TRABALHO DE GRADUAÇÃO

AVALIAÇÃO E IMPLEMENTAÇÃO DE TÉCNICAS DE CONTROLE PARA CONDICIONAMENTO TÉRMICO DE AMBIENTES PREDIAIS

Rodrigo Fontes Souto

Brasília, 24 de março de 2006

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA – ENGENHARIA MECATRÔNICA

UNIVERSIDADE DE BRASILIA
Faculdade de Tecnologia

TRABALHO DE GRADUAÇÃO

AVALIAÇÃO E IMPLEMENTAÇÃO DE TÉCNICAS DE CONTROLE PARA CONDICIONAMENTO TÉRMICO DE AMBIENTES PREDIAIS

Rodrigo Fontes Souto

Banca Examinadora

Prof. Geovany Araújo Borges, UnB/ Elétrica (Orientador)	____________________
Prof. Flávia Maria de Guerra Sousa, UnB/ Elétrica	____________________
Prof. João Ishihara, UnB/ Elétrica	____________________

DEDICATÓRIA

Pensamentos elevados devem ter linguagem elevada (Aristófanes)

Potência não é nada sem controle (Slogan da Pirelli)

Aos meus pais, Maria de Fátima e Gilson, por simplesmente tudo.

Aos meus irmãos, Rafael e Bruno.

A toda minha família.

Aos meus amigos.

AGRADECIMENTOS

Aos meus orientadores, Geovany Araújo Borges e Adolfo Bauchspiess, por suas valiosas orientações, bem como por suas amizades.

Aos professores da UnB e do Colégio Marista, que são muitos para listar aqui, por suas aulas e conhecimentos imprescindíveis à conclusão deste trabalho.

Aos funcionários da UnB, em especial aos do SG-11, que deram uma importante contribuição neste trabalho.

Aos meus colegas de faculdade, em especial Luciano Ginani e Gustavo Amaral, por suas amizades tão importantes.

E, sobretudo, agradecer a Deus pela oportunidade de experimentar esta chance única e maravilhosa que se chama viver.

RESUMO

O condicionador de ar é um dos maiores responsáveis pelo consumo de energia em prédios comerciais. Diferentes algoritmos de controle têm sido propostos para controlar a temperatura de um ambiente de forma a minimizar o consumo de energia, sem deixar de atingir o conforto térmico almejado. Contudo, muitos projetos são feitos de maneira mais qualitativa, sem um componente matemático de explicação. Neste trabalho, é avaliada a utilização de algumas técnicas analíticas para o controle de temperatura em um processo reduzido. Tais técnicas empregam a identificação e o controle baseado no modelo identificado do sistema.

As técnicas de controle avaliadas são o controle polinomial *self tuning*, o controle ótimo e o controlador mais difundido em automação predial: o controlador liga/desliga. Para se avaliar o desempenho de cada método, utilizam-se dois parâmetros: o consumo de energia e o erro da resposta do sistema em relação à referência.

Para avaliar os algoritmos de controle, foi utilizado um ambiente de salas em tamanho reduzido (maquete). O objetivo é selecionar a técnica de controle com melhor desempenho para que possa, futuramente, ser empregada em aparelhos de ar condicionado comerciais.

Os resultados obtidos mostraram que há um compromisso entre um menor consumo de energia e um menor erro em relação a um valor de referência para a temperatura. Observa-se, neste estudo, que a técnica de controle mais indicada depende do critério a ser priorizado. Utilizando o critério de consumo de energia, o controle *self tuning* de segunda ordem apresentou o melhor resultado dentre todas as técnicas de controle implementadas. Para o critério do erro, em relação ao valor de temperatura, o controle ótimo mostrou-se o método mais eficiente. No entanto, no caso de se almejar um resultado que leve em conta ambos os critérios, o controlador com estrutura variável apresentou um bom resultado e um compromisso satisfatório entre consumo e erro em relação à temperatura de referência.

ABSTRACT

The use of air-conditioning devices is one of the greatest responsible for the energy consumption in commercial buildings. Different control algorithms have been considered to control the temperature of an environment in order to minimize the energy consumption, while still maintains the desired comfort level. However, many projects are made in a qualitative way, without a mathematical component of explanation. In this work there were evaluated analytical techniques for temperature control in a reduced model. Such techniques involve system identification and control based on the identified model of the system.

The evaluated control techniques are the polynomial control, the optimum control, besides another one well known in building automation: the on/off controller. To evaluate the performance of each method, two parameters are used. One is the energy consumption and the other one is the steady-state error.

To validate the control algorithms, a room environment in reduced size was used (mockup). The objective is to select the technique with better performance so that it can, in future, be used in commercial air conditional devices.

The results had shown that there is a trade off between energy consumption and steady-state error. It is observed, in this study, that the indicated control technique depends on the criterion to be prioritized. Using the consumption energy criterion, the second order self tuning control presented the best result among all the implemented techniques. For the steady-state error criterion, the optimum control revealed the most efficient method. However, in case to get a result that has taken in account both criteria, the variable structure controller presented a good result and a satisfactory commitment between energy consumption and steady-state error.

