus antecedentes satisfeitos. Pode ocorrer que algumas das subcondições das regras estejam indefinidas (não se possui suficiente informação do problema tratado), nesta situação o sistema solicita ao usuário a informação necessária para avaliar as regras que ainda não foram acionadas para a resolução do problema.

Esta estratégia de controle também é chamada de *controle dirigido por antecedente* ou *controle "bottom-up"*. Com o auxílio do pseudocódigo definido na seção 1.1 observa-se na listagem PSC.1.1 uma implementação possível desta estratégia.

## PROCEDIMENTO ENCADEAMENTO DIRETO;

### COMEÇO

OBTER a Meta-objetivo;

Conjunto-de-Discórdia := vazio;

### REPETIR

Regra-Atual := 1ª-Regra-da-BG;

### REPETIR

SE (Antecedentes-Regra-Atual satisfeitos OU  
Antecedentes-Regra-Atual satisfeitos pelo usuário) E

Conclusão-Regra-Atual não está na Base-de-dados  
ENTÃO

Conjunto-de-Discórdia :=

Conjunto-de-Discórdia + Regra-Atual;

Regra-Atual := Proxima-Regra-da-BG;

ATÉ Regra-Atual = vazia

SE o Conjunto-de-Discórdia não está vazio

ENTÃO

### REPETIR

SELECIONAR Uma-Regra do

Conjunto-de-Discórdia;

Conjunto-de-Discórdia :=

Conjunto-de-Discórdia - Uma-Regra;

SE a Conclusão-de-Uma-Regra é a  
Meta-Objetivo

ENTÃO

INDICAR Solução-Achada;

SENÃO

APLICAR Uma-Regra na  
Base-de-dados;

ATÉ Solução-Achada OU

Conjunto-de-Discórdia = vazio

SENÃO

INDICAR Problema-Sem-Solução;

ATÉ Solução-Achada OU Problema-Sem-Solução

FIM;

PSC. 1.1 - Máquina de Inferência Com Encadeamento Direto

### 1.4.3.2.-ESTRATÉGIA DE CONTROLE POR ENCADEAMENTO REVERSO

Nesta estratégia de controle, a partir de uma ação a ser executada (ou uma conclusão possível), a máquina de inferência procura, por meio de um movimento "reverso" sobre a Base de Conhecimento, obter uma base de dados que corresponda ao objetivo associado à solução do problema.

A forma como a máquina de inferência obtém a solução de um problema é descrito a seguir: A máquina de inferência seleciona uma ou várias regras da base de conhecimento cujo conseqüente (<Conclusão> ou <Ação>) coincide com o objetivo procurado, formando assim o conjunto de discordia. Deste conjunto escolhe-se (por algum dos mecanismos já indicados na estratégia de controle com encadeamento para frente) uma das regras. A seguir analisam-se os seus antecedentes para verificar a sua validade (na base de dados). Se eles forem válidos ou validados pelo usuário, a regra é executada e é achada a solução do problema. Se os antecedentes não são satisfeitos por que uma ou mais subcondições de uma regra com estrutura conjuntiva, ou todas as disjunções de uma regra com estrutura mista, são falsas, então a máquina de inferência seleciona uma outra regra (pelo mecanismo anteriormente utilizado) do conjunto de discórdia, e verifica a validade dos seus antecedentes. Se algumas das subcondições são indefinidas, estas tornam-se sub-objetivos a serem provados pela máquina de inferência. Este processo é repetido até que não seja possível obter regras que satisfaçam o objetivo e/ou seus sub-objetivos, ou até que todos os dados necessários para satisfazer o objetivo sejam definidos. Quando um sub-objetivo permanece indefinido por que não se dispõe de um valor associado na base de dados, o usuário é requerido para fornecê-lo.

A estratégia de controle por encadeamento reverso também é conhecida como *controle dirigido por conseqüente*, ou *controle "Top-Down"*. Na listagem PSC.1.2 apresenta-se um possível algoritmo de controle baseado na estratégia descrita.

## PROCEDIMENTO ENCADEAMENTO REVERSO;

**FUNÇÃO PROCURA-SOLUÇÃO**(Meta): {verdadeiro, falso};  
**COMEÇO**

Conjunto-de-Discórdia := vazio;

Regra-Atual := 1ª-Regra-da-BC;

**REPETIR**

SE Conclusão-Regra-Atual é a Meta

ENTÃO

Conjunto-de-Discórdia :=

Conjunto-de-Discórdia + Regra-Atual;

Regra-Atual := Proxima-Regra-da-BC;

**ATÉ** Regra-Atual = vazia

SE o Conjunto-de-Discórdia não está vazio

ENTÃO

**REPETIR**

**SELECIONAR** Uma-Regra do

Conjunto-de-Discórdia;

Conjunto-de-Discórdia :=

Conjunto-de-Discórdia - Uma-Regra;

SE Antecedentes-Regra-Atual satisfeitos **OU**

Antecedentes-Regra-Atual satisfeitos pelo  
usuário

ENTÃO

Solução = verdadeiro;

**SENÃO**

**COMEÇO**

Seguinte-Meta :=

1ª-Antecedente-Uma-Regra;

**REPETIR**

Solução :=

**PROCURA-SOLUÇÃO**(Seguinte-Meta);

Seguinte-meta :=

Seguinte-Antecedente-Uma-Regra;

**ATÉ** Seguinte-Meta = vazia **OU**

Solução = falso

**FIM;**

**ATÉ** Solução = verdadeiro **OU**

Conjunto-de-Discórdia = vazio

**SENÃO**

Solução := falso;

**PROCURA-SOLUÇÃO** := Solução ;

**FIM;**

**COMEÇO** (Encadeamento-Reverso)

**OBTER** Meta-Objetivo;

Solução := **PROCURA-SOLUÇÃO**(Meta-Objetivo);

SE Solução = verdadeiro

ENTÃO

**INDICAR** Solução-Achada;

**SENÃO**

**INDICAR** Problema-Sem-solução;

**FIM;**

PSC. 1.2 - Máquina de Inferência Com Encadeamento Reverso

#### 1.4.3.3.-ESTRATÉGIA DE CONTROLE POR ENCADEAMENTO MISTO

Esta estratégia utiliza uma combinação das duas estratégias descritas anteriormente. Neste caso a máquina de inferência combina as conclusões obtidas por encadeamento direto com os sub-objetivos obtidos por encadeamento reverso. Este mecanismo de controle procura resolver o problema da explosão combinatorial, diminuindo o espaço de busca dos dois encadeamentos.

#### 1.4.4.- INTERFACE HOMEM-MÁQUINA

As partes constituintes de um sistema especialista, até agora descritas, são comuns na maioria dos sistemas baseados em conhecimento. A interface homem-máquina permite ao usuário interagir com o sistema como se ele fosse um especialista humano.

Os métodos pelo quais o sistema comunica-se com o usuário são basicamente quatro:

- i).-Solicitando dados para o funcionamento do sistema;
- ii).-Apresentando os resultados obtidos;
- iii).-Permitindo concordar ou não com a solução obtida. Nesta circunstância, o usuário pode obrigar o sistema a procurar uma outra solução.
- iv).-Explicando como foi obtida a solução, isto é, a linha de raciocínio utilizada. Esta capacidade, denominada *transparência*, permite ao usuário conhecer ou verificar a validade da solução apresentada, e aprender o método utilizado pelo Sistema Especialista.

## 15.-RESUMO

Neste capítulo abordou-se a utilização de Sistemas Especialistas na área de controle adaptativo e identificação. Também foram apresentadas as características dos sistemas especialistas, ressaltando-se a escolha da representação de conhecimento utilizado pelo sistema a ser descrito no capítulo III.

## **CAPÍTULO**

### **II**

## **CONTROLE ADAPTATIVO**

## 2.1-INTRODUÇÃO

Durante muitos anos, permaneceu na mente dos projetistas de sistemas de controle a idéia de sistemas de controle mais sofisticados, os que pudessem funcionar corretamente sobre uma ampla faixa de condições de operação.

Com o advento dos computadores digitais nos anos 50, estes investigadores acharam-se frente a um dispositivo poderoso para ser utilizado como controlador, como até então nunca tinha existido. Mas os primeiros esforços foram infrutíferos pelo que foi perfeitamente definido por Astrm [AST83]

"Muito entusiasmo, hardware ruim, e teoria inexistente"

Nos vinte anos seguintes conseguiu-se reverter esta situação (sem se perder o entusiasmo), com o desenvolvimento da *Teoria de Controle Moderno*, e os avanços na eletrônica. Os anos 70 estavam prontos para receberem toda uma avalanche de novos controladores.

Muitos destes algoritmos de controle desenvolvidos desde então, estão baseados em:

i) Uma representação computacional (interna) do modelo do processo. Na seção 2.2. são discutidos os modelos utilizados neste trabalho.

ii) A estimação dos parâmetros desse modelo, considerada na seção 2.3.

iii) A detecção de não estacionariedade e condições de excitação do processo, desenvolvidos nas seções 2.4 e 2.5.

iv) O cálculo do sinal de controle a ser aplicado ao processo para se obter a evolução desejada, tema tratado nas seções 2.7 e 2.8.

Na seção 2.6, apresenta-se o mais clássico dos algoritmos de controle, o PID, que é incluído para fins de comparação com os outros controladores desenvolvidos neste trabalho. Na seção 2.9 são apresentados os algoritmos descritos nas seções anteriores na linguagem de pseudocódigo descrita no capítulo 1. Finalmente na seção 2.10 é apresentado um resumo do capítulo.

## 2.2-MODELO DO PROCESSO

O modelo do processo utilizado para a obtenção dos algoritmos de controle adaptativo é descrito por uma equação a diferenças conforme descrito a seguir:

$$A(z^{-1}) y(t) = B(z^{-1}) u(t-d) + x(t) \quad (2.2.1)$$

onde  $A(z^{-1})$  e  $B(z^{-1})$  são polinômios no operador atraso  $z^{-1}$  de ordens  $n_a$  e  $n_b$  respectivamente, com

$$\begin{aligned} A(z^{-1}) &= 1 + a_1 z^{-1} + \dots + a_{n_a} z^{-n_a} \\ B(z^{-1}) &= b_0 + b_1 z^{-1} + \dots + b_{n_b} z^{-n_b} \end{aligned} \quad (2.2.2)$$

sendo  $u(\cdot)$  a entrada,  $y(\cdot)$  a saída e  $x(\cdot)$  a perturbação que atua no processo. O parâmetro "d" denomina-se atraso do processo, e inclui o atraso de transporte e o atraso incorporado devido ao segurador de ordem zero utilizado na discretização do modelo contínuo. A modelagem normalmente utilizada da perturbação é [AST73] :

$$x(t) = C(z^{-1}) \xi(t) \quad (2.2.3)$$

onde  $C(z^{-1})$ <sup>1</sup> é um polinômio de ordem  $n_c$  da forma

---

<sup>1</sup> Assume-se que os zeros do polinômio  $C$  estão dentro do círculo unitário do plano complexo  $Z$ .

$$C(z^{-1}) = 1 + c_1 z^{-1} + \dots + c_{nc} z^{-nc} \quad (2.2.4)$$

e  $\xi(\cdot)$  é uma sequência aleatória não-correlata com variância  $\sigma^2$  e média nula. Com esta representação da perturbação a equação (2.2.1) denomina-se o modelo "CARMA"<sup>2</sup> do processo

$$A(z^{-1}) y(t) = B(z^{-1}) u(t-d) + C(z^{-1}) \xi(t) \quad (2.2.5)$$

Na literatura podem também ser encontradas as seguintes representações para o modelo do processo

$$A(z^{-1}) y(t) = z^{-d} B(z^{-1}) u(t) + C(z^{-1}) \xi(t) \quad (2.2.6)$$

e

$$A(z^{-1}) y(t) = B(z^{-1}) u(t-1) + C(z^{-1}) \xi(t) \quad (2.2.7)$$

Nesta última equação a ordem do polinômio  $B(z^{-1})$  é incrementada em  $d-1$  para incluir os  $d-1$  elementos (nulos) correspondentes ao atraso de transporte do processo [WAL87].

Embora o modelo CARMA seja amplamente utilizado para o desenvolvimento dos algoritmos de controle auto-ajustáveis em aplicações industriais, muitas das perturbações atuantes neste tipo de ambiente não são corretamente modeladas desta forma. Em certos processos a perturbação não tem média nula, como por exemplo nas estruturas flexíveis [LAM87]. Neste caso pode-se utilizar o seguinte modelo, denominado CARMA estendido

$$A(z^{-1}) y(t) = B(z^{-1}) u(t-d) + C(z^{-1}) \xi(t) + dc \quad (2.2.8)$$

onde todos os parâmetros são definidos como no modelo "CARMA" dado pela equação (2.2.5), e "dc" é o parâmetro que modela o valor não nulo da média da perturbação. Nos algoritmos adaptativos este parâmetro é estimado "on-line" e, geralmente, recorre-se na estimação deste modelo ao algoritmo dos Mínimos Quadrados com "1-no-vetor-de-dados" [CLKB3]. Esta medida prática, embora

---

<sup>2</sup> CARMA : Do inglês Controlled Auto-Regressive Moving Average.

realizável, necessita de uma modificação no algoritmo de identificação que permita *fatores de esquecimento* diferentes (seção 2.3) para a estimação do parâmetro constante e para a estimação dos outros parâmetros. Além disto, é mais sensível aos problemas de identificação por falta de excitação na entrada constante [CLK84].

O modelo dado pela equação (2.2.8) também não representa adequadamente uma perturbação encontrada frequentemente nos ambientes industriais denominado *movimento browniano* [PET84, CLK84], nem representa perturbações na saída do tipo degrau de amplitude aleatória acontecendo em instantes aleatórios, produzidos por exemplo por mudanças na qualidade do material [TUF85]. Um modelo para este tipo de perturbações é dado por

$$x(t) = x(t-1) + G(z^{-1}) \xi_1(t) \quad (2.2.9)$$

Definindo  $\Delta$  como o operador *diferença*

$$\Delta = (1 - z^{-1})$$

$$x(t) = \frac{G(z^{-1}) \xi_1(t)}{\Delta} \quad (2.2.10)$$

Utilizando a representação da perturbação dada pela equação (2.2.10) o modelo do processo pode ser descrito como segue (modelo "CARIMA"<sup>3</sup>):

$$A(z^{-1}) y(t) = B(z^{-1}) u(t-d) + \frac{C(z^{-1}) \xi(t)}{\Delta} \quad (2.2.11)$$

Neste modelo é necessário estimar o mesmo número de parâmetros do modelo CARMA.

Este modelo também permite derivar algoritmos de controle com

---

<sup>3</sup> CARIMA: Do inglês Controlled Auto-Regressive Integrated Moving Average.

ação integral (necessária para a eliminação do erro em regime entre a saída e a referência para uma variação degrau), e evita a utilização de um integrador na malha de controle.

Neste texto também pode-se encontrar a equação (2.2.11) nas seguintes formas:

$$\alpha(z^{-1}) y(t) = B(z^{-1}) \Delta u(t-d) + C(z^{-1}) \xi(t) \quad (2.2.12)$$

$$\alpha(z^{-1}) y(t) = z^{-d} B(z^{-1}) \Delta u(t) + C(z^{-1}) \xi(t) \quad (2.2.13)$$

ou

$$\alpha(z^{-1}) y(t) = B(z^{-1}) \Delta u(t-1) + C(z^{-1}) \xi(t)^4 \quad (2.2.14)$$

com

$$\Delta u(t) = u(t) - u(t-1) \quad (2.2.15)$$

$$\alpha(z^{-1}) = \Delta A(z^{-1}) = 1 + \alpha_1 z^{-1} + \dots + \alpha_{n\alpha} z^{-n\alpha} \quad (2.2.16)$$

e  $n\alpha = n_a + 1$

## 2.3.- IDENTIFICAÇÃO DO PROCESSO

### 2.3.1.- INTRODUÇÃO

Na seção anterior adotou-se um modelo (discreto) do processo, que fornece uma representação adequada para a sua manipulação matemática. A seguir são discutidos diferentes métodos para a determinação dos coeficientes dos modelos apresentados na seção 2.2.

---

<sup>4</sup> São observadas as mesmas considerações para o polinômio  $B$  que no caso do modelo CARMA.

Um dos métodos mais conhecidos e utilizados para a modelagem de processos é o dos Mínimos Quadrados, que será explicado a seguir.

### 2.3.2.- ESTIMADOR DOS MÍNIMOS QUADRADOS PONDERADOS RECURSIVO

Para a obtenção deste algoritmo assume-se que  $C(z^{-1}) = 1$ , e que  $\xi(\cdot)$  é uma sequência aleatória  $N(0, \sigma^2)$ . Sob estas hipóteses, pode-se reescrever a equação (2.2.14) como<sup>5</sup>:

$$A \Delta y(t) = B \Delta u(t-1) + \xi(t) \quad (2.3.1)$$

ou

$$\begin{aligned} \Delta y(t) = & -a_1 \Delta y(t-1) \dots - a_n \Delta y(t-n) + \\ & + b_0 \Delta u(t-1) \dots + b_n \Delta u(t-n-1) + \xi(t) \end{aligned} \quad (2.3.2)$$

O modelo estimado do processo terá a seguinte forma

$$\begin{aligned} \Delta y(t) = & -\hat{a}_1 \Delta y(t-1) \dots - \hat{a}_n \Delta y(t-n) + \\ & + \hat{b}_0 \Delta u(t-1) \dots + \hat{b}_n \Delta u(t-n-1) + e(t) \end{aligned} \quad (2.3.3)$$

onde o acento circunflexo (^) indica parâmetros estimados, e a variável  $e(\cdot)$  inclui a perturbação não mensurável e os erros de estimação. Definindo os vetores  $\varphi(\cdot)$  e  $\hat{\theta}(\cdot)$  como segue

$$\varphi_{(t)}^T = \left[ -\Delta y(t-1) \dots -\Delta y(t-n) \Delta u(t-1) \dots \Delta u(t-n-1) \right] \quad (2.3.4)$$

$$\hat{\theta}_{(t)}^T = \left[ \hat{a}_1(t) \dots \hat{a}_n(t) \quad \hat{b}_0(t) \dots \hat{b}_n(t) \right] \quad (2.3.5)$$

o erro de previsão é dado por

$$e(t) = \Delta y(t) - \varphi_{(t)}^T \hat{\theta}_{(t-1)} \quad (2.3.6)$$

---

<sup>5</sup> Por simplicidade e clareza o fator  $(z^{-1})$  é eliminado dos polinômios quando não apresentem ambiguidades.

Quando os parâmetros estimados coincidem com os valores reais, o erro de previsão é uma sequência de variáveis aleatórias não-correlatas (sequência branca).

O estimador do Mínimos Quadrados é obtido pela minimização do seguinte critério quadrático

$$J_N(\theta) = 1/N \sum_{t=1}^N W(t) \left[ \Delta y(t) - \varphi_{(t)}^T \hat{\theta}_{(t-1)} \right]^2 \quad (2.3.7)$$

em relação ao vetor de  $\hat{\theta}_{(t)}$ .  $W(t)$  é uma ponderação variável, escolhida de tal forma que

$$W(t) = \mu(t) \prod_{i=t}^N \lambda_e(i) \geq 0 \quad (2.3.8)$$

onde  $\mu(t)$  é o coeficiente que pondera a qualidade das medidas realizadas, e  $\lambda_e(t)$  é o chamado *fator de esquecimento* que permite rastrear variações nos parâmetros do processo.

Da minimização da equação (2.3.7) obtém-se

$$\hat{\theta}_{(t)} = \left[ \sum_{i=1}^N W(i) \varphi_{(i)} \varphi_{(i)}^T \right]^{-1} \cdot \left[ \sum_{i=1}^N W(i) \varphi_{(i)} \Delta y(i) \right] \quad (2.3.9)$$

ou na sua forma recursiva [G0077, SIN83]

$$\hat{\theta}_{(t)} = \hat{\theta}_{(t-1)} + K_{(t)} \left[ \Delta y(t) - \varphi_{(t)}^T \hat{\theta}_{(t-1)} \right] \quad (2.3.10)$$

$$K_{(t)} = P_{(t)} \varphi_{(t)} = \frac{P_{(t-1)} \varphi_{(t)}}{1/\mu(t) + \varphi_{(t)}^T P_{(t-1)} \varphi_{(t)}} \quad (2.3.11)$$

$$P_{(t)} = \left[ P_{(t-1)} - \frac{P_{(t-1)} \varphi_{(t)} \varphi_{(t)}^T P_{(t-1)}}{1/\mu(t) + \varphi_{(t)}^T P_{(t-1)} \varphi_{(t)}} \right] \frac{1}{\lambda_e(t)} \quad (2.3.12)$$

sendo  $K(t)$  o ganho do estimador (ganho de Kalman) e  $P(t)$  a matriz

de covariância do erro de estimação

$$P_{(t)} \in \mathbb{R}^{(na+nb+1) \times (na+nb+1)}$$

Como a perturbação atuante no processo é considerada "branca", pode-se demonstrar [G0077] que o vetor  $\hat{\theta}(t)$  converge em probabilidade para o vetor de parâmetros corretos  $\theta(t)$ . Quando a perturbação não satisfaz esta condição, os parâmetros identificados não coincidem com os valores reais. Nestas condições diz-se que o estimador está *polarizado*. Diversos métodos têm sido propostos para estimação de parâmetros não polarizados quando a perturbação não é branca [G0077, SIN83]. Dos métodos discutidos na literatura adotou-se neste trabalho o método da Matriz Estendida.

### 2.3.3.- ESTIMADOR DE MATRIZ ESTENDIDA RECURSIVO

Neste algoritmo modificam-se os vetores  $\varphi(t)$  e  $\hat{\theta}(t)$  da seguinte forma:

$$\varphi_{(t)}^T = \begin{bmatrix} -\Delta y(t-1) \dots -\Delta y(t-na) & \Delta u(t-1) \dots \Delta u(t-nb-1) & \xi(t-1) \dots \xi(t-nc) \end{bmatrix} \quad (2.3.13)$$

$$\hat{\theta}_{(t)}^T = \begin{bmatrix} \hat{a}_1(t) \dots \hat{a}_{na}(t) & \hat{b}_0(t) \dots \hat{b}_{nb}(t) & \hat{c}_1(t) \dots \hat{c}_{nc}(t) \end{bmatrix} \quad (2.3.14)$$

Como a sequência  $\{\xi(t)\}$  não é conhecida, substitui-se  $\varphi(t)$  pela sua estimação  $\hat{\varphi}(t)$ , isto é:

$$\hat{\varphi}_{(t)}^T = \begin{bmatrix} -\Delta y(t-1) \dots -\Delta y(t-na) & \Delta u(t-1) \dots \Delta u(t-nb-1) & \hat{e}(t-1) \dots \hat{e}(t-nc) \end{bmatrix} \quad (2.3.15)$$

onde  $\{\hat{e}(t)\}$  é a sequência de resíduos dada por:

$$\hat{e}(t) = \Delta y(t) - \hat{\varphi}_{(t)}^T \hat{\theta}_{(t-1)} \quad (2.3.16)$$

e  $\hat{e}(t)$  é o erro de previsão a priori [SIN83]. Desta forma obtém-se

o seguinte algoritmo recursivo para o estimador da matriz estendida

$$\hat{\theta}_{(t)} = \hat{\theta}_{(t-1)} + K_{(t)} \hat{e}(t) \quad (2.3.17)$$

$$K_{(t)} = P_{(t)} \hat{\phi}_{(t)} = \frac{P_{(t-1)} \hat{\phi}_{(t)}}{1/\mu(t) + \hat{\phi}_{(t)}^T P_{(t-1)} \hat{\phi}_{(t)}} \quad (2.3.18)$$

$$P_{(t)} = \left[ P_{(t-1)} - \frac{P_{(t-1)} \hat{\phi}_{(t)} \hat{\phi}_{(t)}^T P_{(t-1)}}{1/\mu(t) + \hat{\phi}_{(t)}^T P_{(t-1)} \hat{\phi}_{(t)}} \right] \frac{1}{\lambda_0(t)} \quad (2.3.19)$$

### 2.3.3.1.-RÚIDO COLORIDO E PERTURBAÇÕES NA CARGA

Na prática os elementos do polinômio  $C(z^{-1})$  convergem lentamente. Isto pode provocar problemas no estimador devido a estimação de raízes do polinômio  $C(z^{-1})$  fora do círculo unitário no plano  $z$  [AST70]. Para evitar o problema da estimação do polinômio  $C(z^{-1})$ , neste trabalho utiliza-se um filtro  $T(z^{-1})$  que representa o conhecimento a priori do usuário sobre as perturbações do processo. O filtro  $T(z^{-1})$  é utilizado para modelar as perturbações na banda passante do processo, sendo então a sua estrutura a de um filtro passa baixo ou passa faixa.

Para a inclusão deste filtro na estimação dos parâmetros simplesmente filtram-se os sinais  $\Delta y(t)$  e  $\Delta u(t)$  com  $1/T$ , obtendo-se assim o seguinte modelo para o estimador (veja também a figura 2.1).

$$A \Delta y'(t) = B \Delta u'(t-1) + \xi(t) \quad (2.3.20)$$

onde o acento agudo (') indica quantidade filtrada por  $1/T$ , isto é  $x'(t) = x(t)/T$ .

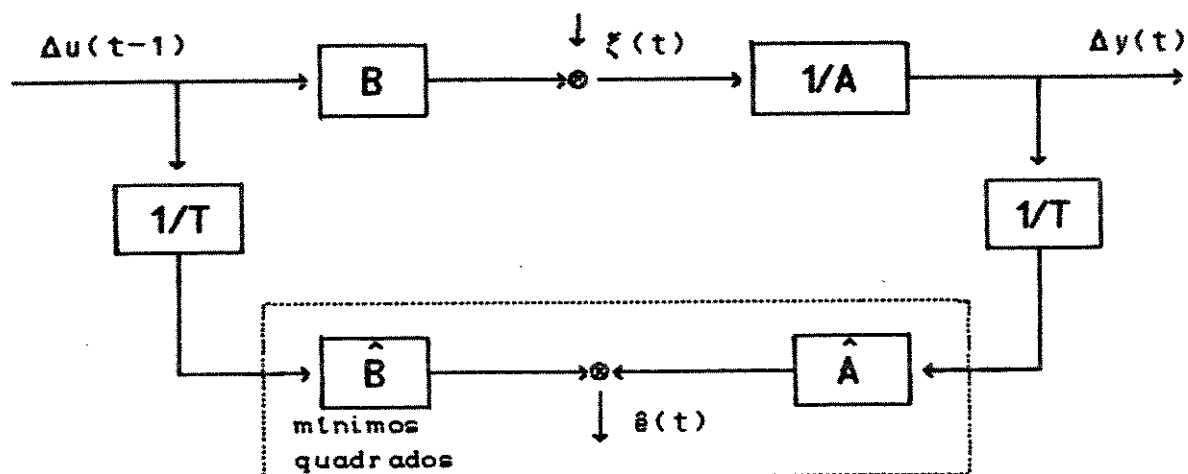


FIG 2.1 Diagrama de blocos do estimador incluindo o filtro  $T(z^{-1})$

Neste modelo assume-se que o polinômio  $C(z^{-1})$  foi modelado pelo filtro  $T$ . Em condições ideais com  $T=C$ , os parâmetros estimados convergem para os valores reais. Em aplicações práticas o filtro  $T$  é considerado fixo, sendo  $1/T$  um filtro passa baixo, insensibilizando o estimador a perturbações de alta frequência.

## 2.4.-IDENTIFICAÇÃO DE SISTEMAS NÃO-ESTACIONÁRIOS

Nos algoritmos de identificação apresentados anteriormente, os elementos da matriz de covariância ( $P(.)$ ) diminuem à medida que os parâmetros estimados convergem para os seus valores reais, e diminui-se o ganho dado pela equação (2.3.11) denominado ganho de correção ou de Kalman. Na identificação de processos invariantes no tempo esta diminuição é desejada, pois ela insensibiliza o estimador às perturbações na carga. Entretanto quando os processos são não-estacionários, com parâmetros variantes no tempo, essa diminuição impede o rastreamento dos parâmetros após o período de convergência inicial. Como todas as estratégias de controle auto-ajustáveis a serem apresentadas baseiam-se no cálculo do sinal de controle em função dos parâmetros estimados, faz-se imprescindível assegurar que o estimador convirja para o modelo correto sob pena de se ter um controle que produza instabilidade no sistema. Das equações (2.3.10) e (2.3.17) nota-se que para

permitir o seguimento correto de não estacionariedades no modelo é suficiente manter  $K(\cdot)$  em níveis apropriados [ARR88].

Os métodos mais comumente utilizados para evitar que  $K(\cdot)$  diminua, atuam sobre os coeficientes da matriz de covariância  $P(\cdot)$ , evitando a sua redução.

Na equação (2.3.19) nota-se que o fator  $\lambda_e(t)$  influencia diretamente no cálculo da matriz  $P(\cdot)$ , assim com  $\lambda_e(t) = \mu(t) = \text{cte.} = 1$  a norma  $\|P(\cdot)\|$  decresce com o tempo tendendo assintoticamente a zero. Nestas circunstâncias, da equação (2.3.18) verifica-se que a norma  $\|K(\cdot)\|$  também aproxima-se assintoticamente a zero. Por outro lado, com  $\lambda_e(t) < 1$  permite-se que as medidas recentes tenham uma maior ponderação do que as mais antigas, provocando que as normas anteriormente citadas tendam assintoticamente a valores maiores que zero, mantendo a "sensibilidade" do algoritmo à variações dos parâmetros. Necessita-se portanto a escolha correta do fator de esquecimento  $\lambda_e(t)$ . Esta escolha não é trivial como se ilustra à seguir.

Na equação (2.3.19), o termo negativo no lado direito representa a redução da incerteza nos parâmetros devido às últimas medidas realizadas. Se estas medidas não possuírem nova informação, isto é, se o sistema não estiver corretamente excitado, então os elementos do vetor  $(P(t) \varphi(t))$  tenderão a zero [WIT84], e a equação (2.3.19) reduz-se a:

$$P(t) = P(t-1) / \lambda(t) \quad (2.4.1)$$

Com  $\lambda_e(t)$  constante e menor que a unidade, os elementos de  $P(t)$  crescerão exponencialmente até que o vetor  $\varphi(t)$  possua nova informação. Portanto se o sistema não é corretamente excitado durante longos períodos de tempo, os elementos de  $P(t)$  e de  $K(t)$  podem assumir valores elevados, provocando, nestas condições, variações excessivas nos parâmetros quando nova informação chega ao processo. Este problema é conhecido na literatura como

"wind-up" do estimador [WIT84], que também pode provocar oscilações no sinal de saída, conhecidas como "burst". Outro problema encontrado no estimador quando a matriz de covariância é muito grande, é a sua maior sensibilidade as perturbações [WEL82].

Para analisar a influência da escolha de  $\lambda_e(t)$  sobre as medidas realizadas, considere-se a equação (2.3.8) onde é definido o fator de ponderação  $W(t)$ . Adotando-se  $\mu(t) = \mu = 1$ , e  $\lambda_e(t) = \lambda_e < 1$ ,  $W(t)$  pode ser aproximado [BUE89] por

$$W(t) = \lambda_e^{t-1} \approx e^{-(t-1)\lambda_e} \quad (2.4.2)$$

Definindo a variável  $C_N$  como

$$C_N = \frac{1}{1 - \lambda_e} \quad (2.4.3)$$

Se  $C_N$  assume o valor  $t-1$ , a ponderação para as últimas "1" medidas é  $W(.) = e^{-1} \cong 36\%$ , isto é,  $\lambda_e(t)$  determina a *constante de tempo da memória* da função de custo, sendo  $C_N$  a "janela" de medidas mais recentes do estimador. Dependendo do valor de  $\lambda_e(t)$  adotado pode-se qualificar a memória da função de custo como na tabela 2.1 ([BUE89, WEL82]):

| $\lambda_e$ | Memória  | $C_N$    |
|-------------|----------|----------|
| 1           | infinita | infinita |
| 0,995       | media    | 200      |
| 0,98        | curta    | 50       |

TABELA 2.1 :  $\lambda_e$  versus Tamanho da memória da função de custo

Normalmente na literatura escolhe-se os métodos para o seguimento de variações nos parâmetros do modelo de acordo com as características das variações dos parâmetros, que são (i) Variações lentas e (ii) Variações rápidas.

#### 2.4.1.-PARÂMETROS COM VARIAÇÕES LENTAS

Para o rastreamento de parâmetros que variam lentamente com o tempo adota-se a mais simples das estratégias possíveis, que é a escolha de um fator de esquecimento constante e menor que 1. Evitando-se fatores de esquecimentos menores que 0,98, diminui-se o risco de "wind-up", desde que o sistema esteja corretamente excitado.

#### 2.4.2.-PARÂMETROS COM VARIAÇÕES RÁPIDAS

Quando ocorrem mudanças abruptas nos parâmetros do processo, estas podem ser detectadas a partir de medidas na saída. Após detectada uma não estacionariedade, deve-se atuar sobre o ganho de correção para permitir a correta estimação dos parâmetros.

Devido às características indicadas pode-se decompor o problema em duas etapas: (1) a detecção da não estacionariedade, e (2) atuação sobre o ganho de correção.

##### 2.4.2.1.-DETECÇÃO DE NÃO ESTACIONARIEDADES

Uma das variáveis do estimador mais sensível às variações dos parâmetros é o erro de previsão dado pela equação (2.3.6), repetida a seguir com  $C(z^{-1})=1$  por simplicidade.

$$e(t) = \Delta y(t) - \varphi_{(t)}^T \hat{\theta}_{(t-1)} \quad (2.4.4)$$

Substituindo o sinal de saída pelo da equação (2.3.2) na sua forma vetorial (com  $\theta(\cdot)$  o vetor de parâmetros reais do processo), obtém-se:

$$e(t) = \varphi_{(t)}^T \theta_{(t-1)} + \xi(t) + \varphi_{(t)}^T \hat{\theta}_{(t-1)} \quad (2.4.5)$$

$$e(t) = \varphi_{(t)}^T \left[ \theta_{(t-1)} - \hat{\theta}_{(t-1)} \right] + \xi(t) \quad (2.4.6)$$

Quando o vetor de parâmetros estimados  $\hat{\theta}(\cdot)$  converge para o vetor de parâmetros reais  $\theta(\cdot)$ , o erro de previsão aproxima-se da perturbação existente no processo  $\zeta(t)$ . Se ocorre uma variação repentina de algum parâmetro do vetor  $\theta(\cdot)$ , verifica-se um aumento do erro de previsão. Logo esta é uma variável apropriada para a detecção de variações no estimador [ISE85].

Entretanto, a detecção de não estacionariedades do modelo através da observação do valor instantâneo do erro de previsão apresenta alguns inconvenientes tais como [BUE89]:

- i)-É preciso conhecer a relação sinal/ruído existente no processo para poder tomar decisões.
- ii)-Se acontecerem variações na relação sinal/ruído, elas podem ser interpretadas como não estacionariedades.
- iii)-Apenas podem ser detectadas as variações de parâmetros que provoquem uma grande influência no sinal de saída.

Pelos inconvenientes citados anteriormente, adotou-se um método de detecção de não estacionariedade baseado na variância do erro de previsão calculada sobre um intervalo de tempo. O método foi proposto por Favler [FAV87a] e Arruda [ARR88]. Um método similar, mas baseado no valor absoluto do erro de previsão computado sobre intervalos de tempo, também pode ser encontrado em [BUE89].

Para a detecção de uma não estacionariedade compara-se a relação entre a variância do erro de previsão e a variância da perturbação existente no processo. Para a obtenção da variância do erro de previsão utiliza-se da seguinte equação:

$$\sigma_c^2(t) = \frac{1}{N_c} \sum_{i=t-N_c}^t e_{(1)}^2 \quad (2.4.7)$$

onde  $N_c$  é denominado horizonte curto e é escolhido pelo usuário em função da velocidade de detecção desejada. Este intervalo não pode ser muito curto para evitar detecções falsas de variação de modelo. A variância da perturbação é obtida a partir da mesma

variável, (equação 2.4.8) assumindo que nos instantes anteriores a  $\tau+Nc$  o estimador tenha convergido para os parâmetros corretos. " $\tau$ " é um intervalo de tempo para desacoplar as medidas do erro de previsão para o cálculo das duas variâncias. O coeficiente  $Nl$  conhecido como horizonte longo é escolhido pelo usuário em função das características da perturbação.

$$\sigma_l^2(t) = \frac{1}{Nl} \sum_{i=t-Nc-\tau-Nl}^{t-Nc-\tau} e_{(1)}^2 \quad (2.4.8)$$

A seguir calcula-se a variável  $J(t)$  como:

$$J(t) = \sigma_c^2(t) / \sigma_l^2(t) \quad (2.4.9)$$

com este índice determina-se se o modelo tem sofrido uma não estacionaridade. Esta variável é utilizada na seção seguinte para se obter o ganho de correção do estimador.

#### 2.4.2.2.-ATUAÇÃO SOBRE O GANHO DE CORREÇÃO

Da mesma forma que no caso de variações lentas, atua-se sobre o fator de esquecimento para aumentar os elementos da matriz  $P(\cdot)$ , e com isto o ganho de adaptação.

A cada nova iteração é calculado um novo valor de  $\lambda(t)$  da seguinte equação:

$$\lambda_0(t) = \text{traço}(\tilde{P}(t)) / \text{tro} \quad (2.4.10)$$

com

$$\tilde{P}(t) = \left[ P(t-1) - \frac{P(t-1) \varphi(t) \varphi^T(t) P(t-1)}{1/\mu(t) + \varphi^T(t) P(t-1) \varphi(t)} \right] \quad (2.4.11)$$

sendo o fator " $\text{tro}$ " o *traço de regulagem* que é escolhido como é indicado a seguir:

i)-Se  $0 \leq J(t) < J_{\min}$ , não ocorreu nenhuma variação de modelo, sendo  $J_{\min}$  o limite mínimo para a detecção de uma não estacionariedade (equação 2.4.9). Nestas condições utiliza-se um traço de regulagem mínimo  $tro = t_{\min}$ .

ii)-Se  $J_{\min} \leq J(t) < J_{\max}$ , onde  $J_{\max}$  é o limite máximo para a detecção de não estacionariedade, a detecção é incerta e utiliza-se um traço de regulagem intermediário  $tro = t_{\text{int}}$ .

iii)-Se  $J(t) \geq J_{\max}$ , ocorreu a detecção de uma não estacionariedade e deve-se utilizar um traço elevado  $tro = t_{\max}$ .

Em quaisquer dos casos anteriores se o traço de  $\tilde{P}_{(t)} \geq tro$  então utiliza-se  $\lambda_e(t) = \lambda_{\max}$ , sendo este  $\lambda_{\max}$  o fator de esquecimento utilizado para o rastreamento de variações lentas.

Finalmente calcula-se  $P_{(t)}$  como

$$P_{(t)} = \tilde{P}_{(t)} / \lambda_e(t) \quad (2.4.12)$$

#### 2.4.2.3.-CONSIDERAÇÕES NO ALGORITMO DE DETECÇÃO DE NÃO ESTACIONARIEDADES

Para se obter um rastreamento adequado das variações dos parâmetros do processo, e evitar detecções falsas, deve-se considerar as seguintes condições na implementação do algoritmo.

i)-Como o índice  $J(.)$  é obtido como função da variância da perturbação, e esta é calculada através da variância do erro de previsão, para se ter uma estimação adequada deste valor, os parâmetros estimados devem ter convergido para os parâmetros reais. Na ausência de um teste de convergência, aplica-se a detecção de não estacionariedade somente após um tempo igual ou superior ao tempo de estabilização do estimador indicado na figura 2.2 pelo intervalo TE.

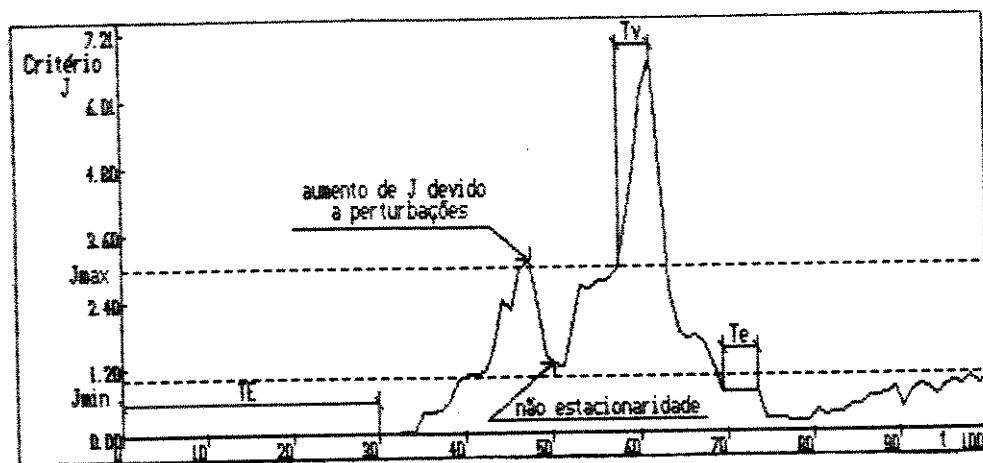


Fig 2.2a Critério  $J(t)$

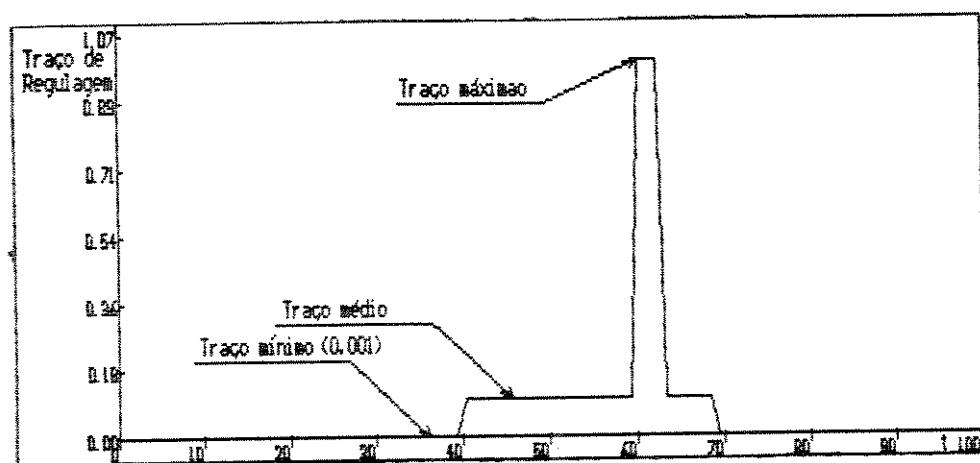


Fig 2.2b Traço de regulagem

II)-O traço de regulagem máximo só é aplicado após um certo número de iterações de "validação", que são utilizadas para evitar detecções falsas, devidas por exemplo a perturbações impulsivas na carga, na figura 2.2 este intervalo é indicado como  $T_v$ .

III)-Uma vez detectada uma não estacionariedade, se o critério  $J(.)$  decresce abaixo do limite mínimo, ele não é mais calculado até que se ultrapasse um tempo mínimo de estabilização representado na figura 2.2 por  $T_e$ , para evitar oscilações fortes dos parâmetros estimados após a detecção de uma não estacionariedade.

## 2.5.-EXCITAÇÃO DO PROCESSO

Como foi indicado na seção 2.4, se a excitação do processo não é persistente, isto é, se não existe nova informação nas últimas medidas realizadas, a norma do produto  $P_{(t)}\varphi(t)$  tende a zero, prejudicando a convergência do estimador, e em presença de um factor de esquecimento menor que 1, pode provocar o "wind-up" do estimador.

Uma solução para este problema é incorporar no sinal de excitação um sinal de baixa amplitude e com características tais que excite todos os modos do processo. Tal sinal idealmente é um ruído branco, que sendo computacionalmente custoso de obter é substituído por uma *Sequência Binária Pseudo-Randômica* (PRBS) [BUE89], que deve exercer uma influência no sinal de saída maior que a da perturbação presente no processo.

A PRBS utilizada é um vetor com 31 componentes dado por

$$\text{PRBS}^T = [1, 1, -1, 1, 1, -1, 1, 1, 1, 1, -1, -1, -1, 1, -1, 1, \\ -1, 1, 1, 1, -1, -1, -1, -1, 1, -1, -1, 1, -1, -1, 1] \quad (2.5.1)$$

e é aplicado ao sinal de controle segundo a equação (2.5.2)

$$u(t) = u(t) + \text{PRBS}(t \bmod 31) \quad (2.5.2)$$

### 2.5.1.-DETECÇÃO DE EXCITAÇÃO NÃO PERSISTENTE

Como em muitas ocasiões o sistema está corretamente excitado, pode não ser necessário aplicar continuamente o sinal PRBS, que prejudica o sinal de saída aumentando a sua variância.

O teste para a detecção de falta de excitação persistente é feito através da monitoração do traço da matriz de covariância  $P(\cdot)$ , no qual se  $\text{traço}(P_{(t)}) > \text{Limite}$  então o sistema não se encontra corretamente excitado, e faz-se necessário somar a sequência PRBS à excitação.

## 2.6.-ALGORITMO DE CONTROLE PID

O Algoritmo de controle PID é até hoje o mais utilizado dentro da indústria, devido principalmente à sua simplicidade, o que lhe permite ser implementado em forma tanto analógica quanto digital, e a que a grande maioria dos processos industriais podem ser controlados com leis de controle simples.

Embora já se tenham desenvolvido trabalhos sobre este tipo de controlador, que o provém de uma estrutura auto-ajustável [WIT79, BUE89], optou-se por implementá-lo com estrutura fixa, devido à existência no sistema de algoritmos auto-ajustáveis que são utilizados quando os processos controlados forem não estacionários.

A equação que descreve o comportamento dinâmico em função da entrada e a saída de um controlador com estrutura PID contínuo é dada por:

$$u(t) = K_P e(t) + \left[ \frac{1}{T_I} \int_0^t e(\tau) d\tau + T_D \frac{d e(t)}{dt} \right] \quad (2.6.1)$$

onde  $e(.)$  é o erro entre o sinal de referência  $w(.)$  e a saída do processo  $y(.)$ .

$$e(t) = w(t) - y(t) \quad (2.6.2)$$

O ganho proporcional  $K_P$ , o tempo integral  $T_I$  e o tempo derivativo  $T_D$ , permitem o ajuste do controlador para um desempenho especificado quando se controla um determinado processo. Um problema apresentado por este controlador é que o erro atual  $e(t)$  atua diretamente sobre os termos proporcional e diferencial, comprometendo a estabilidade da malha de controle quando a referência apresenta variações muito rápidas (i.e. degraus) [KUC88]. Estes problemas fazem que em muitas implementações práticas, as estruturas difiram da apresentada na equação (2.6.1) [WIT79].

Das diferentes estruturas existentes para os algoritmos contínuos surgem os equivalentes discretos, em número maior devido aos diferentes métodos de discretização utilizados, (por exemplo, aproximação trapezoidal para a integral, no lugar da simples somatória). Como é destacado em [WIT79], as diferentes estruturas obtidas apresentam diferentes propriedades com respeito à evolução do sistema em malha fechada.

Por simplicidade o algoritmo aqui obtido baseia-se na discretização da equação (2.6.1) aproximando o termo integrativo contínuo a:

$$\frac{T_S}{T_I} \sum_{i=1}^t e(i) \quad (2.6.3)$$

e a diferenciação contínua a:

$$\frac{T_D}{T_S} \left[ e(t) - e(t-1) \right] \quad (2.6.4)$$

Nas equações anteriores a variável  $T_S$  é o período de amostragem utilizado para a discretização do modelo do processo.