SUMÁRIO

1 INTRODUÇÃO....1
1.1 O MODELO REDUZIDO.... 2
1.2 CONTROLADORES DE TEMPERATURA.... 3
2 FUNDAMENTOS TEÓRICOS....5
2.1 CONTROLADOR LIGA/DESLIGA.... 5
2.2 CONTROLADOR COM ESTRUTURA VARIÁVEL.... 5
2.3 O FILTRO DE KALMAN.... 6
2.4 ESTIMAÇÃO DOS PARÂMETROS COM FILTRO DE KALMAN.... 7
2.5 CONTROLADOR POLINOMIAL.... 8
2.6 CONTROLE ÓTIMO.... 9
3 METODOLOGIAS E RESULTADOS....13
3.1 CONTROLADOR LIGA/DESLIGA.... 13
3.2 CONTROLADOR COM ESTRUTURA VARIÁVEL.... 14
3.3 CONTROLE *SELF TUNING* PARA MODELO DE PRIMEIRA ORDEM.... 15
3.4 CONTROLADOR *SELF TUNING* COM CANAL INTEGRAL.... 22
3.5 CONTROLADOR *SELF TUNING* PARA MODELO DE SEGUNDA ORDEM COM CANAL INTEGRAL.... 23
3.6 CONTROLADOR ÓTIMO E IDENTIFICAÇÃO *OFFLINE*.... 24
3.7 COMPARAÇÃO DE DESEMPENHO DOS CONTROLADORES.... 28
4 CONCLUSÕES....33
REFERÊNCIAS BIBLIOGRÁFICAS....35
APÊNDICE I – CÓDIGO DO MATLAB....37
I.1 – CONTROLE LIGA/DESLIGA.... 37
I.2 – CONTROLE *SELF TUNING* DE PRIMEIRA ORDEM.... 38
I.3 – CONTROLE *SELF TUNING* DE SEGUNDA ORDEM.... 40
I.4 – CONTROLE ÓTIMO.... 42
APÊNDICE II – CÓDIGO DO MICROCONTROLADOR....44

LISTA DE TABELAS

Tabela 3.1 – Parâmetros de desempenho das técnicas de controle................................31
Tabela 3.2 – Classificação pelo erro em relação à referência..31
Tabela 3.3 – Classificação de acordo com o consumo médio. 31
Tabela 3.4 – Métodos de controle indicados para cada critério. 32

LISTA DE FIGURAS

Figura 1.1 – Modelo reduzido a ser utilizado para o estudo do processo térmico. 2
Figura 1.2 – Planta baixa do modelo reduzido. 3
Figura 1.3 – Circuito montado para o acionamento do motor do aquecedor. 3

Figura 2.1 – Controlador com estrutura variável. 6

Figura 3.1 – Fluxo de informações no modelo reduzido...13
Figura 3.2 – Controlador “liga/desliga”: resposta do sistema. 14
Figura 3.3 – Controlador com estrutura variável: resposta do sistema. 15
Figura 3.4 – Controlador polinomial de primeira ordem: simulação a parâmetros constantes. 16
Figura 3.5 – Controlador polinomial de primeira ordem: simulação com variação linear dos parâmetros...17
Figura 3.6 – Controlador polinomial de primeira ordem: simulação com variação quadrática dos parâmetros. 18
Figura 3.7 – Fluxograma do sistema do controle *self tuning*. 18
Figura 3.8 – Controlador polinomial de primeira ordem: resposta do sistema para referência em 45ºC. 19
Figura 3.9 – Controlador polinomial de primeira ordem: resultado da estimação de parâmetros para referência em 45ºC. 19
Figura 3.10 – Controlador polinomial de primeira ordem: resposta do sistema para referência em 35ºC. 19
Figura 3.11 – Controlador polinomial de primeira ordem: resultado da estimação de parâmetros para referência em 35ºC. 20
Figura 3.12 – Controlador polinomial de primeira ordem: resposta do sistema para referência em 55ºC. 20
Figura 3.13 – Controlador polinomial de primeira ordem: resultado da estimação de parâmetros para referência em 55ºC. 20
Figura 3.14 – Controlador polinomial de primeira ordem: resposta à onda quadrada de 43ºC a 47ºC. 21
Figura 3.15 – Controlador polinomial de primeira ordem: resposta à onda quadrada de 33ºC a 37ºC. 21
Figura 3.16 – Controlador polinomial de primeira ordem: resposta à onda quadrada de 52ºC a 56ºC. 22
Figura 3.17 – Controlador polinomial de primeira ordem com canal integral: resposta à onda quadrada...22
Figura 3.18 – Controlador polinomial de segunda ordem com canal integral: resposta à onda quadrada...23
Figura 3.19 – Controlador polinomial de segunda ordem com canal integral: resposta utilizando pólos mais lentos..24
Figura 3.20 – Controlador ótimo: resultado da simulação. 25
Figura 3.21 – Fluxograma do controle ótimo. 25
Figura 3.22 – Resposta a um sinal pseudo-aleatório. 26
Figura 3.23 – Controle ótimo: resposta do sistema. 27
Figura 3.24 – Controle ótimo: resposta do sistema com alteração dos parâmetros. 27
Figura 3.25 – Controle polinomial de primeira ordem com canal integral: resposta do sistema a um longo intervalo de tempo. 28
Figura 3.26 – Controle polinomial de segunda ordem com canal integral: resposta do sistema a um longo intervalo de tempo. 29
Figura 3.27 – Controle liga/desliga: resposta do sistema a um longo intervalo de tempo.29

Figura 3.28 – Controle com estrutura variável: resposta do sistema a um longo intervalo de tempo......30
Figura 3.29 – Controle ótimo: resposta do sistema a um longo intervalo de tempo..........30

Figura 4.1 – Topologia possível para a aplicação em aparelhos de ar condicionado.......34

1 INTRODUÇÃO

Uma das maiores preocupações da sociedade moderna é o consumo de energia. A geração de energia elétrica, nos moldes em que é realizada hoje, é um recurso escasso. A demanda por energia elétrica é crescente e cíclica. O crescimento populacional aumenta a demanda por energia elétrica. Por sua vez, um aumento na oferta de energia elétrica implica em maiores impactos ambientais que, por sua vez, tendem a aumentar a temperatura global. Este aumento na temperatura acarreta em um aumento na demanda por ar-condicionado. O ar-condicionado representa cerca de 20% do consumo de energia elétrica das edificações do setor público e comercial de grande porte (Signor, 1999). Desta forma, o aumento nas vendas de aparelhos de ar-condicionado aumenta a demanda por energia elétrica, que por sua vez causa poluição, aumentando a temperatura global e fazendo subir novamente a demanda por aparelhos de ar-condicionado e assim sucessivamente.

O que se espera de um ambiente é conforto. Conforto é um conceito mais abrangente, incluindo a adequação da temperatura ambiente, iluminação, umidade e qualidade do ar. O conforto também é um fator individual, ou seja, depende do metabolismo de cada pessoa e das roupas que ela está usando (Fanger 1974). As roupas que as pessoas vestem, por sua vez, estão intimamente relacionadas com a temperatura externa. Por outro lado, o metabolismo é uma função natural e não apresenta muita correlação com variáveis do ambiente. Estas e outras questões são discutidas com mais detalhes por Kwok e Wai (2003).

Neste contexto, diversas providências podem ser tomadas com o objetivo de se reduzir o consumo de energia sem reduzir o conforto. Tais medidas vão desde o projeto arquitetônico, com o objetivo de se utilizar melhor a iluminação natural, a escolha de cores, materiais e espessuras das edificações (funcionando como membranas por onde ocorrem as trocas de calor), até o projeto de aparelhos mais eficientes no que tange ao consumo de energia elétrica.

Tipicamente em ambientes condicionados, têm-se variações das temperaturas externa e interna ao longo do dia. A temperatura externa varia de acordo com a posição do sol e com as condições do tempo na região. Fatores como o número de pessoas, computadores ligados, bem como portas e janelas abertas, são fatores que contribuem para a variação da temperatura interna do recinto ao longo do dia. Destarte, ambientes condicionados podem ser considerados como sistemas não-lineares e variantes no tempo. Portanto, a minimização do consumo de energia, sem que haja perda no conforto, é considerada como um problema de otimização.

Por meio de técnicas de controle, almeja-se emular um ambiente condicionado capaz de manter uma referência de temperatura, seja robusto às imperfeições do modelo e rejeite perturbações externas. Deseja-se, então, projetar um controlador que seja capaz de se “adaptar” a cada variação da planta, mantendo a temperatura desejada e fazendo isso com o menor consumo possível. Todavia, neste estudo, não serão implementadas técnicas de controle que já foram analisadas para o problema de controle de temperatura em ambiente reduzido, tais como o controle PID e controlador baseado em lógica *Fuzzy* (Santos, 2005).

O primeiro capítulo deste trabalho faz referência ao modelo reduzido utilizado para a verificação do funcionamento dos algoritmos propostos. Ainda neste mesmo capítulo, tem-se uma pesquisa sobre controladores de temperatura e, no capítulo 2, a fundamentação teórica dos algoritmos. Enquanto o capítulo 3 explica a metodologia utilizada, bem como os resultados obtidos tanto em simulações quanto no modelo reduzido, o capítulo 4 apresenta as conclusões feitas a partir dos resultados, assim como propostas futuras para a continuidade do trabalho.

1.1 O MODELO REDUZIDO

Com o objetivo de se estudar o processo térmico, foi construído em laboratório um modelo reduzido de um ambiente a ser condicionado (Filho & Pereira, 2003). O modelo é composto por cinco salas conectadas entre si por meio de portas e/ou janelas, além de dois túneis externos às cinco salas, os quais podem ser utilizados para a representação das diversas condições externas, como variações na temperatura externa e diferentes temperaturas externas no ambiente, representando a situação em que um lado do ambiente está no sol e o outro na sombra. O modelo reduzido pode ser visualizado na Figura 1.1.

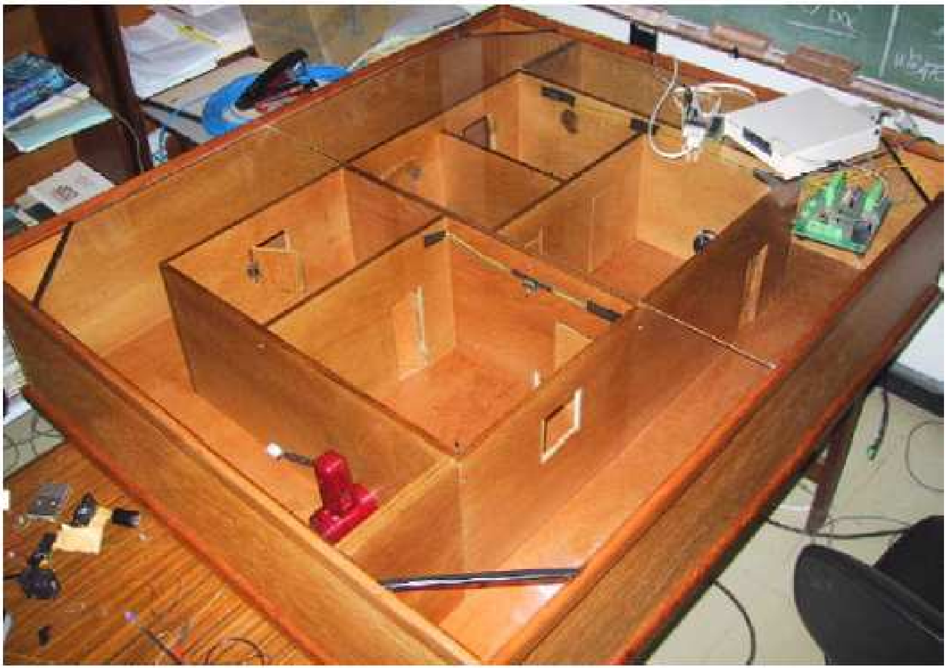


Figura 1.1 – Modelo reduzido a ser utilizado para o estudo do processo térmico. [1]

Para a medição de temperatura, utilizaram-se sensores LM35 (*National Semiconductor*). Devido à dificuldade de se conseguir representar de maneira análoga o funcionamento de um ar-condicionado, utilizaram-se aquecedores de ar que seriam o complementar dos condicionadores de ar. A planta baixa do modelo é mostrada na Figura 1.2.