Logo a equação do sinal de controle em tempo discreto é:

$$u(t) = K_P \left[ e(t) + \frac{T_S}{T_I} \sum_{i=1}^t e(i) + \frac{T_D}{T_S} \left[ e(t) - e(t-1) \right] \right] \quad (2.6.5)$$

reescrevendo esta equação para o instante  $t-1$  obtém-se

$$u(t-1) = K_P \left[ e(t-1) + \frac{T_S}{T_I} \sum_{i=1}^t e(i) + \frac{T_D}{T_S} \left[ e(t-1) - e(t-2) \right] \right] \quad (2.6.6)$$

Subtraindo a equação (2.6.6) da equação (2.6.5) obtém-se a forma recursiva do controlador PID discreto, denominado incremental ou de velocidade.

$$\Delta u(t) = K_P \left[ e(t) - e(t-1) + \frac{T_S}{T_I} e(t) + \frac{T_D}{T_S} \left[ e(t) - 2e(t-1) + e(t-2) \right] \right] \quad (2.6.7)$$

Os termos  $T_i/T_i$  e  $T_d/T_s$ , são denominados Ganho Integral e Ganho Derivativo respectivamente, e assim são considerados no sistema implementado.

## 2.7.-CONTROLADOR DE VARIÂNCIA MÍNIMA GENERALIZADA

### 2.7.1.-INTRODUÇÃO

Como já foi descrito na seção anterior, os controladores PID não-adaptativos são aplicáveis à grande maioria dos processos industriais, mas ... a grande maioria não são todos. Já na década dos 50', começaram a aparecer aplicações onde os controladores clássicos não apresentavam desempenhos satisfatórios. Estes processos tinham como característica comum a variação dos seus parâmetros. Este problema levou à procura de controladores que pudessem ajustar automaticamente os seus parâmetros para acomodar às mudanças no processo ou no ambiente. Nesta época ocorrem dois problemas, a falta de uma base teórica, e a inexistência de hardware apropriado.

Nos anos 60, as estratégias de desenvolvimento de controladores auto-ajustáveis foi fundamentada numa base teórica mais segura, devido à introdução dos conceitos de *Controle Moderno*, especialmente os da *Teoria da Estabilidade*. Junto com este desenvolvimento, o hardware disponível tornou-se mais rápido e barato, e conseqüentemente mais fácil de ser utilizado.

Assim, estes dois fatores, permitiram que nos começos dos anos 70 desenvolvessem algoritmos de controle auto-ajustável aplicáveis industrialmente. O primeiro deles, foi o Regulador Auto-Ajustável ou STR<sup>6</sup> [AST73], que veio a demonstrar a viabilidade destes controladores. Porém, ele possuía alguns problemas que incentivaram o desenvolvimento de outros

---

<sup>6</sup>STR: Do inglês Self-Tuning Regulator.

controladores baseados na mesma teoria. O primeiro passo foi dado por Clarke e Gawthrop [CLK75] com o Controlador de Variância Mínima Generalizada ou GMVC, que é desenvolvido a seguir.

## 2.7.2.-ALGORITMO DE CONTROLE GMV

Como acontece com a grande maioria dos controladores auto-ajustáveis, desenvolve-se o presente controlador assumindo que os parâmetros estimados são os reais do processo, o que denomina-se o princípio de equivalência certa. Assim de agora em diante assume-se que os polinômios do processo indicados nas seguintes equações são, ou fornecidos pelo usuário, ou estimados pela equação (2.3.17).

Diferentes formas tem sido propostas para o desenvolvimento deste algoritmo, o aqui apresentado é similar ao desenvolvido em [MEN84], com a diferença de que o processo assumido é tipo CARIMA e não CARMA.

O sinal de controle é obtido pela minimização do seguinte critério quadrático:

$$J = \varepsilon \left\{ \left[ \left( P y(t+d) - R w(t) \right)^2 + \left( Q' \Delta u(t) \right)^2 \right] / \mathcal{M}_t \right\} \quad (2.7.1)$$

sendo " $\varepsilon$ " o operador esperança matemática com respeito ao conjunto de medidas  $\mathcal{M}_t$  disponíveis no instante " $t$ ". " $d$ " é o atraso de transporte identificado do processo, e  $P(z^{-1})$ ,  $Q'(z^{-1})$  e  $R(z^{-1})$  filtros fornecidos pelo usuário que determinam o desempenho do sistema em malha fechada, e representam o conhecimento do usuário a respeito do processo, o ambiente, e a evolução desejada do sistema.

### 2.7.2.1.-PREVISÃO DA SAÍDA "d" PASSOS À FRENTE

A saída do processo dada pela equação (2.2.13), filtrada pelo polinômio  $P(z^{-1})$  é dada por (o argumento  $z^{-1}$  é eliminado por simplicidade):

$$P y(t) = z^{-d} \frac{PB}{\alpha} \Delta u(t) + \frac{PC}{\alpha} \xi(t) \quad (2.7.2)$$

No desenvolvimento do algoritmo de controle assume-se que a perturbação atuando no processo  $\xi(t)$  é um ruído branco ( $\mathcal{N}(0, \sigma^2)$ ). A saída filtrada do processo "d" passos à frente é dada por:

$$P y(t+d) = \frac{PB}{\alpha} \Delta u(t) + \frac{PC}{\alpha} \xi(t+d) \quad (2.7.3)$$

Na equação (2.7.3) nota-se que a saída filtrada "d" passos à frente depende da entrada atual e passada, e da perturbação passada e futura; procedendo na forma clássica, utiliza-se a seguinte identidade polinomial [AST70] para separar a perturbação nos seus valores passados e futuros.

$$\frac{PC}{\alpha} = F + z^{-d} \frac{G}{\alpha} \quad (2.7.4)$$

com a ordem do polinômio  $F$  dada por  $n_f = d-1$ ; e a ordem do polinômio  $G$  dada por  $n_g = \max(n_a, n_c + n_p - d)$ .

Aplicando a equação (2.7.4) na equação (2.7.3) obtém-se a seguinte saída filtrada:

$$P y(t+d) = \frac{PB}{\alpha} \Delta u(t) + F \xi(t+d) + \frac{G}{\alpha} \xi(t) \quad (2.7.5)$$

No instante "t" a melhor previsão que se pode fazer com as medidas conhecidas é dada por:

$$P \hat{y}(t+d/t) = \frac{PB}{\alpha} \Delta u(t) + \frac{G}{\alpha} \xi(t) \quad (2.7.6)$$

onde o acento circunflexo (^) indica previsão. A saída é dada por:

$$P y(t+d) = P \hat{y}(t+d/t) + F \xi(t+d) \quad (2.7.7)$$

ou seja a soma da saída prevista mais a perturbação futura não mensurável filtrada pelo polinômio  $F(z^{-1})$ .

A perturbação atual  $\xi(t)$  pode ser obtida da equação do modelo do processo ( equação 2.2.13) como:

$$\xi(t) = - z^{-d} \frac{B}{C} \Delta u(t) + \frac{\alpha}{C} y(t) \quad (2.7.8)$$

onde todas as medidas são conhecidas no instante "t". Substituindo a equação (2.7.8) na equação (2.7.6) obtém-se:

$$\begin{aligned} P \hat{y}(t+d/t) &= \frac{PB}{\alpha} \Delta u(t) + \frac{G}{\alpha} \left[ \frac{\alpha}{C} y(t) - z^{-d} \frac{B}{C} \Delta u(t) \right] \\ P \hat{y}(t+d/t) &= \frac{G}{C} y(t) + \Delta u(t) \left[ \frac{PB}{\alpha} - z^{-d} \frac{BG}{C\alpha} \right] \end{aligned} \quad (2.7.9)$$

Recorrendo novamente à identidade polinomial (2.7.4), e multiplicando ambos os membros por  $B/C$  obtém-se:

$$\begin{aligned} \frac{PB}{\alpha} &= \frac{FB}{C} + z^{-d} \frac{GB}{C\alpha} \\ \text{ou} \\ \frac{FB}{C} &= \frac{PB}{\alpha} - z^{-d} \frac{GB}{C\alpha} \end{aligned} \quad (2.7.10)$$

Substituindo esta equação na equação (2.7.9) obtém-se a previsão da saída filtrada "d" passos à frente como:

$$P \hat{y}(t+d/t) = \frac{G}{C} y(t) + \frac{FB}{\alpha} \Delta u(t) \quad (2.7.11)$$

Nesta equação deve-se observar que todas as medidas são conhecidas no instante "t".

## 2.7.2.2.-MINIMIZAÇÃO DA FUNÇÃO DE CUSTO

Utilizando-se a equação (2.7.7) a função de custo dada na equação (2.7.1) pode ser reescrita como segue:

$$J = E \left\{ \left[ \left( P \hat{y}(t+d/t) + F \xi(t+d) - R w(t) \right)^2 + \left( Q' \Delta u(t) \right)^2 \right] / \mathcal{M}t \right\} \quad (2.7.12)$$

Desenvolvendo-se os quadrados tem-se:

$$\begin{aligned} J &= E \left\{ \left[ \left( P \hat{y}(t+d/t) - R w(t) \right)^2 + \right. \right. \\ &\quad + 2 F \xi(t+d) \left[ P \hat{y}(t+d/t) - R w(t) \right] + \\ &\quad \left. \left. + \left( F \xi(t+d) \right)^2 + \left( Q' \Delta u(t) \right)^2 \right] / \mathcal{M}t \right\} \\ J &= E \left\{ \left[ \left( P \hat{y}(t+d/t) - R w(t) \right)^2 + \left( Q' \Delta u(t) \right)^2 \right] / \mathcal{M}t \right\} + \\ &\quad + E \left\{ \left[ 2 F \xi(t+d) \left[ P \hat{y}(t+d/t) - R w(t) \right] \right] / \mathcal{M}t \right\} + \\ &\quad + E \left\{ \left[ \left( F \xi(t+d) \right)^2 \right] / \mathcal{M}t \right\} \quad (2.7.13) \end{aligned}$$

Dos três termos do lado direito da equação (2.7.13) obtém-se:

$$\begin{aligned} 1) \quad E \left\{ \left[ \left( P \hat{y}(t+d/t) - R w(t) \right)^2 + \left( Q' \Delta u(t) \right)^2 \right] / \mathcal{M}t \right\} &= \\ \left[ \left( P \hat{y}(t+d/t) - R w(t) \right)^2 + \left( Q' \Delta u(t) \right)^2 \right] &\quad (2.7.14) \end{aligned}$$

pois todas as medidas são conhecidas no instante "t".

$$2) \quad E \left\{ \left[ 2 F \xi(t+d) \left[ P \hat{y}(t+d/t) - R w(t) \right] \right] / \mathcal{M}t \right\} = 0 \quad (2.7.15)$$

pois  $F \xi(t+d)$  é não correlato com as medidas.

ii)

$$\varepsilon \left\{ \left[ F \xi(t+d) \right]^2 / Mt \right\} = \varepsilon \left\{ \left[ F \xi(t+d) \right]^2 \right\} \quad (2.7.16)$$

ou seja, a perturbação  $F\xi(t+d)$  é independente do instante "t", já que o polinômio  $F(z^{-1})$  é de ordem  $d-1$ .

Desta forma a função de custo é reduzida a:

$$J = \left[ \left( P \hat{y}(t+d/t) - R w(t) \right)^2 + \left( Q' \Delta u(t) \right)^2 \right] + \varepsilon \left\{ \left[ F \xi(t+d) \right]^2 \right\} \quad (2.7.17)$$

Derivando a equação (2.7.17) com respeito a  $\Delta u(t)$ , isto é, o incremento da excitação no instante "t", obtém-se:

$$\begin{aligned} \frac{dJ}{d\Delta u(t)} &= 2 \left[ P \hat{y}(t+d/t) - R w(t) \right] \left[ \frac{\partial(P \hat{y}(t+d/t))}{\partial \Delta u(t)} - \frac{\partial(R w(t))}{\partial \Delta u(t)} \right] + \\ &+ 2 Q' \Delta u(t) \frac{\partial(Q' \Delta u(t))}{\partial \Delta u(t)} + \frac{\partial(\varepsilon \{ (F \xi(t+d))^2 \})}{\partial \Delta u(t)} = 0 \end{aligned} \quad (2.7.18)$$

Analisando-se cada termo da equação anterior obtém-se as seguintes conclusões:

i) da equação (2.7.6)

$$\frac{\partial(P \hat{y}(t+d/t))}{\partial \Delta u(t)} = \frac{\partial \left[ \frac{PB}{\alpha} \Delta u(t) \right]}{\partial \Delta u(t)} + \frac{\partial \left[ \frac{G}{\alpha} \xi(t) \right]}{\partial \Delta u(t)} = p_0 \quad (2.7.19)$$

ii)

$$\frac{\partial(Q' \Delta u(t))}{\partial \Delta u(t)} = q_0' \quad (2.7.20)$$

e finalmente

iii)

$$\frac{\partial(\varepsilon \{ (F \xi(t+d))^2 \})}{\partial \Delta u(t)} = 0 \quad (2.7.21)$$

Substituindo estas derivadas na equação (2.7.18), tem-se:

$$\frac{dJ}{d\Delta u(t)} = 2 \left[ P \hat{y}(t+d/t) - R w(t) \right] p_0 b_0 + 2 Q' \Delta u(t) q_0' = 0 \quad (2.7.22)$$

$$P \hat{y}(t+d/t) - R w(t) + \frac{q_0'}{p_0 b_0} Q' \Delta u(t) = 0 \quad (2.7.23)$$

Definindo  $Q = \frac{q_0'}{p_0 b_0} Q'$   
obtem-se que

$$P \hat{y}(t+d/t) - R w(t) + Q \Delta u(t) = 0 \quad (2.7.24)$$

Substituindo  $P \hat{y}(t+d/t)$  na equação anterior pelo obtido na equação (2.7.11) tem-se a seguinte equação de controle:

$$G y(t) + (FB + QC) \Delta u(t) - RC w(t) = 0 \quad (2.7.25)$$

Definindo os termos

$$H = FB + QC$$

e

$$E = -RC \quad (2.7.26)$$

obtem-se a equação de controle do GMV (equação de velocidade)

$$G y(t) + H \Delta u(t) + E w(t) = 0 \quad (2.7.27)$$

O controlador GMV já foi aplicado no controle de diferentes tipos de processos, além do que, a sua simplicidade permite que ele possa ser implementado em pequenos computadores, o que fez este algoritmo muito atrativo para a substituição dos controladores clássicos (PID) em processos historicamente controlados por estes.

O principal problema enfrentado pelos projetistas de sistemas de controle é assegurar a estabilidade do sistema em malha fechada. No algoritmo apresentado anteriormente, a robustez fica determinada, principalmente, por dois fatores:

i)-A correta estimação dos parâmetros do processo a ser controlado (principalmente em processos instáveis em malha aberta) e do valor correto do atraso de transporte, utilizado no cálculo do sinal de controle.

ii)-Da escolha dos filtros P, Q e R. Em [CLK79] pode-se encontrar algumas sugestões para a escolha destes filtros, porém elas não garantem a estabilidade do sistema em malha fechada.

Devido a estes problemas, analisa-se a seguir a influência dos filtros na função de transferência em malha fechada.

### 2.7.2.3.-FUNÇÃO DE TRANSFERÊNCIA EM MALHA FECHADA

Para se obter a função de transferência em malha fechada reescreve-se a equação (2.7.27) como segue:

$$\Delta u(t) = - \frac{G}{H} y(t) - \frac{E}{H} w(t) \quad (2.7.28)$$

Substituindo esta equação no modelo do processo dado pela equação (2.2.13) obtém-se:

$$\begin{aligned} H \alpha y(t) &= - z^{-d} B G y(t) - z^{-d} B E w(t) + H G \xi(t) \\ (H \alpha + z^{-d} B G) y(t) &= - z^{-d} B E w(t) + H G \xi(t) \end{aligned} \quad (2.7.29)$$

Analisando o termo do lado esquerdo pode-se ver que:

$$\begin{aligned} H \alpha + z^{-d} B G &= \alpha (F B + Q C) + z^{-d} B (z^d (P C - \alpha F)) \\ &= C (B P + \alpha Q) \end{aligned} \quad (2.7.30)$$

logo

$$C (B P + \alpha Q) y(t) = z^{-d} B E w(t) + H G \xi(t) \quad (2.7.31)$$

Substituindo o polinômio E pelo seu valor, obtém-se a função de transferência em malha fechada dada a seguir:

$$y(t) = z^{-d} \frac{B R}{B P + \alpha Q} w(t) + \frac{H}{B P + \alpha Q} \xi(t) \quad (2.7.32)$$

A equação anterior mostra a influência dos filtros na evolução e estabilidade do controle, e também fornece um método para o seu cálculo.

Definindo o polinômio S como os polos desejados em malha fechada, os polinômios P e Q podem ser calculados a partir da resolução da seguinte equação:

$$B P + \alpha Q = S \quad (2.7.33)$$

A equação (2.7.32) mostra também a necessidade de uma estimativa correta dos parâmetros do processo. Nos primeiros instantes da estimativa eles podem estar muito longe dos valores corretos, é de se esperar que nestas condições ocorram grandes oscilações na saída do processo. Uma solução a este problema é a estimativa inicialmente em malha aberta, ou com um controlador mais robusto.

#### 2.7.2.4.-ELIMINAÇÃO DO ERRO DE REGIME

Depois da robustez do controlador, o segundo maior problema a ser resolvido pelos projetistas de sistemas de controle é a eliminação do erro de regime [TUF85]

Para se analisar quando é satisfeita a condição de erro de regime nulo para uma entrada degrau, utiliza-se o teorema do valor final, que fornece o valor de uma variável quando o tempo tende a  $\infty$ , como função do valor obtido de

$$\lim_{t \rightarrow \infty} y(t) = \lim_{z \rightarrow 1} \frac{z - 1}{z} Y(z)$$

onde  $Y(z)$  é a transformada z do sinal temporal  $y(t)$ .

No desenvolvimento do algoritmo foi assumido que a perturbação é um ruído branco, (i.e.  $\mathcal{N}(0, \sigma^2)$ ), portanto para obter o valor de regime da saída analisa-se o primeiro termo do lado direito da equação (2.7.32). Assim a transformada z desta parte da

equação (2.7.32) é:

$$Y(z) = z^{-d} \frac{B R}{B P + \alpha Q} W(z) \quad (2.7.34)$$

A transformada z de um degrau de amplitude "h" é dado por:

$$W(z) = h \frac{z}{z - 1}$$

Aplicando o teorema do valor final obtém-se

$$\lim_{t \rightarrow \infty} y(t) = \lim_{z \rightarrow 1} \frac{z - 1}{z} z^{-d} \frac{B R}{B P + \alpha Q} h \frac{z}{z - 1} \quad (2.7.35)$$

Mas  $\alpha = \Delta A$ , e  $\Delta(1) = 0$ , portanto

$$\lim_{t \rightarrow \infty} y(t) = \lim_{z \rightarrow 1} \frac{B R}{B P} h = \frac{R(1)}{P(1)} H \quad (2.7.36)$$

Tem-se erro de regime nulo se

$$R(1) = P(1) \quad (2.7.37)$$

Os resultados anteriores demostram que o modelo CARIMA permite derivar algoritmos de controle com ação integral em forma natural, já que no algoritmo posicional (isto é, derivado do modelo GARMA), o filtro Q deve necessariamente incluir o fator  $(1-z^{-1})$  [CLK79, MEN84].

#### 2.7.2.5.-RUÍDO COLORIDO E PERTURBAÇÕES NA CARGA

Quando na estimação dos parâmetros é utilizado o filtro T, evitando-se a estimação do polinômio  $C(z^{-1})$ , este filtro também pode ser incluído no cálculo do sinal de controle (apêndice B), obtendo-se a seguinte equação de controle:

$$G_1 y'(t) + H_1 \Delta u'(t) + E w(t) = 0 \quad (2.7.38)$$

onde o acento (') indica sinal filtrado por  $1/T$ . Logo, para se obter o incremento do sinal de controle a ser aplicado, precisa-se recuperar este sinal como indicado a seguir:

$$\Delta u(t) = t_0 \Delta u'(t) + t_1 \Delta u'(t-1) + \dots + t_{nt} \Delta u'(t-nt) \quad (2.7.39)$$

Como pode ser verificado no apêndice B, o filtro  $T(z^{-1})$  não afeta a alocação dos pólos em malha fechada (isto é, comportamento determinístico). Entretanto a sua presença permite melhorar o desempenho do algoritmo em relação às perturbações da carga. Este filtro é escolhido para atenuar as perturbações de alta e baixa frequência, assim deverá possuir características passa faixa <sup>7</sup>.

## 2.8.-CONTROLADOR PREDICTIVO GENERALIZADO

### 2.8.1.-INTRODUÇÃO

Os primeiros algoritmos de controle auto-ajustável tais como o STR [AST73] ou o GMV visto na seção anterior, tinham como particularidade a previsão da saída em "d" passos à frente, onde "d" é o atraso de transporte do processo. Entretanto para a correta operação destes controladores, é necessário um bom conhecimento a priori do atraso de transporte [CLK79].

Quando o valor do atraso não é conhecido, ou varia no tempo pode-se realizar a previsão da saída sobre um horizonte igual ou maior que o atraso de transporte [PET84, CLK84, LEL87]. Esta prática tem-se mostrado efetiva no controle de diferentes processos [CLK84, MOH86, FAV87b, LEL87, etc].

---

<sup>7</sup> Ainda que em ambientes onde as perturbações são melhor conhecidas, deve-se adequar as características deste filtro a essas perturbações. Por exemplo, quando é conhecida a ausência de perturbações de baixa frequência, pode-se escolher o filtro  $T$  como passa baixo.

Nestes algoritmos a lei de controle é obtida pela minimização de um critério quadrático genericamente indicado por:

$$J = E \left\{ \left[ \sum_{i=N1}^{N2} \gamma_i \phi_y^2(t+i) + \sum_{i=1}^{Nu} \lambda_i \phi_u^2(t+i-1) \right] / \mathcal{M}t \right\} \quad (2.8.1)$$

onde:

- .- $E(.)$  é o operador esperança matemática, condicionado ao conjunto de medidas  $\mathcal{M}t$  disponíveis no instante "t".
- .- $\phi_y$  e  $\phi_u$  são os termos que consideram a saída e a excitação respectivamente.
- .- $\gamma_i$  e  $\lambda_i$  são os coeficientes de ponderação da saída e da excitação respectivamente.
- .- $N1$  é o denominado horizonte inicial de previsão da saída ou horizonte mínimo.
- .- $N2$  horizonte final de previsão da saída ou horizonte máximo.
- .- $Nu$  o horizonte de controle.

Dependendo da escolha dos termos  $\phi_y(.)$  e  $\phi_u(.)$ , obtém-se os diferentes algoritmos de controle preditivos encontrados na literatura [CLK84, LEL87]. Desta equação também é evidente que, ao contrário dos controladores baseados somente em uma previsão, nestes dispõem-se de um horizonte maior para que a saída atinja os valores requeridos, obtendo como consequência imediata deste fato, a diminuição dos incrementos no sinal de controle, o que evita a saturação dos atuadores.

A seguir são apresentados dois métodos para a dedução do algoritmo, que somente diferem quanto ao método utilizado para a obtenção da previsão da saída. O primeiro está baseado no método clássico de decomposição por identidade polinomial, útil para a análise do sistema em malha fechada, porém que requer muito tempo para o cálculo do sinal de controle. O segundo, baseado na iteração do modelo do processo, é mais eficiente computacionalmente, mas pouco útil para se realizar uma análise de malha fechada.

### 2.8.2.-PREVISÃO DA SAÍDA "J" PASSOS À FRENTE

Como no caso do controlador de variância mínima generalizada, para a obtenção da lei de controle, obtém-se inicialmente a previsão da saída filtrada, sendo ela dada por:

$$y_f(t) = P(z^{-1}) y(t) \quad (2.8.2)$$

Substituindo-se o valor da saída pelo valor filtrado, obtém-se da equação (2.2.14) o modelo filtrado dado a seguir:

$$y_f(t) = \frac{PB}{\alpha} \Delta u(t-1) + \frac{PC}{\alpha} \xi(t) \quad (2.8.3)$$

Para a obtenção da previsão procede-se como na seção 2.7.2.1, deslocando a equação anterior no tempo "J" passos à frente, obtém-se:

$$y_f(t+j) = \frac{PB}{\alpha} \Delta u(t+j-1) + \frac{PC}{\alpha} \xi(t+j) \quad (2.8.4)$$

Utilizando a seguinte identidade polinomial pode-se decompor a perturbação nas medidas passadas e as futuras

$$\frac{PC}{\alpha} = FJ + z^{-j} \frac{GJ}{\alpha} \quad (2.8.5)$$

com as ordens dos polinômios  $nfj = j-1$  e  $ngj = \max(na, nc+np-j)$ .

Logo

$$y_f(t+j) = \frac{PB}{\alpha} \Delta u(t+j-1) + FJ \xi(t+j) + \frac{GJ}{\alpha} \xi(t) \quad (2.8.6)$$

A melhor previsão da saída filtrada "J" passos à frente é dada por:

$$\hat{y}_f(t+j/t) = \frac{PB}{\alpha} \Delta u(t+j-1) + \frac{GJ}{\alpha} \xi(t) \quad (2.8.7)$$

com o seguinte erro de previsão

$$e(t+j/t) = FJ \xi(t+j) \quad (2.8.8)$$

Da equação (2.2.14) pode-se obter a perturbação no instante atual como:

$$\xi(t) = \frac{\alpha}{C} y(t) - \frac{B}{C} \Delta u(t-1) \quad (2.8.9)$$

Substituindo este valor na equação (2.8.7) obtém-se:

$$\begin{aligned} \hat{y}_f(t+j/t) &= \frac{PB}{\alpha} \Delta u(t+j-1) + \frac{GJ}{C} y(t) - \frac{BGJ}{C\alpha} \Delta u(t-1) \\ &= \frac{GJ}{C} y(t) + \Delta u(t+j-1) \left[ \frac{PB}{\alpha} - z^{-j} \frac{BGJ}{C\alpha} \right] \end{aligned} \quad (2.8.10)$$

Da equação (2.8.5) multiplicando ambos os membros por B/C obtém-se:

$$\frac{BFJ}{C} = \frac{PB}{\alpha} - z^{-j} \frac{BGJ}{C\alpha} \quad (2.8.11)$$

Assim a previsão "J" passos à frente é dada por:

$$\hat{y}_f(t+j/t) = \frac{GJ}{C} y(t) + \frac{BFJ}{C} \Delta u(t+j-1) \quad (2.8.12)$$

Esta equação é utilizada para se obter as previsões da saída (filtrada) desde um horizonte mínimo N1 até o horizonte máximo N2. Para cada uma destas previsões deve-se calcular os polinômios FJ e GJ, trabalho que pode ser simplificado através de um cálculo recursivo conforme apresentado no apêndice A.

A outra solução baseia-se no cálculo direto das previsões da saída  $y_f(\cdot)$  sem o cálculo intermediário dos polinômios FJ e GJ. Para tanto deve-se iteragir o modelo dado pela equação (2.8.3) com

uma decomposição especial de  $G$ . Este método foi apresentado por Favler em [FAV87b], e é denominado "previsão a múltiplos passos".

Neste método o polinômio  $G$  é decomposto na soma de dois termos como se indica a continuação

$$G = \Gamma = \overline{\Gamma}_J + z^{-J} \tilde{\Gamma}_J \quad (2.8.13)$$

com  $n_{\Gamma} = n_G$  e

$$\overline{\Gamma}_J = 1 + \Gamma_1 z^{-1} + \dots + \Gamma_{J-1} z^{-J+1} \quad (2.8.14)$$

$$\tilde{\Gamma}_J = \begin{cases} \Gamma_J + \Gamma_{J+1} z^{-1} + \dots + \Gamma_{n_{\Gamma}} z^{-n_{\Gamma}+J} & \text{para } J \leq n_{\Gamma} \\ 0 & \text{para } J > n_{\Gamma} \end{cases} \quad (2.8.15)$$

logo a equação (2.8.4) pode ser reescrita como:

$$\alpha y_f(t+J) = PB \Delta u(t+J-1) + \overline{\Gamma}_J \xi_f(t+J) + \tilde{\Gamma}_J \xi_f(t) \quad (2.8.16)$$

A melhor previsão da saída filtrada " $J$ " passos à frente é dada por:

$$\alpha \hat{y}_f(t+J/t) = PB \Delta u(t+J-1) + \tilde{\Gamma}_J \xi_f(t) \quad (2.8.17)$$

sendo  $\xi_f(t)$  obtido como:

$$\xi_f(t) = P \xi(t) = y_f(t) - \hat{y}_f(t/t-1) \quad (2.8.18)$$

Chamando de  $\beta$  o produto polinomial  $PB$  com  $n_{\beta} = n_P + n_B$  a equação (2.8.17) pode ser reescrita como:

$$\begin{aligned} \hat{y}_f(t+J/t) = & - \sum_{i=1}^{n_{\alpha}} \alpha_i \hat{y}_f(t+J-i/t) + \sum_{i=0}^{n_{\beta}} \beta_i \Delta u(t+J-i-1) + \\ & + \sum_{i=0}^{n_{\Gamma}-J} \Gamma_{i+J} \left[ y_f(t-i) - \hat{y}_f(t-i/t-1-i) \right] \end{aligned} \quad (2.8.19)$$

onde é utilizado

$$\hat{y}_f(t+j-1/t) = y_f(t+j-1) \quad \text{se } i \geq j \quad (2.8.20)$$

ou seja, são utilizados os valores da saída obtidos do processo (filtrados por P).

### 2.8.3.-DECOMPOSIÇÃO DA SAÍDA

Como se pode observar das equações (2.8.11) e (2.8.19), as previsões da saída dependem tanto do sinal de controle já exercido ( $j < i$ ,  $i=0..n/\beta$ ) como do sinal de controle a ser calculado até o instante "j". Baseado nessa separação pode-se assumir a previsão da saída como composta de duas partes, como se indica na figura (2.3) e na equação (2.8.21).

$$\hat{y}_f(t+j/t) = \hat{y}_{ff}(t+j/t) + \hat{y}_{fl}(t+j/t) \quad (2.8.21)$$

onde  $\hat{y}_{ff}(\cdot)$  é a "saída filtrada forçada", que é função dos futuros incrementos do sinal de controle a serem otimizados,

$$\Delta u(t), \dots, \Delta u(t+Nu).$$

e  $\hat{y}_{fl}(\cdot)$  é a "saída filtrada livre" que é função dos incrementos passados do sinal de controle

$$(\Delta u(t-1), \Delta u(t-2), \dots),$$

do estado do processo no instante "t"

$$(y_f(t), y_f(t-1), \dots)$$

e dos erros de previsão anteriores ao instante "t".

Para a obtenção das previsões destas duas saídas, no método baseado na identidade polinomial, recorre-se novamente a uma outra identidade polinomial dada por:

$$B F J = C H J + z^{-j} I J \quad (2.8.22)$$

com  $nhj = j-1$  e  $nij = \max(nb+nfj-1, nc-1)$ . Logo a equação (2.8.12) fica

$$\hat{y}_f(t+J/t) = \frac{GJ}{C} y(t) + HJ \Delta u(t+J-1) + \frac{IJ}{C} \Delta u(t-1) \quad (2.8.23)$$

Separando a parte da equação (2.8.23) que depende dos futuros incrementos do sinal de controle obtém-se:

$$\hat{y}_{ff}(t+J-1/t) = HJ \Delta u(t+J-1) \quad (2.8.24)$$

e

$$\hat{y}_{f1}(t+J/t) = \frac{GJ}{C} y(t) + \frac{IJ}{C} \Delta u(t-1) \quad (2.8.25)$$

Para a obtenção das previsões pelo método de "múltiplos passos", reescreve-se a equação (2.8.19) para os instantes  $J = 1$  até  $N2$ , ou seja até o horizonte máximo a que é necessária a previsão, como é feito a seguir:

$$\begin{aligned} \hat{y}_f(t+1/t) &= -\alpha_1 y_f(t) - \dots - \alpha_{n_\alpha} y_f(t-n_\alpha+1) + \\ &\quad + \beta_0 \Delta u(t) + \dots + \beta_{n_\beta} \Delta u(t-n_\beta) + \\ &\quad + \Gamma_1 \left[ y_f(t) - \hat{y}_f(t/t-1) \right] + \dots + \\ &\quad + \Gamma_{n_\Gamma} \left[ y_f(t-n_\Gamma+1) - \hat{y}_f(t-n_\Gamma+1/t-n_\Gamma) \right] \\ &\quad \vdots \\ \hat{y}_f(t+N2/t) &= -\alpha_1 \hat{y}_f(t+N2-1/t) - \dots - \alpha_{n_\alpha} y_f(t-n_\alpha+N2/t) + \\ &\quad + \beta_0 \Delta u(t+N2-1) + \dots + \beta_{n_\beta} \Delta u(t-n_\beta+N2-1) + \\ &\quad + \Gamma_{N2} \left[ y_f(t) - \hat{y}_f(t/t-1) \right] + \dots + \\ &\quad + \Gamma_{n_\Gamma} \left[ y_f(t-n_\Gamma+N2) - \hat{y}_f(t-n_\Gamma+N2/t-n_\Gamma+N2-1) \right] \end{aligned} \quad (2.8.26)$$

Separando destas previsões as partes livre e forçada, obtém-se o seguinte conjunto de previsões para a saída livre:

$$\begin{aligned}
\hat{y}_{f1}(t+1/t) &= -\alpha_1 y_f(t) - \dots - \alpha_n y_f(t-n\alpha+1) + \\
&\quad + \beta_1 \Delta u(t-1) + \dots + \beta_n \Delta u(t-n\beta) + \\
&\quad + \Gamma_1 \left[ y_f(t) - \hat{y}_f(t/t-1) \right] + \dots + \\
&\quad + \Gamma_n \left[ y_f(t-n\Gamma+1) - \hat{y}_f(t-n\Gamma+1/t-n\Gamma) \right] \\
&\quad \vdots \\
\hat{y}_{f1}(t+N2/t) &= -\alpha_1 \hat{y}_{f1}(t+N2-1/t) - \dots - \alpha_n \hat{y}_{f1}(t-n\alpha+N2/t) + \\
&\quad + \beta_{N2} \Delta u(t-1) + \dots + \beta_n \Delta u(t-n\beta+N2-1) + \\
&\quad + \Gamma_{N2} \left[ y_f(t) - \hat{y}_f(t/t-1) \right] + \dots + \\
&\quad + \Gamma_n \left[ y_f(t-n\Gamma+N2) - \hat{y}_f(t-n\Gamma+N2/t-n\Gamma+N2-1) \right]
\end{aligned} \tag{2.8.27}$$

e a previsão da saída filtrada forçada como:

$$\begin{aligned}
\hat{y}_{ff}(t+1/t) &= \beta_0 \Delta u(t) \\
\hat{y}_{ff}(t+2/t) &= -\alpha_1 \hat{y}_{ff}(t+1/t) + \beta_0 \Delta u(t+1) + \beta_1 \Delta u(t) \\
&\quad \vdots \\
\hat{y}_{ff}(t+N2/t) &= -\alpha_1 \hat{y}_{ff}(t+N2-1/t) - \dots - \alpha_n \hat{y}_{ff}(t-n\alpha+N2/t) + \\
&\quad + \beta_0 \Delta u(t+N2-1) + \dots + \beta_{N2-1} \Delta u(t)
\end{aligned} \tag{2.8.28}$$

As previsões da saída forçada  $\hat{y}_{ff}(t+j/t)$  podem ser reescritas com a ajuda das previsões dos instantes anteriores  $\hat{y}_{ff}(t+j-1/t)$ , assim

$$\begin{aligned}
\hat{y}_{ff}(t+2/t) &= \left[ \beta_1 - \beta_0 \alpha_1 \right] \Delta u(t) + \beta_0 \Delta u(t+1) \\
&= m_1 \Delta u(t) + m_0 \Delta u(t+1) \\
\hat{y}_{ff}(t+3/t) &= -\alpha_1 \left[ \left[ \beta_1 - \beta_0 \alpha_1 \right] \Delta u(t) + \beta_0 \Delta u(t+1) \right] - \\
&\quad - \alpha_2 \left[ \beta_0 \Delta u(t) \right] + \beta_2 \Delta u(t) + \beta_1 \Delta u(t+1) + \beta_0 \Delta u(t+2) \\
&= \left[ \beta_2 - \alpha_1 m_1 - \alpha_2 m_0 \right] \Delta u(t) + m_1 \Delta u(t+1) + m_0 \Delta u(t+2) \\
&= m_2 \Delta u(t) + m_1 \Delta u(t+1) + m_0 \Delta u(t+2) \\
&\quad \vdots \\
\hat{y}_{ff}(t+N2/t) &= m_{N2-1} \Delta u(t) + \dots + m_0 \Delta u(t+N2-1)
\end{aligned} \tag{2.8.29}$$

sendo

$$m_0 = \beta_0 = b_0 p_0$$

e

$$m_j = \beta_j - \sum_{i=1}^{\min(n, \alpha_j)} \alpha_i m_{j-i} \quad (2.8.30)$$

Da equação (2.8.30) pode-se verificar que os coeficientes  $m_i$  são as ordenadas da resposta ao impulso unitário de

$$B(z^{-1}) P(z^{-1}) / \alpha(z^{-1})$$

Obtidas as previsões das saídas filtradas livre e forçada, as equações anteriores podem ser reescritas em forma vetorial, para isto definem-se os vetores

$$\begin{aligned} \hat{\underline{y}}_f &= \begin{bmatrix} \hat{y}_f(t+1/t) & \dots & \hat{y}_f(t+N2/t) \end{bmatrix}^T \\ \hat{\underline{y}}_{f1} &= \begin{bmatrix} \hat{y}_{f1}(t+1/t) & \dots & \hat{y}_{f1}(t+N2/t) \end{bmatrix}^T \\ \hat{\underline{y}}_{ff} &= \begin{bmatrix} \hat{y}_{ff}(t+1/t) & \dots & \hat{y}_{ff}(t+N2/t) \end{bmatrix}^T \end{aligned} \quad (2.8.31)$$

$$\underline{\hat{\delta}}u = \begin{bmatrix} \Delta u(t) & \dots & \Delta u(t+N2-1) \end{bmatrix}^T \quad (2.8.32)$$

e a matriz

$$M = \begin{bmatrix} m_0 & & & 0 \\ & \ddots & & \\ & & \ddots & \\ m_{N2-1} & & & m_0 \end{bmatrix} \quad (2.8.33)$$

O vetor de previsões forçadas é dado na forma vetorial por:

$$\hat{\underline{y}}_{ff} = M \underline{\hat{\delta}}u \quad (2.8.34)$$

e a previsão total da saída filtrada é

$$\hat{\underline{y}}_f = M \underline{\hat{\delta}}u + \hat{\underline{y}}_{f1} \quad (2.8.35)$$

A última equação é utilizada para a obtenção dos incrementos futuros do sinal de controle otimizado. Deve-se observar que apesar da previsão da saída ter sido obtida utilizando o método de "previsão a múltiplos passos", as equações (2.8.34) e (2.8.35) também são aplicáveis no primeiro método, com a mesma definição de vetores anteriormente dada, sendo que os coeficientes  $h_{i+1}$  da equação (2.8.24) são os mesmos coeficientes  $m_i$  dados na equação (2.8.29).

#### 2.8.4.-FUNÇÃO DE CUSTO GENERALIZADA

O sinal de controle segundo a estratégia do Controlador Preditivo Generalizado (GPC) é obtido através da minimização da função de custo dada na equação (2.8.1), reescrita a seguir:

$$J = \varepsilon \left\{ \left[ \sum_{i=N_1}^{N_2} \gamma_i \phi_y^2(t+i) + \sum_{i=1}^{N_u} \lambda_i \phi_u^2(t+i-1) \right] / M_t \right\} \quad (2.8.1)$$

Nesta estratégia consideram-se as seguintes definições para  $\phi_y$  e  $\phi_u$

- i)  $\phi_y = y_{f1}(t) - y_m(t)$
  - ii)  $\phi_u = \Delta u(t)$
  - iii)  $\gamma_i = \gamma = 1$ , i.e., não ponderação no erro previsto.
  - iv)  $\lambda_i = \lambda$ , i.e., ponderação constante na excitação.<sup>8</sup>
- (2.8.36)

sendo

$$y_f(t) = P(z^{-1}) y(t) \quad (2.8.37)$$

a saída filtrada, e

$$y_m(t) = R(z^{-1}) w(t)$$

a referência filtrada.

---

<sup>8</sup> Embora  $\lambda$  possa ser variante no tempo, para acompanhar as variações dos parâmetros do processo e insensibilizar o cálculo do sinal de controle às variações do ganho, para o desenvolvimento do algoritmo ele é assumido constante.

iii)  $Nu \leq N2$ , esta restrição é equivalente a uma ponderação infinita nos incrementos do sinal de controle além de  $t+Nu-1$ . Desta forma são disponíveis somente  $Nu$  sinais de controle para levar a saída à referência especificada.

Assim o vetor de incrementos no sinal de controle é

$$\underline{\delta u} = \begin{bmatrix} \Delta u(t) & \dots & \Delta u(t+Nu-1) \end{bmatrix}^T \quad (2.8.38)$$

e a matriz  $M$  é modificada como segue:

$$M_L = \begin{bmatrix} m0 & & & \\ & \ddots & & \\ & & m0 & \\ mN2-1 & \dots & mN2-Nu & \end{bmatrix} \quad (2.8.39)$$

Desta forma o conjunto de previsões da saída filtrada em forma vetorial é dada por:

$$\hat{y}_f = M_L \underline{\delta u} + \hat{y}_{f1} \quad (2.8.40)$$

A função de custo a ser minimizada pode ser reescrita como:

$$J = E \left\{ \left[ \sum_{i=N1}^{N2} \left( y_f(t+i) - y_m(t+i) \right)^2 + \sum_{i=1}^{Nu} \lambda \left( \Delta u(t+i-1) \right)^2 \right] / M_t \right\} \quad (2.8.41)$$

Pelas propriedades da esperança matemática pode-se reescrever a equação anterior como:

$$J = \sum_{i=N}^{N2} E \left\{ \left( y_f(t+i) - y_m(t+i) \right)^2 / M_t \right\} + \sum_{i=1}^{Nu} E \left\{ \lambda \left( \Delta u(t+i-1) \right)^2 / M_t \right\} \quad (2.8.42)$$

Analisando o termo da primeira somatória, substituindo  $y_f$  pela soma do valor previsto mais a perturbação atuante no processo tem-se que:

$$\begin{aligned}
\varepsilon \left\{ \left[ y_f(t+1) - y_m(t+1) \right]^2 / \mathcal{M}t \right\} &= \varepsilon \left\{ \left[ \hat{y}_f(t+1/t) + F I \xi(t+1) - y_m(t+1) \right]^2 / \mathcal{M}t \right\} \\
&= \varepsilon \left\{ \left[ \left( \hat{y}_f(t+1/t) - y_m(t+1) \right)^2 - \right. \right. \\
&\quad \left. \left. - 2 F I \xi(t+1) \left[ \hat{y}_f(t+1/t) - y_m(t+1) \right] + \right. \right. \\
&\quad \left. \left. + \left[ F I \xi(t+1) \right]^2 \right] / \mathcal{M}t \right\} \\
&= \varepsilon \left\{ \left[ \hat{y}_f(t+1/t) - y_m(t+1) \right]^2 / \mathcal{M}t \right\} - \\
&\quad - \varepsilon \left\{ 2 F I \xi(t+1) \left[ \hat{y}_f(t+1/t) - y_m(t+1) \right] / \mathcal{M}t \right\} + \\
&\quad + \varepsilon \left\{ \left[ F I \xi(t+1) \right]^2 / \mathcal{M}t \right\}
\end{aligned}
\tag{2.8.43}$$

A equação (2.8.43) pode ser simplificada devido às seguintes hipóteses e considerações:

i) Como a perturbação atuando no processo é considerada não correlata com o sinal de controle, nem com a saída ou a referência pode-se concluir que:

$$\varepsilon \left\{ 2 F I \xi(t+1) \left[ \hat{y}_f(t+1/t) - y_m(t+1) \right] / \mathcal{M}t \right\} = 0 \tag{2.8.44}$$

ii) Como a ordem do polinômio  $F I$  é  $1-1$  obtém-se:

$$\varepsilon \left\{ \left[ F I \xi(t+1) \right]^2 / \mathcal{M}t \right\} = \varepsilon \left\{ \left[ F I \xi(t+1) \right]^2 \right\} \tag{2.8.45}$$

iii) Utilizando o valor de  $\hat{y}_f(\cdot)$  dado pela equação (2.8.23) obtém-se:

$$\varepsilon \left\{ \left[ \hat{y}_f(t+1/t) - y_m(t+1) \right]^2 / \Delta t \right\} =$$

$$\varepsilon \left\{ \left[ \frac{G1}{C} y(t) + H1 \Delta u(t+1-1) + \frac{11}{C} \Delta u(t-1) - y_m(t+1) \right]^2 / \Delta t \right\}$$

(2.8.46)

onde fica claro que todos os termos são conhecidos no instante "t", exceto  $H1 \Delta u(t+1-1)$ , mas ele é totalmente determinístico, e o mesmo pode-se dizer da segunda somatória da equação (2.8.42). Logo a função de custo a ser minimizada toma a seguinte forma:

$$J = \sum_{i=N}^{N2} \left[ \hat{y}_f(t+1/t) - y_m(t+1) \right]^2 + \sum_{i=1}^{Nu} \lambda \left[ \Delta u(t+1-1) \right]^2 + \varepsilon \left\{ \left[ F1 \xi(t+1) \right]^2 \right\}$$

(2.8.47)

Esta última equação foi obtida assumindo que as previsões da saída foram obtidas pelo método da equação diofantina. No caso da previsão a múltiplos passos chega-se a um resultado similar dado pela seguinte equação:

$$J = \sum_{i=N}^{N2} \left[ \hat{y}_f(t+1/t) - y_m(t+1) \right]^2 + \sum_{i=1}^{Nu} \lambda \left[ \Delta u(t+1-1) \right]^2 + \varepsilon \left\{ \left[ \frac{\overline{F1}}{\alpha} \xi(t+1) \right]^2 \right\}$$

(2.8.47')

Para a minimização assume-se, por simplicidade,  $N1 = 1$ . Definindo os vetores

$$\underline{F\xi} = \left[ F1\xi(t+1) \dots FN2\xi(t+N2) \right]^T$$

(2.8.48)

ou

$$\underline{F\xi} = \left[ \frac{\overline{F1}}{\alpha} \xi(t+1) \dots \frac{\overline{FN2}}{\alpha} \xi(t+N2) \right]^T$$

(2.8.48')

e

$$\underline{y}_m = \left[ R w(t+1) \dots R w(t+N2) \right]^T$$

(2.8.49)

Se as futuras referências são conhecidas, como acontece em

certos tipos de processos a referência denomina-se "pré-especificada", caso contrário, assume-se que as futuras referências são iguais à atual, isto é

$$w(t+1) = w(t)$$

$$N1 \leq i \leq N2$$

denominando-se como referência "normal".