Sabe-se que a temperatura da sala é diferente em diversas regiões. Regiões mais próximas ao aquecedor, por exemplo, apresentarão uma temperatura mais elevada. Desta forma, a posição dos sensores é de grande interesse para um controle mais eficiente. Riederer (2000) propõe a utilização de um sensor no centro do recinto e outro próximo às paredes, pois a diferença de temperaturas é responsável pela dinâmica de circulação de ar dentro do ambiente e, conseqüentemente, do calor transportado por ele. Todavia, neste trabalho, será utilizado apenas um sensor no recinto, a fim de simplificar o cálculo do sinal de controle, além de reduzir o custo do projeto.

Ressalta-se a modificação feita nos aquecedores, com o intuito de manter constante o fluxo de ar insuflado. Nos modelos de fábrica, uma queda na tensão de alimentação implica na diminuição da temperatura do potenciômetro e da velocidade angular do motor, o que faz variar o fluxo de ar insuflado para dentro do recinto. Assim, separaram-se os circuitos de aquecimento e do motor. A partir desta modificação, a velocidade de rotação do motor mantém-se constante independentemente da alimentação de entrada, a qual será utilizada apenas para variar a temperatura da resistência. Com este objetivo, foi montado o circuito mostrado na Figura 1.3 para o acionamento do motor.

Para uma descrição mais detalhada do funcionamento do modelo reduzido, ver Santos (2005).

[1] Retirado de Santos, 2005.

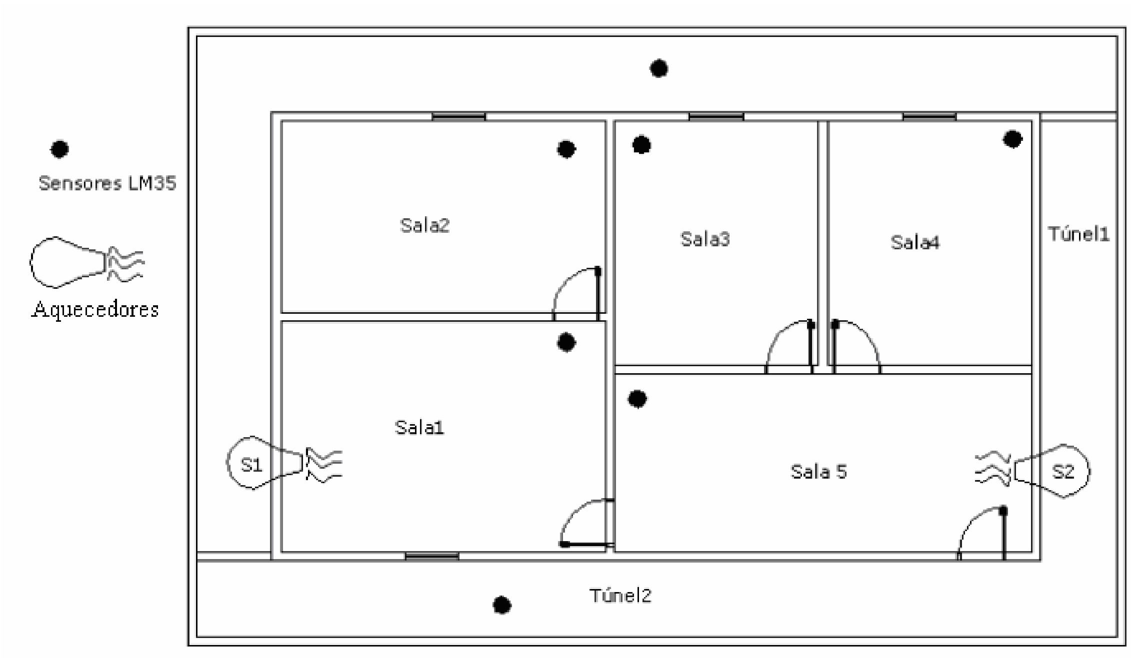


Figura 1.2 – Planta baixa do modelo reduzido. [2]

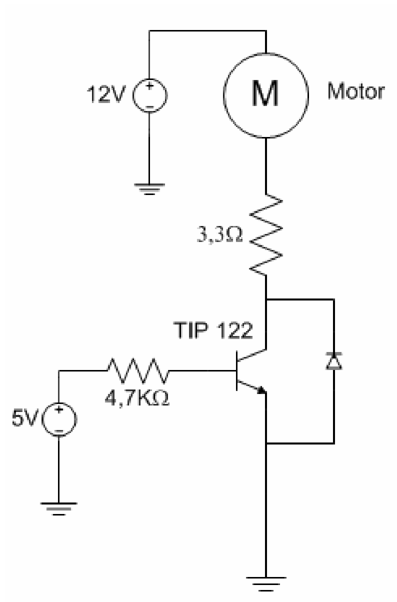


Figura 1.3 – Circuito montado para o acionamento do motor do aquecedor.

1.2 CONTROLADORES DE TEMPERATURA

O processo térmico dentro de um recinto está constantemente mudando sua dinâmica devido aos fatos mencionados na introdução deste trabalho. Portanto, é intuitiva a utilização de um controlador capaz de lidar com tais mudanças. De fato, diversos tipos de controladores adaptativos são encontrados na literatura para este fim. Têm-se projetos utilizando controladores PID até projetos utilizando Redes Neurais, bem como projetos mais elaborados utilizando modelos híbridos de tais técnicas, tal como o controlador PID ajustado por meio de redes neurais proposto por Zaheeruddin e Tudoroiu (2004).

[2] Adaptado de Santos, 2005.

Dentre os controladores propostos na literatura, vários propõem a utilização de Lógica Fuzzy. Alguns controladores *Fuzzy* apresentam o controle de temperatura com economia de energia (Bauchspiess, 2003), além de outros que também controlam a qualidade do ar (Kolokotsa, 2001). Da mesma forma, Zupančič e Škrjanc (1998) utilizam um controlador *Fuzzy* para o controle de um modelo reduzido, levando-se em consideração diversos fatores, como temperatura externa, temperatura do terreno, radiação solar, ventilação e o uso de cortinas.

Neste trabalho, serão propostos e avaliados outros controladores de temperatura mais analíticos e com menor visibilidade na comunidade acadêmica, tais como o *self tuning* com controle polinomial e o controlador ótimo.

2 FUNDAMENTOS TEÓRICOS

Deseja-se realizar o controle da temperatura de um ambiente reduzido. Este sistema, devido a características intrínsecas e extrínsecas, possui parâmetros que variam ao longo do tempo. O problema consiste em fazer a identificação e/ou a estimação, de tais parâmetros. A partir dos novos parâmetros do sistema, o controlador é sintonizado novamente de forma a se adequar ao estado atual do sistema. Os próximos tópicos têm por objetivo dar os fundamentos necessários para que se possa acompanhar e derivar as diversas técnicas de controle utilizadas neste trabalho para o controle da temperatura em um ambiente reduzido.

2.1 CONTROLADOR LIGA/DESLIGA

É considerado um dos mais simples controladores que possam ser implementados. Justamente por esta razão, é largamente utilizado pela indústria para o controle de temperatura, pois este processo é considerado razoavelmente lento, permitindo, assim, este tipo de abordagem.

Tal tipo de controlador pode ter diversos níveis, sendo o mínimo de dois. Desta forma, o controlador compara a saída do sistema com o valor de referência. Se o valor estiver acima da referência, coloca-se o sinal de controle em seu valor mínimo. Estando abaixo, atribui-se o valor máximo ao sinal de controle, conforme mostrado na equação (2.1). Percebe-se que este é um sistema que oscila permanentemente, não alcançando em momento algum um valor em regime.

$$u(k)=\begin{cases} u_{max}, & se \quad y \le y_{ref} \\ u_{min}, & se \quad y > y_{ref} \end{cases} \tag{2.1}$$

2.2 CONTROLADOR COM ESTRUTURA VARIÁVEL

A fim de se minimizar tais oscilações, acrescenta-se mais uma faixa de operação ao controlador. Deste modo, tem-se um controlador com estrutura variável concebido a partir do controlador liga/desliga. Assim sendo, têm-se três faixas de operação: uma com um o valor máximo, outra com o valor mínimo e uma terceira com um valor intermediário, como mostra a equação (2.2).

$$u(k)=\begin{cases} u_{max}\ , & se \quad y \le y_{ref}-\Delta_{y1} \\ f(y), & se \quad y_{ref}-\Delta_{y1} < y < y_{ref}+\Delta_{y2} \\ u_{min}\ , & se \quad y \ge y_{ref}+\Delta_{y2} \end{cases} \tag{2.2}$$

De fato, percebe-se que este sinal de controle é uma comutação entre o controlador liga/desliga e um outro tipo de controlador. Esta idéia pode ser mais bem compreendida observando-se a Figura 2.1.

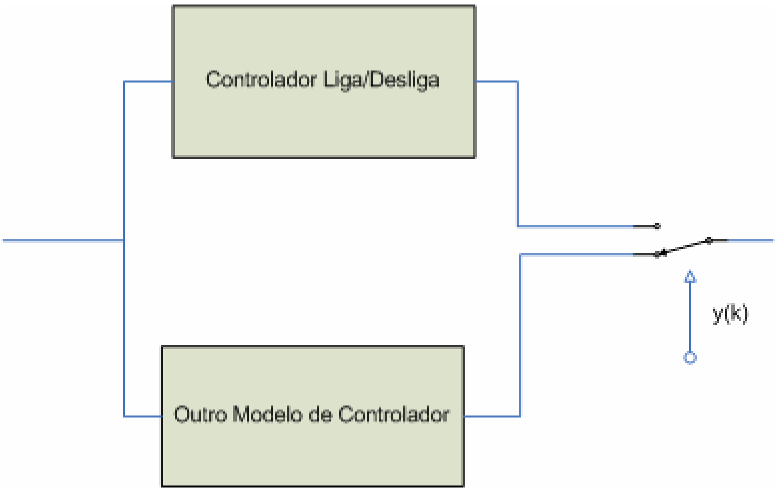


Figura 2.1 – Controlador com estrutura variável.