Com os vetores assim definidos pode-se reescrever a função de custo em forma vetorial como:

$$J = \left[ \hat{y}_f - y_m \right]^T \cdot \left[ \hat{y}_f - y_m \right] + \lambda \underline{\hat{u}}^T \underline{\hat{u}} + \varepsilon \left\{ \underline{F}^T \underline{F} \right\} \quad (2.8.50)$$

Substituindo o vetor de previsões da saída pela equação (2.8.40) obtém-se o seguinte critério a ser minimizado:

$$J = \left[ M_k \underline{\hat{u}} + \hat{y}_{f1} - y_m \right]^T \cdot \left[ M_k \underline{\hat{u}} + \hat{y}_{f1} - y_m \right] + \lambda \underline{\hat{u}}^T \underline{\hat{u}} + \varepsilon \left\{ \underline{F}^T \underline{F} \right\} \quad (2.8.51)$$

A minimização do critério J resulta em:

$$\frac{d J}{d \underline{\hat{u}}} = 2 M_k^T M_k \underline{\hat{u}} + 2 M_k^T \left[ \hat{y}_{f1} - y_m \right] + 2 \lambda \underline{\hat{u}} + \frac{\partial}{\partial \underline{\hat{u}}} \varepsilon \left\{ \underline{F}^T \underline{F} \right\} = 0 \quad (2.8.52)$$

como

$$\frac{\partial}{\partial \underline{\hat{u}}} \varepsilon \left\{ \underline{F}^T \underline{F} \right\} = 0 \quad (2.8.53)$$

logo a equação (2.8.52) torna-se:

$$2 M_k^T M_k \underline{\hat{u}} + 2 M_k^T \left[ \hat{y}_{f1} - y_m \right] + 2 \lambda \underline{\hat{u}} = 0 \quad (2.8.54)$$

O vetor de incrementos futuros no sinal de controle é dado por:

$$\underline{\Delta u} = \left[ M_k^T M_k + \lambda I \right]^{-1} M_k^T \left[ y_m - \hat{y}_{f1} \right] \quad (2.8.55)$$

Nesta equação "I" é a matriz identidade de ordem  $(N_u \times N_u)$ . Sendo o GPC uma estratégia de controle de deslocamento de horizonte ("receding horizon"), isto é, só é exercido a cada iteração o primeiro dos sinais de controle da sequência computada pela equação (2.8.55), é necessário calcular somente o primeiro elemento do vetor  $\underline{\Delta u}$ . Para isto define-se a matriz V como:

$$V = \left[ M_k^T M_k + \lambda I \right]^{-1} M_k^T \quad (2.8.56)$$

Sendo  $v_1$  a primeira linha da matriz V, obtém-se o incremento no sinal de controle no instante seguinte, a partir da equação (2.8.57), isto é:

$$\Delta u(t) = v_1^T \left[ y_m - \hat{y}_{f1} \right] \quad (2.8.57)$$

Da equação (2.8.55) pode-se verificar que com  $N_u < N_2$  reduz-se o esforço computacional necessário para o cálculo do sinal de controle, diminuindo o tempo de inversão da matriz. Quando o horizonte de controle é igual a 1 obtém-se um escalar. O coeficiente  $\lambda \neq 0$  além de ponderar o sinal de controle, garante a inversão da matriz quando que  $M_k^T M_k$  é singular.

#### 2.8.5.-FUNÇÃO DE TRANSFERÊNCIA EM MALHA FECHADA

Denominando cada elemento da matriz V como  $v_{ij}$ , a equação (2.8.57) pode ser reescrita como segue:

$$\Delta u(t) = \sum_{i=1}^{N_2} v_{1i} \left[ y_m(t+1) - \hat{y}_{f1}(t+1/t) \right] \quad (2.8.59)$$

Substituindo  $y_m(\cdot)$  pelo valor dado na equação (2.8.37),  $\hat{y}_{f1}(\cdot)$  pelo valor obtido na equação (2.8.25), e multiplicando ambos membros da equação anterior por  $C(z^{-1})$  tem-se:

$$C \Delta u(t) = \sum_{i=N_1}^{N_2} v_{1i} \left[ RC w(t+1) - GI y(t) - II \Delta u(t-1) \right] \quad (2.8.60)$$

Considerando a referência "normal", a equação anterior pode ser reescrita como:

$$\left[ RC \sum_{i=N_1}^{N_2} v_{1i} \right] w(t) - \left[ \sum_{i=N_1}^{N_2} v_{1i} GI \right] y(t) - \left[ C + z^{-1} \sum_{i=N_1}^{N_2} v_{1i} II \right] \Delta u(t) = 0 \quad (2.8.61)$$

ou descrito por:

$$Geq y(t) + Heq \Delta u(t) - Eeq w(t) = 0 \quad (2.8.62)$$

onde

$$\begin{aligned} Geq &= \left[ \sum_{i=N_1}^{N_2} v_{1i} GI \right] \\ Heq &= \left[ C + z^{-1} \sum_{i=N_1}^{N_2} v_{1i} II \right] \\ Eeq &= \left[ RC \sum_{i=N_1}^{N_2} v_{1i} \right] \end{aligned} \quad (2.8.63)$$

Procedendo de forma similar ao realizado no caso do GMV substitui-se o sinal de controle calculado pela equação (2.8.62) no modelo do processo dado pela equação (2.2.14), e obtém-se:

$$\alpha Heq y(t) = - z^{-1} B Geq y(t) + z^{-1} B Eeq w(t) + C Heq \xi(t) \quad (2.8.64)$$

Resulta a seguinte equação em malha fechada:

$$y(t) = \frac{z^{-1} B Eeq w(t)}{\alpha Heq + z^{-1} B Geq} + \frac{C Heq \xi(t)}{\alpha Heq + z^{-1} B Geq} \quad (2.8.65)$$

Analisa-se a seguir o polinômio característico do sistema descrito pela equação (2.8.65), que determina a dinâmica do sistema em malha fechada.

$$\alpha H_{eq} + z^{-1} B G_{eq} = \alpha \left[ C + z^{-1} \sum_{i=N_1}^{N_2} v_{1i} \mid \mid \right] + z^{-1} B \sum_{i=N_1}^{N_2} v_{1i} G \mid \quad (2.8.66)$$

Substituindo  $G \mid$  pelo valor dado pela equação (2.8.5) obtém-se:

$$\alpha H_{eq} + z^{-1} B G_{eq} = \alpha \left[ C + z^{-1} \sum_{i=N_1}^{N_2} v_{1i} \mid \mid \right] + z^{-1} B \sum_{i=N_1}^{N_2} v_{1i} z^i (P C - \alpha F \mid) \quad (2.8.67)$$

e substituindo  $B F \mid$  pela equação (2.8.22) pode se escrever que:

$$\begin{aligned} \alpha H_{eq} + z^{-1} B G_{eq} &= \alpha C + \alpha z^{-1} \sum_{i=N_1}^{N_2} v_{1i} \mid \mid + z^{-1} B \sum_{i=N_1}^{N_2} v_{1i} z^i P C - \\ &\quad - \alpha C z^{-1} \sum_{i=N_1}^{N_2} v_{1i} z^i H \mid - \alpha z^{-1} \sum_{i=N_1}^{N_2} v_{1i} z^i z^{-i} \mid \mid \\ \alpha H_{eq} + z^{-1} B G_{eq} &= P B C \sum_{i=N_1}^{N_2} v_{1i} z^{i-1} + \alpha C \left[ 1 - \sum_{i=N_1}^{N_2} v_{1i} z^{i-1} H \mid \right] \end{aligned} \quad (2.8.68)$$

Esta é a equação do polinômio característico da função de transferência em malha fechada. Pode-se verificar que é uma função complexa dos horizontes  $N_1$ ,  $N_2$  e  $N_u$  porque os parâmetros  $v_{ij}$  dependem destes horizontes. Porém, sob certas condições este polinômio pode ser simplificado. Estas condições, principalmente como escolhas dos horizontes, são indicadas a seguir:

- $N_1 = 1,$
- $N_2 = \text{Qualquer},$
- $N_u = N_2,$
- $\lambda = 0,$

Nestas condições a matriz  $M$  é a matriz quadrada ( $N2 \times N2$ )  $M$  (equação (2.8.33)), pós-multiplicando a matriz  $V$  também quadrada ( $N2 \times N2$ ) vê-se que:

$$V M = \left[ M^T M \right]^{-1} M^T M = I \quad (2.8.69)$$

Resolvendo o sistema de equações formado pelo produto  $V.M$  obtém-se a primeira fila da matriz  $V$  como:

$$v_1 = \begin{bmatrix} v_{11} & v_{12} & \dots & v_{1N2} \end{bmatrix} = \begin{bmatrix} \frac{1}{m_0} & 0 & \dots & 0 \end{bmatrix}$$

e como  $m_0 = \beta_0$  tem-se:

$$v_1 = \begin{bmatrix} \frac{1}{\beta_0} & 0 & \dots & 0 \end{bmatrix} \quad (2.8.70)$$

Assim a parte determinística da equação (2.8.65) pode ser reescrita como:

$$y(t) = \frac{C B R w(t-1) (1/\beta_0)}{P B C (1/\beta_0) + \alpha C (1-h_1/\beta_0)} \quad (2.8.71)$$

Das equações 2.8.24 e 2.8.34 pode-se verificar que os coeficientes  $h_{i+1}$  correspondem aos  $m_i$ . Assim, tem-se que  $h_1 = \beta_0$ , e a equação (2.8.71) é dada por:

$$y(t) = \frac{R}{P} w(t-1) \quad (2.8.72)$$

Nesta última equação fica claro que (sob as condições impostas) o filtro  $R/P$  deve ser escolhido como o modelo desejado em malha fechada, e que o GPC torna-se um controlador do tipo "dead-beat".

## 2.8.6.-RUIDO COLORIDO E PERTURBAÇÕES NA CARGA

Como no caso do algoritmo GMV, a presença do filtro T na estimação dos parâmetros do processo, também pode ser incluído no cálculo do sinal de controle. Como no caso do GMV, o algoritmo pode ser desenvolvido a partir da modificação da identidade polinomial (2.8.5). Entretanto como o cálculo é feito com base na previsão a múltiplos passos, modifica-se o modelo a ser iteragido, ficando o modelo filtrado dado por:

$$\alpha y'_f(t+j) = PB \Delta u'(t+j-1) + P \xi'(t+j) \quad (2.8.73)$$

onde o acento (') indica a quantidade filtrada por  $1/T$ . Esta equação pode ser decomposta nas partes forçada e livre, sendo que as previsões da saída livre são obtidas pela iteração do modelo (2.8.73) desde o instante atual filtrado<sup>p</sup>

$$[ y'_f(t) \ y'_f(t-1) \ \dots \ \Delta u'(t-1) \ \Delta u'(t-2) \ \dots ]$$

com os futuros incrementos no sinal de controle igual a zero (passados pelo filtro  $1/T$ ) até o horizonte máximo  $N2$ . A matriz  $M$  permanece inalterada, já que os coeficientes desta matriz são obtidos como a resposta ao impulso do modelo  $PB/\alpha$ , que independe do filtro  $T(z^{-1})$ .

Para se obter as previsões livres reais, procede-se a filtrar as previsões livres (filtradas) como se indica a seguir

$$\begin{aligned} \hat{y}_{f1}(t+1/t) &= t_0 \hat{y}'_{f1}(t+1/t) + t_1 y'_f(t) + \dots + t_{n1} y'_f(t-nt+1) \\ \hat{y}_{f1}(t+2/t) &= t_0 \hat{y}'_{f1}(t+2/t) + t_1 \hat{y}'_{f1}(t+1/t) + \dots + t_{n1} y'_f(t-nt+2) \\ &\vdots \qquad \qquad \qquad \vdots \end{aligned}$$

até o instante " $t+N2$ ". O cálculo do sinal de controle continua sendo realizado utilizando a equação (2.8.58), sendo que os pólos em malha fechada não são modificados pela inclusão do filtro T.

---

<sup>p</sup>O "f" está indicando que a saída já foi filtrada com o filtro F.

## 2.9.-ALGORITMOS DE CONTROLE E IDENTIFICAÇÃO

Os algoritmos implementados têm as seguintes funções: Identificação, Seguimento de ruptura de Modelo, Controle PID, Controle GMV, Controle GPC. Estes algoritmos são descritos em continuação utilizando-se a linguagem de pseudocódigo apresentada no capítulo 1.

1).-Algoritmo de Identificação: o algoritmo implementado foi o estimador de matriz estendida descrito na seção 2.3.3.

PROCEDIMENTO EME( $\theta$ );

COMEÇO

ATUALIZAR do vetor de medidas ( $\varphi$ ) com  $y(t)$ ,  $u(t-1)$  e  $\hat{\theta}(t)$ ;

CALCULAR  $K(t)$ ;

CALCULAR  $P'(t)$ ;

SE o Seguimento de Ruptura de Modelo está ativado

ENTÃO  $SRM(\lambda_0)$ ;

$P(t) := P'(t)/\lambda_0$ ;

CALCULAR

FIM;

PSC. 2.1 - Algoritmo de Identificação

11).-Algoritmo de Seguimento de Ruptura: Foi implementado o método do erro previsto, apresentado na seção 2.4.2.

PROCEDIMENTO  $SRM(\lambda_0)$ ;

COMEÇO

CALCULAR Sigma-Curto;

CALCULAR Sigma-Longo;

CALCULAR índice de desempenho  $J(t)$ ;

CALCULAR Traço-da-Matriz-P;

SE  $J(t) \leq J\text{-mínimo}$

ENTÃO Traço-auxiliar := Traço-mínimo;

SENÃO SE  $J\text{-mínimo} \leq J(t) < J\text{-máximo}$

ENTÃO Traço-auxiliar := Traço-médio;

SENÃO SE  $J(t) \geq J\text{-máximo}$

ENTÃO Traço-auxiliar := Traço-máximo;

SE Traço-da-Matriz-P  $\geq$  Traço-auxiliar

ENTÃO  $\lambda_0 := \lambda_0\text{-seguimento-variações-lentas}$ ;

SENÃO  $\lambda_0 := \text{Traço-da-Matriz-P} / \text{Traço-auxiliar}$ ;

FIM;

PSC. 2.2 - Algoritmo de seguimento de ruptura de modelo

III).-Controlador GMV: permite o cálculo do sinal de controle em forma clássica ou por alocação de pólos.

PROCEDIMENTO GMV(P, Q, R, S, d, T,  $\Delta u(t)$ );

COMEÇO

OBTER  $y(t)$  do processo;

OBTER  $y'(t)$ ,  $\Delta u'(t-1)$  e  $w'(t)$  filtrando  $y(t)$ ,  $\Delta u(t-1)$  e  $w(t)$  com o filtro T;

SE a identificação está ativada

ENTÃO EME( $\hat{\theta}$ );

CALCULAR  $\alpha(z^{-1}) := \Delta \hat{A}(z^{-1})$ ;

SE a Alocação-de-Pólos está ativada

ENTÃO,

COMEÇO

RESOLVER  $\alpha P + B Q := S$

CALCULAR  $R(z^{-1})$  de  $P(z^{-1})$  (  $r_0 := P(1)$  );

FIM

RESOLVER  $F \alpha + z^{-d} G := P C$ ;

CALCULAR  $H := F B + Q C$ ;

CALCULAR  $E := - R C$ ;

CALCULAR  $\Delta u'(t) := - \frac{G y'(t) + E w'(t)}{H}$ ;

CALCULAR  $\Delta u(t) := \frac{\Delta u'(t)}{T}$ ;

FIM;

PSC. 2.3 - Algoritmo do controlador GMV

III).-Controlador PID:

PROCEDIMENTO PID(KP, KI, KD,  $\Delta u(t)$ );

COMEÇO

OBTER  $y(t)$  do processo;

CALCULAR  $e(t) := w(t) - y(t)$ ;

CALCULAR  $\Delta u(t) := KP \left[ \Delta e(t) + KI e(t) + KD \Delta^2 e(t) \right]$ ;

FIM;

PSC. 2.4 - Algoritmo do controlador PID

v).-Controlador GPC: O cálculo das previsões livres faz-se pelo método iterativo.

PROCEDIMENTO GPC(N1, N2, Nu, P, R,  $\lambda$ , T,  $\Delta u(t)$ );

COMEÇO

OBTER  $y(t)$  do processo;  
 OBTER  $y'(t)$  e  $\Delta u'(t-1)$  filtrando  $y(t)$ ,  $\Delta u(t-1)$  com o filtro T;  
 SE a identificação está ativada

ENTÃO EME( $\hat{\theta}$ );

CALCULAR  $\alpha(z^{-1}) := \Delta \hat{A}(z^{-1})$ ;

CALCULAR  $\beta(z^{-1}) := P(z^{-1}) \hat{B}(z^{-1})$ ;

PARA I := 0 ATÉ N2 - 1 FAÇA

CALCULAR  $m_i$ ; (equação 2.8.30)

CALCULAR  $v_1$ ; ( $V = (M^T M + \lambda I)^{-1} M^T$ )

PARA := 1 ATÉ N2 FAÇA

CALCULAR  $\hat{y}'_{f1}(t+1/t)$ ; (equação 2.8.27)

CALCULAR  $\hat{y}_{f1} := \hat{y}'_{f1} / T$ ;

CALCULAR  $y_m := w R(z^{-1})$ ;

CALCULAR  $\Delta u(t) := v_1^T \left[ y_m - \hat{y}_{f1} \right]$ ;

FIM;

PSC. 2.5 - Algoritmo do controlador GPC

Outros algoritmos implementados coletam os índices de desempenho que são testados pelo Sistema Especialista; Como a sua estrutura é simples, eles não são apresentados.

## 2.10.-RESUMO

Neste capítulo foram apresentados os algoritmos de identificação e controle utilizados no sistema. Sendo que os algoritmos de identificação são muito conhecidos, não foi apresentado o desenvolvimento destes. Entretanto os algoritmos de controle GMV, e GPC apresentam algumas diferenças com relação ao descritos na literatura, razão pela qual eles foram explicados mais em detalhe.

Os algoritmos apresentados revelam que quanto mais poderosos, maior quantidade de parâmetros que necessitam ser ajustados, consumindo em geral maior tempo de cálculo. Este último problema vem sendo eliminado dia a dia pelo surgimento de dispositivos de computação mais poderosos. Entretanto o problema do ajuste de parâmetros é relativamente independente das tecnologias disponíveis. Assim a aplicação de Sistemas Especialistas para a resolução dos problemas de controle adaptativo aparece como um enfoque natural.

## **CAPÍTULO**

### **III**

## **SISTEMA ESPECIALISTA EM CONTROLE**

### 3.1-INTRODUÇÃO

No capítulo 1 foi apresentada a estrutura dos *Sistemas Especialistas*, enfatizando-se os que utilizam a *representação do conhecimento* na forma de *regras de produção*. Esta ênfase deve-se a que, no sistema implementado este tipo de representação do conhecimento é intensamente utilizada. A outra representação de conhecimento usada neste sistema, baseada em procedimentos, é o suporte dos algoritmos de *identificação e controle*.

O sistema base deste trabalho é um sistema de controle com ajuda especialista nas tarefas de configuração e supervisão do mesmo. O sistema deve executar as seguintes tarefas:

- 1).-Identificação do modelo do processo
  - 1a).-Identificador
  - 1b).-Detecção de não estacionariedade
  - 1c).-Configuração de dados para a tarefa 4a
- 2).-Controle do processo
  - 2a).-PID, GMV, GPC
  - 2b).-Configuração de dados para a tarefa 4b
- 3).-Configuração do sistema de controle
  - 3a).-Inicialização geral
  - 3b).-Configuração do identificador
    - 3ba).-do algoritmo de identificação
    - 3bb).-do algoritmo de detecção de não estacionariedades
  - 3c).-Escolha e Configuraração do algoritmo de controle
- 4).-Supervisão do sistema de controle
  - 4a).-Supervisão do identificador
  - 4b).-Supervisão do controlador

Para se implementar os módulos que devem realizar as tarefas 3 e 4, foi necessário inicialmente escolher uma *representação do conhecimento* e depois as estratégias de *manipulação do conhecimento* (tipo de inferência). A escolha feita é justificada na seção 3.2.

Como os sistemas baseados em conhecimento são fortemente influenciados pela linguagem da implementação, na seção 3.3. é definida a linguagem utilizada no sistema.

Na seção 3.4 são descritas as estruturas de armazenamento do conhecimento, e na seção 3.5 são apresentadas as estruturas que manipulam este conhecimento.

A seção 3.6. discorre sobre a interface Homem-Máquina utilizada. Finalmente na seção 3.7 é apresentado um resumo do capítulo.

## 3.2-REPRESENTAÇÃO DO CONHECIMENTO

Nos últimos anos aumentou-se o interesse no estudo dos *Sistemas Integrados de Informação e Controle* [GOM86]. Nestes sistemas podem-se distinguir vários níveis hierárquicos de informação e controle de acordo com os objetivos, escopo e escalas de tempo das tarefas a serem realizadas em cada um destes níveis (figura 3.1).

| NÍVEIS                    | OBJETIVO                  | ESCOPO                 | TEMPO              |
|---------------------------|---------------------------|------------------------|--------------------|
| 5 GERENCIAMENTO           | GESTÃO DA EMPRESA         | OPERAÇÃO DA EMPRESA    | ANOS E MESES       |
| 4 PLANEJAMENTO            | GERENCIAMENTO DA FÁBRICA  | OPERAÇÃO DA FÁBRICA    | MESES E SEMANAS    |
| 3 COORDENAÇÃO             | GERENCIAMENTO DO PROCESSO | OPERAÇÃO DAS ÁREAS     | SEMANAS E DIAS     |
| 2 SUPERVISÃO              | SUPERVISÃO DOS PROCESSOS  | OPERAÇÃO DAS UNIDADES  | HORAS E MINUTOS    |
| 1 CONTROLE DIGITAL DIRETO | CONTROLE DOS PROCESSOS    | OPERAÇÃO DOS PROCESSOS | MINUTOS E SEGUNDOS |

FIG 3.1 Níveis hierárquicos de informação e controle

A utilização de sistemas baseados em conhecimento pode e tem sido feita em todos esses níveis hierárquicos, porém o

conhecimento adequado para a otimização das tarefas de cada nível tem características particulares que devem ser levadas em conta na escolha de uma determinada representação do conhecimento. A primeira diferença é devida à forma em que o conhecimento é descrito pelos "especialistas humanos" existentes nos diferentes níveis, nos primeiros voltados à ciências exatas, e nos últimos, voltados à organização empresarial, com o consequente "aprendizado" de diferentes formas de representação do conhecimento. Outra diferença ocorre porque os objetos<sup>1</sup> considerados nos distintos níveis possuem características diferentes, pelas funções que desempenham dentro do sistema, e pela complexidade e relações com os outros objetos (do seu nível e dos outros). Outra característica a ser considerada na escolha da representação do conhecimento é o tempo que o sistema pode esperar por uma resposta, que depende dos tempos das tarefas de cada nível.

As tarefas a serem realizadas no sistema proposto são, conforme listado na seção 3.1, de quatro tipos: Identificação, Controle, Configuração e Supervisão. As duas primeiras tarefas são encontradas no primeiro nível (Controle Digital Direto) na hierarquia dos sistemas integrados, enquanto que as duas seguintes estão no segundo nível, denominado de Supervisão [GOMB5].

Analisando estes dois níveis verifica-se que a representação do conhecimento deve possuir as seguintes características:

- I).-Representar objetos "simples" encontrados no nível de Controle Digital Direto.
- II).-Operar com a representação dos objetos do nível de Controle Digital Direto, para realizar funções de otimização; função encontrada no nível de Supervisão.

---

<sup>1</sup>Por objetos entende-se as variáveis observadas e controladas no nível específico considerado.

iii).-Representação em uma linguagem semelhante à dos especialistas em controle.

iv).-Eficiente para obter uma resposta confiável em tempo adequado.

v).-Facilidade de incorporação e modificação do conhecimento sem grandes alterações do sistema.

A representação do conhecimento que mais se aproxima das características citadas é a de *Regras de Produção* (capítulo 1). Embora possa-se arguir da eficiência quanto ao tempo de resposta dessa representação, as regras podem ser "compiladas"<sup>2</sup> para se obter maior velocidade, necessidade esta já claramente reconhecida na literatura ([GAR88, SAN86, WR186]). Este tipo de representação favorece a aquisição do conhecimento do especialista porque na área de automação é comum que as pessoas possuam o conhecimento na forma de regras em linguagem natural. A representação por regras também é uma escolha adequada quando se deseja uma representação com maior poder procedural que declarativo, isto é, quando se deseja simular as respostas dos especialistas quando ocorrem mudanças no ambiente [HER88].

### 3.2.1-ESTRATÉGIAS DE MANIPULAÇÃO DO CONHECIMENTO

Para realizar as tarefas supervisão partiu-se do trabalho realizado por Arruda [ARR88], que desenvolveu um *supervisor para modelagem de processos* baseado em uma máquina de inferência com encadeamento direto.

A escolha da estratégia de inferência por encadeamento direto para as tarefas do item 4 (seção 3.1), Justifica-se devido a que a tarefa de supervisão é inerentemente uma tarefa de diagnóstico. Embora ambos tipos de encadeamentos possam ser aplicados para

---

<sup>2</sup>Um modo simples de compilação de regras é dado na forma de matrizes, estrutura largamente utilizada nos sistemas com lógica difusa [INA85, SHABAI].

realizar esta tarefa, no encadeamento reverso deve-se "assumir um diagnóstico provável", e comprovar se a Base de Dados confirma-o. O encadeamento direto trabalha em forma mais natural, analisando os fatos e tirando conclusões (obtendo os diagnósticos) destes fatos.

Para realizar as tarefas de configuração (item 3 seção 3.1), como no caso analisado anteriormente, podem ser utilizados quaisquer dos dois tipos de encadeamento mencionados (e ainda também o misto). Porém, a falta de um dado na cadeia de inferência provocará a solicitação do seu valor, a um procedimento ou ao usuário. A forma como é realizada a solicitação depende do tipo de máquina de inferência utilizada.

No encadeamento reverso toda regra que forma parte do conjunto de discórdia tem o seu consequente como a conclusão requerida para satisfazer a "meta". Desta forma os objetos que formam o antecedente destas regras são úteis para a solução do problema.

No caso do encadeamento direto, para se chegar à solução deve-se formar o conjunto de discórdia com todas as regras que tem os seus antecedentes satisfeitos, (ou com possibilidades de serem satisfeitos pelos dados obtidos dos procedimentos ou fornecidos pelo usuário). Neste caso, quando os consequentes das regras do conjunto de discórdia não satisfazem a "meta" procurada, os objetos dos antecedentes desta regras provavelmente não sejam úteis para a solução do problema.

Pelo exposto anteriormente optou-se por uma máquina com encadeamento reverso para as tarefas de configuração dos dados (denominadas genericamente tarefas de "configuração").

### 3.3.-ESCOLHA DA LINGUAGEM DE IMPLEMENTAÇÃO

Na seção 3.2.1 verificou-se a necessidade de serem implementadas duas máquinas de inferência para as tarefas "inteligentes" do sistema. Porém a simplicidade nestas implementações depende em grande parte da linguagem escolhida. Como já foi dito, dispunha-se de um supervisor baseado em conhecimento com encadeamento direto, que tinha sido implementado na linguagem PASCAL. Pelas facilidades numéricas, o PASCAL é extremamente útil para a implementação de sistemas com grande necessidade de cálculo. Embora possam ser construídas nesta linguagem máquinas de inferência com qualquer tipo de encadeamento, ela possui algumas desvantagens e limitações que serão consideradas a seguir.

1).-PASCAL não é uma linguagem adequada para o processamento de estruturas da linguagem natural, como as encontradas nas regras de produção fornecidas pelos especialistas. Isto obriga a se dispor de uma rede de tradução do conhecimento representado na figura 3.2.

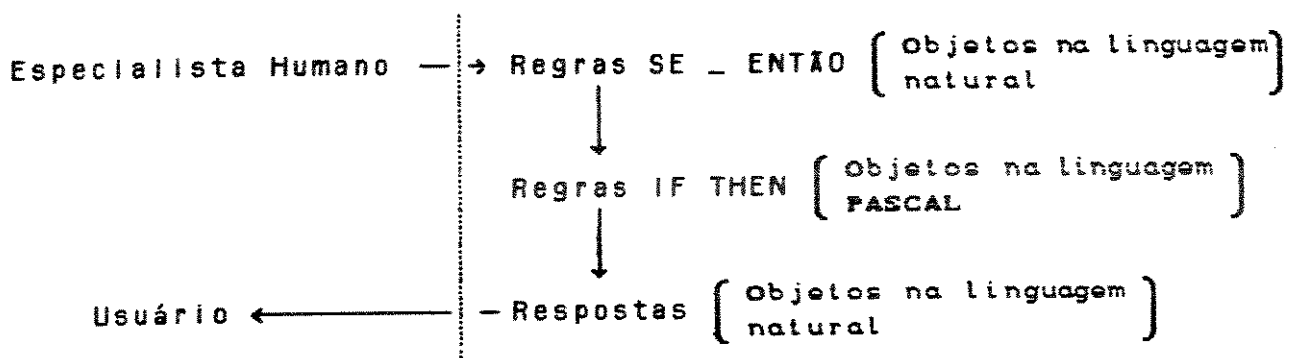


FIG 3.2 Rede de Traduções de Representação

Estas traduções além de aumentar apreciavelmente o tamanho do sistema, já que o conhecimento fica praticamente duplicado (ver [ARR88, TH085]), diminuem a "flexibilidade" do sistema, pois ao se modificar uma regra, deve-se ter o cuidado de fazê-lo na tradução,

Isto é, todo o conhecimento num nível deve ser a imagem do outro nível<sup>3</sup>.

II).-A implementação de sistemas baseados em conhecimento por regras de produção com inferência realizada por encadeamento direto, é facilitada com a utilização desta linguagem por que as estruturas "SE \_ ENTÃO \_" têm tradução direta à estruturas IF-THEN da linguagem PASCAL ([ARR88, RED84]). Esta facilidade não existe na implementação de máquinas de inferência com encadeamento reverso, necessitando-se da utilização de apontadores e estruturas tipo pilha [TH085]. Com a linguagem PASCAL para o sistema poder-se-ia ter representações diferentes das regras no encadeamento direto e no encadeamento reverso, com o consequente aumento na complexidade do sistema.

III).-Pelas características do sistema, os objetos utilizados são simples, mas de características variadas (ver seção 3.4.1). Neste caso, se as estruturas de armazenamento de conhecimento fossem implementadas na linguagem PASCAL elas serão seguramente complexas.

Devido aos problemas apresentados optou-se pela utilização de uma combinação das linguagens PROLOG e PASCAL, que fornecem as seguintes facilidades:

I).- PROLOG é uma linguagem não tipada. Assim os objetos definidos podem ser armazenados em forma similar. Os objetos definidos em linguagem natural, podem ser tratados pelo sistema sem traduções, diminuindo as repetições do conhecimento.

II).-Como PROLOG não é uma linguagem adequada para operações matemáticas, optou-se por realizar a parte numérica (identificação e controle) na linguagem PASCAL.

---

<sup>3</sup> Esta característica, deve também ser mantida entre os fatos da Base de Dados e as variáveis a que eles se referem nos algoritmos de identificação e controle [WRIB83].

iii).-A combinação entre estas duas linguagens foi favorecida pelas facilidades previstas pelo PROLOG de ter interface com PASCAL, e pela estrutura quase independente das funções realizadas, que podem ser resumidas na seguinte figura na qual se mostra o *Ciclo de Consulta* ao sistema.

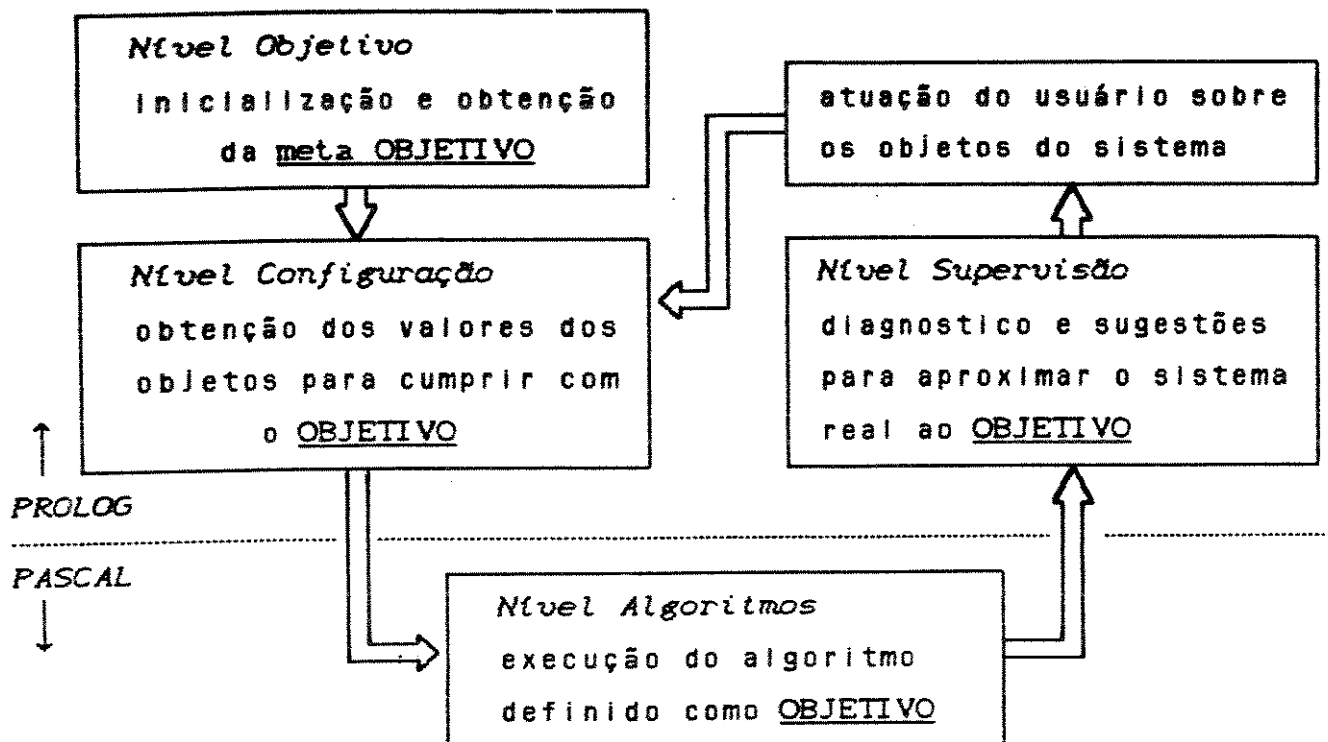


FIG 3.3 Ciclo de Consulta

### 3.4.-ESTRUTURAS DE ARMAZENAMENTO DO CONHECIMENTO

No capítulo I, ao serem apresentadas as partes constituintes dos Sistemas Especialistas, destacaram-se duas estruturas que têm como função o armazenamento do conhecimento. A Base de Conhecimento onde é armazenado o conhecimento do domínio específico da aplicação (i.e., como realizar o controle e a identificação), e a Base de Dados onde é armazenado o conhecimento sobre as características particulares da implementação e evolução do sistema (processo + identificador + controlador). A implementação destas duas estruturas é detalhada a seguir.

### 3.4.1.-ESTRUTURA DA BASE DE DADOS

No sistema implementado, como na maioria dos sistemas de controle, os objetos sobre os quais se deve fazer as inferências têm estruturas simples. Como pode ser observado nas equações dadas no capítulo II, os parâmetros que o usuário deve fornecer ao sistema, têm como estrutura mais complexa os polinômios. Embora os objetos a serem tratados possuam estruturas simples, na implementação do sistema observou-se que eles eram de características variadas, como pode se observado na tabela 3.1.

| OBJETO                         | TIPO       | LIMITE/VALORES           |
|--------------------------------|------------|--------------------------|
| a ordem do polinômio A         | inteiro    | 0 $\longrightarrow$ 10   |
| o polinômio do filtro Pf       | polinômio  | [Pfo,Pf1,...,Pfnp]       |
| o identificação do modelo      | lógica     | ativada,desativada       |
| o tipo de controlador desejado | multivalor | PID, GMV, GPC            |
| o fator de esquecimento        | real       | 0.98 $\longrightarrow$ 1 |
| os parâmetros do modelo        | não-tipado | Qualquer                 |

TABELA 3.1 : OBJETOS e TIPOS do sistema

No capítulo I foi ressaltado que uma característica desejável da Base de Dados, é que a estrutura dos fatos armazenados seja compatível com a estrutura das regras, e próximo da linguagem natural. Esta condição evita as traduções entre as representações, e conseqüentemente permite maior velocidade na verificação da validade das regras, embora seja requerida uma maior quantidade de memória para armazenar os fatos. Por se ter escolhido a linguagem PROLOG, esta estrutura foi simplificada, tendo todos os fatos a mesma estrutura, independentemente da complexidade do tipo do objeto armazenado. Esta estrutura é a seguinte

fato : <objeto> <predicado> <valor>

onde pode ser observado que a estrutura dos objetos adotada é a de dupla <objeto-valor> (capítulo I).

Na representação dos fatos, <objeto> é um dos objetos definidos nos antecedentes ou consequentes das regras (veja a próxima seção). <valor> é uma das instâncias possíveis do objeto (ver LIMITE/VALOR na tabela 3.1), e deve pertencer a algum dos tipos indicados na tabela 3.1.

A seguir são exemplificados alguns dos fatos obtidos numa consulta ao sistema na forma em que eles ficam realmente armazenados na Base de Dados.

fato : 'o tempo de transitorio negativo do modelo' 'eh maior que' 0.  
fato : 'os parametros para a simulacao' estao disponiveis.  
fato : 'a ordem do polinomio A1' 'tem o valor' 2.  
fato : 'o polinomio A1' 'tem o valor' [1,-1.5,0.7].

Destes fatos podem ser observadas algumas das características do sistema. A linguagem é quase natural (note a ausência de acentuação, e a incorporação de aspas para delimitar os <objetos>, <predicados> e <valores>). Os objetos com estruturas mais complexas são representados por mais de um par <objeto-valor>. No caso de polinômios, utiliza-se duas estruturas, uma para a ordem e outra para os coeficientes, com estes últimos armazenados na forma de listas. Finalmente deve-se destacar que não existe nenhuma ordem no armazenamento dos fatos (depende somente de quando os objetos foram requeridos pelo sistema). Embora esta condição permita simplificar a estrutura da Base de Dados e a estrutura das máquinas de Inferências, quando a quantidade de fatos for muito elevada o sistema pode se tornar lento.

#### 3.4.2.-ESTRUTURA DA BASE DE CONHECIMENTO

É um fato conhecido que a representação do conhecimento na forma de regras de produção é um método útil em sistemas em que se deseja simular as respostas humanas à situações provenientes de condições dinâmicas [HERBB]. Entretanto esta representação pode ser implementada de forma diferente dependendo dos requisitos do

sistema. Como já foi explicado, uma característica desejável é a de flexibilidade do sistema às modificações. Assim sendo, optou-se por implementar as regras na forma mais próxima possível à linguagem natural, evitando-se as traduções indicadas na figura 3.2.

A estrutura genérica das regras é a indicada a seguir.

```
<identificador de regra> <identificador de encadeamento>  
se <antecedentes>  
entao <consequente>.
```

Devido as dificuldades de acentuação no ambiente PROLOG disponível, optou-se por não incluí-los na linguagem compreendida pelo sistema (i.e., escritura das regras e fatos, e interação com o usuário). Na descrição da estrutura das regras dadas anteriormente têm-se que o <identificador de regras> deve ser um átomo no sentido PROLOG, isto é, um elemento reconhecido como único pelo sistema, que geralmente possui a forma "regra\_XXX". O <identificador de encadeamento> determina se a regra é utilizada no encadeamento direto ":", ou no encadeamento reverso "<:". Esta distinção é necessária porque ainda que as duas Bases de Conhecimento ao serem implementadas fiquem em arquivos diferentes, ao serem carregadas pelo sistema compartilham a mesma área de memória com o resto do sistema. <antecedentes> é o conjunto de fatos associados em forma conjuntiva (conectivo "e"), que devem ser satisfeitos para que a regra seja considerada válida, e os <consequentes> são um conjunto de fatos associados em forma conjuntiva, que são validados pelas máquinas de inferência (de serem válidos os <antecedentes>), e assim executados (se eles implicarem algum procedimento a ser executado), ou aplicados sobre a Base de Dados a medida que se progride numa inferência (tanto direta quanto reversa).

Embora nas Bases de Conhecimento não se faça uma distinção explícita entre as regras, pode-se classificá-las em função da sua utilização dentro do sistema [SAN86]. Na Base de Conhecimento para o encadeamento reverso podem ser distinguidas três classes de regras que são indicadas a seguir.

### 3.4.2.1.-TIPOS DE REGRAS DA BASE DE CONHECIMENTO DE CONFIGURAÇÃO

Regras de decisão: Seus consequentes são utilizados para chamar procedimentos, tais como a modificação de variáveis ou a chamada dos algoritmos de controle e identificação, por exemplo:

regra\_1<:

```
se 'os parametros para a simulacao' estao disponiveis
e 'os parametros de resposta de servo' estao disponiveis
e 'o modelo do processo' esta disponivel
e 'a lei de controle' esta disponivel
e 'os parametros para verificacao de desempenho do controlador'
    estao disponiveis
entao acao de controle.
```

Nesta regra pode ser observada uma característica do sistema implementado, a de que o valor dos objetos é requerido pedindo a sua "disponibilidade". Desta forma cada objeto tem um valor (que depende do tipo), e a informação da sua disponibilidade. Logo, são necessários dois fatos (como mínimo) para cada objeto.<sup>4</sup>

Regras de definição: Estas regras são utilizadas para definição dos valores requeridos por alguns objetos que permitem obter a disponibilidade de outro objeto, por exemplo:

regra\_15<:

```
se 'os parametros do processo' sao variaveis
e 'as variacoes dos parametros' 'nao sao' rapidas
e 'o fator de esquecimento do identificador':= 0.996
entao 'o fator de esquecimento do identificador'
    esta disponivel.
```

---

<sup>4</sup> Porém os objetos com estruturas simples (i.e., inteiros, reais, etc) são reconhecidos como disponíveis se existe um fato dentro da Base de Dados que define um valor para esses objetos, sem a necessidade do fato indicando a sua disponibilidade.

Na regra anterior pode-se verificar que o operador (<predicado>) "==" é utilizado para a atribuição de valores aos objetos.

Regras de valores pré-especificados ("Default"): São regras com estrutura similar à das regras de definição, mas sua função é a de fornecer os valores dos objetos quando não é disponível mais informação, nem do usuário nem de procedimentos. A implementação de regras "default" permite o raciocínio não monotônico [HERBB]. Elas são ativadas somente quando forem esgotadas todas as outras possibilidades, e a prioridade das regras é dada pela disposição dentro da Base de Conhecimento. Estas regras são localizadas após as regras de definição principais. Como um exemplo tem-se:

regra\_73<:

```
se 'nao eh possivel fornecer' 'o maximo erro de previsao'
e 'o maximo erro de previsao' := 0.2
entao 'o maximo erro de previsao' esta disponivel.
```

Deste último exemplo pode-se observar outra característica do sistema, a de que alguns fatos possuem uma estrutura diferente da discutida na seção 3.4.1, isto é, o <valor> está incluído na estrutura do <predicado>. Esta estrutura torna o sistema mais complexo, mas facilita a transcrição das regras, permitindo manter a linguagem (quase) natural.

Também existe a possibilidade de combinar as funções das regras de decisão com as das regras de definição, mas este tipo de regra é considerado principalmente de decisão, já que esta é a mais importante dentro do sistema (as regras pertencentes a este tipo têm como consequentes as metas finais a serem obtidas, isto é, "identificação e controle").

### 3.4.2.2.-TIPOS DE REGRAS DA BASE DE CONHECIMENTO DE SUPERVISÃO

Na Base de Conhecimento de supervisão, podem ser identificadas as seguintes classes de regras:

Regras de estado: Seus consequentes indicam i).-o estado dos objetos utilizados para determinar o estado do sistema a ser controlado e/ou identificado, e ii).- a condição dos algoritmos de identificação e controle. Por exemplo a estacionariedade e convergência do algoritmo de identificação, e a estabilidade do sistema. Como exemplo tem-se a seguinte regra:

```
regra_22_ef :>
```

se 'a media do erro de saída' 'é menor que'  
'a media maxima do erro de saída'  
e 'o índice ITAE' 'é maior que' 'o máximo índice ITAE'  
e 'o primeiro sobrepico' 'é menor que' 'o terceiro sobrepico'  
então 'o sistema em malha fechada' é 'instável oscilatório'.

Regras de sugestões: Estas regras são utilizadas para indicar as ações que são necessárias para melhorar o desempenho dos algoritmos. Exemplo:

```
regra_GPC_2_ef :>
```

se 'o processo' eh 'de fase mínima e estável em malha aberta'  
e 'o sistema' eh 'instável'  
e  $N_u$  'eh igual a' 1  
entao sugere-se aumentar  $N_2$ .

Regra mistas: Nestas regras o consequente é uma conjunção de fatos indicando o estado do sistema (regras da primeira classe) e de sugestões (regras da segunda classe), por exemplo

regra\_15\_ef :>

se 'o seguimento de ruptura de modelo' 'tem o valor' desativado  
e 'o estado' eh 'nao convergente'  
entao 'os parametros' 'nao tem' 'os valores corretos'  
e 'sugere-se ativar o modulo de seguimento de ruptura de modelo'.

Esta classificação das regras permite verificar que o tipo de regras depende da função definida na seção 3.2 para as máquinas de inferência. Assim as regras para configuração são "ativas" e ao serem aplicadas modificam o estado dos objetos do sistema, e portanto o desempenho do sistema. As regras da Base de Conhecimento de supervisão são fundamentalmente "passivas", sua função é a de diagnóstico e "sugestões". Elas não modificam o estado dos objetos nem o desempenho do sistema, como pode-se verificar na seção seguinte, onde são descritas as máquinas de inferências.

### 3.5.-MANIPULAÇÃO DO CONHECIMENTO

Por *manipulação do conhecimento* entende-se como são obtidos os resultados a partir de uma *representação do conhecimento*. No capítulo I foi detalhada como as máquinas de inferência com encadeamento direto e reverso desempenham esta tarefa, quando a representação de conhecimento é feita na forma de regras de produção. Nesta seção são descritas as implementações das máquinas de inferência utilizadas pelo sistema.

Antes de ser chamada a máquina de inferência com encadeamento reverso é necessário que o usuário forneça a *Meta objetivo*. Esta parte do sistema corresponde ao nível OBJETIVO mostrado na figura 3.1, e o seu funcionamento é ilustrado na listagem PSC.3.1 utilizando o pseudocódigo definido no capítulo I.

**SEC:**

**COMEÇO**

**REPETIR**

**CARREGAR** Bases-de-Conhecimento;

**VERIFICAR-OBJETOS;**

**ESCOLHER** Função;

**CASO** Função **DE**

Incorporar : **CHAMAR** editor de regras;

Consulta : **COMEÇO**

**ESCOLHER** Meta-OBJETIVO;

Resp := **MIER**(Meta-OBJETIVO),

**MOSTRAR** Resp;

**FIM;** (CASO)

**ATE** Função = Terminar;

**FIM.**

### **PSC. 3.1 - Inicialização e obtenção da Meta objetivo**

Como pode ser observado na listagem anterior, no início de uma consulta ao sistema dispõe-se de três opções:

I).-Incorporar, que permite modificar as Bases de Conhecimento.

II).-Terminar, finalizando a consulta.

III).-Consulta, onde é procurada a Meta OBJETIVO (i.e., controle e/ou identificação), ativando-se na continuidade a máquina de inferência com encadeamento reverso, que é descrita na seção seguinte.

### **3.5.1.-MÁQUINA DE INFERÊNCIA COM ENCADEAMENTO REVERSO**

Uma vez obtido o objetivo ativa-se o nível de Configuração indicado na figura 3.2. Neste nível a máquina de inferência com encadeamento reverso procura preencher o campo de <valor> dos objetos com os valores que permitem (com base ao conhecimento

disponível no momento) obter o objetivo na melhor forma possível. O funcionamento desta máquina de inferência é ilustrado através do pseudocódigo da listagem PSC.3.2. dada a seguir.