2.3 O FILTRO DE KALMAN

Dado um sistema discreto e observável que é regido pela seguinte equação estocástica a diferenças:

$$\mathbf{x}(k+1) = \mathbf{\Phi}(k)\mathbf{x}(k) + \mathbf{\Gamma}(k)\mathbf{u}(k) + \mathbf{w}(k+1). \tag{2.3}$$

As saídas medidas são da seguinte forma:

$$\mathbf{y}(k) = \mathbf{M}(k)\mathbf{x}(k) + \mathbf{v}(k-1). \tag{2.4}$$

As variáveis aleatórias $\mathbf{w}(k+1)$ e $\mathbf{v}(k-1)$ representam os ruídos no processo e na medida respectivamente. Assume-se que são independentes entre si, são ruídos brancos e possuem distribuição normal da forma:

$$\mathrm{p}(w) \approx \mathrm{N}(0, \mathbf{Q}(k)), \tag{2.5}$$
$$\mathrm{p}(v) \approx \mathrm{N}(0, \mathbf{R}(k)), \tag{2.6}$$

em que $\mathbf{Q}(k)$ é a covariância do ruído do processo e $\mathbf{R}(k)$ é a covariância do ruído da medida. Na prática, seus valores mudam com o tempo, mas, neste trabalho, assume-se que são valores constantes.

A partir do modelo representado pelas equações (2.3) e (2.4), com uma estimativa do vetor de estado em um instante (k-1) e os sinais de entrada, pode-se estimar o vetor de estado no instante k. Possuindo o valor da medição também no instante k, deseja-se obter uma estimativa do vetor de estado, levando-se em consideração esta informação. Tal estimativa é dita ótima no sentido que minimiza a seguinte função de custo:

$$E\left\{\left(\hat{\mathbf{x}}(k) - \mathbf{x}(k)\right)^T \left(\hat{\mathbf{x}}(k) - \mathbf{x}(k)\right) \middle| \mathbf{y}(k)\right\}, \tag{2.7}$$

em que $\hat{\mathbf{x}}(k)$ é o vetor de estado estimado. Segundo Aguirre (2000), resolvendo o Filtro de Kalman para o modelo variante no tempo representado pelas equações (2.3) e (2.4):

$$
\begin{aligned}
&\hat{\mathbf{x}}(k \mid k-1) = \mathbf{\Phi}(k)\hat{\mathbf{x}}(k-1) + \mathbf{\Gamma}(k)\mathbf{u}(k). \\
&\mathbf{P}(k \mid k-1) = \mathbf{\Phi}(k)\mathbf{P}(k)\mathbf{\Phi}^T(k) + \mathbf{Q}(k), \\
&\mathbf{K}(k) = \mathbf{P}(k \mid k-1)\mathbf{M}^T(k)\left\{\mathbf{M}(k)\mathbf{P}(k \mid k-1)\mathbf{M}^T(k) + \mathbf{R}(k)\right\}^{-1}, \\
&\hat{\mathbf{x}}(k) = \hat{\mathbf{x}}(k \mid k-1) + \mathbf{K}(k)\left(\mathbf{y}(k) - \mathbf{M}(k)\hat{\mathbf{x}}(k \mid k-1)\right), \\
&\mathbf{P}(k) = \left(\mathbf{I} - \mathbf{K}(k)\mathbf{M}(k)\right)\mathbf{P}(k \mid k-1),
\end{aligned}
\tag{2.8}
$$

em que $\mathbf{P}(k)$ é a estimativa da covariância do erro e $\mathbf{K}(k)$ é a matriz de ganhos, também conhecida como Ganho de Kalman. $\mathbf{Q}(k)$ e $\mathbf{R}(k)$ são matrizes referentes a incertezas do processo e da medição. Também são necessárias duas condições iniciais: uma para a estimativa do vetor de estados e outra condição para a matriz inicial de covariância do erro. As duas primeiras equações são relativas à estimação inicial dos parâmetros. Contudo, percebe-se que é uma estimação em malha aberta. Desta forma, as três últimas equações são responsáveis pela correção da estimativa por meio da realimentação de saída.

2.4 ESTIMAÇÃO DOS PARÂMETROS COM FILTRO DE KALMAN

Deseja-se estimar os parâmetros de um processo linear de primeira ordem sem atrasos, conforme mostrado na equação (2.9). Este modelo é proposto inicialmente por se tratar de um modelo mais simples para o controle adaptativo. De fato, modelos incluindo atrasos e/ou modelos de segunda ordem são mais aceitos para processos térmicos. Ainda neste trabalho, será considerado um modelo de segunda ordem sem atrasos.

$$
\frac{Y(s)}{U(s)} = \frac{K_p}{sT+1}. \tag{2.9}
$$

Ambiciona-se, então, estimar os parâmetros T e Kp. Tais parâmetros estão constantemente mudando de valor, seja por características próprias do sistema ou por perturbações externas ao sistema. Em um sistema térmico, por exemplo, o sistema tem determinado comportamento durante o aquecimento e outro durante o resfriamento. Perturbações externas típicas são o aumento de carga térmica na sala, a decisão de abrir ou fechar portas e/ou janelas do ambiente e a variação da temperatura externa durante o dia. Contudo, devido ao desconhecimento preciso do modelo do processo, também serão estimadas as derivadas dos parâmetros.