**FUNÇÃO MIER** ( Meta ): (verdadeiro,falso);

**PROCEDIMENTO CÁLCULO** ( Meta );

**COMEÇO** ( CÁLCULO )

SE a Meta é um procedimento de IDENTIFICAÇÃO e/ou CONTROLE  
ENTÃO COMEÇO

REPETIR

CHAMAR o PROCEDIMENTO; ( PASCAL )

CHAMAR MIEF; ( List. 3.3 )

SE o usuário deseja modificar  
algum parâmetro

ENTÃO MODIFICAR o Parâmetro;

ATÉ que o usuário concorde com os resultados  
obtidos;

FIM;

SENÃO CHAMAR o PROCEDIMENTO; ( PASCAL ou PROLOG )

FIM; ( CÁLCULO )

**COMEÇO** ( MIER )

SE a Meta é vazia ( i.e., fim da recursão )

ENTÃO RETORNAR verdadeiro;

SE a Meta é formada pela disjunção de Condições

ENTÃO COMEÇO (recursão)

Resp1 := MIER (Primeira-Condição);

Resp2 := MIER (Resto-das-Condições);

RETORNAR (Resp1 AND Resp2);

FIM;

SE a Meta se encontra na Base-de-Dados

ENTÃO RETORNAR verdadeiro;

SE a Meta se encontra "negada" na Base-de-Dados

ENTÃO RETORNAR falso;

SE a Meta se encontra como Conclusão de uma ou mais regras na  
Base-de-Conhecimento-de-Configuração

ENTÃO COMEÇO

FORMAR o Conjunto-de-Discórdia

REPETIR

SELECIONAR a 1ª Regra do  
Conjunto-de-Discórdia;

TIRAR a 1ª Regra do Conjunto-de-Discórdia;

Resp := MIER(Condições-da-1ª Regra);

ATÉ Conjunto-de-Discórdia = vazio OU  
Resp = verdadeiro;

SE Resp = verdadeiro

ENTÃO COMEÇO

SE a Meta é um procedimento  
de cálculo

```

                                ENTÃO CHAMAR CÁLCULO(Meta);
                                SALVAR a Meta na Base-de-Dados;
                                RETORNAR verdadeiro;
                                FIM;
                                FIM;
SE a Meta é um procedimento de cálculo
    ENTÃO CHAMAR CÁLCULO(Meta);
SE a Meta é uma comparação entre dois Objetos
    ENTÃO RETORNAR (o resultado da comparação);
SE a Meta é a solicitação de um Dado
    ENTÃO COMEÇO
        PEDIR ao usuário para fornecer o Dado;
        RETORNAR verdadeiro;
    FIM;
SE não tem acontecido nenhuma das condições anteriores
    ENTÃO COMEÇO
        PEDIR ao usuário para confirmar a validade
            da Meta;
        SE a Meta é confirmada como válida
            ENTÃO COMEÇO
                SALVAR a Meta na
                    Base-de-Dados;
                RETORNAR verdadeiro;
            FIM
        SENÃO COMEÇO
            SALVAR a Meta "negada" na
                Base-de-Dados;
            RETORNAR falso;
    FIM;
FIM; ( MIER )

```

### PSC. 3.2 - Máquina de Inferência com Encadeamento Reverso

No pseudocódigo anterior, por Meta "negada" deve-se entender a negação do predicado (p.e., 'tem', 'nao tem', etc.)

### 3.5.2.-MÁQUINA DE INFERÊNCIA COM ENCADEAMENTO DIRETO

Na implementação atual do sistema, o nível de supervisão, isto é, a máquina de inferência com encadeamento direto é chamada pela máquina com encadeamento reverso, após realizada a identificação e/ou o controle do processo (ver listagem PSC.3.2). Esta máquina de inferência é apresentada no pseudocódigo da listagem PSC.3.3.

## PROCEDIMENTO MIEF;

### COMEÇO

FORMAR o Conjunto-de-Discórdia com todas as regras da Base-de-Conhecimento-de-Supervisão;

### REPETIR

SELECIONAR a 1ª Regra do Conjunto-de-Discórdia;

TIRAR a 1ª Regra do Conjunto-de-Discórdia;

SE a Conclusão-da-1ª Regra não se encontra na Base-de-Dados

### ENTÃO COMEÇO

Resp := VERIFICA(Condição-da-1ª Regra);

SE resp = valida

ENTÃO SALVAR Conclusão-da-1ª Regra na Base-de-Dados;

### FIM;

ATE Conjunto-de-Discórdia = vazio;

FIM;(MIEF)

FUNÇÃO VERIFICA(Meta):(verdadeiro,falso);

### COMEÇO

SE a Meta é vazia ( i.e., fim da recursão )

ENTÃO RETORNAR verdadeiro;

SE a Meta é formada pela disjunção de Condições

ENTÃO COMEÇO (recursão)

Resp1 := VERIFICA(Primeira-Condição);

Resp2 := VERIFICA(Resto-das-Condições);

RETORNAR (Resp1 AND Resp2);

### FIM;

SE a Meta se encontra na Base-de-Dados

ENTÃO RETORNAR verdadeiro;

SE a Meta se encontra "negada" na Base-de-Dados

ENTÃO RETORNAR falso;

SE a Meta é uma comparação entre dois Objetos

ENTÃO RETORNAR (o resultado da comparação);

FIM;(VERIFICA)

## PSC. 3.3.- Máquina de Inferência com Encadeamento Direto

Nas listagens fornecidas pode ser notado que as duas máquinas realizam a inferência em forma similar, isto é, a ordem de procura para a validação das metas é mantida. Entretanto, a máquina de inferência com encadeamento direto é mais simples do que a outra máquina de inferência porque na tarefa de supervisão não se tem necessidade de modificar os objetos do sistema (seção 3.4.2).

### 3.6.-INTERFACE HOMEM-MÁQUINA

Por comunicação entre sistemas entende-se como a interação entre estes sistemas. Porém, quando um destes sistemas é o ser humano, as características dessa comunicação são bem particulares. O ser humano pode "aprender" os métodos de comunicação de outros sistemas, sendo que o tempo necessário para o aprendizado diminui a medida que o método de comunicação se aproxima da linguagem natural.

Implementar um processador de linguagem natural não é o objetivo deste trabalho; Contudo, a forma em que o conhecimento é armazenado no sistema facilita a implementação de uma interface entre o sistema e o usuário com características de linguagem natural.

Como foi descrito na seção 3.4, as Bases de Conhecimento são formadas por conjuntos de regras escritas numa linguagem quase natural. Analogamente, os fatos da Base de Dados, possuem estruturas compatíveis com estas regras. Assim, quando o usuário o requisitar, o sistema pode apresentar estes conhecimentos em forma facilmente compreensível, sem a necessidade de tradução.

A tarefa a ser realizada pelas máquinas de inferência (seção 3.5) é a de demonstrar a validade de certas metas (fatos). Para tanto na configuração (encadeamento reverso) procura-se inicialmente verificar se a meta existe como um fato válido na Base de Dados. A seguir procura-se verificar se a meta é deduzível das regras da Base de Conhecimento de configuração (i.e., a meta é conclusão de uma ou mais regras). Em sequência procura-se a existência de algum procedimento (em PROLOG ou PASCAL) que valide a meta<sup>5</sup>. Finalmente a máquina de inferência procura provar a validade da meta através do usuário. Para conseguí-la o sistema possui dois mecanismos.

---

<sup>5</sup> Isto acontece, por exemplo, quando a meta é a obtenção de algum valor para um OBJETO do sistema, que depende de um cálculo de certa complexidade.

1).-O primeiro, e mais importante, é uma pergunta em linguagem quase natural, da forma

você poderia dizer se <meta>.

Por exemplo, numa consulta ao sistema, ele apresentou a seguinte pergunta:

x-----x

voce poderia dizer se  
pode fornecer o tempo de subida  
s.\n.\p.\como sabe que META.\v.\a.  
==>

x-----x

Na última linha do texto são indicadas as respostas possíveis que o sistema espera receber do usuário, as letras são, abreviações das palavras sim, não porque?, ajuda e ver objetos. Desta forma o usuário também responde em linguagem natural. A função destas respostas é a seguinte:

sim: indica que o usuário concorda com o fato (no caso do exemplo, que ele pode fornecer o dado solicitado). Este é armazenado como fato válido na Base de Dados, e a inferência continua.

não: indica que o usuário não concorda com o fato. Neste caso, armazena-se a negação do fato, e obriga-se o sistema a procurar outro caminho de inferência.

Nas duas respostas anteriores, o usuário tem uma participação ativa no processo de inferência. Porém, também é permitido que ele atue de forma passiva, somente verificando como as soluções são ou podem ser obtidas. Neste caso o sistema também aceita as seguintes perguntas:

porque?: Indicando que se deseja saber porque o sistema está fazendo essa pergunta. Neste caso o sistema mostra o fato que se

quer provar a validade, a regra que está sendo utilizada para a prova, e os <antecedentes> dessa regra que já foram validados. No exemplo dado anteriormente pode se obter o seguinte diálogo:

x-----x

==>porque.

estou tentando demonstrar pela regra\_45 que  
o tempo de subida do modelo esta disponivel  
voce concorda com o que se esta procurando?  
s.\n.\p.\como sabe que Meta.

==>

x-----x

Novamente o sistema fornece ao usuário uma pergunta que ele pode responder em forma ativa ou passiva. Respodendo **sim**, o sistema volta à primeira pergunta do exemplo, já que o usuário pode concordar com o que se deseja obter, mas não com a validade do fato. Se responder com **não**, ele indica que o caminho de Inferência está errado, obrigando o sistema a procurar um outro caminho (mesma condição que na resposta **não** dada anteriormente). Se o usuário responder novamente com a pergunta **porque?**, o sistema procura na cadeia de inferência a regra que disparou a solicitação do fato que é o consequente da regra atual (no exemplo regra-45), continuando com a pergunta já feita. Este método permite ao usuário percorrer toda a cadeia de Inferência, função indicada como útil no capítulo I para o treinamento. No exemplo o sistema apresenta a seguinte tela:

x-----x

==>porque.

estou tentando demonstrar pela regra\_MOD\_1 que  
os parametros do modelo estao disponiveis  
ja foi provado que o atraso do modelo esta disponivel  
voce concorda com o que se esta procurando?  
s.\n.\p.\como sabe que Meta.

==>

x-----x



x-----x

==> como sabe que 'o horizonte N2' esta disponivel?.  
pode-se provar pela regra\_25  
que diz:

se 'o processo' eh 'de fase minima'  
e 'os parametros do modelo' estao disponiveis  
e 'o atraso do processo' eh conhecido  
e 'o horizonte N2' := 'o atraso do processo'  
entao 'o horizonte N2' esta disponivel.

e tambem pela regra\_26

se 'o processo' eh 'de fase minima'  
e 'os parametros do modelo' estao disponiveis  
e 'o atraso do processo' eh variavel  
e 'o maximo atraso do processo' eh conhecido  
e 'o horizonte N2' := 'o maximo atraso do processo'  
entao 'o horizonte N2' esta disponivel.

e tambem pela regra\_27

.  
.  
.

e tambem pela regra\_45

que diz:

se 'pode fornecer' 0\_parametro  
e ingresse 0\_parametro  
entao 0\_parametro esta disponivel.

e também pela regra\_49

que diz:

se 'nao pode fornecer' 'o horizonte N2'  
e 'o horizonte N2' := 10  
entao 'o horizonte N2' esta disponivel.

não ha mais regras.

x-----x

4).-Se a **META** não cumpre com nenhuma das condições anteriores, o sistema não tem forma de prová-lo, por exemplo

x-----x

**==>como sabe que 'hoje vai chover'?**

**não tenho forma de prova-lo.**

x-----x

Outra forma de intervenção passiva do usuário no sistema ocorre quando ele deseja conhecer os valores dos objetos do sistema. Desde que os algoritmos implementados possuem grande quantidade de variáveis, mas nem todas são necessárias para obter o controle e/ou a identificação de um processo determinado, optou-se por fornecer somente aqueles que pertencem à solução do problema (i.e., aqueles objetos não complexos segundo a tabela 3.1 que existem na Base de Dados). Além disto, eles são apresentados só por solicitação explícita do usuário através da opção **ver objetos** (v.), que permite não carregar a tela com dados que o usuário não deseja conhecer no momento.

A última opção de resposta oferecida pelo sistema é a de **ajuda**, e fornece ao usuário um resumo das possibilidades que ele tem para responder (basicamente o descrito nesta seção).

O segundo método de interação entre o sistema e o usuário na configuração, é a solicitação do valor de um objeto por parte do primeiro. Neste caso o sistema faz a seguinte solicitação:

**ingresse o valor do <objeto>, (<limites> ou <valores possíveis>).**

**==>**

Após o usuário fornecer o valor solicitado, o sistema verifica se ele pertence ao mesmo tipo que o <objeto>, e se ele está entre os <limites> ou é um dos <valores possíveis>, (ver tabela 3.1).

Quando o sistema encontra-se no nível de supervisão, a interação com o usuário, é mais simples, por que a máquina de inferência de encadeamento direto também é simples. Só existe o primeiro tipo de interação, pois a inferência direta é "passiva", e não modifica os objetos, logo não existe a solicitação de dados ao usuário.

Da mesma forma que no encadeamento reverso, a interação realiza-se por meio de perguntas e respostas. Cada vez que a máquina de inferência prova a validade de uma regra, ela apresenta ao usuário uma mensagem como a seguinte:

Pela <regra\_xx> concluiu-se que:  
<consequente da regra\_xx>  
voce concorda:  
s.\n.\p.\como sabe que META.\v.\a.  
==>

As respostas possíveis para esta pergunta têm o mesmo significado que na interação no encadeamento reverso, mas o seu funcionamento difere ligeiramente. A resposta **sim**, é tratada na mesma forma, com a diferença agora que **META** é a conclusão da regra validada. **não**, indica que se deve pesquisar um outro caminho, o que é similar ao encontrado na interação do usuário no nível de configuração. O funcionamento das respostas **como sabe que META**, **ver objetos** e **ajuda**, é idêntico ao das respostas no nível de configuração. Porém o modo em que o sistema trata a resposta **porque**, difere pelas características da máquina de inferência. Neste caso o sistema mostra como se chegou à conclusão, isto é, qual regra foi utilizada; se o usuário requerer, analogamente à inferência reversa, pode-se pesquisar toda a corrente de inferência repetindo a pergunta **porque**. Por exemplo, em uma consulta ao supervisor obteve-se o seguinte diálogo:

x-----x

pela regra\_4\_ef concluiu-se que:

o estado eh convergente.

voce concorda:

s.\n.\p.\como sabe que META.\v.\a.

==>porque.

porque a regra\_4\_ef diz que

se 'o sistema' eh estacionario

e 'a norma de variacao dos parametros'

'eh menor que' 'a variacao máxima dos parametros'

entao 'o estado' eh convergente.

sendo as condicoes provadas pelos fatos da Base de Dados,

exceto o fato 'o sistema' eh estacionario

que foi concluido da regra\_3\_ef.

voce concorda com esta conclusao?.

s.\n.\p.\como sabe que META.\v.\a.

==>porque.

essa foi a primeira regra a ser disparada.

x-----x

Uma característica adicional para se ter a interface amigável foi a utilização de Janelas do tipo "popup", com as que se oferece ao usuário uma melhor visão do estado atual da inferência, ou face ativa do sistema.

### 3.7.-RESUMO

Neste capítulo foi apresentada a estrutura do Sistema Especialista. Foi destacada a vantagem da utilização de linguagens diferentes dependendo da tarefa realizada por cada parte do sistema. Para cálculos o PASCAL, para a manipulação simbólica PROLOG. Também foi ressaltada a utilização de duas máquinas de

inferência, uma para as tarefas de configuração e outra para as tarefas de supervisão, indicando a sua estrutura como as vantagens da sua utilização. Foram apresentadas as Bases de Conhecimento assim como a Base de Dados na que se verifica a utilização de uma linguagem quase-natural. Esta linguagem facilita a construção das partes anteriormente indicadas como a da interface homem-máquina a que também foi apresentada neste capítulo.

A implementação do Sistema Especialista apresentada, ainda que simples, mostrou ser um método válido para o escopo das tarefas a ele designadas. O sistema desenvolvido permitiu explorar as vantagens e desvantagens da utilização de ferramentas baseadas em conhecimento na área de controle adaptativo.

## CAPÍTULO

### IV

## BASES DE CONHECIMENTO

## 4.1-INTRODUÇÃO

Neste capítulo é apresentada a obtenção do conhecimento para as bases de conhecimento. Como foi descrito no capítulo anterior, o conhecimento do sistema pode ser dividido em duas partes:

i).- Conhecimento para a tarefa de *Configuração*, ou seja, conhecimento sobre os valores que devem assumir os objetos para que, frente a uma situação particular (i.e., um processo particular), a tarefa seja realizada de forma adequada. Este é armazenado na Base de Conhecimento para encadeamento reverso.

ii).-Conhecimento para a tarefa de *Supervisão*, que é utilizado para determinar o estado global do processo, e sugerir ações a serem tomadas para melhorar o desempenho dos algoritmos. Este conhecimento é armazenado na Base de Conhecimento para encadeamento direto.

Na ausência de um módulo dedicado à aquisição do conhecimento requerido, este foi obtido de três formas: Fornecido por um especialista através de consulta, por pesquisa de bibliografia dedicada na área, e principalmente por testes orientados a ressaltar a influência dos objetos no desempenho dos algoritmos.

Devido ao fato de que parte deste trabalho é baseado parcialmente no trabalho desenvolvido por Arruda, o conhecimento para a supervisão do algoritmo de identificação, é o discutido em [ARR88].

Este capítulo é dedicado à obtenção do conhecimento para os algoritmos de controle, tentando mostrar por meio de testes, a validade deste conhecimento. Devido à grande quantidade de objetos envolvidos numa tarefa qualquer do sistema, é apresentado somente o conhecimento sobre os objetos considerados mais importantes, que justificam as regras constituintes das bases de conhecimento criadas.

Os processos sobre os quais se adquire conhecimento são lineares e de três tipos:

- i) Estáveis em malha aberta, e de fase mínima.
- ii) Instáveis em malha aberta, e de fase mínima.
- iii) Estáveis em malha aberta, e de fase não mínima.

Os processos simulados são descritos por uma equação à diferenças dada em forma genérica a seguir:

$$A(z^{-1}) y(t) = B(z^{-1}) u(t-1)$$

a perturbação é adicionada ao processo quando é testada a influência dos parâmetros do controlador nas perturbações.

Os processos simulados são descritos na tabela 4.1. O processo 1 é de fase mínima e estável em malha aberta, o lugar de raízes deste processo é mostrado na figura 4.1.1. O processo 2 é estável em malha aberta e de fase mínima, sendo o seu atraso maior que a unidade ( $d = 5$ ). O lugar de raízes no plano Z é apresentado na figura 4.1.2. O processo 3 é de fase mínima, mas instável em malha aberta como pode-se verificar pelo lugar de raízes mostrado na figura 4.1.3, e o processo 4, cujo lugar de raízes é mostrado na figura 4.1.4, é também instável em malha aberta e de fase mínima. O 5 é um processo estável e de fase não mínima (lugar de raízes da figura 4.1.5). Finalmente os processos 6 e 7 são de fase não mínima, e instáveis em malha aberta, como pode ser verificado pelo lugar de raízes apresentado nas figuras 4.1.6 e 4.1.7

| Proc. N° | MODELO                                                                                                           |
|----------|------------------------------------------------------------------------------------------------------------------|
| 1        | $A = 1 - 1.036 z^{-1} + 0.2636 z^{-2}$<br>$B = 0.1387 + 0.0889 z^{-1}$                                           |
| 2        | $A = 1 - 1.5 z^{-1} + 0.7 z^{-2}$<br>$B = z^{-4} (1 + 0.9 z^{-1})$                                               |
| 3        | $A = 1 - 1.4 z^{-1} + 1.44 z^{-2}$<br>$B = 0.14 + 0.09 z^{-1}$                                                   |
| 4        | $A = 1 - 0.3 z^{-1} + 1.5 z^{-2}$<br>$B = z^{-3} (1 - 0.8 z^{-1})$                                               |
| 5        | $A = 1 - 1.425 z^{-1} + 0.496 z^{-2}$<br>$B = -0.102 + 0.173 z^{-1}$                                             |
| 6        | $A = 1 - 0.4 z^{-1} + 1.9 z^{-2} + 2.1 z^{-3} + 1.3 z^{-4} - 1.2 z^{-5}$<br>$B = -0.4 - 0.6 z^{-1} - 0.6 z^{-2}$ |
| 7        | $A = 1 - 1.44 z^{-1} + 1.4 z^{-2}$<br>$B = -0.102 + 0.173 z^{-1}$                                                |

TABELA 4.1 Processos simulados

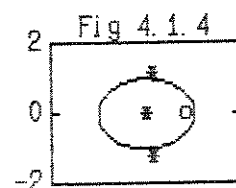
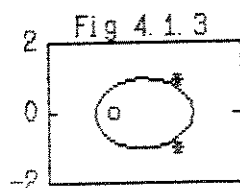
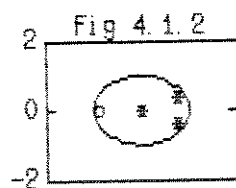
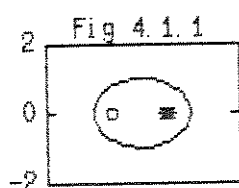


Fig 4.1.1 a 4.1.4 Lugar de raízes dos Processos simulados

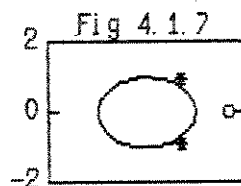
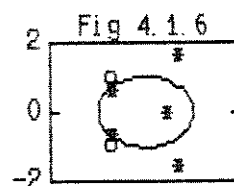
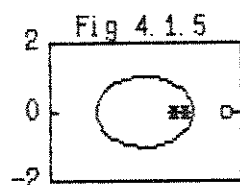


Fig 4.1.5 a 4.1.7 Lugar de raízes dos Processos simulados

As simulações são realizadas conforme é descrito a seguir:

- fase 1) 50 iterações em malha aberta para identificação do modelo, com as ordens corretas, (Não mostradas nas figuras).
- fase 2) 100 iterações em malha fechada com o controlador calculado com o modelo identificado (controle adaptativo).

A referência é o sinal mostrado na figura 4.1.8, e apresenta uma mudança de 0 a 10 na iteração 10, e na iteração 50 ela volta a zero. Na iteração 66 sofre novamente uma mudança para 10, voltando a zero na iteração 75 e ficando nesse valor até o fim da simulação.

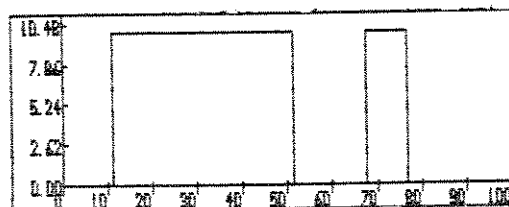


Fig 4.1.8 Sinal de referência

As figuras apresentadas devem ser "lidas", quando não

Indicado de outra forma, como : a figura superior (Fig X.XXa) é a saída do processo, sendo a figura inferior (Fig X.XXb), o sinal de controle do processo, (sinal calculado pelo controlador). Da mesma forma por razões de clareza, a referência não é superposta à saída do processo, a menos que essa prática auxilie ao entendimento dos gráficos.

A divisão deste capítulo é a seguinte:

Na seção 4.2 é apresentada a aquisição do conhecimento para a configuração e supervisão do controlador GPC. A primeira parte é com referência "normal" (capítulo 11), sendo a segunda com referência pré-especificada.

Na seção 4.3 obtém-se o conhecimento necessário para o funcionamento do controlador de Variância Mínima Generalizado (GMV), e GMV com alocação de pólos.

O conhecimento de configuração e supervisão do controlador PID é apresentado na seção 4.4.

Na seção 4.5 apresenta-se um exemplo de aplicação que demonstra o funcionamento do sistema. Finalmente na seção 4.6 são apresentadas as conclusões do capítulo.

## **4.2.-CONTROLADOR PREDITIVO GENERALIZADO**

### **4.2.1.-HORIZONTES DE PREVISÃO DA SAÍDA E DE CONTROLE**

Antes de serem apresentadas as simulações serão feitos alguns comentários a respeito das características dos parâmetros do GPC, assim como o seu estado particular nestas simulações. Também serão apresentadas aqui as idéias básicas da escolha dos horizontes.

Como foi visto no Capítulo II, ao ser apresentado o Controlador Preditivo Generalizado (GPC), este controlador possui, como o controlador GMV, filtros que modificam as características do controle realizado (evolução da saída). Porém, diferente do que ocorre no GMV, onde estes filtros garantem a estabilidade do sistema em malha fechada, no GPC sua função é secundária, já que a estabilidade pode ser assegurada por uma escolha correta dos horizontes do controlador (Inicial, Final e de controle). A tarefa a ser realizada nesta seção é a obtenção destes horizontes em função das características significativas dos processos a serem controlados.

Para a obtenção dos horizontes, os filtros são desativados, isto é, eles são configurados como:

$$\begin{aligned} \deg P &= \deg T = \deg R = 0 \\ P_0 &= R_0 = T_0 = 1 \end{aligned}$$

e a ponderação da excitação é fixada como :

$$\lambda = 1e^{-8} \text{ (i.e., zero)}$$

Uma característica do GPC é a sua capacidade de utilizar a referência previamente conhecida, no cálculo do sinal de controle. Porém, devido que o conhecimento antecipado da referência é particularidade de alguns processos, será considerado principalmente neste trabalho a referência "normal" (seção 2.8), sendo a referência "pré-especificada" considerada posteriormente como uma melhora possível no desempenho de algum sistema.

Ao ser analisado o algoritmo de controle observa-se que o tempo de cálculo depende principalmente de dois fatores:

- i).-O tempo necessário para o cálculo das previsões da saída livre.
- ii).-A inversão da matriz  $(M^T M + \lambda I)^{-1}$ .

O primeiro fator é diretamente influenciado pelo valor do horizonte final de previsão  $N_2$ . O segundo fator depende do valor do horizonte de controle  $N_u$ , como já foi indicado no capítulo 11. Assim, estes horizontes devem ser o menor possível para permitir uma maior velocidade de cálculo.

Finalmente deve-se indicar que nas simulações, foi utilizada a seguinte notação para os horizontes de previsão inicial, final e de controle:  $(N_1, N_2, N_u)$ .

#### 4.2.1.1.-PROCESSOS ESTÁVEIS EM MALHA ABERTA E DE FASE MÍNIMA

A primeira simulação será utilizando o processo 1, que possui polos bem amortecidos como pode ser verificado pelo seu lugar de raízes apresentado na figura 4.1.1.

Ainda que algum conhecimento sobre os valores possíveis dos horizontes possa ser obtido da literatura [CLK89, MOH86], os horizontes indicados nesses trabalhos geralmente procuram ser genéricos para poderem ser utilizados na maioria dos processos. Também nesses trabalhos, sugerem-se valores de horizontes que permitem que o GPC tenha desempenho análogo a outro controlador, por exemplo ("Dead Beat", etc). Uma abordagem inicial para a escolha dos horizontes é selecioná-los o menor possível (como foi indicado anteriormente), para obter o menor esforço computacional. Na figura 4.2.1a mostra-se a evolução da saída deste processo para os horizontes  $(1,1,1)$  e na 4.2.1b o sinal de controle. Desta figuras pode-se observar que a saída acompanha a referência. A diferença inicial deve-se ao processo não estar ainda corretamente identificado. Esta característica poderá ser observada com maior ou menor intensidade em todas as simulações realizadas.

(1,1,1)

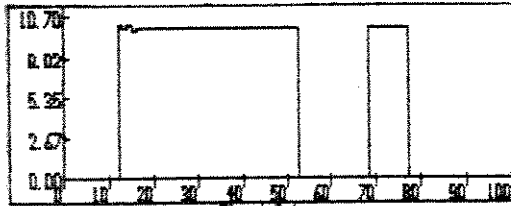


Fig. 4.2.1a

(1,2,1)

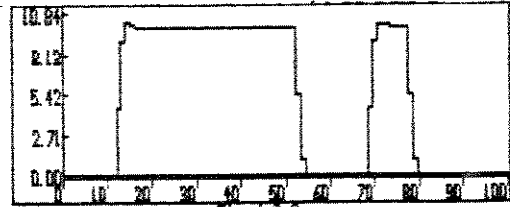


Fig. 4.2.2a

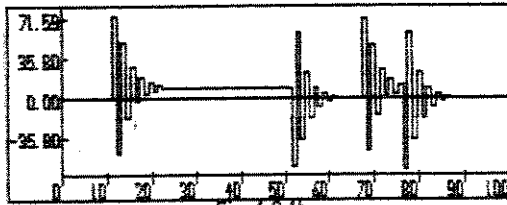


Fig. 4.2.1b

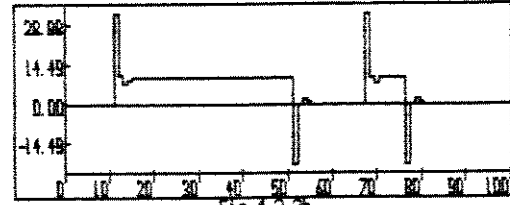


Fig. 4.2.2b

Nas figuras 4.2.2, 4.2.3 e 4.2.4 pode-se verificar a influência do horizonte final de previsão ( $N_2$ ) para os valores 2, 3 e 8 respectivamente. O horizonte inicial e o horizonte de controle  $N_u$  são fixados em 1. Das quatro simulações apresentadas para este processo, pode-se verificar que o aumento do horizonte final de previsão torna a resposta do sistema mais lenta, com um menor nível de sinal de controle.

(1,3,1)

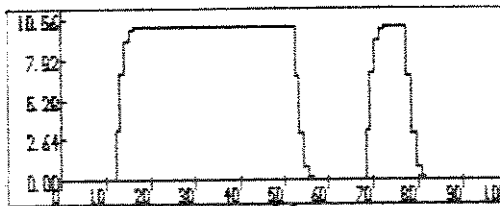


Fig. 4.2.3a

(1,8,1)

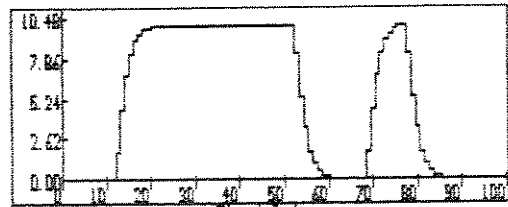


Fig. 4.2.4a

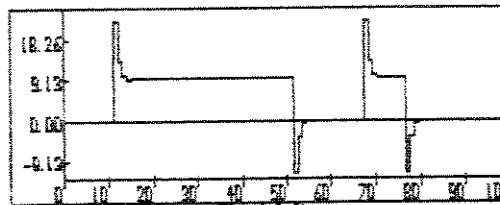


Fig. 4.2.3b

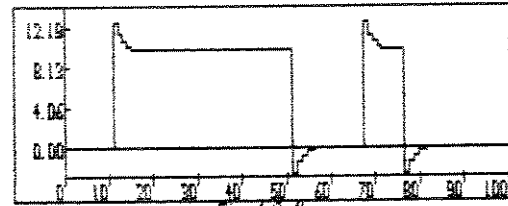


Fig. 4.2.4b

A influência do horizonte de controle é analisada utilizando-se os resultados ilustrados nas figuras 4.2.5 e 4.2.6, para os valores 2 e 3 respectivamente, sendo  $N_1 = 1$  e  $N_2 = 3$ . Na primeira figura pode-se observar que os sinais de saída e de controle tendem para os obtidos com os horizontes unitários, sendo

que a correspondência exata é obtida para o horizonte  $N_u = 3$ . Este fato deve-se a que tanto com este conjunto de horizontes quanto com o conjunto  $(1,1,1)$ , tem-se um desempenho do tipo "dead-beat". Das ilustrações do sinal de controle apresentadas nas figuras 4.2.5b e 4.2.6b pode-se observar que a excitação depende mais da diferença entre os horizontes  $N_u$  e  $N_2$ , que do valor de  $N_2$  isoladamente ainda que não exista uma relação direta da diferença entre estes horizontes e a amplitude do sinal de excitação. Esta particularidade de *suavizar as ondas* diminuindo a atividade nos atuadores por meio dos horizontes final e de controle, pode ser obtida (como se verá) também pelo coeficiente  $\lambda$ , pelo filtro da saída  $P$ , e sob algumas circunstâncias pelo horizonte  $N_1$ . Esta redundância de "tuning-knobs" permite por um lado uma flexibilidade para as tarefas de controle, não fornecida por quase nenhum dos controladores adaptativos desenvolvidos anteriormente, mas apresenta a desvantagem de não permitir um entendimento simples por pessoas não especializadas.

$(1,3,2)$

$(1,3,3)$

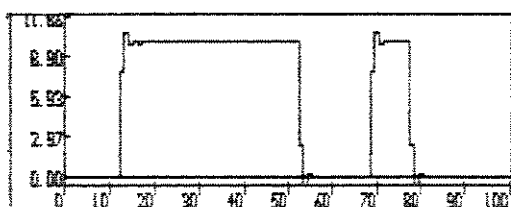


Fig. 4.2.5a

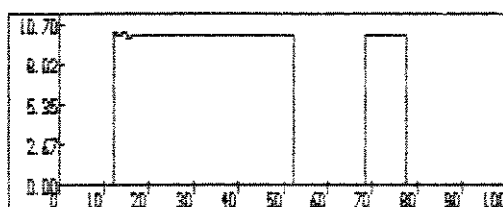


Fig. 4.2.5b

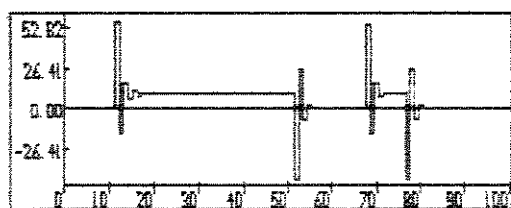


Fig. 4.2.5c

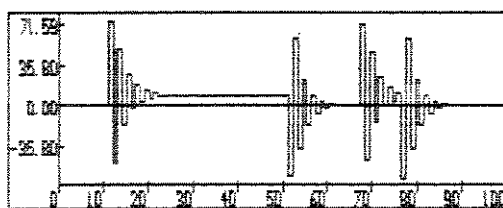


Fig. 4.2.6a

Nas figuras 4.2.7 e 4.2.8 são apresentadas as simulações com o horizonte de controle  $N_1$  em 2 e 3 respectivamente, sendo que os outros horizontes possuem os valores:  $N_2 = 3$  e  $N_u = 3$ . Estas simulações mostram que o horizonte de previsão inicial maior que a unidade permite obter uma ação de controle mais suave com uma deterioração da saída. Entretanto este efeito é mais difícil de quantificar que o efeito da variação dos outros horizontes.

(2,3,3)

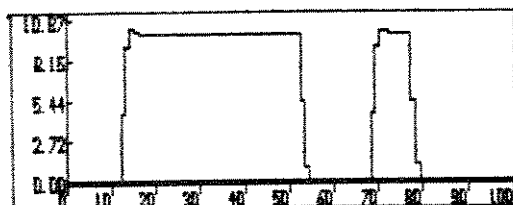


Fig. 4.2.7a

(3,3,3)

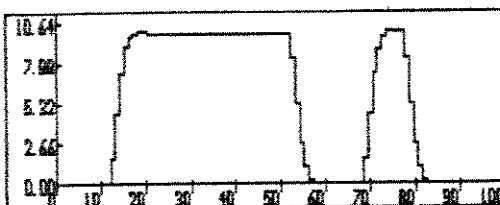


Fig. 4.2.8a

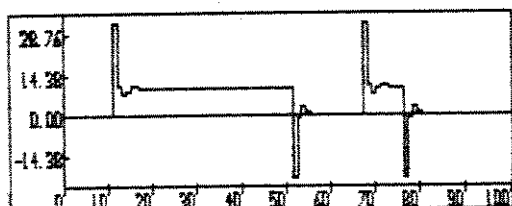


Fig. 4.2.7b

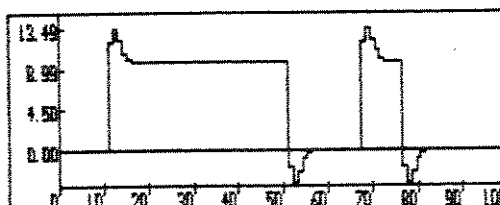


Fig. 4.2.8b

Destas simulações pode-se propor os seguintes valores para os horizontes para este processo:

- i). -  $N1 = 1$ ;
- ii). -  $N2 = 1$ ;
- iii). -  $Nu = 1$ ;

No processo simulado o atraso é unitário, e como foi indicado na seção 2.3, em processos com atrasos maiores que a unidade, deve-se aumentar a ordem do polinômio  $\hat{B}$  para que os primeiros coeficientes deste, compreendam o atraso de transporte. Este procedimento tem como vantagem, a de considerar o atraso do processo como unitário, o que diminui a quantidade de informação a ser fornecida pelo usuário aos algoritmos (prevendo que a ordem de  $\hat{B}$  seja igual ou maior que " $d+nb$ ", onde " $d$ " é o atraso real do processo e  $nb$  a ordem do polinômio  $B$  do processo). Ainda que os primeiros " $d-1$ " coeficientes após a convergência do estimador, possuam valores muito pequenos, estes valores geralmente são diferentes de zero, acrescentando zeros de fase não mínima no processo estimado, e provocando problemas ao tentar obter um desempenho tipo "dead-beat" (i.e.,  $N1=1$  e  $N2=Nu$ ).

Para analisar a influência dos horizontes no desempenho de processos com atraso maior que a unidade, simula-se o processo 2. Neste sistema o atraso real é 5, e a ordem do polinômio  $B$  é 1.

Utilizando o procedimento de calcular o polinômio estendido, são estimados 5 coeficientes do polinômio  $\hat{B}$ . O lugar de raízes do processo real é mostrado na figura 4.1.2. Na figura 4.2.9 pode-se observar o lugar de raízes do processo identificado. No modelo identificado existem zeros de fase não mínima porque os d-1 coeficientes iniciais do polinômio  $\hat{B}$  não são nulos.

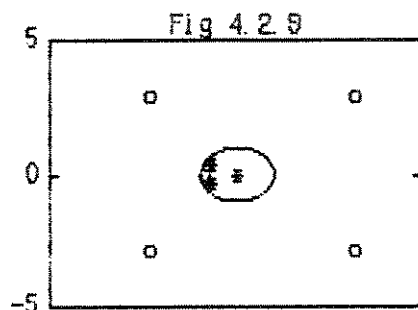


Fig 4.2.9 Lugar de raízes estimadas do processo No 2

Como no caso anterior, inicia-se os testes com o conjunto de horizontes mínimo, isto é  $(1,1,1)$ , mostrando-se a evolução do sistema em malha fechada na figura 4.2.10, evidentemente instável. Aumentando o horizonte  $N_2$ , Obteve-se estabilidade com para  $N_2 = 5$ , como ilustra a figura 4.2.11.

$(1,1,1)$

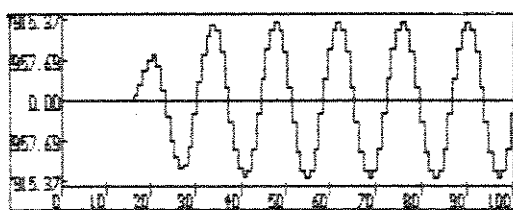


Fig 4.2.10a

$(1,5,1)$

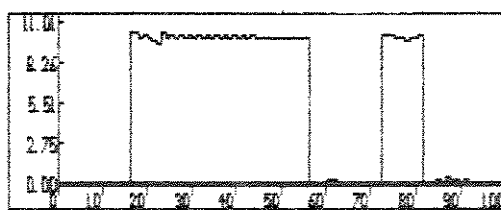


Fig 4.2.11a

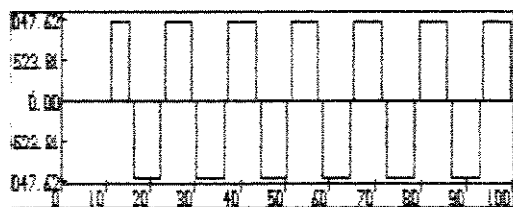


Fig 4.2.10b

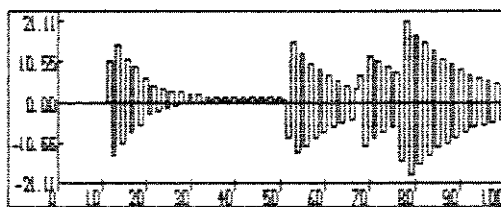


Fig 4.2.11b

Na figura 4.2.12 pode-se observar a evolução do processo mantendo as condições dos horizontes de previsão inicial e final como na simulação anterior, sendo que o horizonte de controle encontra-se com o valor 2. Este aumento sensibiliza o controlador

a erros no modelo (principalmente nos  $d-1$  primeiros coeficientes do polinômio B) pelo que o sistema ficou instável.

(1,5,2)

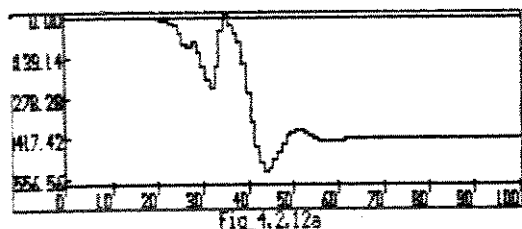


Fig. 4.2.12a

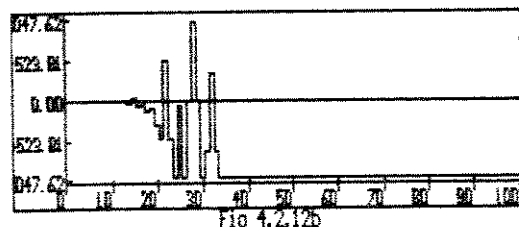


Fig. 4.2.12b

Quando o horizonte final é maior que o atraso do processo (pode acontecer quando o atraso não é conhecido e/ou variável), os resultados obtidos são em geral similares aos obtidos  $N_2 = 5$ . Por exemplo considere-se um horizonte de previsão final  $N_2 = 7$ ,  $N_1 = 1$  e  $N_u$  variando de 1 a 4. O desempenho do sistema nestas condições pode ser observado na sequência de figuras 4.2.13 a 4.2.16. Nestas é evidente que para valores do horizonte de controle menores que  $N_2 - d + 1$ , o sistema é estável, porém o seu desempenho degradou-se com respeito ao obtido com o conjunto de horizontes (1,5,1). Este mesmo desempenho foi obtido com o horizonte  $N_u = N_2 - d + 1 = 3$  como é observado na figura 4.2.15. Finalmente o aumento do horizonte  $N_u$  além desse valor provoca valores elevados da saída durante o regime transitório do processo (Figura 4.2.16).

(1,7,1)

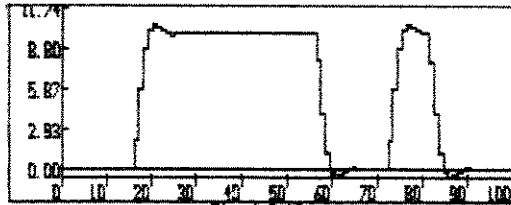


Fig. 4.2.13a

(1,7,2)

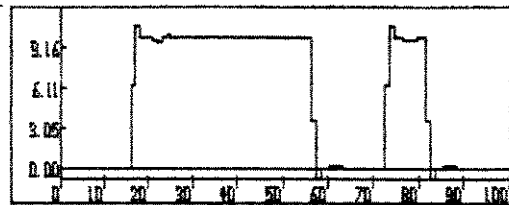


Fig. 4.2.14a

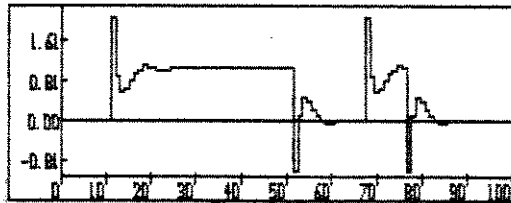


Fig. 4.2.13b

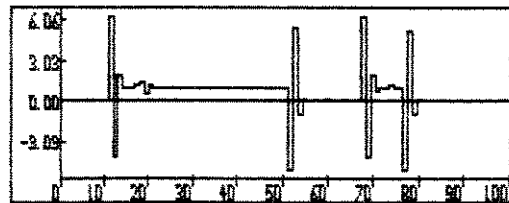


Fig. 4.2.14b

(1,7,3)

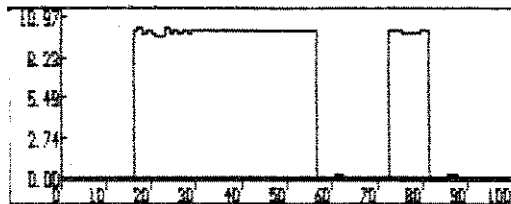


Fig. 4.2.15a

(1,7,4)

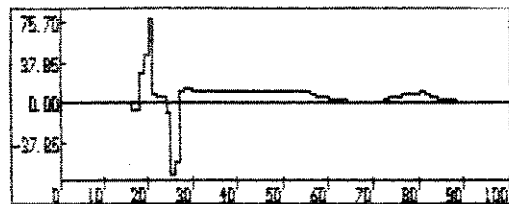


Fig. 4.2.16a

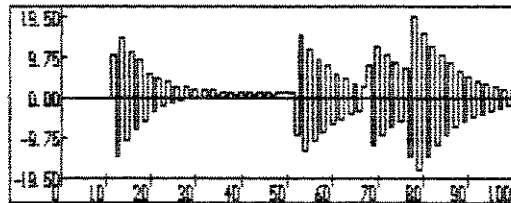


Fig. 4.2.15b

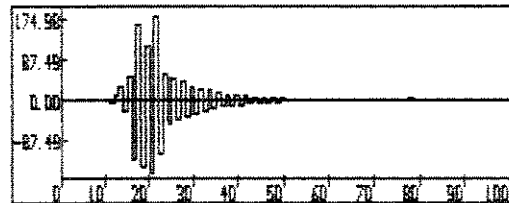


Fig. 4.2.16b

Nas figuras 4.2.17 e 4.2.18 observa-se a evolução apresentada pelo sistema para os conjuntos de horizontes (3,5,1) e (3,5,2) respectivamente. Na última das simulações pode-se observar instabilidade no processo provocada por um horizonte  $N_u$  maior que  $N_2 - d + 1$ . Na sequência de figuras 4.2.19 a 4.2.22 é apresentada a influência do horizonte de controle variando de 1 a 4 sendo o horizonte inicial  $N_1 = 3$  e o final  $N_2 = 7$ , onde pode-se observar que o horizonte  $N_1$  maior que a unidade não provocou alteração do desempenho do sistema.

(3,5,1)

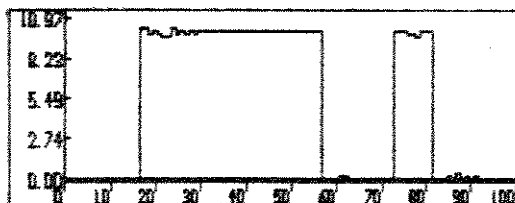


Fig. 4.2.17a

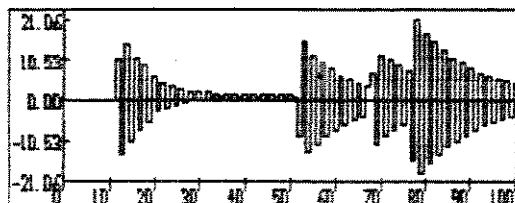


Fig. 4.2.17b

(3,5,2)

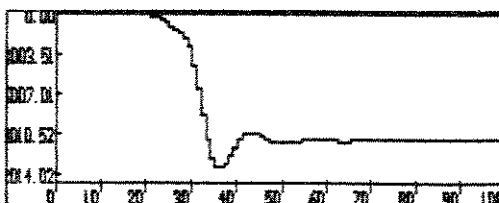


Fig. 4.2.18a

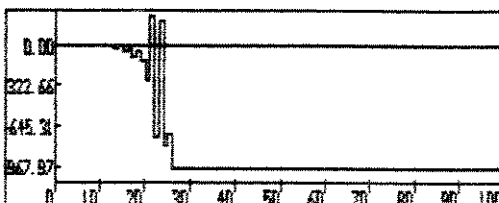


Fig. 4.2.18b

(3,7,1)

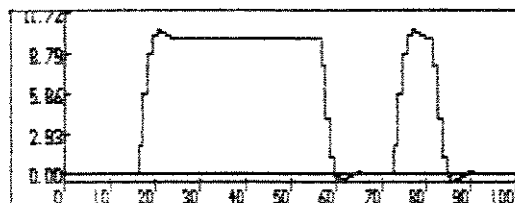


Fig. 4.2.19a

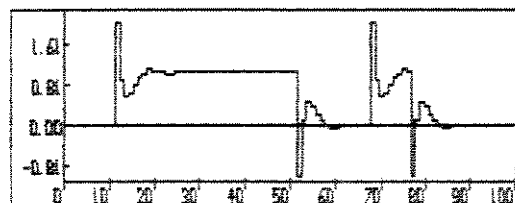


Fig. 4.2.19b

(3,7,2)

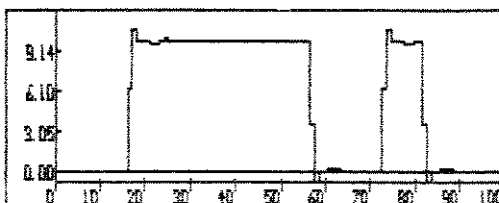


Fig. 4.2.20a

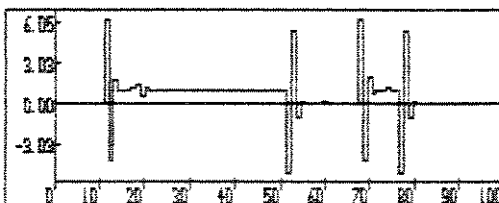


Fig. 4.2.20b

(3,7,3)

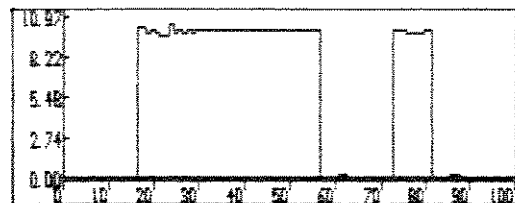


Fig. 4.2.21a

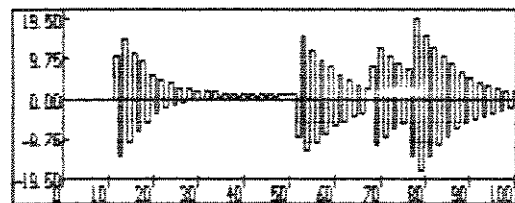


Fig. 4.2.21b

(3,7,4)

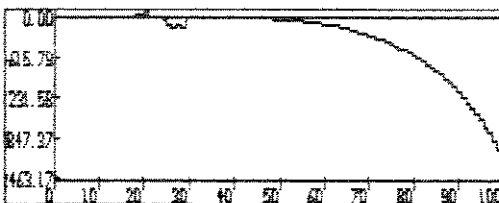


Fig. 4.2.22a

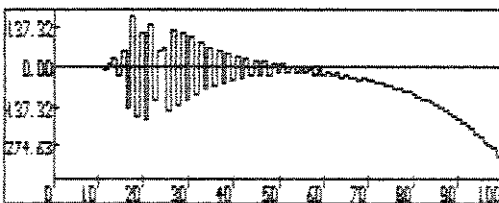
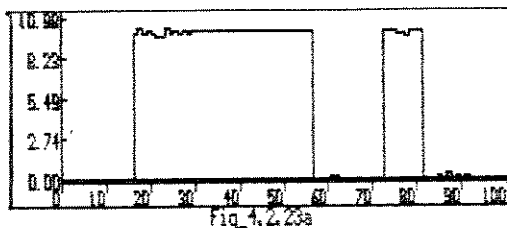


Fig. 4.2.22b

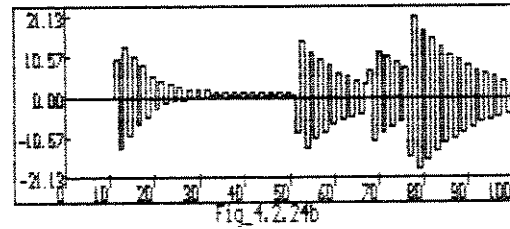
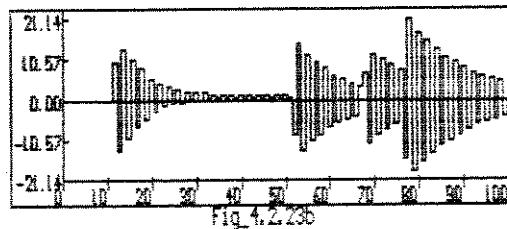
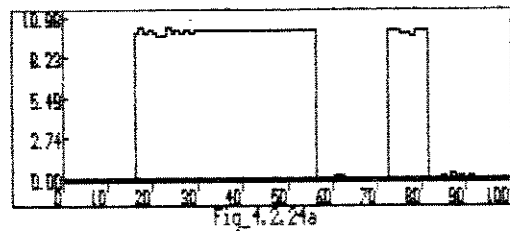
Quando o horizonte inicial é igual ao atraso do processo, as

condições requeridas para um bom desempenho nos outros horizontes são relaxadas. Nas figuras 4.2.23 a 4.2.25 observa-se a saída e a entrada do processo para os horizontes  $N1 = 5$ ,  $N2 = 5$  e  $Nu = 1, 3$  e 5 respectivamente. Desta sequência pode-se concluir que com  $N1 = N2 = d$  o horizonte de controle pode adquirir qualquer valor (menor ou igual a  $N2$ ) sem deteriorar o desempenho do controlador. Se o horizonte de previsão final for maior pode-se observar nas figuras 4.2.26 a 4.2.29, onde é simulado o processo para os horizontes  $N1 = 5$ ,  $N2 = 7$  e  $Nu = 1, 3, 6$  e 7, que para valores de  $Nu < N2 - d + 1$  a evolução é estável mas deteriorada com respeito à obtida para o conjunto  $(1,5,1)$ . Se  $Nu$  for igual ou maior do valor  $N2 - d + 1$ , a evolução do processo permanecerá inalterada.

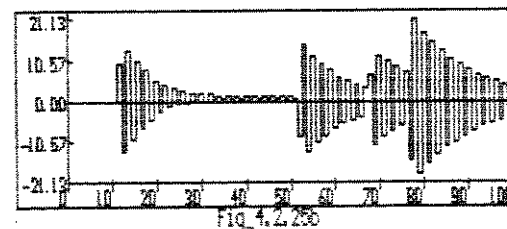
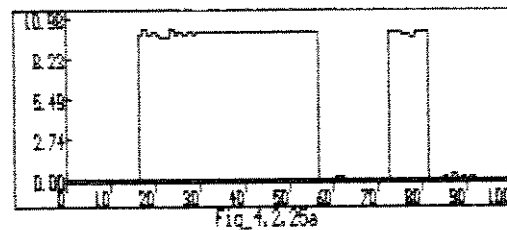
(5,5,1)



(5,5,3)



(5,5,5)



(5,7,1)

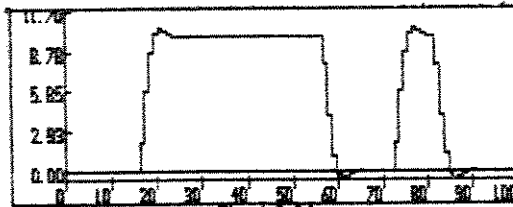


Fig. 4.2.26a

(5,7,3)

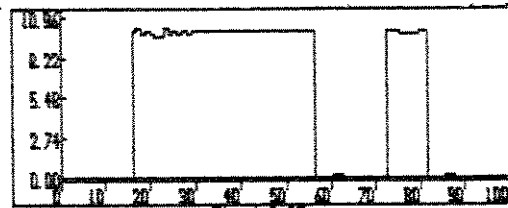


Fig. 4.2.27a

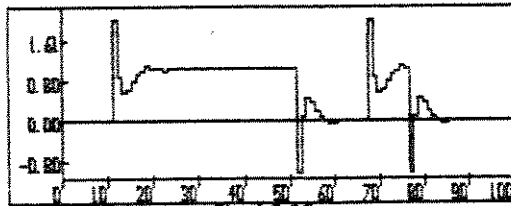


Fig. 4.2.26b

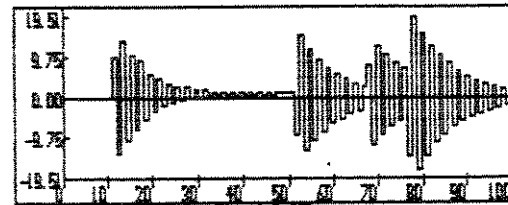


Fig. 4.2.27b

(5,7,6)

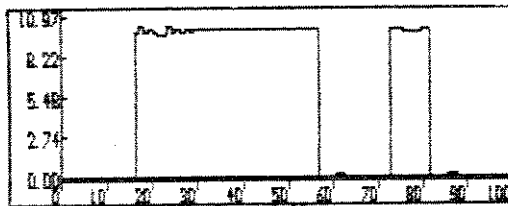


Fig. 4.2.28a

(5,7,7)

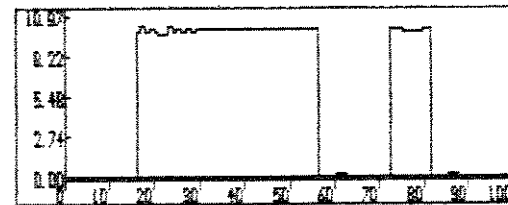


Fig. 4.2.29a

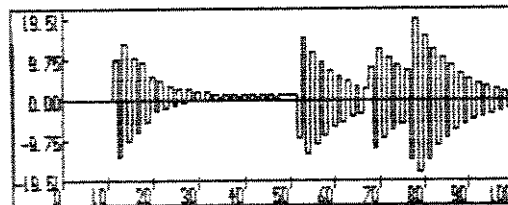


Fig. 4.2.28b

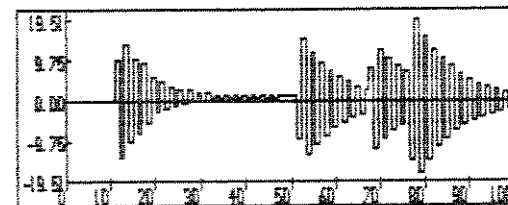


Fig. 4.2.29b

Se o horizonte inicial de previsão for maior que o atraso do processo, observa-se uma deterioração da saída que não pode ser corrigida com nenhuma seleção dos outros horizontes. Por exemplo para o horizonte inicial  $N_1 = 6$  e  $N_2 = 6$ , o horizonte de controle pode variar entre 2 e 6 sem alterar o desempenho (figuras 4.2.30, e 4.2.31), mesmo desempenho obtido com o conjunto de horizontes (6,10,10) mostrado na figura 4.2.32.

(6,6,2)

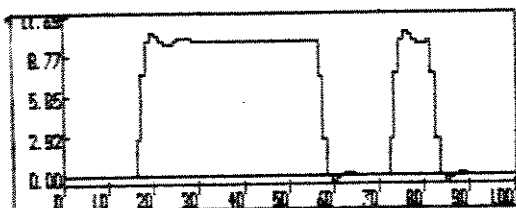


Fig. 4.2.30a

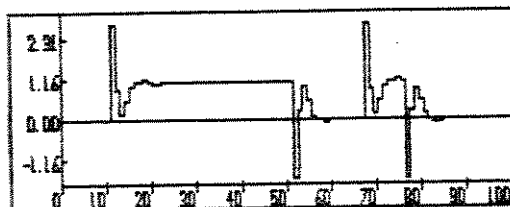


Fig. 4.2.30b

(6,6,6)

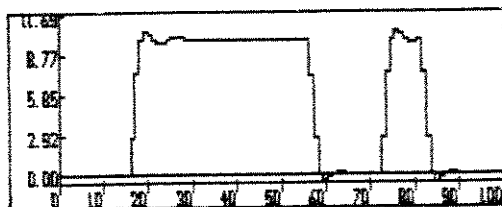


Fig. 4.2.31a

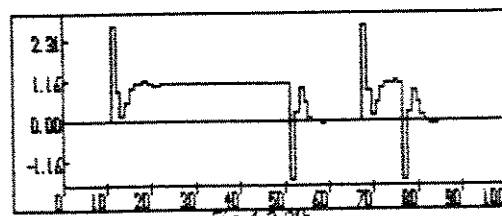


Fig. 4.2.31b

(6,10,10)

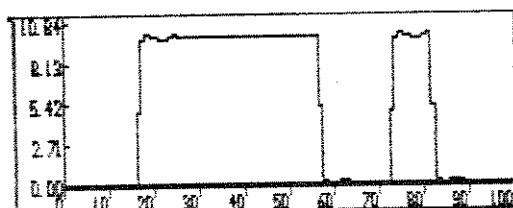


Fig. 4.2.32a

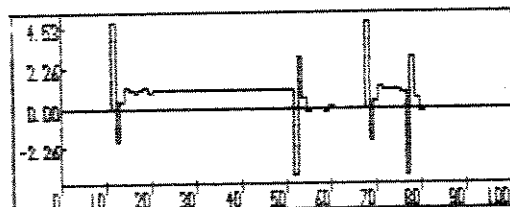


Fig. 4.2.32b

(1,9,9)

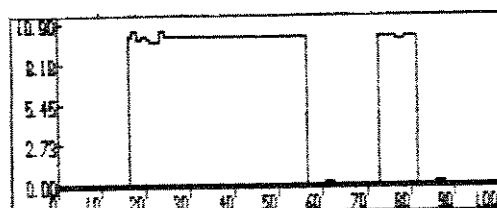


Fig. 4.2.33a

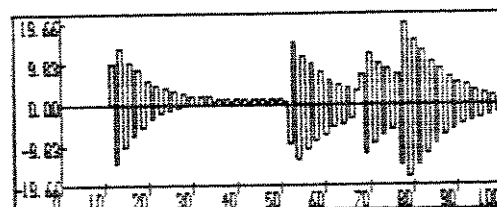


Fig. 4.2.33b

Finalmente foi tentado ajustar o controlador para as condições de "dead-beat" (i.e.,  $N1 = 1$ ,  $N2 = Nu$ ). Esta condição foi obtida para o conjunto de horizontes (1,9,9) mostrado na figura 4.2.33. O valor dos horizontes de controle e final corresponde a  $2d - 1$ , mas este desempenho não apresenta melhora com relação aos resultados obtidos com horizontes menores.

Da sequência de simulações anteriores pode se obter as seguintes condições a serem satisfeitas pelos horizontes para obter uma evolução adequada.

|                                                                             |                                                                                    |
|-----------------------------------------------------------------------------|------------------------------------------------------------------------------------|
|                                                                             | i). $N1 < d$ ;<br>ii). $N2 \geq d$ ;<br>iii). $Nu = N2 - d + 1$ .                  |
| ou                                                                          | <hr/> i). $N1 = d$ ;<br>ii). $Nu \geq d$ ;<br>iii). $N2 - d + 1 \leq Nu \leq N2$ . |
| ou                                                                          | <hr/> i). $N1 = 1$ ;<br>ii). $N2 \geq 2d - 1$ ;<br>iii). $Nu = N2$ .               |
| TABELA 4.2 escolha dos horizontes para processos estáveis e de fase mínima. |                                                                                    |

Como se observa da tabela 4.2, as condições de bom desempenho encontradas para o controle do processo 1 correspondem ao segundo caso da tabela.

Dos resultados obtidos anteriormente pode-se propor o seguinte conjunto de regras da base de conhecimento para configuração do Controlador Preditivo Generalizado:

SE o processo *É* de fase mínima e estável em malha aberta  
 E o atraso do processo *É* constante  
 E o atraso do processo *É* conhecido  
 ENTÃO  $N1 =$  o atraso do processo.

SE o processo *É* de fase mínima e estável em malha aberta  
 E (o atraso do processo *É* variável  
   OU o atraso do processo *NÃO É* conhecido)  
 ENTÃO  $N1 = 1$  (default).

SE o processo *É* de fase mínima e estável em malha aberta  
 E o atraso do processo *É* conhecido  
 ENTÃO  $N2 =$  atraso do processo (menor valor possível).

SE o processo *É* de fase mínima e estável em malha aberta  
E (o atraso do processo *É* variável  
OU o atraso do processo *NÃO É* conhecido)  
E o máximo atraso do processo *É* conhecido  
ENTÃO  $N2 = \text{o máximo atraso do processo.}$

SE o processo *É* de fase mínima e estável em malha aberta  
E (o atraso do processo *É* variável  
OU o atraso do processo *NÃO É* conhecido)  
E o máximo atraso do processo *NÃO É* conhecido  
ENTÃO  $N2 = 10$  (default).

SE o processo *É* de fase mínima e estável em malha aberta  
E o atraso do processo *É* conhecido  
E  $N1$  *É IGUAL AO* atraso do processo  
ENTÃO  $Nu = N2 - \text{o atraso do processo} + 1.$

SE o processo *É* de fase mínima e estável em malha aberta  
E (o máximo atraso do processo *NÃO É* conhecido  
OU  $N1$  *É MENOR QUE* o atraso máximo do processo)  
ENTÃO  $Nu = N2 - \text{o atraso máximo do processo} + 1.$ <sup>1</sup>

Um valor pre-estabelecido de 10 além de ser o valor default sugerido pela maioria dos trabalhos sobre GPC, parece ser um valor razoável, devido a que como é notado em [BUE89], não é aconselhável que na discretização de um processo, o tempo de amostragem provoque valores de atraso muito altos.

Para a base de conhecimento de supervisão (encadeamento direto) pode-se propor as seguintes regras:

---

<sup>1</sup> Esta condição não garante o melhor desempenho, mas garante a estabilidade do processo, e ainda a ausência de amplitudes elevadas da saída durante o regime transitório.

SE o processo É de fase mínima e estável em malha aberta  
E o sistema É instável  
E  $Nu$  É IGUAL A 1  
ENTÃO sugere-se aumentar  $N2$ .

SE o processo É de fase mínima e estável em malha aberta  
E (o sistema É instável  
OU a variância da saída É MAIOR QUE  
a máxima variância permitida)  
E  $Nu$  É MAIOR QUE 1  
ENTÃO sugere-se diminuir  $Nu$ .

SE o processo É de fase mínima e estável em malha aberta  
E o sistema É estável  
E tempo de subida É MAIOR QUE o tempo de subida desejado  
E a ordem do filtro P É IGUAL A 0  
ENTÃO sugere-se aumentar  $Nu$ .

Estas regras são obtidas das análises realizadas anteriormente e resumidas a seguir:

I).-Aumentar o horizonte de previsão final torna a resposta do sistema mais suave.

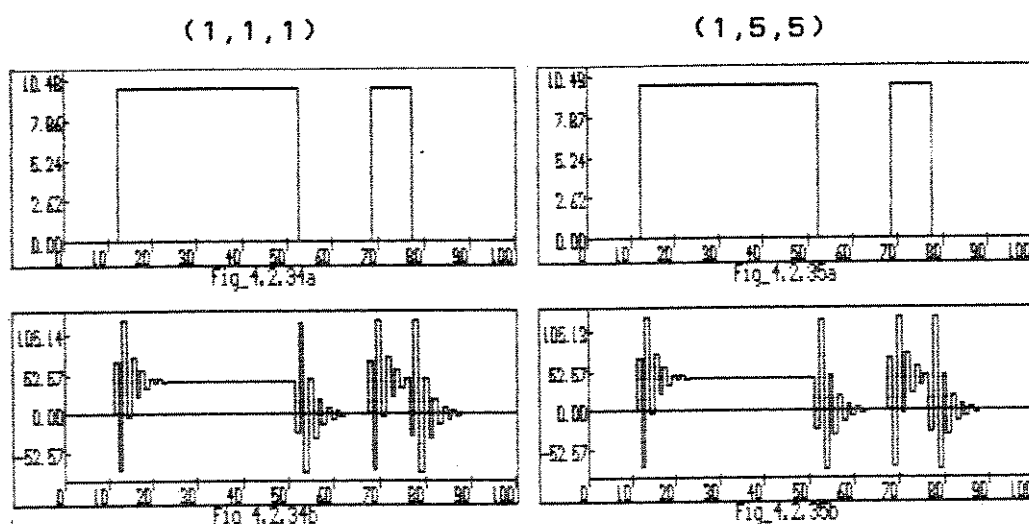
II).-Quando  $Nu < N2 - d + 1$ , a resposta do sistema também é mais lenta, mas aumentar o seu valor além de  $N2 - d + 1$ , geralmente não provoca instabilidade, mas sim amplitudes maiores no regime transitório.

III).-Finalmente a melhor condição sempre é obtida com  $Nu = N2 - d + 1$  (menor  $Nu$  com evolução equivalente à de "dead-beat"). Aumentar este horizonte quando o tempo de crescimento é menor que o desejado (sempre que o filtro P não esteja atuando) fará o sistema se comportar mais ativamente, sem correr o risco de instabilidade que se tem tentando obter o mesmo efeito diminuindo  $N2$ .

#### 4.2.1.2.-PROCESSOS INSTÁVEIS EM MALHA ABERTA E DE FASE MÍNIMA

O processo escolhido para ser simulado com as características indicadas no título, é o número 3 (Tabela 4.1). Nas raízes (mostradas na figura 4.1.3) pode-se observar um par de polos fora do círculo unitário.

A simulação para o conjunto de horizontes (1,1,1) é mostrado na figura 4.2.34, onde observa-se um perfeito seguimento da referência. O aumento dos horizontes de previsão e de controle não acarreta melhora como pode ser observado na figura 4.2.35, onde os horizontes são a tripla (1,5,5).



Como este processo tem atraso unitário, simula-se também o processo 4, também instável em malha aberta mas com um atraso de transporte igual a 4. Na figura 4.2.36 mostra-se a evolução (instável) obtida para o conjunto de horizontes (1,1,1). A estabilidade é obtida para os horizontes (1,4,1), como é mostrado na figura 4.2.37, sendo que nas figuras 4.2.38 e 4.2.39 pode ser observado o desempenho do controlador para o conjunto de horizontes (4,4,1) e (4,4,4) respectivamente. Não são notadas diferenças significativas com os resultados previamente obtidos, o que mostra que as regras citadas para um processo "estável em malha aberta e de fase mínima" também podem ser utilizadas para processos "instáveis em malha aberta e de fase mínima".

(1,1,1)

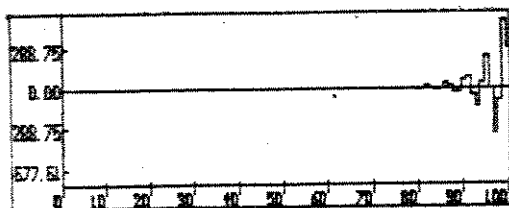


Fig. 4.2.36a

(1,4,1)

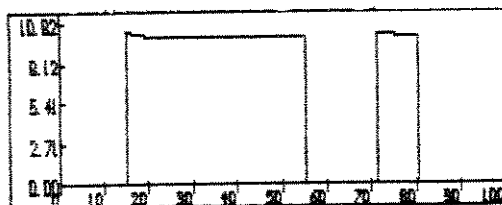


Fig. 4.2.37a

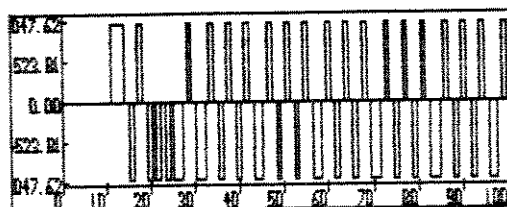


Fig. 4.2.36b

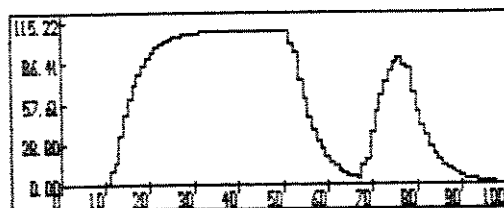


Fig. 4.2.37b

(4,4,1)

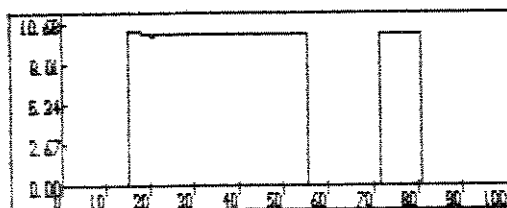


Fig. 4.2.38a

(4,4,4)

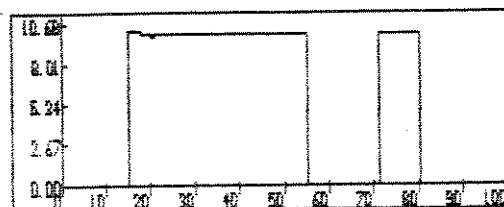


Fig. 4.2.39a

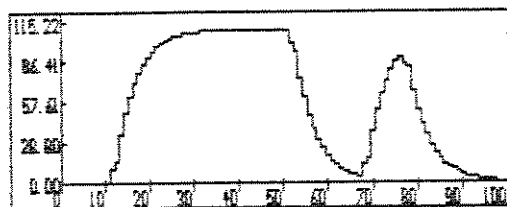


Fig. 4.2.38b

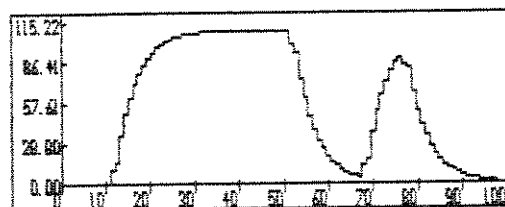


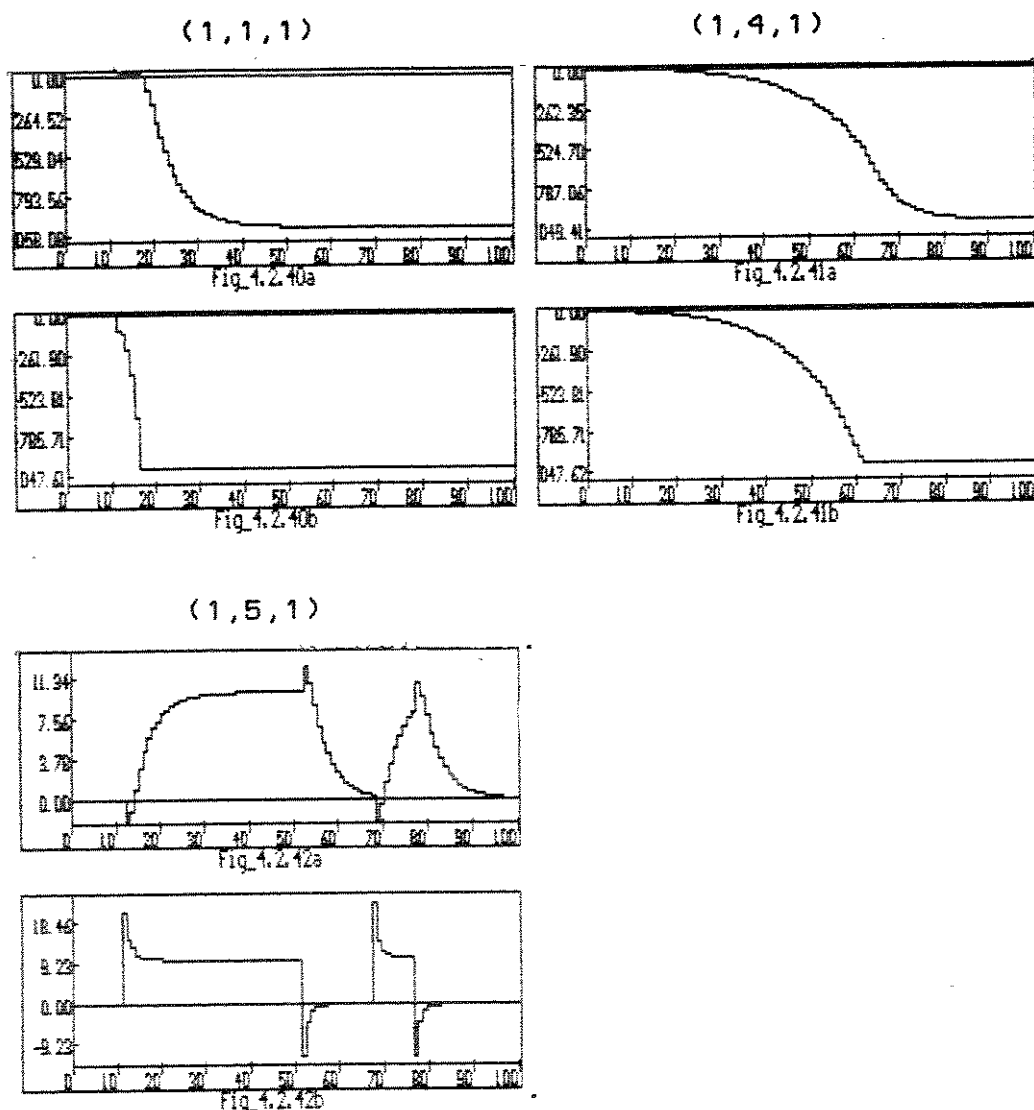
Fig. 4.2.39b

#### 4.2.1.3.-PROCESSOS ESTÁVEIS EM MALHA ABERTA E DE FASE NÃO MÍNIMA

Este tipo de processos são os que apresentam maiores problemas para a seleção correta dos horizontes, sob as condições impostas (i.e., sem ponderações nem filtros). Simula-se inicialmente o processo 5, que possui características de fase não mínima, conforme descrito na tabela 4.1.

Como foi realizado na simulações anteriores, utiliza-se inicialmente o conjunto de horizontes (1,1,1). O resultado desta simulação é mostrado na figura 4.2.40, onde se observa que o sistema não consegue controlar o processo. Na figura 4.2.41, é

mostrada a saída do processo para os horizontes (1,4,1), onde novamente pode-se observar que o processo não foi controlado. Na figura 4.2.42 o sistema é estável, para o conjunto de horizontes (1,5,1). Entretanto não existe nenhum critério que relacione as características do processo com o valor do horizonte final de previsão.



Nas figuras 4.2.43, 4.2.44 e 4.2.45, procura-se analisar a influência do horizonte de controle variando-o de 2 até 4, com  $N_1 = 1$  e  $N_2 = 5$ . Nessa sequência pode-se observar que o sistema fica instável para o horizonte de controle igual a 4, esta característica sempre ocorre quando o horizonte inicial é igual ao atraso do processo, e o horizonte de controle é maior que o

horizonte final menos a quantidade de zeros fora do círculo unitário<sup>2</sup>.

(1,5,2)

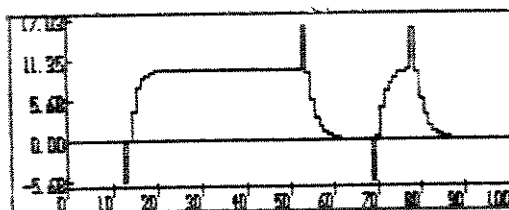


Fig. 4.2.43a

(1,5,3)

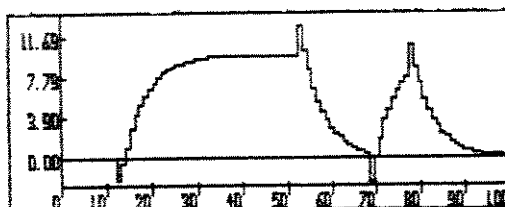


Fig. 4.2.44a

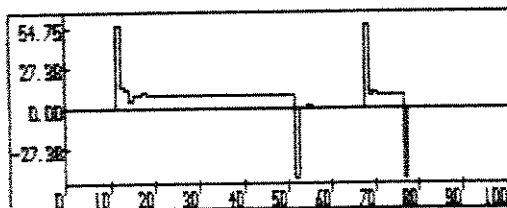


Fig. 4.2.43b

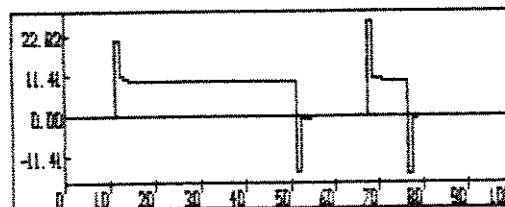


Fig. 4.2.44b

(1,5,4)

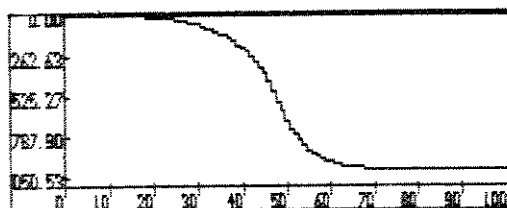


Fig. 4.2.45a

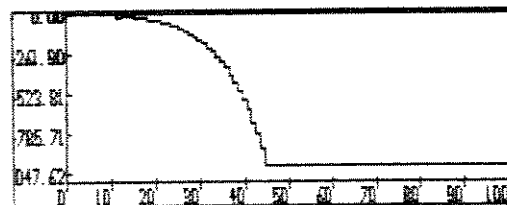


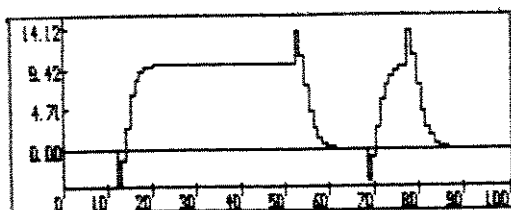
Fig. 4.2.45b

A simulação do processo para os horizontes  $N_1 = 2$ ,  $N_2 = 5$  e  $N_u$  variando de 1 até 5 pode ser observada na sequência de figuras 4.2.46 a 4.2.50. Desta série de simulações pode-se concluir que o aumento do horizonte de controle além da ordem do processo mais uma unidade não provoca nenhum benefício, já que a saída permanece idêntica à obtida para os horizontes de controle 3 e 4. Entretanto, observa-se uma deterioração quando o horizonte de

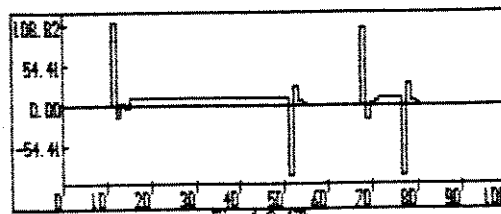
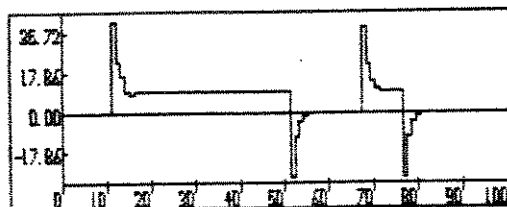
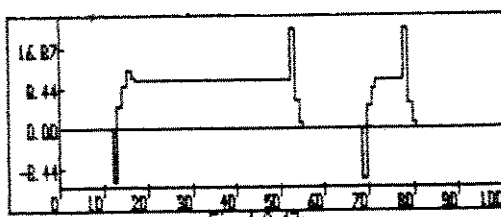
<sup>2</sup> Como uma análise detalhada do processo em malha fechada foge ao escopo deste trabalho, não foi analisado a alocação de pólos obtida nesta situação, porém os valores parecem sugerir que com esses horizontes o controlador acha-se na condição "dead-beat".

controle é igual ao horizonte final de previsão.

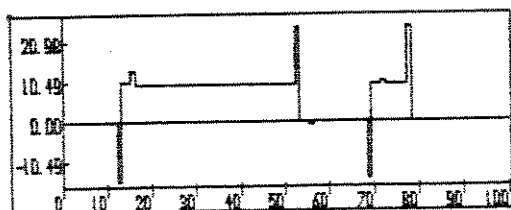
(2,5,1)



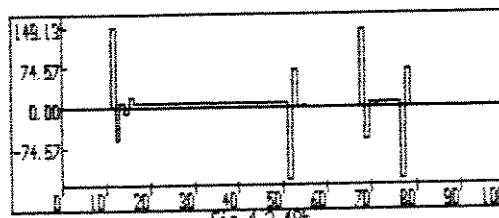
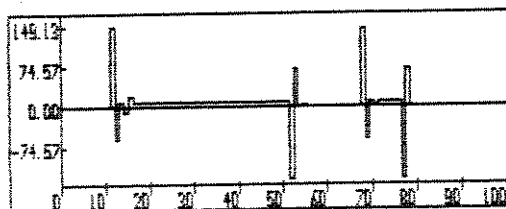
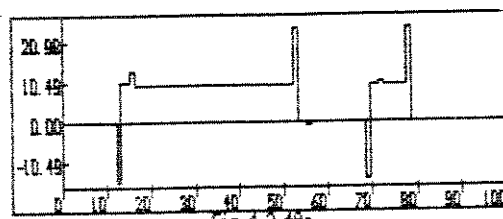
(2,5,2)



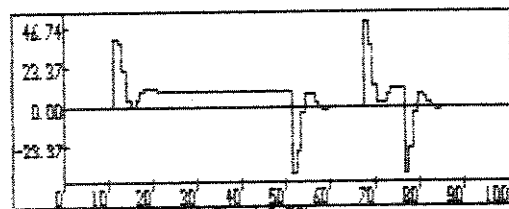
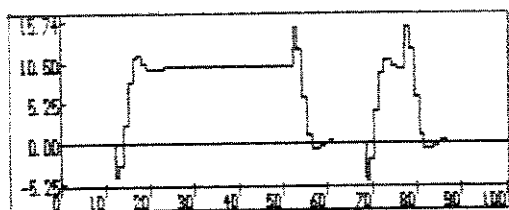
(2,5,3)



(2,5,4)

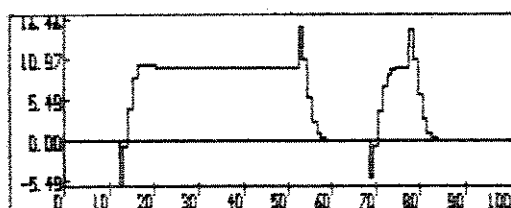


(2,5,5)

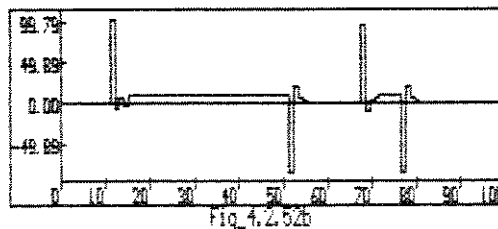
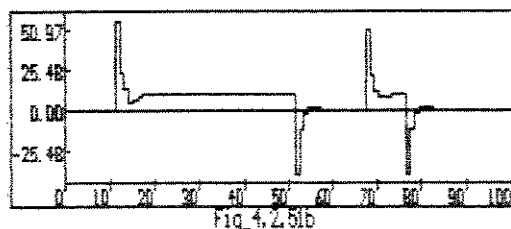
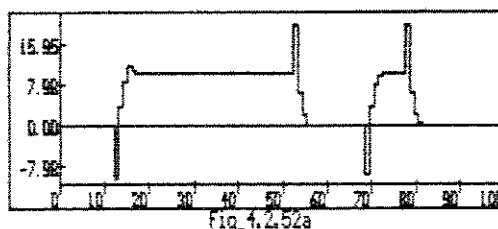


A influência do aumento do horizonte de previsão inicial é ilustrada nas figuras 4.2.51 a 4.2.55, onde se mostra a saída do processo para os horizontes  $N1 = 3$ ,  $N2 = 5$  e  $Nu$  novamente variando entre 1 e 5. Duas conclusões podem ser obtidas destas simulações. i) Não se obtém nenhuma melhora no desempenho quando o horizonte inicial é maior que o atraso mais a quantidade de zeros fora do círculo unitário (proposta). ii) O melhor horizonte de controle para este processo é  $na+1$ .

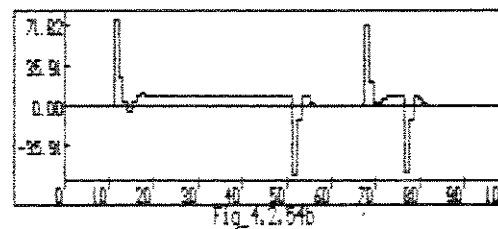
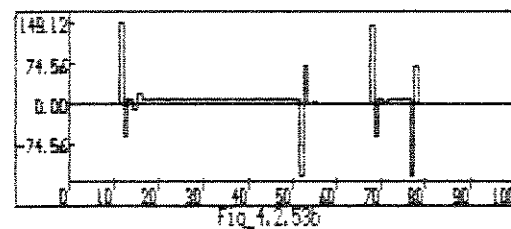
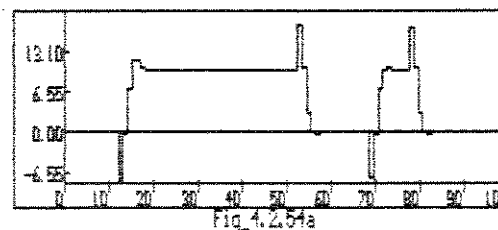
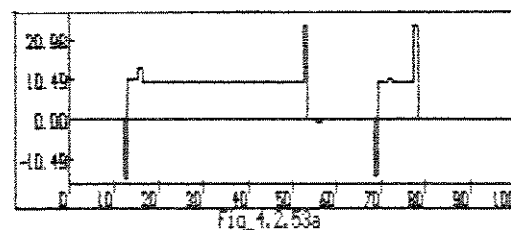
(3,5,1)



(3,5,2)



(3,5,3)



(3,5,5)

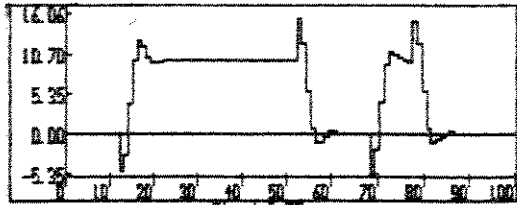


Fig. 4.2.55a

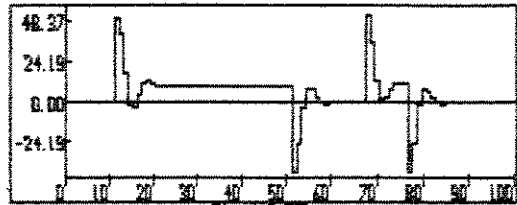


Fig. 4.2.55b

Como o horizonte de previsão final foi obtido sob as condições ( $N_1 = 1$  e  $N_u = 1$ ), deve-se verificar se não existe algum outro valor deste horizonte que permita melhorar o desempenho do sistema ou ainda aumentar a velocidade de cálculo. Assim a sequência de figuras 4.2.56 a 4.2.59 mostra o desempenho do sistema para os horizontes  $N_1 = 2$ ,  $N_2 = 4$  e  $N_u$  variando de 1 a 4. Nesta sequência pode-se observar novamente que o melhor desempenho é obtido para o horizonte de controle 3 ( $n_{a+1}$ ). A estabilidade do sistema é obtida para valores inferiores (i.e.,  $(2,4,3)$ ) ao obtido anteriormente. Também pode-se observar que o desempenho do sistema foi análogo ao obtido com os mesmos horizontes inicial e de controle e com o horizonte final com valor 5 (fig 4.2.48). A simulação com o horizonte final de previsão com valor 3 demonstrou ser instável para todas as combinações possíveis dos outros horizontes.

(2,4,1)

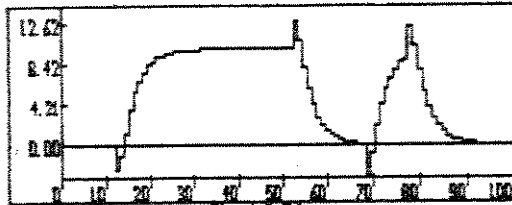


Fig. 4.2.56a

(2,4,2)

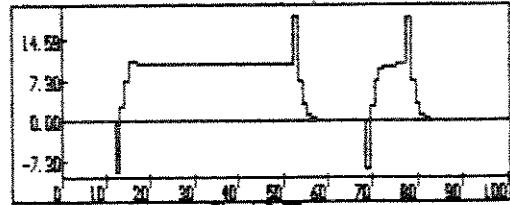


Fig. 4.2.57a

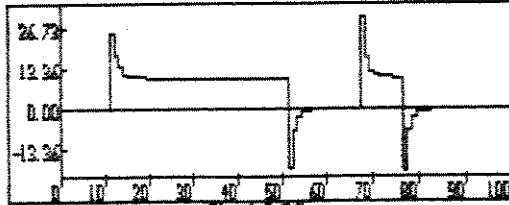


Fig. 4.2.56b

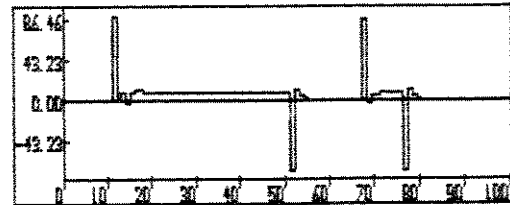


Fig. 4.2.57b

(2,4,3)

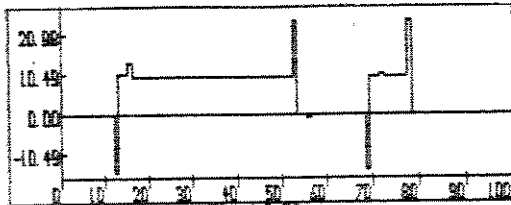


Fig. 4.2.58a

(2,4,4)

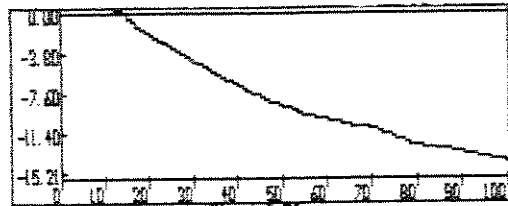


Fig. 4.2.59a

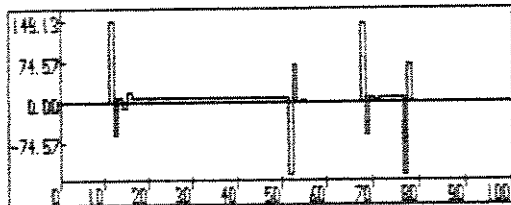


Fig. 4.2.58b

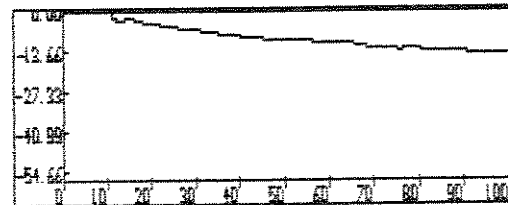


Fig. 4.2.59b

Finalmente na figura 4.2.60 é observada a evolução da saída deste processo para os horizontes  $N_1 = 2$ ,  $N_2 = 8$  e  $N_u = 3$ . Este resultado ilustra que a utilização de um horizonte de previsão final maior que do que o horizonte de controle mais a quantidade de zeros de malha aberta fora do círculo unitário não melhora o desempenho do sistema.