A técnica adotada para a estimação dos parâmetros e suas derivadas foi o rastreamento de Kalman. Assim, tem-se que os parâmetros do sistema podem ser estimados de maneira recursiva. Para a aplicação de identificação do sistema, considera-se um modelo autônomo e linear de evolução dos parâmetros da seguinte forma:

$$
T(k) = T(k-1) + w_T(k), \tag{2.10}
$$

$$
K_p(k) = K_p(k-1) + w_{K_p}(k). \tag{2.11}
$$

Escrevendo este modelo na forma espaço de estados:

$$
\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{w}(k+1). \tag{2.12}
$$

$$
\mathbf{y}(k) = \mathbf{C}(k)\mathbf{x}(k) + \mathbf{v}(k-1). \tag{2.13}
$$

As variáveis aleatórias representam as incertezas das estimações. Adaptando-se o conjunto de equações (2.8) para um modelo autônomo para a evolução dos parâmetros, isto é, sem entradas, representado por (2.10) e (2.11), pode-se fazer o rastreamento de Kalman, o qual será utilizado para a estimação dos parâmetros do sistema. Matematicamente, tem-se o seguinte conjunto de equações:

$$\begin{aligned}
&\hat{\mathbf{x}}(k \mid k-1) = \mathbf{A}\,\hat{\mathbf{x}}(k-1),\\
&\mathbf{P}(k \mid k-1) = \mathbf{A}\mathbf{P}(k-1)\mathbf{A}^T + \mathbf{Q}(k),\\
&\mathbf{K}(k) = \mathbf{P}(k \mid k-1)\mathbf{C}^T(k)\left\{\mathbf{C}(k)\mathbf{P}(k \mid k-1)\mathbf{C}^T(k) + \mathbf{R}(k)\right\}^{-1},\\
&\hat{\mathbf{x}}(k) = \hat{\mathbf{x}}(k \mid k-1) + \mathbf{K}(k)\left(\mathbf{y}(k) - \mathbf{C}(k)\hat{\mathbf{x}}(k \mid k-1)\right),\\
&\mathbf{P}(k) = \left(\mathbf{I} - \mathbf{K}(k)\mathbf{C}(k)\right)\mathbf{P}(k \mid k-1).
\end{aligned} \tag{2.14}$$

Definindo as demais matrizes do conjunto de equações (2.14), em que T_a é a taxa de amostragem, têm-se:

$$\mathbf{C}(k) = \left[y(k-1) \;\; u(k-1) \;\; 0 \;\;\; 0\right]. \tag{2.15}$$

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & T_a & 0 \\ 0 & 1 & 0 & T_a \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \tag{2.16}$$

$$\mathbf{Q}(k) = diag(\nu_{q1}^2,\ \nu_{q2}^2,\ \nu_{q3}^2,\ \nu_{q4}^2). \tag{2.17}$$

$$\mathbf{R}(k) = \nu_R^2. \tag{2.18}$$

2.5 CONTROLADOR POLINOMIAL

Estimados os parâmetros do sistema, almeja-se projetar um controlador que faça o sistema seguir uma entrada, dado um modelo de referência para o comportamento do sistema. Adotou-se, então, o controle polinomial proposto por Ästrom e Wittenmark (1995). Dado o modelo do processo, com A e B sendo polinômios e v um sinal de perturbação, tem-se:

$$y(k) = \frac{B(q)}{A(q)}u(k) + \frac{B(q)}{A(q)}v(k). \tag{2.19}$$

Deseja-se projetar um controlador da seguinte forma:

$$R(q)u(k) = T(q)y_{ref}(k) - S(q)y(k), \tag{2.20}$$

em que R, S e T são polinômios. Desta forma, têm-se as seguintes equações de malha fechada:

$$y(k)=\frac{B(q)T(q)}{A(q)R(q)+B(q)S(q)}y_{ref}(k)+\frac{B(q)R(q)}{A(q)R(q)+B(q)S(q)}v(k), \tag{2.21}$$

$$u(k)=\frac{A(q)T(q)}{A(q)R(q)+B(q)S(q)}y_{ref}(k)+\frac{B(q)S(q)}{A(q)R(q)+B(q)S(q)}v(k). \tag{2.22}$$

A equação característica, neste caso também conhecida como equação de Diophantine, é dada a seguir, em que Ac(q) é o polinômio característico desejado:

$$A(q)R(q)+B(q)S(q)=Ac(q). \tag{2.23}$$

A equação de Diophantine, porém, só resolve os polinômios R e S, sendo necessária mais uma relação, a fim de se determinar o polinômio T. Esta relação é obtida do modelo de referência desejado para a dinâmica do sistema:

$$y_m(k)=\frac{B_m(q)}{A_m(q)}y_{ref}(k)=\frac{B(q)T(q)}{A(q)R(q)+B(q)S(q)}y_{ref}(k)=\frac{B(q)T(q)}{A_c(q)}y_{ref}(k). \tag{2.24}$$

Com o objetivo de se obter um controlador causal no tempo discreto, devem-se impor as seguintes condições:

$$\deg S(q)\le\deg R(q), \tag{2.25}$$

$$\deg T(q)\le\deg R(q). \tag{2.26}$$

Todavia, cobiça-se eliminar o erro em regime do sistema. Novamente, Äström e Wittenmark (1995) propõem a seguinte equação para o projeto do controlador, em que a dedução e o significado de cada termo podem ser encontrados em sua obra:

$$R^*\left(q^{-1}\right)\Delta^*u(k)+S^*\left(q^{-1}\right)\Delta^*y(k)+A_0(1)A_m(1)y(k)=A_0^*\left(q^{-1}\right)A_m(1)y_{ref}(k). \tag{2.27}$$

2.6 CONTROLE ÓTIMO

Uma outra abordagem analítica para o projeto do controlador é por meio do controle ótimo. Sua vantagem está na possibilidade de se atribuir diferentes pesos às variáveis envolvidas. No estudo feito aqui, é interessante considerar tanto a variável de saída do sistema (temperatura) quanto o sinal de controle que dá uma medida do esforço requerido pelo atuador, sendo esta variável adotada como uma referência ao consumo de energia do sistema.