(2,8,3)

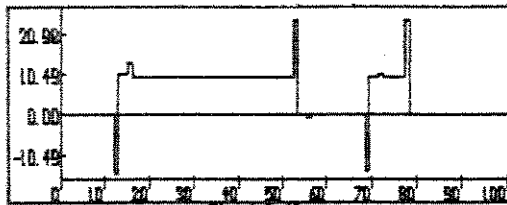


Fig. 4.2.60a

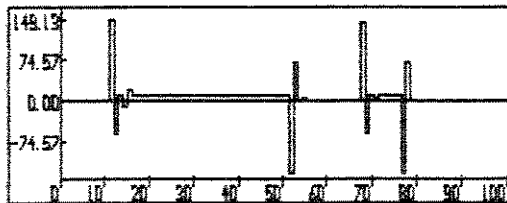


Fig. 4.2.60b

A partir destes resultados de simulação propõe-se a escolha de horizontes para sistema de fase não mínima conforme na tabela 4.3 dada a seguir:

- i).-  $N1 = d + \text{Quantidade de zeros de malha aberta fora do círculo unitário};$
- ii).-  $Nu = na + 1;$
- iii).-  $N2 \geq Nu + \text{quantidade de zeros de malha aberta fora do círculo unitário}.$

TABELA 4.3 escolha dos horizontes para processos estáveis e de fase não mínima.

Outro processo de fase não mínima simulado é o  $N^{\circ} 6$  (tabela 4.1). Utilizando-se os valores descritos na tabela 4.3, pode-se propôr os horizontes (3,8,6) para o controle deste processo. A figura 4.2.61 mostra a saída do sistema quando são utilizados esses horizontes. Duas características podem ser observadas: i) existe uma pequena oscilação após o primeiro degrau devido aos erros de modelagem, que desaparece no segundo degrau. ii) O sistema tem uma resposta relativamente lenta. Este fato ocorre devido a estar-se numa condição que não corresponde à de "dead-beat". Esta condição não pode ser obtida devido ao processo ser de fase não mínima, sendo esta não otimalidade o preço a ser pago pela estabilidade.

(3,8,6)

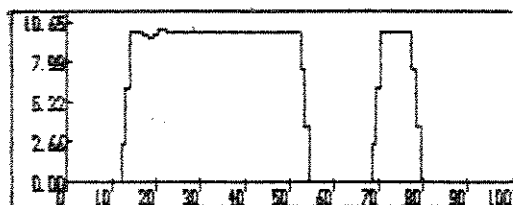


Fig. 4.2.61a

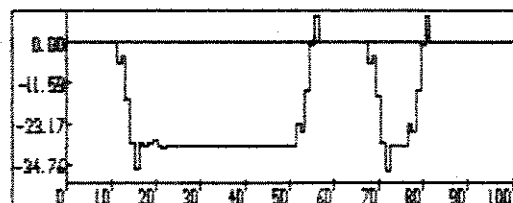


Fig. 4.2.61b

As figuras 4.2.62 e 4.2.63 ilustram a saída do processo para os horizontes  $N1 = 3$ ,  $Nu = 6$  e o horizonte  $N2$  com os valores 7 e 9 respectivamente. Destas figuras pode-se verificar que: i) um horizonte de previsão final menor que 8 ( $Nu - QZFNM^3$ ) provoca uma deterioração da saída do processo, e ii) um horizonte maior que este valor não melhora o desempenho e aumenta-se o tempo de processamento. Nas figuras 4.2.64 e 4.2.65 é mostrada a influência do horizonte de controle para os valores 5 e 7, estando os outros horizontes fixados nos valores  $N1 = 3$  e  $N2 = 8$ . Destas simulações pode-se verificar uma degradação do desempenho quando comparado com o obtido na figura 4.2.61. Finalmente nas figuras 4.2.66 e 4.2.67 é apresentada a influência do horizonte inicial de previsão para os valores 2 e 4, com  $N2 = 8$  e  $Nu = 6$ . Nestas figuras pode-se observar a deterioração da saída com respeito ao valor obtido com o horizonte inicial igual a 3, o que permite afirmar<sup>4</sup> que o melhor desempenho foi obtido no primeiro exemplo de processo de fase não mínima com os horizontes escolhidos conforme descrito na tabela 4.3.

---

<sup>3</sup>Quantidade de Zeros de Fase Não Mínima

<sup>4</sup>Esta afirmação também está baseada em outras simulações realizadas e não descritas neste trabalho.

(3,7,6)

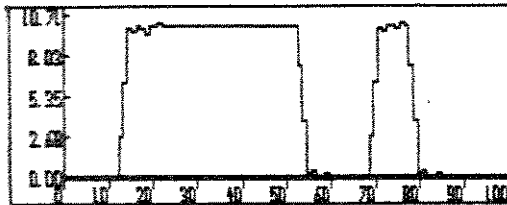


Fig. 4.2.62a

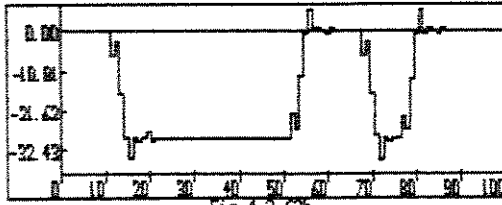


Fig. 4.2.62b

(3,9,6)

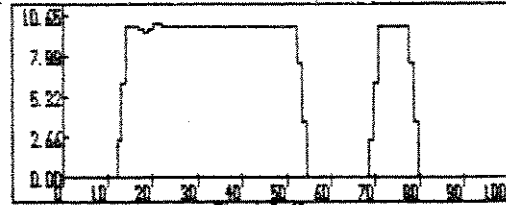


Fig. 4.2.63a

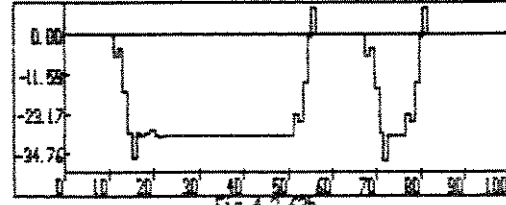


Fig. 4.2.63b

(3,8,5)

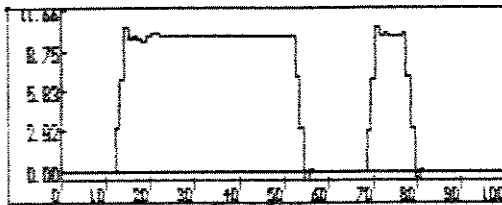


Fig. 4.2.64a

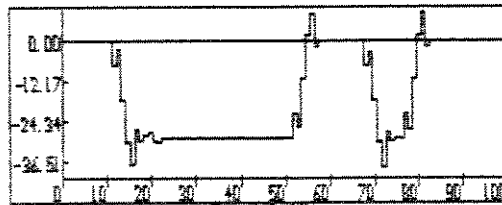


Fig. 4.2.64b

(3,8,7)

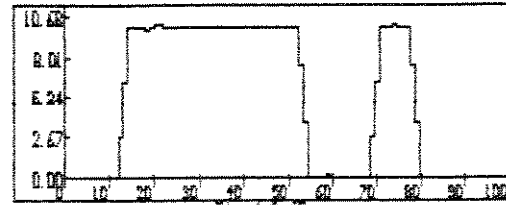


Fig. 4.2.65a

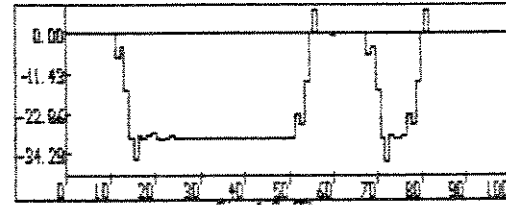


Fig. 4.2.65b

(2,8,6)

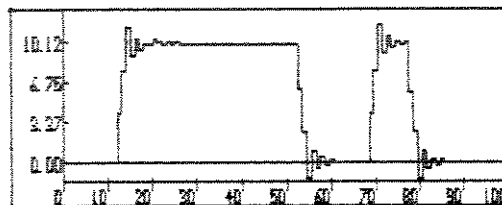


Fig. 4.2.66a

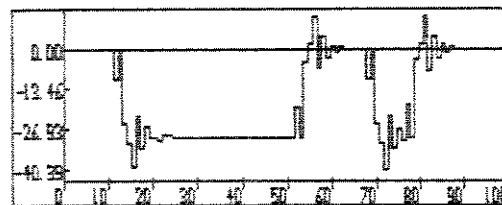


Fig. 4.2.66b

(4,8,6)

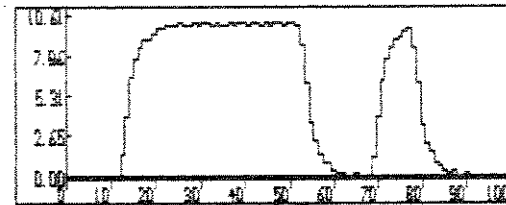


Fig. 4.2.67a

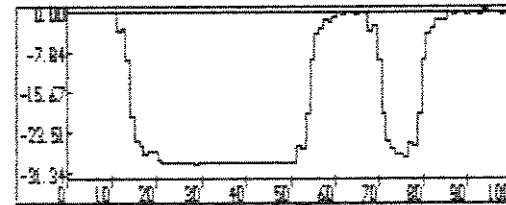


Fig. 4.2.67b

Destes resultados pode-se propor as seguintes regras para a

base de conhecimento de configuração:

SE o processo  $\acute{E}$  de fase não mínima e estável em malha aberta  
E os parâmetros do processo SÃO constantes  
E a quantidade de zeros de fase não mínima  $\acute{E}$  conhecida  
E o atraso do processo  $\acute{E}$  conhecido

ENTÃO  $N1 =$  a quantidade de zeros de fase não mínima +  
o atraso do processo.

SE o processo  $\acute{E}$  de fase não mínima e estável em malha aberta  
E a quantidade de zeros de fase não mínima NÃO  $\acute{E}$  conhecida  
ENTÃO  $N1 = 1$ .

SE o processo  $\acute{E}$  de fase não mínima e estável em malha aberta  
E a ordem do polinômio  $A(z^{-1})$   $\acute{E}$  conhecida  
ENTÃO  $Nu = na + 1$ .

SE o processo  $\acute{E}$  de fase não mínima e estável em malha aberta  
E o horizonte inicial de previsão ( $N1$ ) ESTÁ disponível  
E o horizonte de controle ( $Nu$ ) ESTÁ disponível  
E a quantidade de zeros de fase não mínima  $\acute{E}$  conhecida  
E  $N1 + 1 \acute{E}$  IGUAL OU MAIOR QUE  $Nu +$  a quantidade de zeros de fase  
não mínima

ENTÃO  $N2 = N1 + 1$ .

SE o processo  $\acute{E}$  de fase não mínima e estável em malha aberta  
E o horizonte inicial de previsão ( $N1$ ) ESTÁ disponível  
E o horizonte de controle ( $Nu$ ) ESTÁ disponível  
E a quantidade de zeros de fase não mínima NÃO  $\acute{E}$  conhecida  
E  $N1 + 1 \acute{E}$  MENOR QUE  $Nu +$  a ordem do polinômio  $B(z^{-1})$   
ENTÃO  $N2 = Nu +$  a ordem do polinômio  $B(z^{-1})$ .

SE o processo  $\acute{E}$  de fase não mínima e estável em malha aberta  
E o horizonte inicial de previsão ( $N1$ ) ESTÁ disponível  
E o horizonte de controle ( $Nu$ ) ESTÁ disponível  
E  $N1 + 1 \acute{E}$  MENOR QUE  $Nu +$  a quantidade de zeros de fase não mínima  
ENTÃO  $N2 = Nu +$  a ordem do polinômio  $B$ .

Da mesma forma como foi realizado para processos de fase mínima, das simulações pode ser proposta a seguinte regra para a base de conhecimento de supervisão:

SE o processo  $\hat{E}$  de fase não mínima

E o sistema  $\hat{E}$  instável

E  $N_2 \hat{E}$  MENOR QUE  $N_u$  - a ordem do polinômio B

ENTÃO sugere-se aumentar  $N_2$ .

As simulações realizadas não permitem obter outro conhecimento "confiável" para ampliar a base de conhecimento de supervisão.

#### 4.2.2.-ESCOLHA DOS FILTROS DO CONTROLADOR

Como foi descrito no capítulo II, o GPC similarmente ao GMV possui um conjunto de filtros que permitem alterar a evolução do sistema em malha fechada mas, como já foi dito, ao contrário do que ocorre no GMV, no GPC estes filtros não são imprescindíveis para garantir a estabilidade. As principais funções destes filtros são de:

- i).-Satisfazerem requisitos de desempenho determinístico
- e
- ii).-Diminuir os efeitos das perturbações.

A seguir é analisada a função específica dos filtros P e R discutidos na seção 2.8.

#### 4.2.2.1.-OS FILTROS $P(z^{-1})$ E $R(z^{-1})$

Na seção 2.8 do capítulo II, foi demonstrado que sob certas condições, o GPC tem um desempenho do tipo "dead-beat", e a saída é obtida a partir de um modelo descrito pela função de transferência  $R(z^{-1})/P(z^{-1})$ .

É um fato conhecido que os requisitos de desempenho do sistema em malha fechada são geralmente fornecidos como um conjunto de especificações que determinam o comportamento estocástico e o comportamento determinístico. Este último comportamento é geralmente definido por um dos seguintes fatores:

i).- $\zeta$  e  $\omega$ , Fator de amortecimento e largura de faixa, ambos em malha fechada;

ii).- $p_1$  e  $p_2$ , par de pólos desejados em malha fechada;

iii).- $P_m(z^{-1})$ , um polinômio de ordem dois que determina os pólos em malha fechada;

iv).-tempo de subida ( $T_r$ ) e sobre-elevação ( $S_p$ ) desejados em malha fechada.

sendo estas quatro formas de especificar o comportamento determinístico em malha fechada equivalentes.

Assim parece uma escolha natural que  $P(z^{-1})$  represente o polinômio denominador da função de transferência de malha fechada.  $R(z^{-1})$  é escolhido como um polinômio tal que o ganho em malha fechada seja unitário, isto é  $R(1)/P(1) = 1$ . Neste trabalho adota-se a escolha mais simples<sup>5</sup> com  $n_R = 0$ , e  $r_0 = P(1)$ .

---

<sup>5</sup>O polinômio  $R$  poderia ser escolhido para penalizar variações muito rápidas na referência. Esta função não foi explorada no presente trabalho.

Na figura 4.2.69 é apresentada a evolução de um modelo definido pela equação

$$(1 - 1 z^{-1} + 0.35 z^{-2}) y(t) = 0.35 u(t-1)$$

que possui um par de pólos complexos conjugados em  $0.5 \pm 0.3162i$  como pode ser observado no lugar de raízes é apresentado na figura 4.2.68.

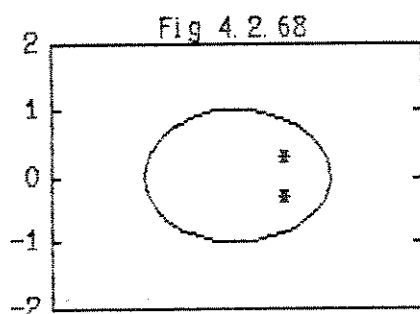


Fig. 4.2.68 Lugar de raízes do modelo

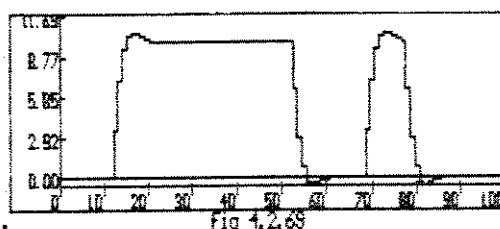


Fig. 4.2.69 Saída do modelo em malha aberta

Na figura 4.2.70 pode-se observar a evolução da saída do processo 1 para o conjunto de horizontes (1,1,1) (note que este conjunto de valores permite uma estratégia de "dead-beat") e com o polinômio  $A(z^{-1})$  do modelo anterior como filtro  $P(z^{-1})$ , e o polinômio  $B(z^{-1})$  como filtro  $R(z^{-1})$ . Nessa figura pode-se observar um desempenho idêntico ao do modelo tanto em relação ao tempo de subida como em relação à sobre-elevação.

O mesmo modelo é utilizado para os filtros P e R do sistema controlando o processo 5 (de fase não mínima) com o conjunto de horizontes (2,4,3) como é sugerido pelas regras obtidas para este tipo de processos. Os resultados desta simulação são apresentados na figura 4.2.71. Nesta pode-se observar que ainda que o pico de transitório negativo não seja eliminado, ele é reduzido. O tempo de subida do modelo e do sistema controlado é quase o mesmo, sendo que a sobre-elevação (após a correta convergência dos parâmetros, ou seja após a iteração 50), foi ligeiramente menor no processo controlado devido a não se ter as condições de "dead-beat" (pela

escolha dos horizontes necessárias para um processo de fase não mínima). Apesar desta diferença neste processo, indubitavelmente, os valores dos horizontes obtidos da tabela 2.3 permitem uma aproximação geralmente aceitável do sistema em malha fechada ao modelo desejado.

Proc. Nº1,  $(1, 1, 1)$   
 $P = 1 - 1z^{-1} + 0.35z^{-2}$   
 $R = 0.35$

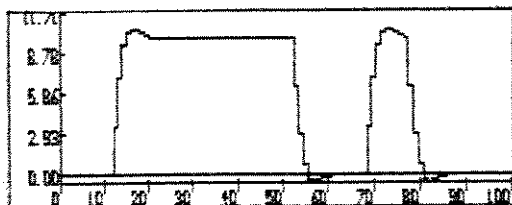


Fig. 4.2.70a

Proc. Nº5,  $(2, 4, 3)$   
 $P = 1 - 1z^{-1} + 0.35z^{-2}$   
 $R = 0.35$

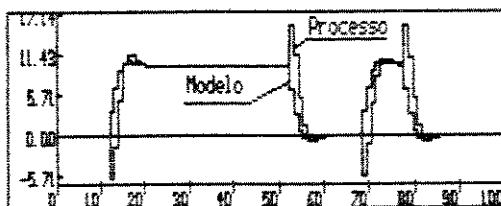


Fig. 4.2.71a

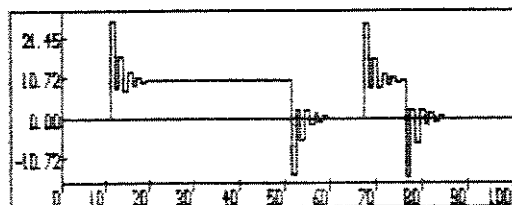


Fig. 4.2.70b

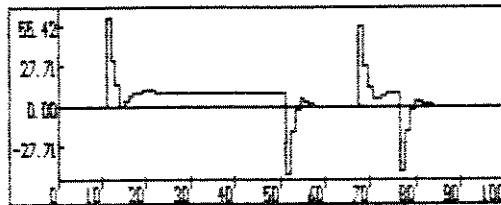


Fig. 4.2.71b (Process)

Mesmo quando se tem uma redução significativa no pico de transitório negativo, pode ser que este valor ainda esteja fora das especificações. Nestas circunstâncias pode ser necessário ou deixar a resposta do sistema mais lenta, atuando sobre os pólos do filtro P ou recorrer a uma outra forma de limitar os transitórios negativos, através do coeficiente  $\lambda$ , a ser discutido mais à frente.

Condicionado ao descrito acima pode-se propor o seguinte conjunto de regras da base de conhecimento para configuração para os filtros P e R:

SE o polinômio do denominador da função de transferência em malha fechada (PM) É fornecido pelo usuário

ENTÃO a ordem do filtro P =  $n_{pm}$

E o polinômio do filtro P = PM.

SE o polinômio do denominador da função de transferência em malha fechada (PM) *NÃO É* fornecido pelo usuário  
ENTÃO a ordem do filtro  $P = 0$   
E o polinômio do filtro  $P = 1$ .

SE o filtro  $P$  *ESTÁ* disponível  
ENTÃO a ordem do filtro  $R = 0$   
E o polinômio do filtro  $R = P(1)$ .

Para a base de conhecimento de supervisão pode ser proposta a seguinte regra:

SE o processo *É* de fase não mínima  
E a estratégia de controle *É* de seguimento de modelo  
E o tempo de subida *É MAIOR QUE* o tempo de subida desejado  
E o sobrepico *É MENOR QUE* o sobrepico desejado  
E o coeficiente  $\lambda$  *É MENOR QUE*  $\lambda_{\min}$   
ENTÃO sugere-se deslocar os pólos do filtro  $P$  mais perto da origem do círculo unitário.

A necessidade que o coeficiente  $\lambda$  seja pequeno como se verá é devida a que este coeficiente atua também sobre o tempo de subida.

#### 4.2.2.2.-ESCOLHA DO FILTRO DAS PERTURBAÇÕES $T(z^{-1})$

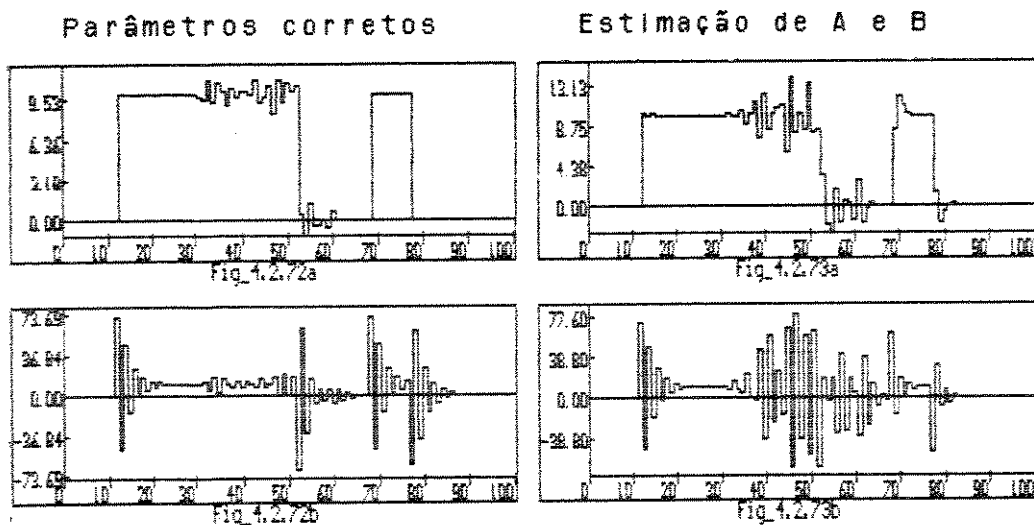
A identificação do polinômio  $C(z^{-1})$  normalmente provoca problemas de convergência do estimador. Assim no capítulo II foi sugerido utilizar um filtro que permita reduzir a influência das perturbações no processo. Para tanto incorpora-se este filtro ao algoritmo de identificação (filtrando a excitação e a saída pelo filtro  $1/T(z^{-1})$ ). O valor ótimo para este filtro seria justamente o do polinômio  $C(z^{-1})$ . Entretanto como este valor é geralmente desconhecido, a estratégia mais comumente utilizada é a de que o filtro  $T(z^{-1})$  represente um observador das perturbações atuante no processo [MOH86]. A seguir mostrar-se-ão algumas simulações do controle com este filtro que permitem obter algumas regras para sua escolha.

O processo escolhido para ser simulado é o processo 1 onde foi adicionado um polinômio  $C(z^{-1})$  que permite obter uma perturbação não branca. Utiliza-se o seguinte valor para este polinômio:

$$C(z^{-1}) = 1 - 1.5 z^{-1} + 0.7 z^{-2}$$

que tem suas raízes dentro do círculo unitário conforme é requerido no algoritmo de identificação.

Na figura 4.2.72 mostra-se a evolução do processo em presença de uma perturbação branca  $N(0, 0.5^2)$  aplicada no intervalo de tempo (30, 60). Neste caso o controle é realizado com os parâmetros corretos (processo conhecido). Os horizontes são a tripla (1, 1, 1), a ponderação  $\lambda = 1e-8$  e o filtro  $P(z^{-1}) = 1$ . Estes valores são utilizados para todas as simulações desta seção. A evolução da saída neste caso corresponde à de "dead-beat".



A figura 4.2.73 mostra o controle do processo identificando-se os polinômios  $A(z^{-1})$  e  $B(z^{-1})$ , com o polinômio  $C(z^{-1})$  considerado igual à unidade. Desta figura pode-se observar um aumento da variância da saída, além de ter ocorrido uma deterioração dos parâmetros estimados do modelo (o que é verificado pelo aumento das sobre-elevações). Na figura 4.2.74 ilustram-se as mesmas saídas estimando-se o polinômio  $C(z^{-1})$ . É

evidente a deterioração provocada no período de convergência dos parâmetros deste polinômio, passado o qual o seguimento foi correto. Ainda que se tenha obtido uma estimativa boa do polinômio das perturbações, a deterioração provocada no período de adaptação é inaceitável para o controle de processos. A seguir utiliza-se um observador<sup>6</sup> do polinômio das perturbações na forma de filtro T, inicialmente como um filtro de primeira ordem dado por

$$T(z^{-1}) = 1 - 0.8 z^{-1} \quad (\text{i.e., um polo em } 0.8)$$

O resultado é apresentado na figura 4.2.75, onde se observa uma melhora com respeito ao obtido nos casos precedentes (adaptativos), ou seja, menos oscilatórios que com  $C = 1$  e que com a estimação do polinômio  $G(z^{-1})$ .

Estimação de A, B e C

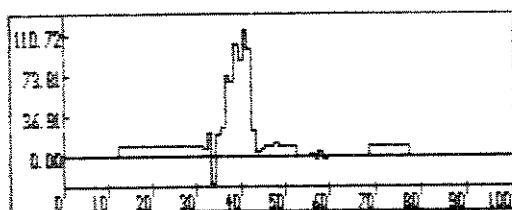


Fig. 4.2.74a

Estimação de A e B  
 $T = 1 - 0.8 z^{-1}$

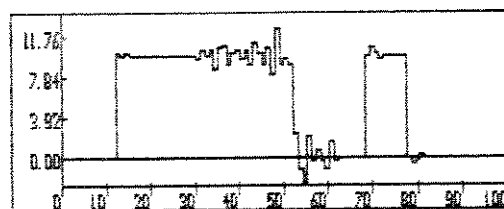


Fig. 4.2.75a

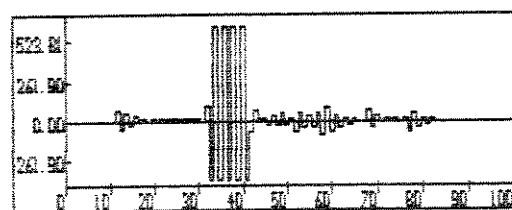


Fig. 4.2.74b

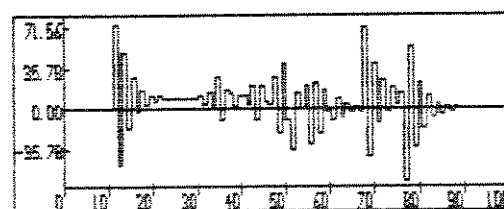


Fig. 4.2.75b

Na figura 4.2.76 é mostrada a evolução do sistema com um filtro de segunda ordem

$$T(z^{-1}) = 1 - 1.6 z^{-1} + 0.64 z^{-2} \quad (\text{i.e., dois polos em } 0.8)$$

Neste caso pode-se observar que o desempenho do sistema é análogo ao do controle tipo "dead-beat". Finalmente, em forma

<sup>6</sup> Esta forma de considerar o filtro T pode ser achada em [MOH80].

ilustrativa é apresentada na figura 4.2.77 a evolução do sistema utilizando o filtro

$$T(z^{-1}) = 1 - 1.5 z^{-1} + 0.7 z^{-2} \text{ (i.e., idêntico a } C(z^{-1}) \text{)}$$

Estimação de A e B  
 $T = 1 - 1.6z^{-1} + 0.64z^{-2}$

Estimação de A e B  
 $T = C$

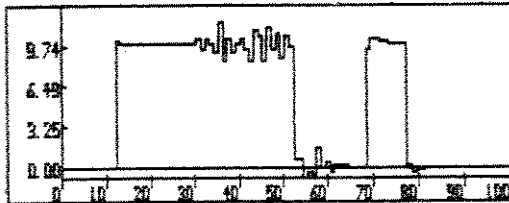


Fig. 4.2.76a

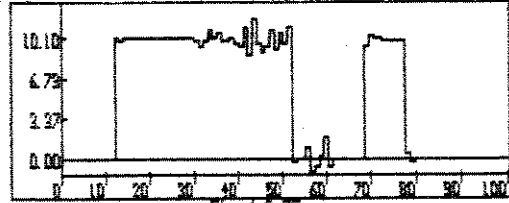


Fig. 4.2.77a

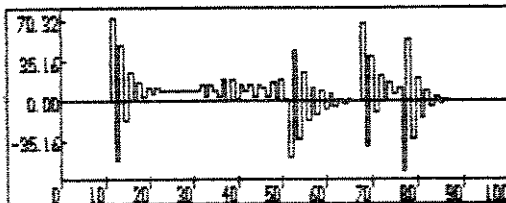


Fig. 4.2.76b

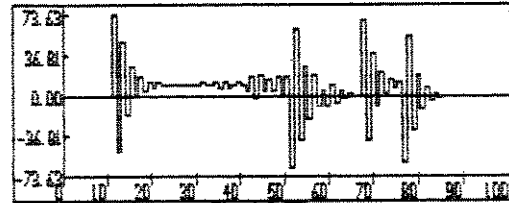


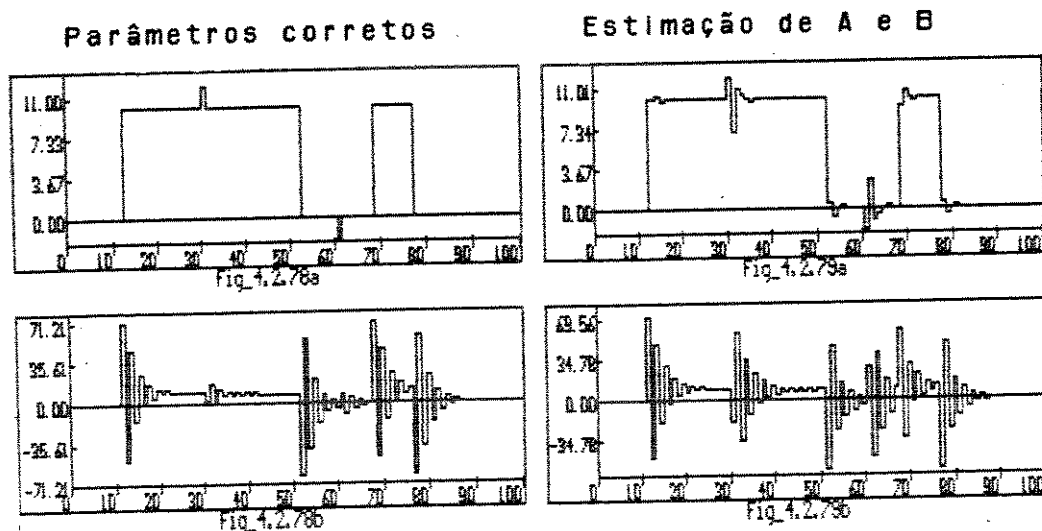
Fig. 4.2.77b

Nessa figura pode-se observar que a evolução novamente ficou semelhante à obtida com conhecimento do processo, sendo que as pequenas deteriorações observadas no seguimento da referência (sobrepicos) são devidas a que os parâmetros estimados (polinômios A e B) sofreram uma pequena variação provocada pela perturbação. Desta figura também fica evidente que o filtro  $T(z^{-1})$  de segunda ordem proposto representa adequadamente o polinômio das perturbações, sendo que o polinômio de primeira ordem não consegue fazê-lo com precisão.

O segundo tipo de perturbação a que foi submetido o processo, também pode ser interpretado como uma perturbação branca colorida com o filtro  $C(z^{-1})$  já apresentado, a diferença é que este ruído branco é descrito como  $N(2,0)$ . Isto é uma perturbação com média 2, e variância zero, que é mais comumente conhecida como uma perturbação "dcy", ou seja, constante sobre a saída.

O primeiro teste realizado para fins comparativos é novamente não adaptativo, com o conhecimento dos parâmetros do processo. Como pode ser observado na figura 4.2.78, esta perturbação é rapidamente eliminada não provocando nenhuma alteração posterior

na evolução do sistema. O caso adaptativo sem a estimação do polinômio  $C(z^{-1})$  é apresentado na figura 4.2.79. Nesta nota-se uma deterioração do desempenho do sistema devido à falta de conhecimento sobre o polinômio da perturbação, que também afeta a estimação correta dos parâmetros, como pode ser verificado pela sobre-elevação no seguimento da referência.



Estimar o polinômio  $C(z^{-1})$  pelo método dos mínimos quadrados apresentado no capítulo II, é inútil<sup>7</sup>, como pode ser demonstrado matematicamente, ou praticamente, o que é feito na figura 4.2.80. Já aplicar um filtro de perturbações pode ser uma política muito mais aconselhável como o demonstra a figura 4.2.81, onde pode se observar a evolução do sistema sob a influência de um filtro  $T(z^{-1})$  de primeira ordem com um pólo em 0.8. Nesta figura a influência da perturbação, ainda que não tenha sido removida totalmente, foi reduzida a valores aceitáveis, permitindo que os parâmetros estimados não sofram grandes deteriorações (note que os sobrepicos de seguimento da referência são pequenos).

<sup>7</sup> Deve-se recorrer a outro método, por exemplo, mínimos quadrados com o valor 1 no vetor de dados (CLK79).

### Estimação de A, B e C

### Estimação de A e B

$$T = 1 - 0.8z^{-1}$$

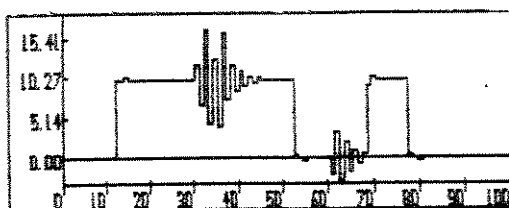


Fig. 4.2.80a

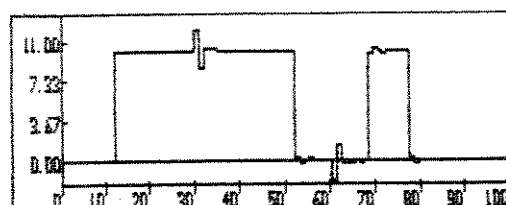


Fig. 4.2.81a

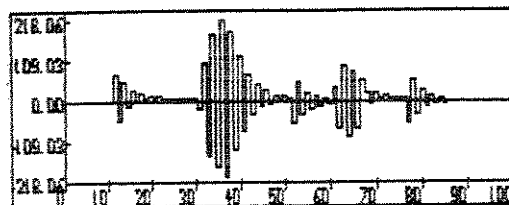


Fig. 4.2.80b

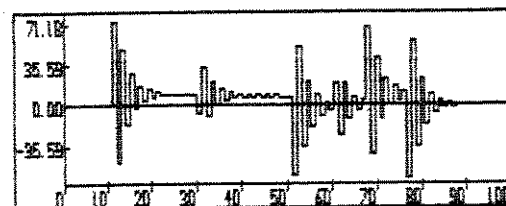


Fig. 4.2.81b

Utilizar dois pólos no filtro  $T(z^{-1})$  (em 0.8) leva a uma diminuição dos picos no início e fim da perturbação, como é observado na figura 4.2.82. Porém a qualidade integrativa apresentada por um filtro com essas características faz com que a influência dessa perturbação, se bem que atenuada, permaneça por mais tempo, o que pode levar a uma deterioração da saída como é possível verificar na figura citada. Finalmente, e novamente com fins ilustrativos, é apresentada a evolução do sistema em presença de um filtro  $T(z^{-1})$  que representa perfeitamente o polinômio das perturbações, isto é,  $T(z^{-1}) = C(z^{-1})$ , o que pode ser observado na figura 4.2.83.

### Estimação de A e B

$$T = 1 - 1.6z^{-1} + 0.64z^{-2}$$

### Estimação de A e B

$$T = C$$

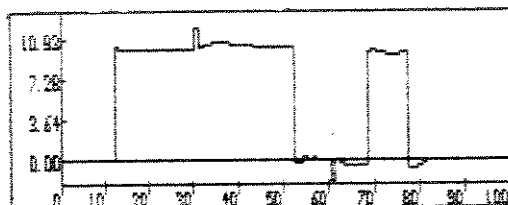


Fig. 4.2.82a

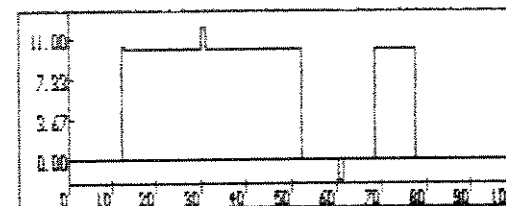


Fig. 4.2.83a

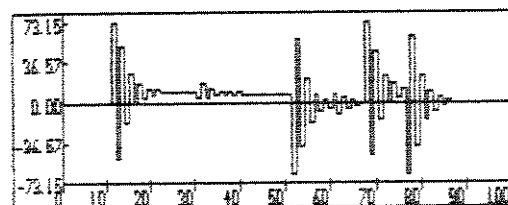


Fig. 4.2.82b

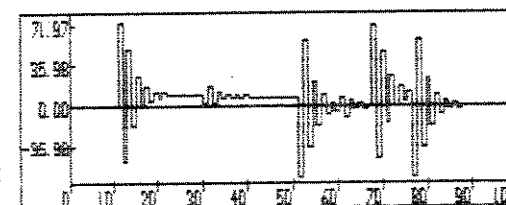


Fig. 4.2.83b

Podem ser ressaltados, neste ponto, alguns indícios a

respeito da influência, e por tanto de valores do filtro procurado. Evidentemente, se o polinômio das perturbações é conhecido,  $T(z^{-1})$  será escolhido conforme este polinômio; porém, esta é uma situação raramente encontrada, pelo que o filtro  $T(z^{-1})$  deve ser escolhido em função da perturbação predominante encontrada no processo. Se a perturbação atuante no processo é predominantemente estocástica, o filtro deve ser passa faixa, sendo a faixa passante entre a mínima e a máxima frequência significativas de resposta do processo. No entanto uma perturbação predominantemente "dcy", um filtro passa baixo, pode ter vantagens com respeito a um passa banda, sobretudo se as variações da referência são frequentes. Mais importante que determinar o tipo de perturbação atuante, é necessário obter os limites de frequência dentro dos quais é esperada a resposta do processo. Tendo sido observado que os valores discutidos dos filtros permitem um desempenho aceitável na maioria dos processo simulados, optou-se por tomar o pólo ou par de pólos em 0.8 como valor central ("default" em presença de perturbações) sugerindo modificações quando os requisitos de projeto não são satisfeitos. Isto permite gerar o seguinte conjunto de regras para a base de conhecimento de configuração:

SE a perturbação atuante sobre o processo É desprezível

ENTÃO a ordem do filtro  $T = 0$

E o polinômio do filtro  $T = 1$ .

SE o polinômio da perturbação atuante sobre o processo É conhecido

ENTÃO a ordem do filtro  $T = nc$

E o polinômio do filtro  $T = C$ .

SE a perturbação atuante sobre o processo É predominantemente estocástica

ENTÃO a ordem do filtro  $T = 2$

E o polinômio do filtro  $T = [1 - 1.6 + 0.64]$ .

SE a perturbação atuante sobre o processo *É* predominantemente "dcy"

ENTÃO a ordem do filtro  $T = 1$

E o polinômio do filtro  $T = (1 - 0.8)$ .

Além destas regras de inicialização do filtro  $T(z^{-1})$ , utilizadas pelo sistema no encadeamento reverso, o seguinte conjunto de regras é utilizado quando o encadeamento ativado é o direto (supervisão):

SE a perturbação atuante sobre o processo *É* predominantemente estocástica

E a variância da saída *É MAIOR QUE* a máxima variância aceitável da saída

ENTÃO sugere-se deslocar o par de pólos do filtro  $T$  para mais perto do círculo unitário no plano  $Z$ .

SE a perturbação atuante sobre o processo *É* predominantemente "dcy"

E a variância da saída *É MENOR QUE* a máxima variância aceitável da saída

ENTÃO sugere-se deslocar o pólo do filtro  $T$  para mais perto da origem do plano  $Z$ .

#### 4.2.3.-ESCOLHA DO FATOR DE PONDERAÇÃO DA EXCITAÇÃO ( $\lambda$ )

Como foi discutido no capítulo II, o fator de ponderação da excitação tem, como uma de suas funções permitir a inversão da matriz  $(M^T M + \lambda I)$  quando o produto  $M^T M$  é nulo. Assim, para aumentar a robustez numérica, em todas as simulações feitas até aqui, foi utilizado  $\lambda = 1e-8$ , isto é, um valor suficientemente pequeno que não influencie o sinal de controle (e permita visualizar a influência dos horizontes), como também a inversão da matriz  $(M^T M + \lambda I)$  nas situações onde a matriz  $M^T M$  é singular, permitindo obter o sinal de controle sem os problemas numéricos associados.

O fator  $\lambda$  também desempenha uma outra importante função no controlador preditivo generalizado, especificamente no controle de processos de fase não mínima. Neste caso os processos apresentam transitório oposto ao do valor de regime ("transitório negativo"), devido aos zeros fora do círculo unitário. Como foi discutido anteriormente, este transitório pode ser melhorado pela utilização do filtro P. Entretanto, pode-se obter redução, e em alguns casos a eliminação total do transitório negativo, com a utilização do fator de ponderação  $\lambda$ , que sendo um valor real tem menos carga computacional associada que o filtro P. Uma regra para a escolha deste fator é complexa devido à interação com os horizontes de previsão, de controle, e do ganho do processo. Lambert [LAM87], Clark [CLKB9], e outros propoem seleccionar  $\lambda$  por meio de um "re-escalamento" como indicado na seguinte equação:

$$\lambda = \lambda_r * |\text{Ganho do processo}|$$

Os outros parâmetros que também exercem influência sobre este coeficiente, são os horizontes do controlador, mas não pode ser obtida alguma relação simples entre estes e o parâmetro  $\lambda$ . Deste modo optou-se por verificar se, quando o controlador possui os horizontes ajustados de acordo com as regras anteriormente desenvolvidas, é possível obter alguma relação entre a evolução do processo e o valor do  $\lambda_r$ .

A seguir analisa-se como seleccionar  $\lambda_r$  de acordo com o tipo de processo. As simulações iniciam novamente com o processo de fase não mínima indicado como N° 5 na tabela 4.1

Nas figura 4.2.84, 4.2.85 e 4.2.86, pode-se observar a evolução da saída do processo com o conjunto de horizontes obtidos como melhores na seção anterior, isto é, (2,4,3), e para os valores de ponderação  $\lambda_r = 1e-4$ ,  $5e-3$  e  $0.5$ . Na primeira das figuras pode se observar que a influência da ponderação na excitação permanece equivalente a com  $\lambda_r = 1e-8$ . Na segunda das figuras o transitório de fase não mínima é ainda importante, no

entanto ele reduziu sensivelmente sem aumentar excessivamente o tempo de subida. Na figura 4.2.86 com um fator de ponderação  $\lambda r$  de 0.5, o transitório negativo é praticamente eliminado mas com prejuízo do tempo de subida. Desta série de simulações pode-se concluir que para este tipo de processos, deve-se ter um compromisso entre o tempo de subida e o transitório negativo. Entretanto os valores obtidos do fator  $\lambda$  para o tempo de subida aceitável, com transitório "médio", isto é,  $\lambda r = 5e-3$ , como para a eliminação do transitório ( $\lambda r = 0.5$ ), mostraram não serem válidos para outros processos simulados. Em particular processos instáveis em malha aberta mostraram-se sensíveis à escolha deste fator. Em processos de fase não mínima que tenham transitório negativo desprezível, como o processo N° 6, pode ser utilizado um fator de ponderação  $\lambda r$  próximo de zero. Em processos de fase não mínima em que o valor do transitório negativo não é desprezível, é conveniente aumentar este fator de ponderação até um valor que não deteriore os outros parâmetros de desempenho do sistema, e mais importante, sem produzir instabilidade. Particularmente nos processos simulados, este valor situou-se em torno de 0.1, como ilustram as simulações com o processo 7 analisadas a seguir:

$$\lambda r = 1e^{-4}$$

$$\lambda r = 5e^{-3}$$

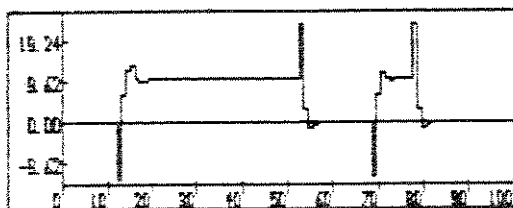


Fig. 4.2.84a

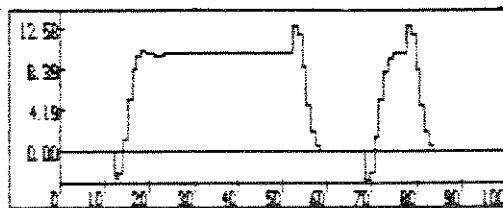


Fig. 4.2.85a

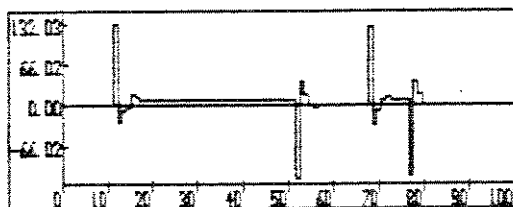


Fig. 4.2.84b

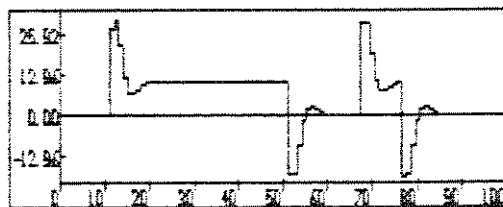


Fig. 4.2.85b

$$\lambda r = .5e^{-1}$$

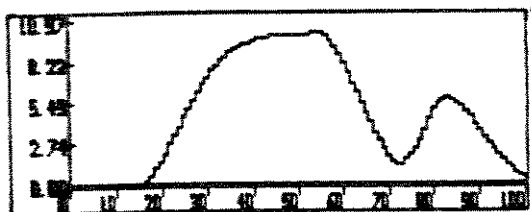


Fig. 4.2.86a

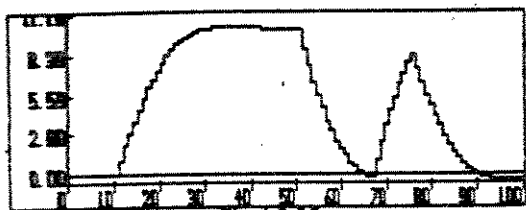


Fig. 4.2.86b

Na figura 4.2.87 mostra-se o desempenho do processo para o conjunto de horizontes (2,4,3) com o fator  $\lambda r = 1e-8$ . Nessa figura pode-se observar um forte transitório negativo, sendo que o tempo de subida é ótimo. Na figura 4.2.88 pode-se observar a saída e excitação do processo para o mesmo conjunto de horizontes e com  $\lambda r = 5e-3$ . Neste caso o tempo de subida aumentou, mas reduziu-se o pico de transitório negativo. O desempenho do controlador com  $\lambda r = 0.1$  e o mesmo conjunto de horizontes é mostrado na figura 4.2.89. Nesta figura pode-se observar novamente o aumento do tempo de subida com a diminuição do pico de transitório negativo. Finalmente na figura 4.2.90 foi simulado o processo para o fator de ponderação  $\lambda r = 0.5$  (com (2,4,3)), o que provocou uma deterioração considerável na saída do processo.

$$\lambda r = 1e^{-8}$$

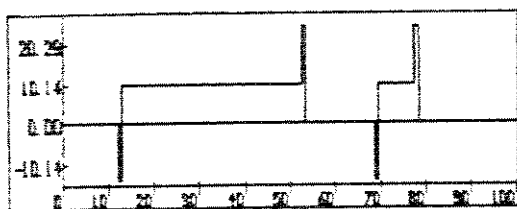


Fig. 4.2.87a

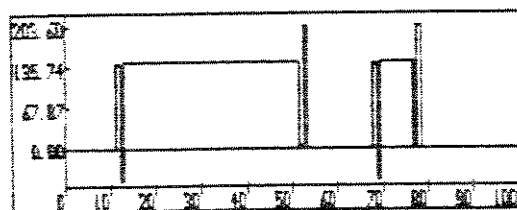


Fig. 4.2.87b

$$\lambda r = 5e^{-3}$$

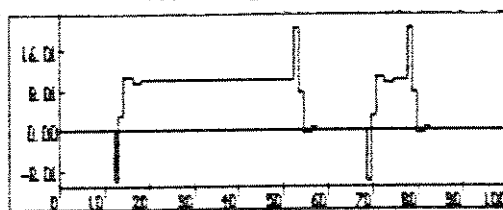


Fig. 4.2.88a

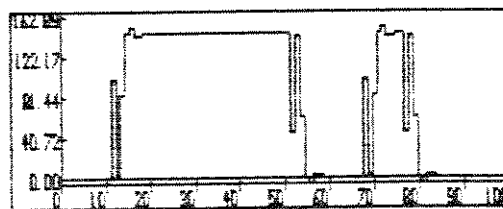
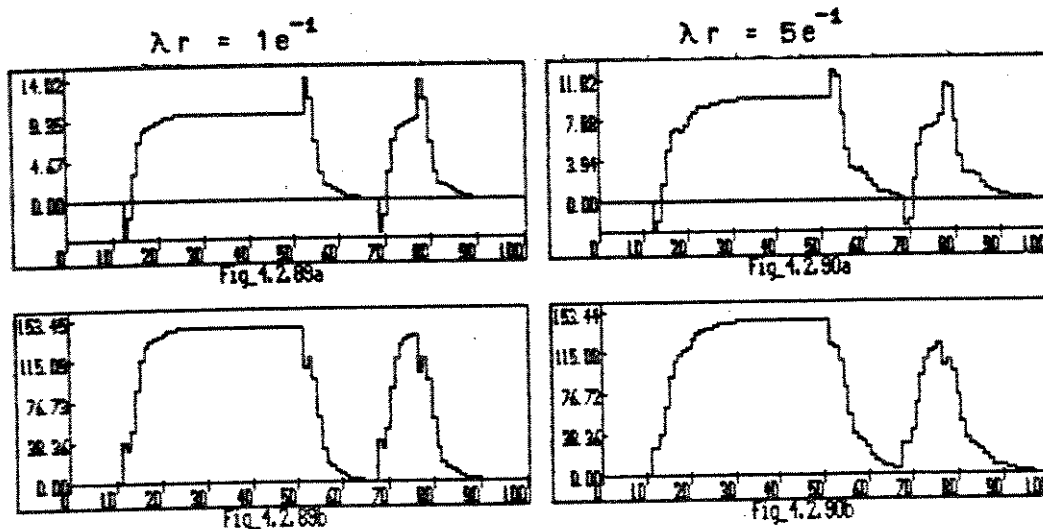


Fig. 4.2.88b



Destas simulações pode-se propor o seguinte conjunto de regras para o fator de ponderação da excitação a serem incluídas na base de conhecimento de configuração.

SE o processo  $\dot{E}$  de fase mínima  
ENTÃO  $\lambda = 1e-8$ . [i.e., zero]

SE o processo  $\dot{E}$  de fase não mínima  
E o processo  $TEM$  transitório negativo desprezível  
ENTÃO  $\lambda = 1e-8$ .

SE o processo  $\dot{E}$  de fase não mínima  
E o processo  $TEM$  transitório negativo  
E o processo  $NÃO\ TEM$  pólos instáveis  
ENTÃO  $\lambda = 0.5$ . [valor alto]

SE o processo  $\dot{E}$  de fase não mínima  
E o processo  $TEM$  transitório negativo  
E o processo  $TEM$  pólos instáveis  
ENTÃO  $\lambda = 0.1$ . [valor médio]

Da mesma forma para a base de conhecimento de supervisão podem ser propostas as seguintes regras:

SE o processo *É* de fase não mínima  
E o processo *TEM* transitório negativo  
E o tempo de subida *É MENOR QUE* o tempo de subida desejado  
E o processo *NÃO TEM* pólos instáveis  
ENTÃO sugere-se aumentar o coeficiente  $\lambda$ .

SE o processo *É* de fase não mínima  
E o processo *TEM* transitório negativo  
E o tempo de subida *É MENOR QUE* o tempo de subida desejado  
E o processo *TEM* pólos instáveis  
E o coeficiente  $\lambda$  *É MENOR QUE* 0.1  
ENTÃO sugere-se aumentar o coeficiente  $\lambda$ .

#### 4.2.4.-REFERÊNCIA PRÉ-ESPECIFICADA

Uma das características do GPC que o diferenciam dos algoritmos adaptativos clássicos (i.e., STR, GMV, PP), é a sua capacidade de aproveitar o conhecimento antecipado da referência.

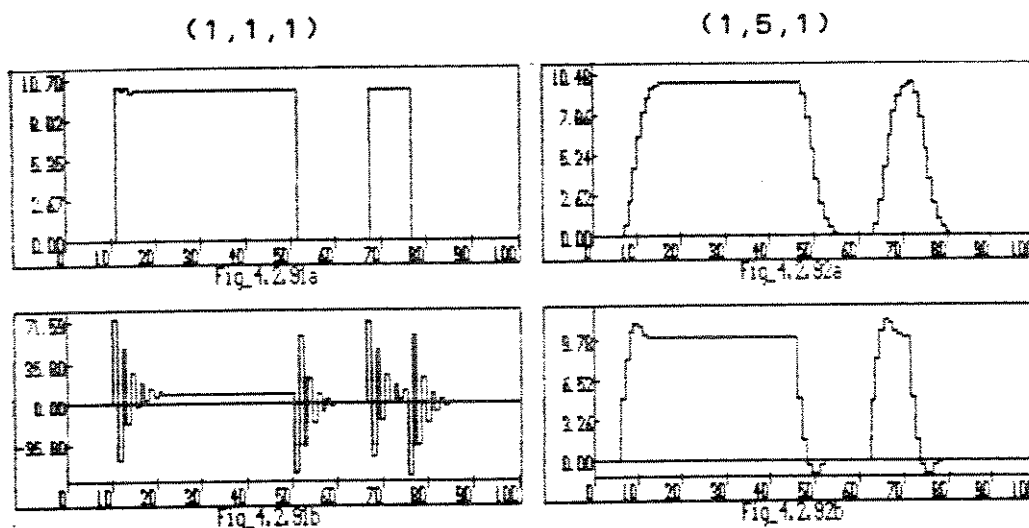
Ainda que este conhecimento, como já foi indicado anteriormente, seja propriedade de alguns poucos processos industriais, o seu conhecimento pode melhorar consideravelmente o desempenho do sistema, especialmente para o controle de processos de fase não mínima.

Analogamente às simulações com referência normal, os resultados obtidos com referência pré-especificada diferem somente quando os processos são de fase mínima ou não mínima, sem importar a sua estabilidade em malha aberta. Assim as simulações dividem-se em processos de fase mínima e em processos de fase não mínima. Os filtros e a ponderação da excitação são desativados.

#### 4.2.4.1.-PROCESSOS DE FASE MÍNIMA

A principal vantagem obtida com a utilização de referência pré-especificada para o processo N°1 da tabela 4.1 é a eliminação do atraso imposto pelo segurador de ordem zero<sup>8</sup>, como pode ser observado na figura 4.2.91. Nesta figura ilustra-se a evolução deste processo sob a influência do controlador com o conjunto de horizontes (1,1,1). O aumento do horizonte final de previsão permite diminuir a atuação da excitação como pode ser observado na figura 4.2.92 para o conjunto de horizontes (1,5,1), porém isto provoca que a saída antecipe à referência. Quando o controlador atinge novamente a condição de "dead-beat" a antecipação desaparece como se pode verificar para o conjunto de horizontes (1,5,5) (figura 4.2.93).

Sendo que as simulações com horizontes diferentes apresentaram resultados similares aos obtidos com referência normal, com a diferença da antecipação da saída, estes resultados não são apresentados.



<sup>8</sup> Este efeito pode ser obtido só com a ajuda de um filtro anticausal.

(1,5,5)

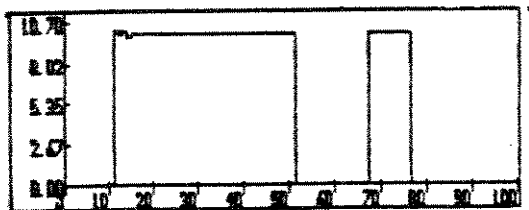


Fig. 4.2.93a

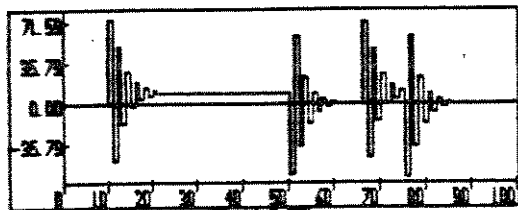
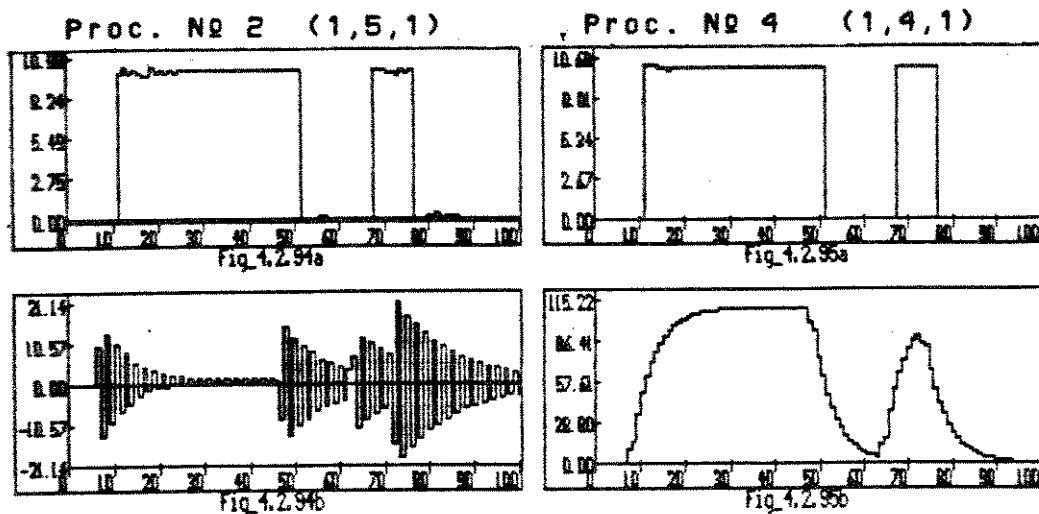


Fig. 4.2.93b

Quando o processo controlado tem atraso maior que a unidade, como o processo N°2 da tabela 4.1, verificou-se que para obter a estabilidade, novamente o horizonte  $N_2$  deve ser igual ou maior do que o atraso do processo. A figura 4.2.94 apresenta os resultados obtidos quando o horizonte  $N_2$  é igual ao atraso do processo, com os outros horizontes com valor 1. Nestas condições, o atraso do processo controlado foi totalmente eliminado. Variando-se os horizontes obteve-se os mesmos resultados que com referência normal pelo que não são apresentados.

Para processos instáveis em malha aberta e de fase mínima, os resultados também coincidiram com os resultados obtidos com referência normal. A eliminação do atraso também pode ser observada na figura 4.2.95, que ilustra a evolução do processo N°4 para o conjunto de horizontes (1,4,1).



Das simulações apresentadas fica evidente que a referência pré-especificada apresenta como única vantagem para processos de fase mínima, a antecipação da resposta o que é um fato geralmente desejável. Assim pode-se propôr as seguintes regras para a base de conhecimento de configuração:

SE o processo *É* de fase mínima  
 E a referência *É* conhecida a priori  
 ENTÃO o modo da referência *É* pré-especificado.

SE o processo *É* de fase mínima  
 E a referência *NÃO É* conhecida a priori  
 ENTÃO o modo da referência *É* normal.

Estas regras permitem que quando a referência é conhecida a priori ela sempre seja utilizada em processos de fase mínima e desta forma elimina-se o atraso.

#### 4.2.4.2.-PROCESSOS DE FASE NÃO MÍNIMA

Novamente este tipo de processo demonstrou ser mais sensível a sintonia dos horizontes.

Baseando-se nas simulações da seção anterior, onde o melhor

desempenho para o processo é obtido com os mesmos horizontes tanto para referência normal como para referência pré-especificada, a primeira simulação é feita com o processo N° 5 para o conjunto de horizontes obtidos com referência normal (2,4,3). Na figura 4.2.96 apresenta-se o desempenho do sistema com os horizontes seleccionados anteriormente. Pode-se observar uma oscilação acentuada na saída do processo iniciando N2 iterações antes que a referência varie. Diminuindo o horizonte final de previsão obtem-se instabilidade para qualquer combinação dos outros dois horizontes. O aumento deste horizonte permite diminuir a amplitude do transitório. As figuras 4.2.97 a 4.2.99 ilustram o sinal de saída e de controle para o processo com o horizonte inicial  $N1 = 2$ , o horizonte de controle  $Nu = 3$  e o horizonte de previsão final  $N2 = 7, 10$  e  $15$  respectivamente. Não obstante o aumento de  $N2$  permite melhorar o desempenho do sistema, esta melhora não pode ser relatada diretamente a algum fator já apresentado do processo (p.e., atraso de transporte, etc). Embora essa afirmação seja válida, nos processos simulados (quando estáveis em malha aberta) observou-se que para valores do horizonte superiores ao tempo de transitório negativo mais a constante de tempo principal do sistema, o transitório foi diminuído fortemente.

(2,4,3)

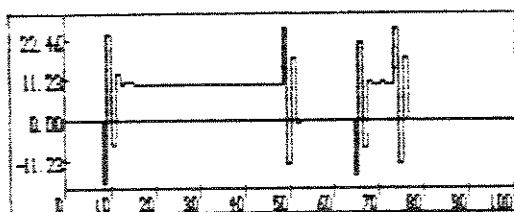


Fig. 4.2.96a

(2,7,3)

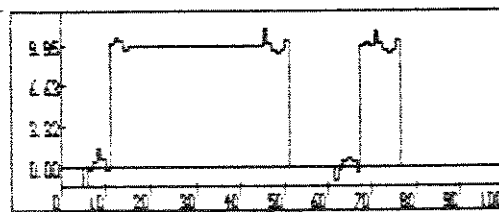


Fig. 4.2.97a

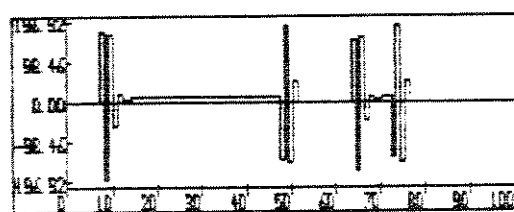


Fig. 4.2.98a

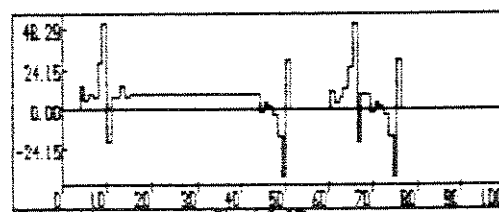


Fig. 4.2.99a

(2,10,3)

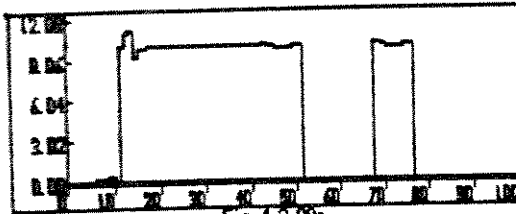


Fig. 4.2.98a

(2,15,3)

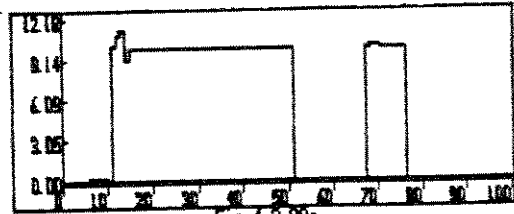


Fig. 4.2.99a

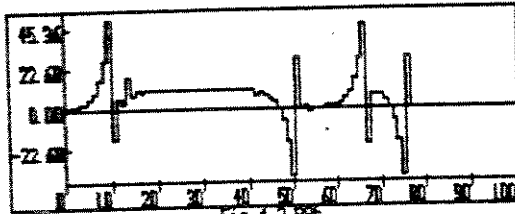


Fig. 4.2.98b

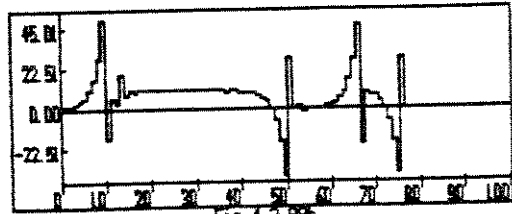


Fig. 4.2.99b

O aumento do horizonte de controle para este processo em particular provocou uma forte deterioração da saída  $N_2$  instantes antes da mudança da referência, como pode ser observado nas figuras 4.2.100 e 4.2.101 nas quais os horizontes são os conjuntos (2,7,6) e (2,10,9).

(2,7,6)

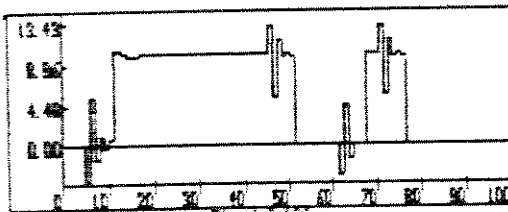


Fig. 4.2.100a

(2,10,9)

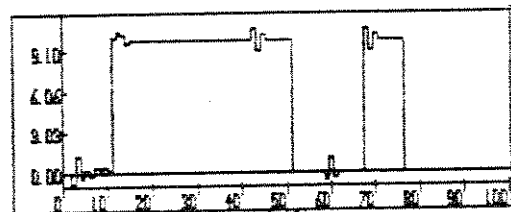


Fig. 4.2.101a

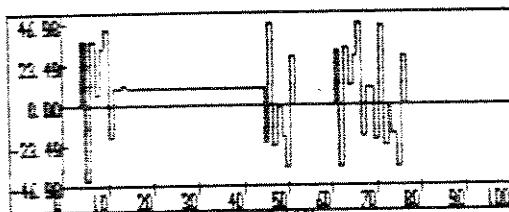


Fig. 4.2.100b

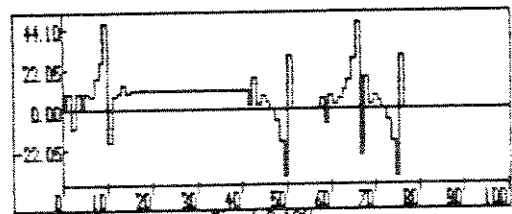


Fig. 4.2.101b

Quando o horizonte de previsão final é maior que 4, a diminuição do horizonte de previsão inicial não gera instabilidade (como acontece com  $N_2 = 4$ ), como pode-se verificar para o conjunto de horizontes (1,5,3) na figura 4.2.102. Entretanto o baixo valor do horizonte  $N_2$  continua deteriorando a saída do processo  $N_2$  intervalos de amostragem antes da mudança da referência. Na figura 4.2.103 com a tripla (1,10,3) observa-se um desenhado similar ao obtido com o conjunto de horizontes (2,10,3). Contudo, o aumento

do horizonte de previsão inicial provoca uma deterioração do desempenho da saída e da excitação, como ilustra a figura 4.2.104 para o conjunto de horizontes (3,7,3).

(1,5,3)

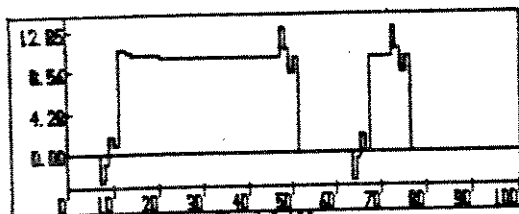


Fig. 4.2.102a

(1,10,3)

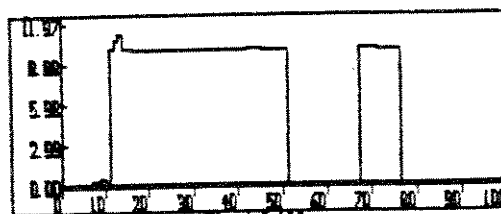


Fig. 4.2.103a

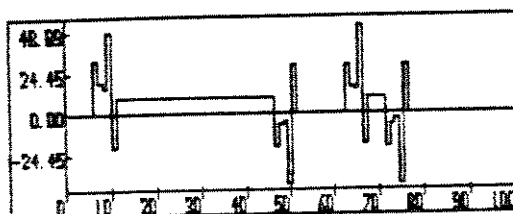


Fig. 4.2.102b

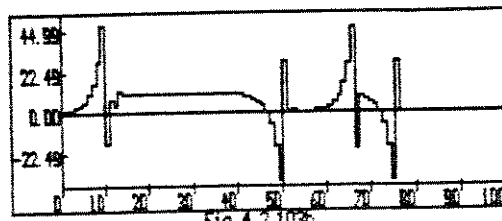


Fig. 4.2.103b

(3,7,3)

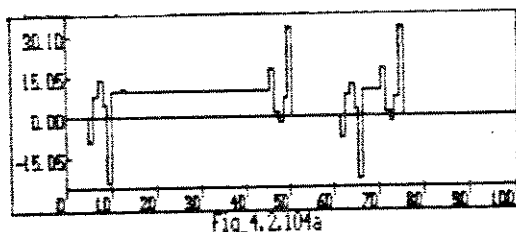


Fig. 4.2.104a

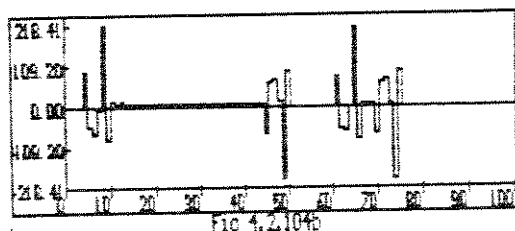


Fig. 4.2.104b

Tratando-se de processos de fase não mínima com transitório negativo desprezível, a influência da referência pré-especificada é similar à apresentada anteriormente. Neste caso, geralmente a referência pré-especificada provoca uma degradação na saída do processo que somente é eliminada utilizando-se valores elevados de horizontes de previsão e controle. O processo N°6 simulado com o conjunto de horizontes fornecidos pelas regras obtidas com referência normal (3,8,6), apresentou fortes transitórios na saída (figura 4.2.105). O desempenho do processo não melhora quando se aumenta somente o horizonte de previsão final como ilustra a

figura 4.2.106 para o conjunto de horizontes (3,15,6). Uma melhora do desempenho da saída é obtida somente quando ambos horizontes  $N_2$  e  $N_u$  possuem valores relativamente altos, como ilustra a figura 4.2.107 para o conjunto (3,15,13). Pode-se obter um sistema com uma resposta que apresente melhor desempenho diminuindo-se o horizonte de previsão inicial como ilustram as figuras 4.2.108 e 4.2.109 para  $N_2 = 15$ ,  $N_u = 13$  e  $N_1 = 1$  e 2 respectivamente.

(3,8,6)

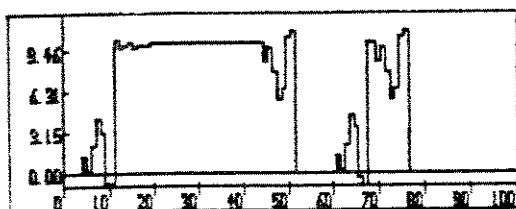


Fig. 4.2.105a

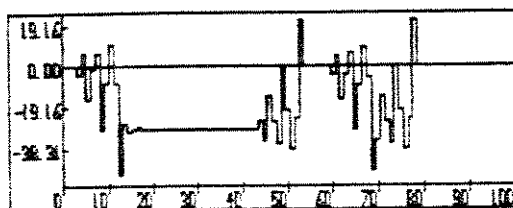


Fig. 4.2.105b

(3,15,6)

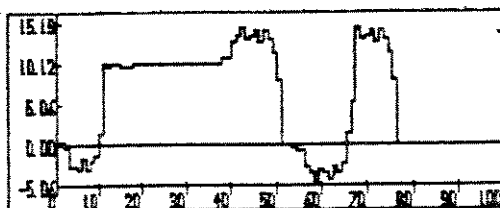


Fig. 4.2.106a

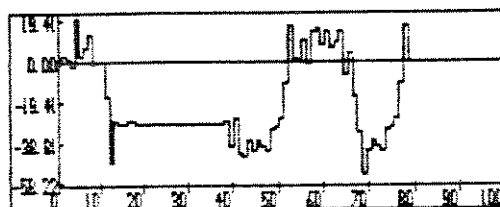


Fig. 4.2.106b

(3,15,13)

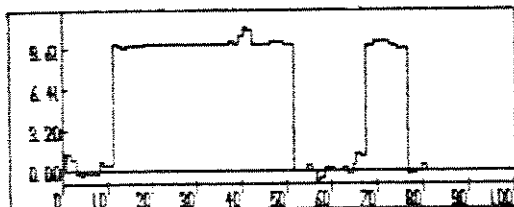


Fig. 4.2.107a

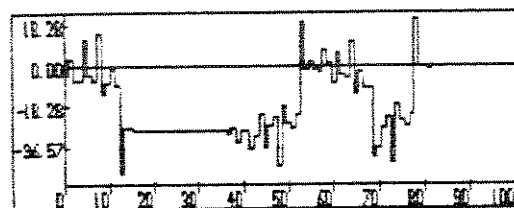


Fig. 4.2.107b

(1,15,13)

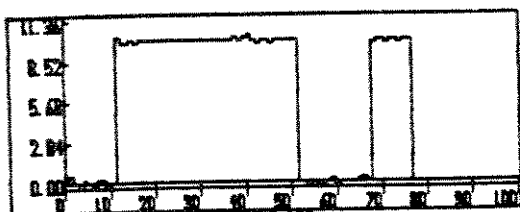


Fig. 4.2.108a

(2,15,13)

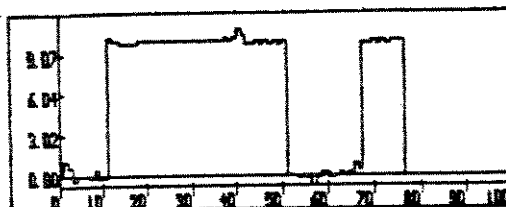


Fig. 4.2.109a

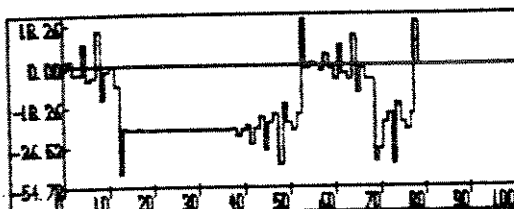


Fig. 4.2.108b

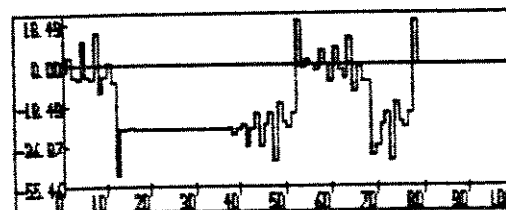


Fig. 4.2.109b

Finalmente quando o processo é de fase não mínima e instável em malha aberta, a melhora de desempenho do sistema é obtida somente com um aumento do valor dos horizontes de previsão final e de controle, conforme pode ser verificado nas figuras 4.2.110, 4.2.111 e 4.2.112 para o processo 7 simulado com o conjunto de horizontes (2,4,3), (2,12,3) e (2,12,11). Embora o horizonte final permita obter uma melhora sensível no desempenho do sistema, é difícil distinguir quais são as características do processo em malha aberta que determinam esta escolha.

(2,4,3)

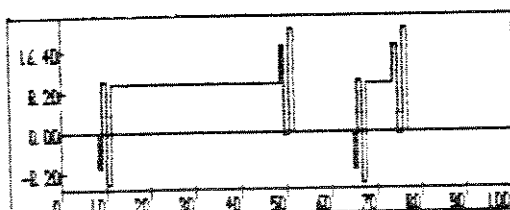


Fig. 4.2.110a

(2,12,3)

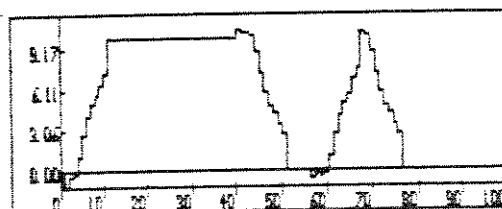


Fig. 4.2.111a

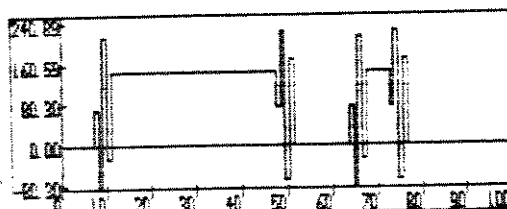


Fig. 4.2.110b

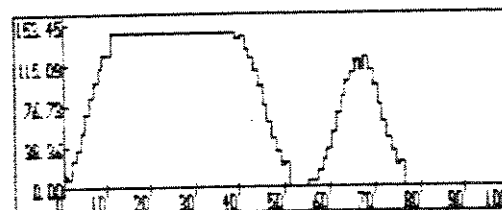


Fig. 4.2.111b

(2,12,11)

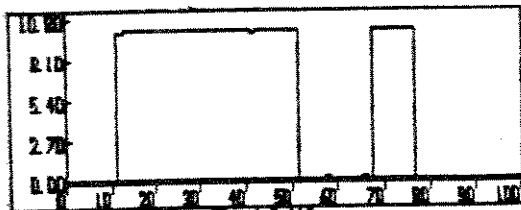


Fig. 4.2.11.2a

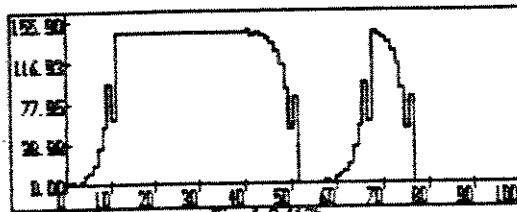


Fig. 4.2.11.2b

Esta dificuldade na obtenção dos horizontes para processos de fase não mínima, é ascentuada pela dificuldade da obtenção da constante de tempo principal em processos instáveis ou de ordem grande, pelo que para este tipo de processos optou-se por não incluir este tipo de referência na etapa de configuração<sup>9</sup>. Na etapa de supervisão ela pode ajudar a diminuir o transitório negativo sem prejudicar o desempenho de processo de fase não mínima que não possuem tal transitório.

As regras a serem incluídas na base de conhecimento de configuração são as seguintes:

SE o processo  $\hat{E}$  de fase não mínima  
E a referência  $\hat{E}$  conhecida a priori  
ENTÃO o modo da referência  $\hat{E}$  normal.

SE o processo  $\hat{E}$  de fase não mínima  
E a referência *NÃO*  $\hat{E}$  conhecida a priori  
ENTÃO o modo da referência  $\hat{E}$  normal.

O conjunto de regras para a base de conhecimento de supervisão é:

---

<sup>9</sup> Espera-se no futuro poder completar estas bases de conhecimento.

SE o processo *É* de fase não mínima  
 E o modo da referência *É* normal  
 E o processo *POSSUI* transitório negativo  
 E a referência *É* conhecida a priori  
 ENTÃO sugere-se referência pré-especificada

SE o processo *É* de fase não mínima  
 E o modo da referência *É* pré-especificada  
 E o processo *É* estável em malha aberta  
 E a variância da saída *É MAIOR QUE* a variância máxima especificada  
 ENTÃO sugere-se aumentar o horizonte de previsão final.

SE o processo *É* de fase não mínima  
 E o modo da referência *É* pré-especificada  
 E o processo *É* instável em malha aberta  
 E a variância da saída *É MAIOR QUE* a variância máxima especificada  
 ENTÃO sugere-se aumentar o horizonte de previsão final  
     E sufre-se aumentar o horizonte de controle mantendo  
         a diferença entre eles constante.

#### 4.2.4.3.-INFLUÊNCIA DA REFERÊNCIA PRÉ-ESPECIFICADA SOBRE OS FILTROS

A influência dos filtros sobre os processo mostrou não ser alterada pela presença de uma referência pré-especificada. Na figura 4.2.113 apresenta-se o processo 1 para o conjunto de horizontes (1,1,1) com  $\lambda_r = 1e-8$ , e com um modelo R/P dado por:

$$\frac{R}{P} = \frac{0.35}{(1 - 1 z^{-1} + 0.35 z^{-2})}$$

Na referida figura pode-se observar o mesmo desempenho já obtido na simulação com o mesmo conjunto de parâmetros com referência normal (figura 4.2.70)

Proc. Nº1, (1,1,1)  
 $P = 1 - 12z^{-1} + 0.35z^{-2}$   
 $R = 0.35$

Estimação de A e B  
 $T = 1 - 1.6z^{-1} + 0.64z^{-2}$

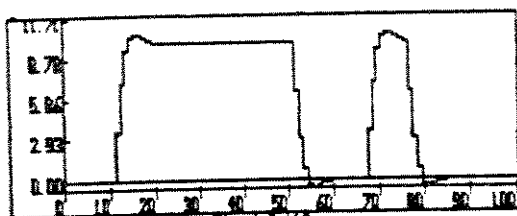


Fig. 4.2.113a

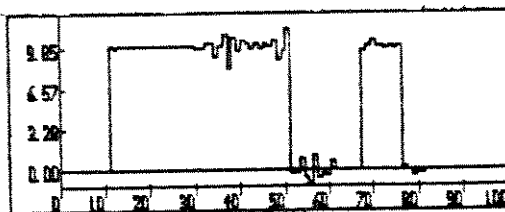


Fig. 4.2.114a

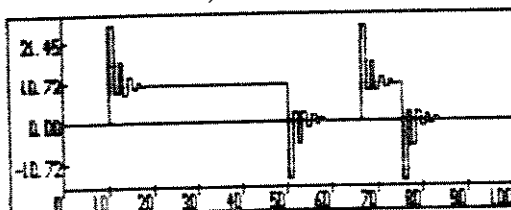


Fig. 4.2.113b

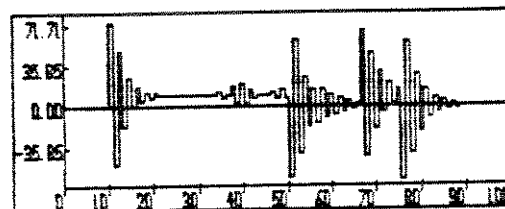


Fig. 4.2.114b

A seguir simula-se o mesmo processo incluindo um polinômio de perturbação dado por:

$$G(z^{-1}) = 1 - 1.5z^{-1} + 0.7z^{-2}$$

Os horizontes são novamente a tripla (1,1,1), estando a ponderação e os filtro P e R desativados. Na figura 4.2.114 ilustra-se a influência do filtro T dado por:

$$T(z^{-1}) = 1 - 1.6z^{-1} + 0.64z^{-2} \text{ (i.e., dois polos em 0.8)}$$

para uma perturbação branca  $N(0, 0.5^2)$  no intervalo (30,60]. O desempenho apresentado nessa figura é equivalente ao obtido quando a referência normal, para as mesmas condições de parâmetros e perturbação, apresentado na figura 4.2.76. Como as simulações têm apresentado resultados similares aos obtidos com referência normal, não apresentam nenhum conhecimento novo e não são apresentadas por carecerem de utilidade.

### 4.3.-CONTROLADOR DE VARIÂNCIA MÍNIMA GENERALIZADA

No capítulo II, ao ser descrito este controlador, mostrou-se uma escolha correta dos polinômios  $P(z^{-1})$ ,  $Q(z^{-1})$  e  $R(z^{-1})$  permite especificar o desempenho determinístico e estocástico<sup>10</sup>. Entretanto estes polinômios podem ser obtidos como solução do problema de alocação de pólos

$$B P + \alpha Q = S$$

$$r_0 = P(1) \text{ com } n_r = 0$$

sendo  $S(z^{-1})$  o polinômio que determina a alocação dos pólos em malha fechada. A seguir apresenta-se algumas simulações que permitem distinguir as características destes dois enfoques, e fornecem o conhecimento para o sistema.

As simulações são feitas com os processos definidos na tabela 4.1, conforme descrito na seção 4.1. Deve-se destacar que sendo este controlador relativamente bem conhecido, e as suas características amplamente discutidas na literatura, as simulações não serão tão extensivas nem intensivas quanto as realizadas para o GPC.

#### 4.3.1.ESCOLHA DOS POLINÔMIOS DO CONTROLADOR

Como no caso do GPC, uma evolução de variância mínima permite obter, no caso não adaptativo, menor tempo de subida e menor sobre-elevação. Entretanto quando o sistema é adaptativo, em processos de fase não mínima esta situação pode acarretar problemas de estabilidade ou degradação de desempenho que podem ser evitados afastando-se desta condição.

O primeiro processo a ser simulado, é o N° 1 da tabela 4.1. Inicialmente observa-se o desempenho para a condição de variância

---

<sup>10</sup> Os polinômios de projeto  $P$  e  $Q$  (principalmente este último) permitem melhorar o desempenho estocástico (ver p. e. [MEN84]). Entretanto neste trabalho recorre-se ao filtro  $T$  para essa função.

mínima, isto é, para:

$$P(z^{-1}) = 1;$$

$$Q(z^{-1}) = 0;$$

$$R(z^{-1}) = 1;$$

Assumindo o atraso conhecido ( $d = 1$ ) obteve-se o desempenho da saída apresentado na figura 4.3.1a. Na figura 4.3.2, é simulado o mesmo processo, assumindo-se que o atraso de transporte é igual a 2. Neste caso o atraso utilizado no controlador é maior que o atraso real do processo e o desempenho do sistema é sumamente deteriorado. Quando o atraso do processo é maior que 1, como no processo 2 (tabela 4.1), se o atraso de transporte não for conhecido corretamente, novamente a saída ficará deteriorada. Nas figuras 4.3.3 ilustra-se o desempenho deste processo com os polinômios como foram definidos anteriormente e para o atraso  $d = 5$ . Assumindo qualquer outro atraso, ou o sistema é instável, já que a solução da identidade polinomial (equação 2.4.7) provoca que o primeiro elemento do polinômio  $H(z^{-1})$  (equação 2.7.27) seja muito pequeno provocando grandes valores da excitação.

$d = 1$

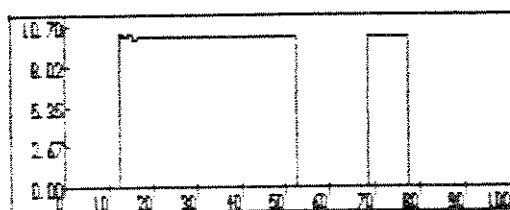


Fig. 4.3.1a

$d = 2$

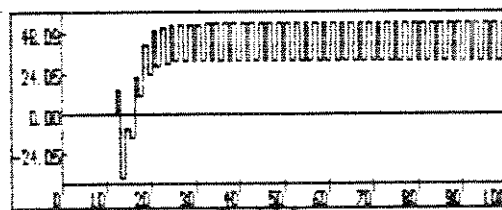


Fig. 4.3.2a

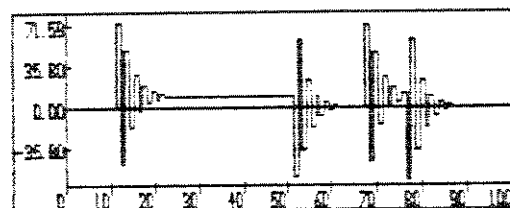


Fig. 4.3.1b

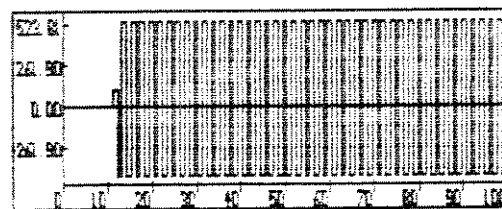


Fig. 4.3.2b

$$d = 5$$

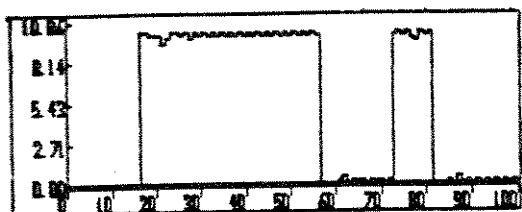


Fig. 4.3.3a

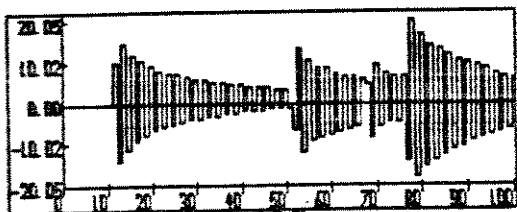


Fig. 4.3.3b

Quando o processo a ser controlado é de fase não mínima, como o N°5 da tabela 4.1, a condição de variância mínima imposta pela escolha particular dos polinômios ( $P(z^{-1}) = R(z^{-1}) = 1$ , e  $Q(z^{-1}) = 0$ ), provoca instabilidade. Entretanto, a escolha dos polinômios depende do processo a ser controlado, pelo que é impossível de se definir valores gerais que forneçam desempenho do sistema aceitável. Na figura 4.3.4. mostra-se o desempenho sistema quando os polinômios tem os valores:

$$P(z^{-1}) = 1 - 0.8 z^{-1};$$

$$Q(z^{-1}) = 0.9;$$

$$R(z^{-1}) = 0.2;$$

Nessa figura pode-se observar um desempenho ainda que fora da condição de variância mínima, pode chegar a ser aceitável<sup>11</sup>. Entretanto para o processo de fase não mínima, e instável em malha aberta indicado como N° 6 da tabela 4.1, estes polinômios de ponderação provocam uma evolução instável do sistema (fig 4.3.5).

<sup>11</sup> Sempre que o tempo de subida e a sobre-elevação não superem os valores desejados.

Proc. Nº 5,  $d = 1$   
 $P = 1 - 0.8 z^{-1}$   
 $Q = 0.9$ ;  $R = 0.2$

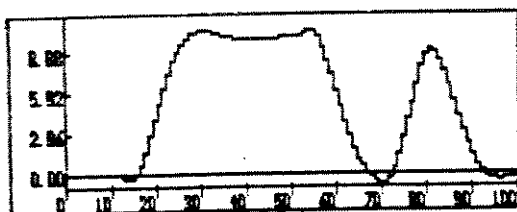


Fig. 4.3.4a

Proc. Nº 6,  $d = 1$   
 $P = 1 - 0.8 z^{-1}$   
 $Q = 0.9$ ;  $R = 0.2$

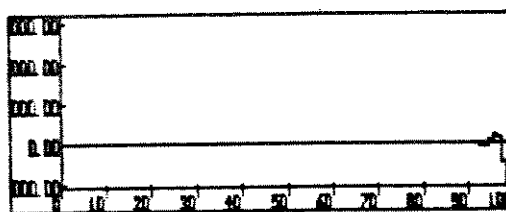


Fig. 4.3.5a

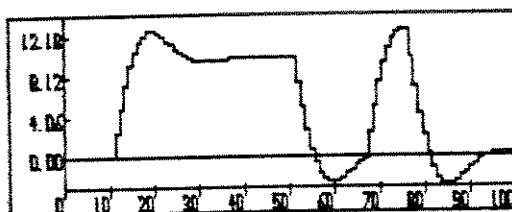


Fig. 4.3.4b

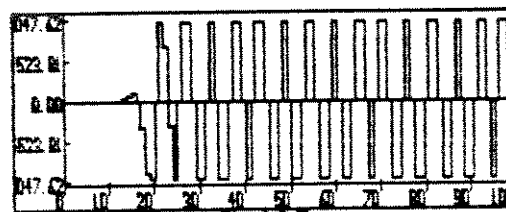


Fig. 4.3.5b

Como foi indicado no começo da seção, uma outra abordagem para a obtenção dos polinômios de ponderação é fornecida pelas equações de alocação de pólos dadas por:

$$B P + \alpha Q = S$$

$$r_0 = P(1) \text{ com } n_r = 0$$

Neste caso a utilização deste controlador é simplificada devido a que:

- i).-Não é necessário conhecer com precisão o atraso do processo;
- ii).-A quantidade de polinômios de projeto é reduzida.

A primeira das afirmações pode ser confirmada com a ajuda do processo Nº 2. Na figura 4.3.6 pode-se observar o desempenho deste processo quando os pólos em malha fechada são especificados pelo polinômio

$$S(z^{-1}) = 1 \text{ (i.e., um pólo na origem)}$$

sendo assumido um atraso igual à unidade. O mesmo desempenho é obtido na figura 4.3.7 com a mesma alocação de pólos mas assumindo o atraso 5 (exato). Se o atraso assumido supera o valor real a

deterioração da saída é muito grande como pode ser observado na figura 4.3.8 para  $d = 6$ .

$d = 1, S = 1$

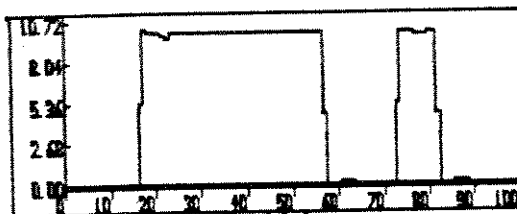


Fig. 4.3.6a

$d = 5, S = 1$

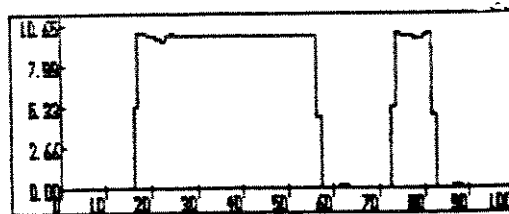


Fig. 4.3.7a

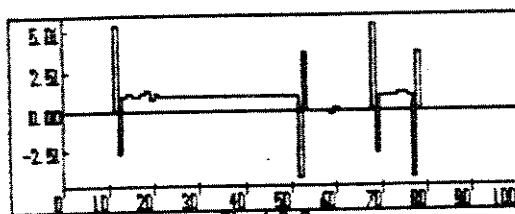


Fig. 4.3.6b

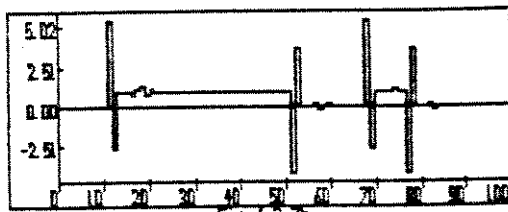


Fig. 4.3.7b

$d = 6, S = 1$

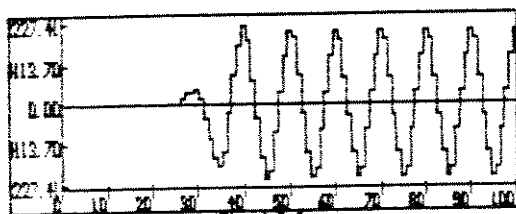


Fig. 4.3.8a

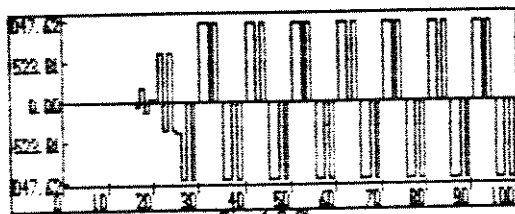


Fig. 4.3.8b

A alocação de pólos permite ainda uma simplificação nos processos de fase não mínima. Na figura 4.3.9 ilustra-se o desempenho do processo 5 quando os pólos alocados ficam na origem ( $S = 1$ ). Ainda que esta alocação não corresponda ao controle de variância mínima, o seu desempenho é similar sem a instabilidade provocada pela não desintonização<sup>12</sup> imposta nessa condição. Na figura 4.3.10 pode-se observar o desempenho do processo N° 6. Na citada figura é evidente um desempenho novamente similar ao de

<sup>12</sup> Desintonização é a condição pela qual se evita a instabilidade produzida pelo cancelamento dos zeros de fase não mínima.

variância mínima. Assim a escolha dos parâmetros de projeto fica mais simples, sendo o polinômio  $S$  ou fornecido pelo usuário para satisfazer requisitos de desempenho (em quaisquer das formas indicadas na seção 4.2), ou então escolhido como a unidade o que permitirá obter um desempenho similar ao de variância mínima. Esta escolha particular não garante a estabilidade do sistema, pelo que como é sugerido em [FRA86]<sup>13</sup>, pode-se utilizar o método do lugar de raízes simétrico.

Proc. Nº 5  
d = 1, S = 1

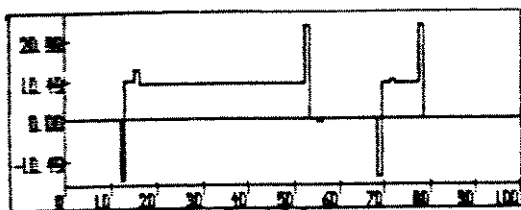


Fig. 4.3.9a

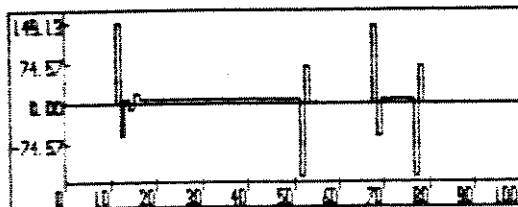


Fig. 4.3.9b

Proc. Nº 6  
d = 1, S = 1

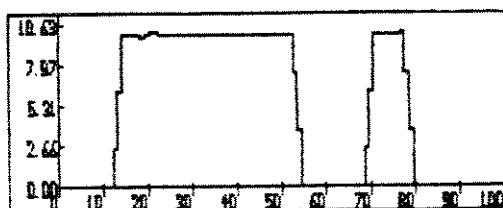


Fig. 4.3.10a

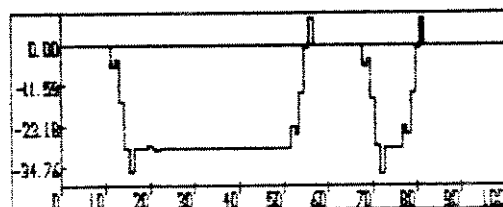


Fig. 4.3.10b

Das simulações apresentadas obtém-se o seguinte conjunto de regras para a base de conhecimento de configuração:

SE o polinômio do denominador da função de transferência em malha fechada (PM)  $\hat{E}$  fornecido pelo usuário  
ENTÃO o modo do GMV  $\hat{E}$  por alocação de pólos.

SE o processo  $\hat{E}$  de fase mínima  
E o atraso do processo  $\hat{E}$  conhecido  
ENTÃO o modo do GMV  $\hat{E}$  normal.

<sup>13</sup> Este enfoque não é tratado no presente trabalho.

SE (o processo *É* de fase não mínima  
OU o atraso do processo *NÃO É* conhecido  
OU o atraso do processo *É* variável)  
ENTÃO o modo do GMV *É* por alocação de pólos.

SE o modo do GMV *É* por alocação de pólos  
E o polinômio do denominador da função de transferência em malha  
fechada (PM) *É* fornecido pelo usuário  
ENTÃO a ordem do polinômio  $S = n_{pm}$   
E o polinômio  $S = PM$ .

SE o modo do GMV *É* por alocação de pólos  
E o polinômio do denominador da função de transferência em malha  
fechada (PM) *NÃO É* fornecido pelo usuário  
ENTÃO a ordem do polinômio  $S = 0$   
E o polinômio  $S = 1$ .

SE o modo do GMV *É* normal  
ENTÃO a ordem dos polinômios  $P$ ,  $Q$  e  $R = 0$   
E  $P = 1$   
E  $Q = 0^{14}$   
E  $R = 1$ .

Com respeito ao supervisor, se os requisitos de desempenho determinísticos não forem satisfeitos só pode-se dever a um conhecimento deficiente do processo (má estimação), e não a problemas no algoritmo de controle pelo que não são propostas regras para este supervisor<sup>15</sup>. Uma opção possível (ainda que não tratada neste trabalho), é a de modificar o polinômio  $S$  segundo critérios de lugar de raízes simétricos [FRAB6], para aumentar a robustez do controlador.

---

<sup>14</sup> Esta condição pode provocar instabilidade por excesso de excitação quando existam dinâmicas não modeladas. Entretanto a especificação de desempenho desejado sempre deve fornecer um filtro  $Q$  diferente de zero.

<sup>15</sup> Entretanto da literatura poderia-se obter uma grande quantidade de regras, mas isso fica fora do escopo deste trabalho.

#### 4.3.2.-ESCOLHA DO FILTRO DAS PERTURBAÇÕES $T(z^{-1})$

Como acontece no GPC, o filtro  $T$  permite melhorar o desempenho do sistema frente a alguns tipos de perturbações, evitando-se a estimação do polinômio  $C(z^{-1})$ . Como se verá a seguir este filtro tem o desempenho similar ao achado no GPC. Em todas as simulações realizadas a seguir faz-se alocação de pólos na origem, isto é  $S = 1$ .

Novamente simula-se o processo 1 adicionado com o polinômio da perturbação dado por:

$$C(z^{-1}) = 1 - 1.5 z^{-1} + 0.7 z^{-2}$$

Na figura 4.3.11 observa-se a evolução do sistema assumindo que o processo é conhecido, e com uma perturbação branca  $N(0, 0.5^2)$  aplicada no intervalo de tempo  $(30, 60]$ . O desempenho observado nessa figura é assim o melhor possível (devido ao conhecimento do polinômio  $C$ ). Assumindo o processo não conhecido, procede-se a estimar 2 coeficientes  $A$  e 2  $B$ , isto é assume-se perturbação branca. O desempenho nestas condições para a perturbação indicada anteriormente é ilustrado na figura 4.3.12. Com a estimação de 2 coeficientes  $C$  é mostrado na figura 4.3.13 onde pode-se observar uma grande oscilação no período de adaptação. Quando se utiliza um filtro  $T$  dado por:

$$T(z^{-1}) = 1 - 1.6 z^{-1} + 0.64 z^{-2} \text{ (i.e., um par de pólos em 0.8)}$$

e sem a estimação do polinômio  $C$ , obtém-se o resultado ilustrado na figura 4.3.14. Este desempenho acha-se entre o obtido com conhecimento total do processo e o mostrado na figura 4.3.12 (sem a estimação do polinômio  $C$ ). Como pode-se verificar pelas figuras a inclusão deste filtro (como acontece também no GPC) não modifica o desempenho determinístico.

### Parâmetros corretos

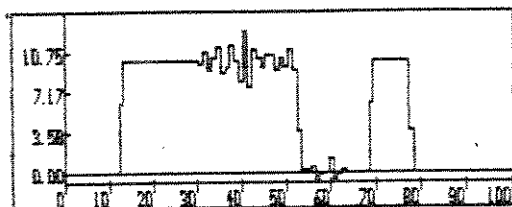


Fig. 4.3.11a

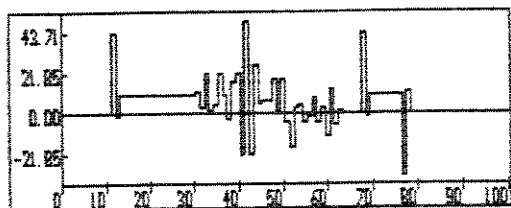


Fig. 4.3.11b

### Estimação de A e B

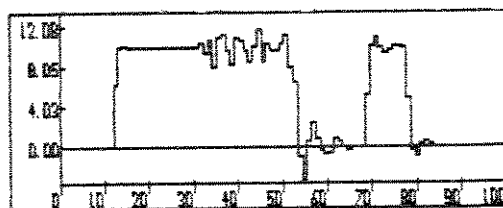


Fig. 4.3.12a

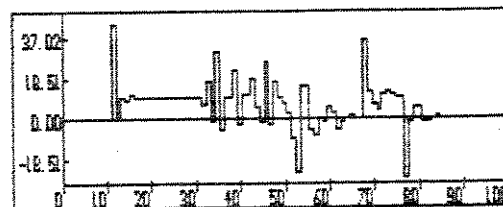


Fig. 4.3.12b

### Estimação de A, B e C

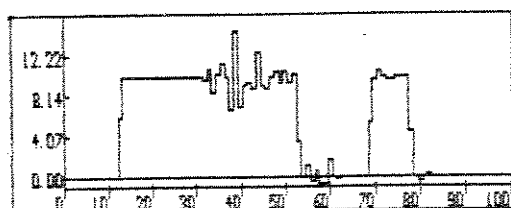


Fig. 4.3.13a

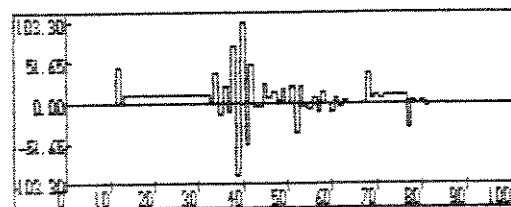


Fig. 4.3.13b

### Estimação de A e B $T = 1 - 1.6z^{-1} + 0.64z^{-2}$

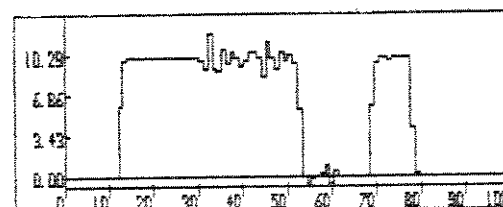


Fig. 4.3.14a

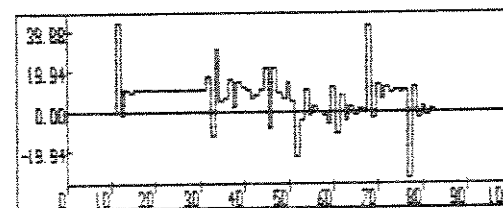


Fig. 4.3.14b

Como no caso do GPC, no GMV a incorporação do filtro das perturbações permite além de uma melhora frente a perturbações estocásticas, melhorar também o desempenho quando as perturbações são do tipo degrau sobre a carga. Isto pode ser verificado nas figuras 4.3.15 (desempenho do processo sem estimação), 4.3.16, (estimação dos polinômios  $A(z^{-1})$  e  $B(z^{-1})$ ) e 4.3.17 (desempenho com estimação e com a incorporação do filtro  $T(z^{-1})$  de segunda ordem definido anteriormente).

#### Parâmetros corretos

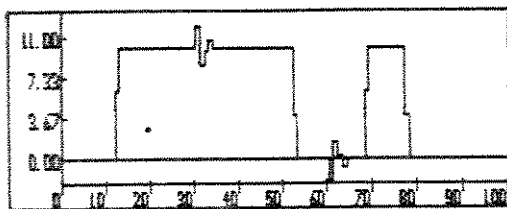


Fig. 4.3.15a

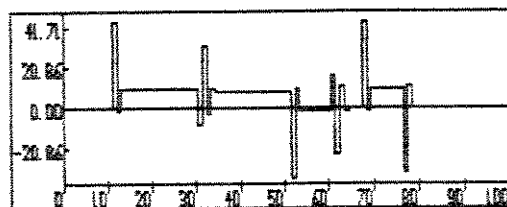


Fig. 4.3.15b

#### Estimação de A e B

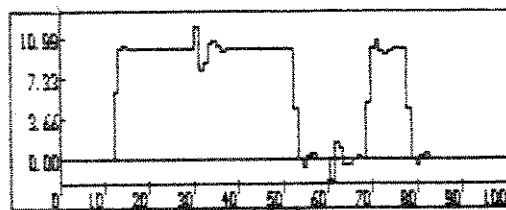


Fig. 4.3.16a

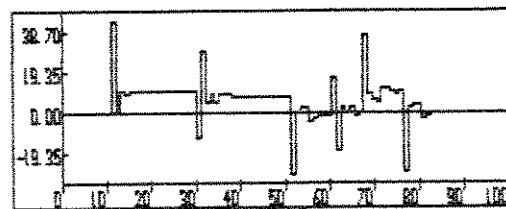


Fig. 4.3.16b

#### Estimação de A e B $T = 1 - 1.6z^{-1} + 0.64z^{-2}$

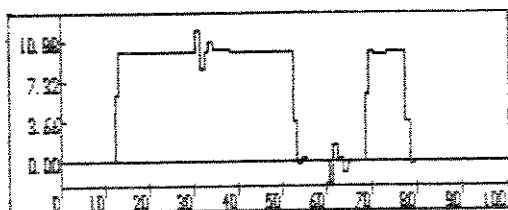


Fig. 4.3.17a

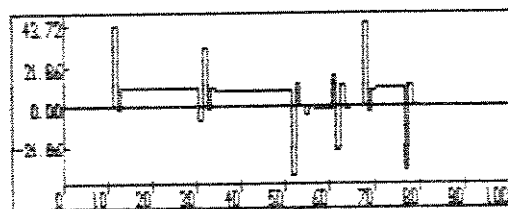


Fig. 4.3.17b

Das simulações pode-se concluir que as regras obtidas anteriormente para o filtro T quando o controlador em uso é o GPC, também são aplicáveis quando o controlador utilizado é o GMV.

#### 4.4.-CONTROLADOR P.I.D

Este tipo de controlador é utilizado com sucesso no controle de diversos processos, e muitos sistemas baseados em conhecimento já foram desenvolvidos para melhorar o desempenho deste controlador [SIL88]. Assim sendo, neste trabalho limitou-se a incluí-lo como uma opção para controlar processos com

características "simples".

A Base de Conhecimento para este controlador foi obtido das tabelas de Takahashi para resposta degrau [KUC88]. Os coeficientes indicados nas regras fornecidas nesta seção são:

Tr: Tempo de subida, constante de tempo principal considerando o sistema de primeira ordem.

d: Atraso de transporte do processo.

Ts: Tempo de amostragem.

KP: Ganho proporcional.

KI: Ganho Integral.

KD: Ganho derivativo.

As regras para a base de conhecimento de configuração são:

SE o controlador desejado *É* do tipo PID  
E os parâmetros do modelo *ESTÃO* disponíveis

$$\text{ENTÃO } KP = \frac{1.2 * Tr}{d+Ts/2} - \frac{0.3*Tr*Ts}{(d+Ts/2)^2}$$

$$E \text{ } KI = \frac{0.6*Tr*Ts}{KP*(d+Ts/2)^2}$$

$$E \text{ } KD = \frac{0.5*Tr}{KP*Ts}$$

SE o controlador desejado *É* do tipo PI  
E os parâmetros do modelo *ESTÃO* disponíveis

$$\text{ENTÃO } KP = \frac{0.9 * Tr}{d+Ts/2} - \frac{0.135*Tr*Ts}{(d+Ts/2)^2}$$

$$E \text{ } KI = \frac{0.27*Tr*Ts}{KP*(d+Ts/2)^2}$$

SE o controlador desejado *É* do tipo P  
E os parâmetros do modelo *ESTÃO* disponíveis

$$\text{ENTÃO } KP = \frac{Tr}{d+Ts}$$

As regras para a base de conhecimento de supervisão surgem das características próprias da influência de cada parâmetro deste controlador no desempenho do sistema.

SE o controlador utilizado *É* do tipo P  
E o sinal de saída *É* estável e com erro de regime  
ENTÃO sugere-se incorporar o ganho integral.

SE o controlador utilizado *É* do tipo P  
E o sinal de saída *É* estável mas oscilatório  
ENTÃO sugere-se reduzir o ganho proporcional.

SE o controlador utilizado *É* do tipo PI  
E o sinal de saída *É* estável e com erro de regime  
E a primeira sobre-elevação *É* desprezível  
ENTÃO sugere-se aumentar o ganho integral.

SE o controlador utilizado *É* do tipo PI  
E o sinal de saída *É* estável e com erro de regime  
E a primeira sobre-elevação *É* MAIOR QUE 0  
ENTÃO sugere-se incorporar o ganho derivativo.

SE o controlador utilizado *É* do tipo PI  
E o sinal de saída *É* estável mas oscilatório  
E o tempo de subida em malha fechada *É* MENOR QUE  
o tempo de subida desejado  
ENTÃO sugere-se reduzir o ganho proporcional.

SE o controlador utilizado *É* do tipo PI  
E o sinal de saída *É* estável mas oscilatório  
E o tempo de subida em malha fechada *É* MAIOR QUE  
o tempo de subida desejado  
ENTÃO sugere-se reduzir o ganho integral.



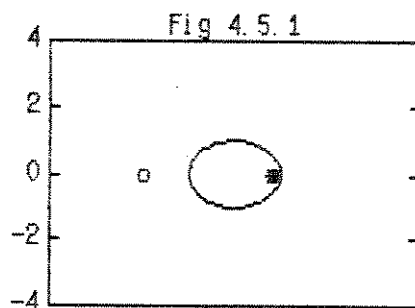


Fig. 4.5.1 Lugar de raízes do processo simulado

Na figura 4.5.1 pode-se verificar que é um processo de fase não mínima. O comportamento dinâmico é definido pelo par "tempo de subida desejado" - "sobre-elevação máxima desejada". Estes valores são inicialmente requeridos pelo sistema com o seguinte diálogo<sup>16</sup>:

voce poderia dizer se

'a estrategia de controle' eh 'de seguimento de modelo'?

voce concorda:

s.\n.\p.\como sabe que META.\v.\a.

==>p.

estou tentando demonstrar pela regra\_3 que:

'os parametros de resposta de servo' estao disponiveis

voce concorda com o que se esta procurando?

s.\n.\p.\como sabe que META.\v.\a.

==> como sabe que 'os parametros de resposta de servo' estao  
disponiveis

esta sendo obtido usando a regra\_3

que diz:

---

<sup>16</sup> Devido a que mostrar toda uma sequência de consulta ocuparia muito espaço, optou-se por indicar os diálogos mais ilustrativos do funcionamento do sistema.

se 'a estrategia de controle' eh 'de seguimento de modelo'  
e 'o tempo de subida desejado (Tr)' esta disponivel  
e 'a sobre-elevacao desejada (Mp)' esta disponivel  
e 'a largura de faixa de malha fechada (w)' :=  
'largura de faixa calculada de Mp e Tr'  
e 'o fator de amortecimento de malha fechada (sita)':=  
'fator de amortecimento calculado de Mp e Tr'  
e 'a ordem do polinomio P' := 2  
e 'o polinomio P' := 'polinomio calculado de sita e w'  
entao 'os parametros de resposta de servo' estao disponiveis.

voce poderia dizer se

'a estrategia de controle' eh 'de seguimento de modelo'?

voce concorda:

s.\n.\p.\como sabe que META.\v.\a.

==>s.

Na sequencia o sistema pede para o usuário fornecer o tempo de subida e a sobre-elevação desejada.

ingresse o valor do 'tempo de subida desejado (Tr)', (0, 10).

==>2.

ingresse o valor da 'a sobre-elevacao desejada (Mp)', (0, 1).

==>0.12.

Destes dois parâmetros são obtidos outros parâmetros que podem ser utilizados no sistema, entre eles o polinômio  $P(z^{-1})$  que determina o par de pólos de malha fechada.

$$P(z^{-1}) = (1 - 1 z^{-1} + 0.35 z^{-2})$$

Para a estimação do modelo do processo assume-se o conhecimento das ordens do modelo, e que este tem parâmetros constantes. Assim, é realizada a estimação dos parâmetros auxiliados pelo seguinte diálogo:

voce poderia dizer se

'os parametros do processo' 'nao sao' variaveis

voce concorda:

s.\n.\p.\como sabe que META.\v.\a.

==>p.

estou tentando demonstrar pela regra\_11 que:

identificacao.

voce concorda com o que se esta procurando?

s.\n.\p.\como sabe que META.\v.\a.

==>como sabe que identificacao.

esta sendo obtido usando a regra\_11

que diz:

se 'os parametros do processo' 'nao sao' variaveis

e 'a identificacao do modelo' := ativada

e 'o seguimento de ruptura de modelo' := desativado

e 'os parametros pre-especificados para

o seguimento de ruptura' estao disponiveis

e 'os parametros gerais de identificacao' estao disponiveis

entao identificacao.

voce poderia dizer se

'os parametros do processo' 'nao sao' variaveis

voce concorda:

s.\n.\p.\como sabe que META.\v.\a.

==>s.

Esta regra permite: i)ativar o módulo identificador; ii)obter parâmetros pré-escolhidos para fazer o seguimento das variações que podem ocorrer no modelo é assim supervisionar (sem atuar) no sistema; iii)Os parâmetros gerais de identificação incluem as ordens do modelo ( $n_a = 2$ ,  $n_b = 1$ ,  $n_c = 0$ ), o fator de esquecimento ( $\xi = 1$ ) e o valor inicial da diagonal da matriz P do identificador ( $\text{diagP} = 100$ )..

Uma vez obtidos estes valores procede-se à identificação do processo. Após 100 iterações obteve-se o seguinte modelo do processo:

$$A(z^{-1}) = 1 - 1.7 z^{-1} + 0.72 z^{-2}$$

$$B(z^{-1}) = 0.1 + 0.2 z^{-1}$$

A estimação foi evidentemente correta, e do modelo pode se obter: atraso = 1; quantidade de zeros de fase não mínima = 1; sistema estável em malha aberta (pólos dentro do círculo unitário do plano  $z$ ).

Na sequência é ativado o módulo supervisor da identificação o qual verifica que:

pela regra\_10\_ef concluiu-se que:

'os parametros' tem 'os valores corretos'.

voce concorda:

s.\n.\p.\como sabe que META.\v.\a.

==>p.

porque a regra\_10\_ef diz que

se 'o estado' eh convergente

e 'a correlacao do erro de previsao' eh branca

entao 'os parametros' tem 'os valores corretos'.

sendo as condicoes provadas pelos fatos da Base de Dados,

exceto o fato 'o estado' eh convergente

que foi concluido da regra\_4\_ef.

voce concorda com esta conclusao?.

s.\n.\p.\como sabe que META.\v.\a.

==>s.

Mais informação sobre a supervisão do algoritmo de modelagem de processos pode ser encontrada em [ARR88].

Na continuação ativa-se a escolha e configuração do algoritmo de controle. Neste exemplo escolheu-se o algoritmo (GPC) mais com fins ilustrativos do que como uma necessidade. Os dados para os horizontes deste algoritmo de controle são obtidos das seguintes regras:

```

regra_10_GPC <:
se 'o processo' eh 'de fase nao minima e estavel em malha aberta'
e 'os parametros do processo' 'nao sao' variaveis
e 'a quantidade de zeros de fase nao minima' eh conhecida
e 'o atraso do processo' eh conhecido
entao N1 := 'a quantidade de zeros de fase nao minima' +
            'o atraso do processo'.

```

o que determina  $N1 = 2$ .

```

regra_11_GPC <:
se 'o processo' eh 'de fase nao minima e estavel em malha aberta'
e 'a ordem do polinomio  $A(z^{-1})$ ' eh conhecida
entao Nu := 'a ordem do polinomio  $A(z^{-1})$ ' + 1.

```

assim  $Nu = 3$ .

```

regra_17_GPC <:
se 'o processo' eh 'de fase nao minima e estavel em malha aberta'
e N1 esta disponivel
e Nu esta disponivel
e  $N1 + 1$  'eh menor que'  $Nu +$ 
            'a quantidade de zeros de fase nao minima'
entao  $N2 = Nu +$  'a quantidade de zeros de fase nao minima'.

```

donde se obtém  $N2 = 4$ .

Desde que todos os coeficientes que determinam os valores destes horizontes devem ser definidos anteriormente, a consulta destas regras é transparente para o usuário.

Da mesma forma são obtidos os valores dos filtros  $P(z^{-1})$  e  $R(z^{-1})$ , que são determinados pelo modelo de malha fechada escolhido inicialmente. Para a determinação do valor do coeficiente  $\lambda$  é necessario conhecer se o sistema possui transitório negativo não desprezível. Para tal é necessário fazer uma simulação em malha aberta. Este teste ainda não foi

considerado no sistema implementado, pelo que se optou por assumir que o transitório negativo do processo é desprezível. Quando esta hipótese estiver errada, a supervisão permitirá corrigir este erro. Nesta etapa o sistema permitirá configurar outros parâmetros utilizados na supervisão tais como o valor máximo do erro de saída e o valor máximo do índice ITAE.

Finalmente é realizada a simulação do processo em malha fechada com o resultado apresentado na figura 4.5.2.

#### Desempenho do processo

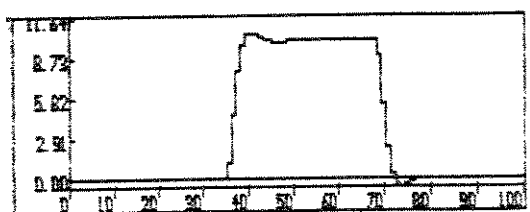


Fig. 4.5.2a

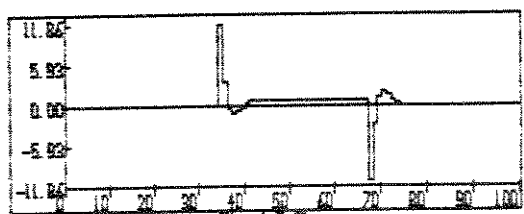


Fig. 4.5.2b

Ativada a supervisão do algoritmo obteve-se a seguinte conclusão:

pela regra\_25\_ef concluiu-se que:

'o sinal de saída' eh

'estavel nao oscilatorio e sem erro de regime'.

voce concorda:

s.\n.\p.\como sabe que META.\v.\a.

==>p.

porque a regra\_25\_ef diz que

se 'a media do erro de saída' 'eh menor que'

'a media maxima do erro de saída'

e 'o indice ITAE' 'eh menor que' 'o maximo indice ITAE'

e 'o erro de regime' 'eh menor que'

$0.01 * \text{'a amplitude da referencia'}$

entao 'o controle' eh 'estavel nao oscilatorio e sem erro de regime'.

sendo as condicoes provadas pelos fatos da Base de Dados,  
voce concorda com esta conclusao?.  
s.\n.\p.\como sabe que META.\v.\a.  
==>s.

pela regra\_26\_ef concluiu-se que:

    '0 controle do sistema' eh satisfatorio.

voce concorda:

s.\n.\p.\como sabe que META.\v.\a.  
==>p.

porque a regra\_26\_ef diz que

se '0 sistema em malha fechada' eh

    'estavel nao oscilatorio e sem erro de regime'

e '0 tempo de crescimento em malha fechada' 'eh menor que'

    '0 tempo de crescimento desejado'

e '0 primeiro sobrepico' 'eh menor que'

$1.1 \times$  '0 sobrepico desejado'

entao '0 controle do sistema' eh satisfatorio.

sendo as condicoes provadas pelos fatos da Base de Dados,  
exceto o fato '0 sinal de saida' eh

    'estavel nao oscilatorio e sem erro de regime'.

que foi concluido da regra\_25\_ef.

voce concorda com esta conclusao?.

s.\n.\p.\como sabe que META.\v.\a.  
==>s.

Desta forma conclui-se a consulta ao sistema ficando a  
possibilidade de uma nova simulacao ou uma reinicializacao do  
sistema por parte do usuario.

## 4.6.-CONCLUSÕES

A obtenção de conhecimento através de simulações pode ser uma ferramenta importante para o desenvolvimento de sistemas especialistas. Entretanto, quando a quantidade de parâmetros é elevada, a explosão combinatória é inevitável, e deve-se recorrer a outros mecanismos para se obter a Base de Conhecimento. Estes mecanismos podem ser ferramentas automáticas, que por meio de simulações adquiram o conhecimento sobre as variáveis do sistema.

Com respeito aos controladores desenvolvidos neste capítulo observou-se que:

- i).-Se o processo é relativamente bem conhecido, a utilização de controladores mais complexos do que o GMV não apresenta grandes vantagens quanto ao melhor desempenho do sistema.
- ii).-Quando os processos a serem controlados possuem características não totalmente identificadas, a utilização do GPC pode melhorar o desempenho do sistema às custas de maior tempo de cálculo (horizontes elevados)
- iii).-Obteve-se considerável conhecimento para a base de configuração para os controladores. Entretanto, para a base de supervisão o conhecimento adquirido foi pequeno. Isto deve-se a que a robustez destes controladores é garantida pela correta estimação dos parâmetros. Para se obter mais conhecimento para a supervisão deva-se analisar o desempenho dos controladores nos limites de funcionamento (saturação de atuadores, etc).

## CAPÍTULO

### V

## CONCLUSÕES

## 5.1.-SINTESE DO TRABALHO

O Sistema Especialista desenvolvido neste trabalho realiza as tarefas de configuração e supervisão dos algoritmos de modelagem e controle de processo. O sistema auxilia inicialmente o usuário na configuração dos diferentes algoritmos de controle implementados. No controlador PID, seleciona os valores dos ganhos: Proporcional, Integral e Derivativo. No GMV escolhe-se os filtros que determinam o desempenho do sistema em malha fechada, e no GPC escolhe os horizontes de previsão e controle, assim como os filtros que alteram o desempenho do sistema em malha fechada.

Na etapa de supervisão o sistema analisa o desempenho dos algoritmos de identificação e controle e, caso necessário, apresenta ao usuário sugestões de modificação dos parâmetros de regulação dos algoritmos, para melhorar o desempenho do sistema em malha fechada.

No Sistema Especialista implementado é possível distinguir três partes principais:

- i).-Os algoritmos de modelagem e controle de processo;
- ii).-O sistema de configuração; e
- iii).-O sistema de supervisão.

O algoritmo de modelagem de processos é o de Matriz Estendida e as variações de parâmetros no processo são detectadas e corrigidas pelo método do erro previsto. Os algoritmos de controle implementados são PID, GMV, GPC. Estes dois últimos foram desenvolvidos a partir de um modelo do processo tipo CARIMA. Todos estes algoritmos foram escritos na linguagem PASCAL devido a necessidade de manipulação numérica.

O sistema de configuração está baseado em uma estrutura de conhecimento em forma de Regras de Produção que são pesquisadas por uma máquina de inferência com encadeamento reverso. O sistema de supervisão também tem como base a estrutura de conhecimento em

forma de regras de produção; neste caso, são pesquisadas por uma máquina de inferência com encadeamento direto. Todas estas estruturas assim como a Base de Dados do Sistema Especialista e a interface com o usuário foram escritas na linguagem PROLOG que facilita a manipulação simbólica.

## 5.2.-CONCLUSÕES I: OS CONTROLADORES E PROPOSTAS DE TRABALHOS FUTUROS

Os controladores adaptativos estudados no presente trabalho mostraram ser suficientemente eficazes para o controle de processos lineares, quando os parâmetros que descrevem estes processos estavam corretamente estimados. No GMV, além desta estimação correta, é necessário o conhecimento do atraso de transporte do processo para que seja garantido o desempenho correto e a estabilidade do sistema. No GMV com alocação de pólos o atraso não desempenha um papel importante; entretanto, a existência de termos comuns no numerador e denominador da função de transferência do processo, fato que pode acontecer pela sobre-estimação das ordens do modelo do processo, pode inviabilizar a utilização deste controlador. O GPC é uma estratégia de controle que permite controlar processos onde não se conhece o valor do atraso de transporte e/ou as ordens do modelo do processo. Contudo, o GPC também é baseado no princípio de equivalência certa; quando se tem estimação incorreta dos parâmetros do modelo, pode ocorrer uma degradação no desempenho do sistema.

Outro fator a ser considerado são as limitações físicas do sistema, que não foram analisadas neste trabalho, mas que devem ser consideradas em qualquer sistema comercial. A análise dos controladores nos limites de funcionamento deve fornecer conhecimento adicional para uma supervisão adequada do sistema.

As propostas de trabalhos futuros em relação a tarefa de controle podem ser sintetizadas em:

I).-Incrementar as Bases de Conhecimento de configuração e supervisão para executar as tarefas de identificação e detecção de não estacionaridade.

II).-Analisar o desempenho dos controladores nos limites do funcionamento.

## 5.3.-CONCLUSÕES II: O SISTEMA ESPECIALISTA E PROPOSTAS DE TRABALHOS FUTUROS

### 5.3.1.-REPRESENTAÇÃO DO CONHECIMENTO

A representação de conhecimento por regras de produção mostrou ser eficiente na área de controle. A geração das bases de conhecimento nessa representação não apresentou problemas e a comunicação com o usuário é transparente. O tempo necessário para a obtenção de uma resposta por parte do Sistema Especialista durante a configuração é aceitável e a inclusão de novo conhecimento não tem prejudicado o desempenho do Sistema Especialista. Entretanto, quando a tarefa desenvolvida pelo Sistema Especialista é a de supervisão, ficou evidente a necessidade de incluir Meta-regras para organizar o conhecimento e para aumentar a velocidade de resposta.

Como a aplicação atual não é em "tempo real", a interação com o usuário não evidenciou problemas de comunicação. Entretanto, para a utilização do Sistema Especialista em "tempo real" será necessário que o sistema execute ações em tempos que inviabilizam a interação com o usuário. Para resolver esta dificuldade sugere-se a separação do conhecimento no aspecto temporal, como é indicado a seguir:

i).-Conhecimento de Longa duração: por exemplo, cargas diárias, desempenho global (medido por parâmetros diários), etc. Sobre estes parâmetros o usuário pode e deve interagir com o Sistema Especialista.

ii).-Conhecimento de Média duração: por exemplo, sobre-elevação, erro de regime, variâncias da entrada e da saída, etc, sobre o qual o usuário pode ou não interagir. A inclusão do usuário no processo de inferência dependerá da dinâmica do processo que se deseja controlar.

iii).-Conhecimento de Curta duração: como saturações, variações abruptas, etc, situações onde é necessária uma ação rápida. Neste caso o usuário deverá somente ser informado do problema e da ação executada.

Outra proposta de trabalhos futuros surge das características próprias do conhecimento na área de controle. É comum serem citadas frases do tipo "o erro de regime é muito alto", ou "o comportamento do sistema é lento". Estas frases utilizam qualificadores do tipo "alto", "baixo", "lento" "rápido", etc. A lógica nebulosa é uma forma de tratar estes qualificadores. A representação por regras de produção utilizada pode ser facilmente adaptada para incluir este tipo de lógica.

### 5.3.2.-MANIPULAÇÃO DO CONHECIMENTO

A escolha do encadeamento reverso para as tarefas de configuração e do encadeamento direto para as de supervisão permitiu simplificar estas tarefas. Entretanto, a máquina de inferência utilizada na supervisão deve ser otimizada para que a adição de novo conhecimento não prejudique o seu desempenho. O método de resolução de conflitos adotado mostrou ser pouco eficiente quando a base de conhecimento adquiriu um volume considerável. A solução proposta é a divisão do conhecimento por "afinidade", isto é, separar o conhecimento de identificação do conhecimento de controle e, dentro destas divisões, separá-los por

métodos. Por exemplo, o conhecimento sobre controle pode ser dividido segundo o tipo de controlador. Esta divisão pode ser feita com a ajuda de meta-regras que permitam o acesso à divisão requerida no momento oportuno.

### 5.3.3.-AQUISIÇÃO DO CONHECIMENTO

O método de aquisição de conhecimento por simulação utilizado neste trabalho mostrou-se eficaz. As dificuldades observadas devem-se principalmente à explosão combinatória quando existem muitas variáveis a serem analisadas (como ocorre com o controlador GPC). Para resolver este problema foram escolhidos processos com características específicas. Utilizou-se o conhecimento público (i.e., da bibliografia) para analisar as condições de desempenho do controlador. A partir deste conhecimento foram modificados os parâmetros do controlador para analisar a sua resposta, o que forneceu mais conhecimento. Finalmente, este conhecimento acumulado definiu as simulações seguintes com o objetivo de analisar condições não verificadas anteriormente.

A simulação mostrou-se apropriada para a geração das bases de conhecimento. Entretanto, pode-se propor métodos automáticos de aquisição de conhecimento através de exemplos de simulações. Para esta tarefa será necessário:

- i).-Identificar as variáveis a serem medidas. Ao ser analisado o desempenho em um determinado exemplo, utilizaram-se expressões como "desempenho bom", "desempenho ruim", "melhor desempenho que", etc. Entretanto, essa classificação é muito subjetiva e geralmente não indica que parâmetros permitem fazer essa afirmação.
- ii).-Identificar os exemplos que gerem novo conhecimento, pois a análise de todas as possibilidades de simulações pode ser inviável devido ao problema da explosão combinatória.

## 5.4.-CONCLUSÕES III: AS FERRAMENTAS UTILIZADAS

A eficiência do PASCAL como linguagem de cálculo pode ser superada por outra linguagem (p.e., C); contudo, considerando que o sistema não é em "tempo real" a velocidade de resposta observada é aceitável. A implementação em linguagem PROLOG permitiu executar as tarefas de manipulação simbólica em um intervalo de tempo aceitável. Entretanto, a versão utilizada necessita de uma grande quantidade de memória, ainda não facilmente disponível em computadores de pequeno porte. Apesar deste problema e da interface com a linguagem PASCAL não prover muitas facilidades, a combinação destas duas linguagens poderia mostrar-se mais adequada com a utilização de outras versões mais eficientes do compilador PROLOG.

## APÊNDICE A

### A.1.-SOLUÇÃO RECURSIVA DA IDENTIDADE POLINOMIAL (EQUAÇÃO DIOFANTINA)

A seguir detalha-se o procedimento para a resolução da identidade polinomial (a extensão para incluir os filtros  $P(z^{-1})$  e  $T(z^{-1})$ , assim como o polinômio  $\alpha$  é direta)

$$C = F_j A + z^{-j} G_j \quad (A.1)$$

em forma recursiva aplicando método proposto por Clarke [CLK84,CLK85] para o cálculo de um predictor "j" passos à frente.

O algoritmo começa considerando  $j = 1$ , assim

$$G_{j=1} = z (C - A F_{j=1}) \quad (A.2)$$

Mas o elemento  $g_{-1}$  está no futuro, portanto deve ser zero, e sendo  $a_0=1$  obtém-se:

$$F_{j=1} = f_0 = C_0 \quad (A.3)$$

Agora, calcula-se recursivamente para  $j = 2$  até  $(d - 1)$  os polinômios  $F_j$  e  $G_j$  das seguintes equações:

$$f_j^{j+1} = g_c^j \quad (A.4)$$

$$F_{j+1} = F_j + z^{-j} f_j^{j+1} \quad (A.5)$$

e

$$g_{l-1}^{j+1} = g_l^j - g_o^j a_l \quad \text{para } l = 1 \text{ até } na-1 \quad (A.6)$$

$$g_{na}^{j+1} = - g_o^j a_{na} \quad (A.7)$$

O subíndice nas letras maiúsculas significa "iteração" e nas letras minúsculas o subíndice indica "tempo", sendo o superíndice que indica a iteração.

## APÊNDICE B

### B.1.- INCORPORAÇÃO DO FILTRO $T(z^{-1})$ NO CONTROLADOR GMV

Para incorporar o filtro  $T(z^{-1})$  no controlador de variância mínima generalizado, recorre-se a uma modificação da equação do modelo filtrado (equação 2.7.2), obtendo-se

$$TP \ y(t) = z^{-d} \frac{TPB}{\alpha} \Delta u(t) + \frac{TPC}{\alpha} \xi(t) \quad (B.1)$$

Assumindo uma perturbação "branca", a saída filtrada do processo "d" passos à frente é então

$$TP \ y(t+d) = \frac{TPB}{\alpha} \Delta u(t) + \frac{TPC}{\alpha} \xi(t+d) \quad (B.2)$$

A identidade polinomial (2.7.4), pode ser modificada para:

$$\frac{TPC}{\alpha} = F_1 + z^{-d} \frac{G_1}{\alpha} \quad (B.3)$$

com a ordem do polinômio  $F_1$  e  $G_1$  respectivamente  $nf_1 = d-1$  e  $ng_1 = \max(na, nc+np+nt-d)$ .

Aplicando esta partição na equação (B.2) obtém-se

$$TP \ y(t+d) = \frac{TPB}{\alpha} \Delta u(t) + F_1 \xi(t+d) + \frac{G_1}{\alpha} \xi(t) \quad (B.4)$$

No instante "t" a melhor previsão que se pode fazer com as medidas conhecidas (exceto a perturbação atual) é dada por:

$$TP \ \hat{y}(t+d/t) = \frac{TPB}{\alpha} \Delta u(t) + \frac{G_1}{\alpha} \xi(t) \quad (B.5)$$

e

$$TP \ y(t+d) = TP \ \hat{y}(t+d/t) + F_1 \xi(t+d) \quad (B.6)$$

Como foi feito na seção 2.7, a perturbação não mensurável é obtida do processo como

$$\xi(t) = -z^{-d} \frac{B}{C} \Delta u(t) + \frac{\alpha}{C} y(t) \quad (B.7)$$

que permite obter

$$TP \hat{y}(t+d/t) = \frac{G_1}{C} y(t) + \Delta u(t) \left[ \frac{TPB}{\alpha} - z^{-d} \frac{BG_1}{C\alpha} \right] \quad (B.8)$$

sendo que da equação B.3 pode-se obter

$$\frac{F_1 B}{C} = \frac{TPB}{\alpha} - z^{-d} \frac{G_1 B}{C\alpha} \quad (B.9)$$

e assim

$$P \hat{y}(t+d/t) = \frac{G_1}{CT} y(t) + \frac{F_1 B}{T\alpha} \Delta u(t) \quad (B.10)$$

onde todas as medidas são conhecidas no instante "t".

A função de custo a ser minimizada, com a ajuda da equação B.6, é:

$$J = E \left\{ \left[ \left( P \hat{y}(t+d/t) + F_1 \xi(t+d) - R w(t) \right)^2 + \left( Q' \Delta u(t) \right)^2 \right] / M_t \right\} \quad (B.11)$$

onde pode-se observar que todos os parâmetros continuam correspondendo à equação 2.7.12, com exceção do polinômio  $F_1$ . Entretanto este polinômio tem a mesma ordem que o polinômio  $F$ , e a minimização realizada no capítulo II continua válida. Logo obtém-se a seguinte equação de controle.

$$P \hat{y}(t+d/t) - R w(t) + Q \Delta u(t) = 0 \quad (B.12)$$

com 
$$Q = \frac{q_0'}{p_0' b_0} Q'$$

Substituindo  $P \hat{y}(t+d/t)$  na equação anterior pelo obtido na equação (B.10) tem-se

$$G_1 \frac{y(t)}{T} + (F_1 B + QCT) \frac{\Delta u(t)}{T} - RC w(t) = 0 \quad (B.13)$$

Chamando

$$H_1 = F_1 B + QC \quad \text{e} \quad E = -RC \quad (B.14)$$

e os sinais filtrados  $x'(t) = x(t)/T$  obtém-se a equação de controle do GMV (equação de velocidade)

$$G_1 y'(t) + H_1 \Delta u'(t) + E w(t) = 0 \quad (B.15)$$

Finalmente o sinal de controle é dado por:

$$\Delta u(t) = T \Delta u'(t) \quad (B.16)$$

## B.2.-FUNÇÃO DE TRANSFERÊNCIA EM MALHA FECHADA DO CONTROLADOR GMV COM FILTRO $T(z^{-1})$

Para se obter a função de transferência em malha fechada procede-se como no capítulo II. Primeiramente reescreve-se a equação (B.15) com a ajuda da equação B.16, como:

$$\Delta u(t) = -\frac{T G_1}{H_1} y(t) - \frac{T E}{H_1} w(t) \quad (B.17)$$

Substituindo esta equação no modelo do processo obtém-se:

$$(H_1 \alpha + z^{-d} B G_1) y(t) = -z^{-d} T B E w(t) + H_1 C \xi(t) \quad (B.18)$$

Analisando o termo do lado esquerdo pode-se ver que

$$\begin{aligned} H_1 \alpha + z^{-d} B G_1 &= \alpha (F_1 B + Q C T) + z^{-d} B (z^d (T P C - \alpha F_1)) \\ &= C T (B P + \alpha Q) \end{aligned} \quad (B.19)$$

Substituindo esta equação e o polinômio E pelo seu valor na equação B.18, obtém-se a função de transferência em malha fechada

$$y(t) = z^{-d} \frac{B R}{B P + \alpha Q} w(t) + \frac{H_1}{T(B P + \alpha Q)} \xi(t) \quad (B.20)$$

Desta equação pode ser observado que o desempenho determinístico não foi alterado. Entretanto, o comportamento estocástico pode ser melhorado por uma correta escolha do filtro  $T(z^{-1})$ .

## REFERÊNCIAS BIBLIOGRÁFICAS

- [ARR88] L. V. R. de Arruda: *Um supervisor baseado em conhecimento para modelagem de processos*. Tese de Mestrado. UNICAMP, 1988.
- [AST70] K. J. strm: *Introduction to stochastic control theory*. Academic Press, 1970.
- [AST73] K. J. strm and B. Wittenmark: *On self-tuning regulator*. Automática, vol. 9, N° 2, pp 185-199, 1973.
- [AST83] K. J. strm: *Theory and applications of adaptive control*. Automatica vol 19, N° \*\*, pp 471-486, 1983.
- [BOR87] J. Borla: *Ingeniería de software*: Edição EBAI, 1987.
- [BUE89] S. S. Bueno: *Controlador auto-ajustável com estrutura PID*. Tese de Mestrado. UNICAMP, 1989.
- [CAR88] R. J. Carmona y A. D. Teszkiewicz: *Sistemas Expertos e representación del conocimiento*. Edição EBAI, 1988.
- [CLK75] D. W. Clarke and P. J. Gawthrop: *Self-tuning controller*. Proc IEE, vol 122, N° 9, pp 929-934, 1975.
- [CLK79] D. W. Clarke and P. J. Gawthrop: *Self-tuning control*. IEE Proceeding, vol 126, Pt D, N° 6, pp 633-640, 1979.
- [CLK83] D. W. Clarke, A. J. F. Hodgson and P. S. Tuffs: *Offset problem and k-incremental predictors in self-tuning control*. IEE Proceeding, vol 130, Pt D, N° 5, pp 217-225, 1983.

- [CLK84] D. W. Clarke, C. Mohtadi and P. S. Tuffs: *Generalized Predictive Control. Part 1: The basic algorithm. Part 2: Extensions and interpretations.* Department of Engineering Science, O.U.E.L. report number 1555/1557. OXFORD, 1984.
- [CLK89] D. W. Clarke and C. Mohtadi: *Properties of Generalized Predictive Control.* Automatica vol 25, N° 6 pp 859-875, 1989.
- [DUD78] P. O. Duda, J. Gasching, P. E. Hart, K. Kolige, R. Rebok, P. Barrett and J. Slocum: *Development of PROSPECTOR Consultation system for mineral exploration.* Final report, SRI projects 5821 and 6415, SRI International INC. CALIFORNIA, 1978.
- [FAV87a] G. Favler: *Computationally efficient adaptive identification algorithms.* CNRS - LASSY, Université de Nice, and GRECO "Sist mes Adaptatif". Internal report, NICE, 1987.
- [FAV87b] G. Favler: *Self-tuning long range predictive controllers.* 10 th. IFAC. 1987.
- [FAV88] J. R. Favila: *Sistema Especialista para diagnóstico de unidades de disco magnético.* Tese de Mestrado UNICAMP, 1988.
- [FRA86] G. F. Franklin, J. D. Powell and A. Emami-Naeini: *Feedback Control od dynamic systems.* Addison-Wesley, 1986.
- [GAR88] A. García Cerezo, A. Ollero and J. Aracil: *Expert Supervision and auto-tuning of digital control system.* Proceedings of the Int. Workshop on Artificial Intelligence for Industrial Applications, pp531-535, 1988.