Para um primeiro desenvolvimento teórico, deseja-se levar o sistema de um estado inicial para um estado nulo, o mesmo ocorrendo com o sinal de controle. A idéia é encontrar uma boa matriz de ganho de realimentação de estados, **K**, de acordo com a seguinte lei de controle:

$$\mathbf{u}=-\mathbf{K}\mathbf{x}. \tag{2.28}$$

Dado o comportamento de um sistema discreto regido pela equação:

$$\mathbf{x}(k+1) = \mathbf{\Phi x}(k) + \mathbf{\Gamma u}(k), \tag{2.29}$$

almeja-se escolher **u**(k) de forma a minimizar a seguinte função de custo:

$$J = \frac{1}{2}\sum_{k=0}^{N}\left[\mathbf{x}^T(k)\mathbf{Q_1}\mathbf{x}(k) + \mathbf{u}^T(k)\mathbf{Q_2}\mathbf{u}(k)\right], \tag{2.30}$$

em que as matrizes $\mathbf{Q_1}$ e $\mathbf{Q_2}$ são simétricas, definidas não-negativas e correspondem aos pesos dados a cada uma das variáveis de estado e de controle. A equação (2.30) está sujeita à restrição:

$$-\mathbf{x}(k+1) + \mathbf{\Phi x}(k) + \mathbf{\Gamma u}(k) = 0 \quad , \quad \forall k = [0, N]. \tag{2.31}$$

A função de custo dada pela equação (2.30) pode ser reescrita de uma maneira expandida, da seguinte forma:

$$J = \frac{1}{2}\sum_{k=0}^{N}\left[\mathbf{x}^T(k)\mathbf{Q_1}\mathbf{x}(k) + \mathbf{u}^T(k)\mathbf{Q_2}\mathbf{u}(k) + \boldsymbol{\lambda}^T(k+1)\left(-\mathbf{x}(k+1) + \mathbf{\Phi x}(k) + \mathbf{\Gamma u}(k)\right)\right]. \tag{2.32}$$

Há várias formas de se resolver a equação de custo dada pela equação (2.32) e sua restrição. Adotou-se o método descrito por Franklin (1997). Inicialmente, considera-se:

$$\boldsymbol{\lambda}(k) = \mathbf{S}(k)\mathbf{x}(k). \tag{2.33}$$

Substituindo a equação (2.33) na função de custo e, em seguida, derivando a equação de custo em relação ao sinal de controle, **u**(k), e igualando a zero, tem-se o seguinte:

$$\mathbf{u}(k) = -\left[\mathbf{Q_2} + \mathbf{\Gamma}^T\mathbf{S}(k+1)\mathbf{\Gamma}\right]^{-1}\mathbf{\Gamma}^T\mathbf{S}(k+1)\mathbf{\Phi x}(k) = -\mathbf{K}(k)\mathbf{x}(k). \tag{2.34}$$

Derivando a equação de custo em relação à variável *x*(k) e igualando-a a zero, obtém-se a relação a seguir:

$$\boldsymbol{\lambda}(k) = \mathbf{\Phi}^T\boldsymbol{\lambda}(k+1) + \mathbf{Q_1}\mathbf{x}(k). \tag{2.35}$$

Substituindo a equação (2.33) na (2.35), tem-se que:

$$\mathbf{S}(k)\mathbf{x}(k) = \mathbf{\Phi}^T\mathbf{S}(k+1)\mathbf{x}(k+1) + \mathbf{Q_1}\mathbf{x}(k). \tag{2.36}$$

Para **x**(k+1), utiliza-se a equação (2.29), resultando em:

$$\mathbf{S}(k)\mathbf{x}(k) = \mathbf{\Phi}^T\mathbf{S}(k+1)\left[\mathbf{\Phi x}(k) + \mathbf{\Gamma u}(k)\right] + \mathbf{Q_1}\mathbf{x}(k). \tag{2.37}$$

Em seguida, substitui-se o sinal de controle, **u**(k), pela relação dada na equação (2.34), assim:

$$\mathbf{S}(k)\mathbf{x}(k) = \mathbf{\Phi}^T\mathbf{S}(k+1)\left[\mathbf{\Phi}\mathbf{x}(k) - \mathbf{\Gamma}\left(\mathbf{Q_2} + \mathbf{\Gamma}^T\mathbf{S}(k+1)\mathbf{\Gamma}\right)^{-1}\mathbf{\Gamma}^T\mathbf{S}(k+1)\mathbf{\Phi}\mathbf{x}(k)\right] + \mathbf{Q_1}\mathbf{x}(k). \quad (2.38)$$

Reescrevendo e agrupando os termos de **x**(k), tem-se:

$$\left[\mathbf{S}(k) - \mathbf{\Phi}^T\mathbf{S}(k+1)\mathbf{\Phi} + \mathbf{\Phi}^T\mathbf{\Gamma}\left(\mathbf{Q_2} + \mathbf{\Gamma}^T\mathbf{S}(k+1)\mathbf{\Gamma}\right)^{-1}\mathbf{\Gamma}^T\mathbf{S}(k+1)\mathbf{\Phi} - \mathbf{Q_1}\right]\mathbf{x}(k) = 0. \quad (2.39)$$

Como o vetor de estados não pode ser nulo, tem-se que os termos dos colchetes, à esquerda, devem sê-lo. Desta forma, chega-se à equação de diferenças de Riccati, a qual pode ser escrita da seguinte forma:

$$\mathbf{S}(k) = \mathbf{\Phi}^T\mathbf{M}(k+1)\mathbf{\Phi} + \mathbf{Q}_1, \quad (2.40)$$

em que:

$$\mathbf{M}(k+1) = \mathbf{S}(k+1) - \mathbf{S}(k+1)\mathbf{\Gamma}\left[\mathbf{Q_2} + \mathbf{\Gamma}^T\mathbf{S}(k+1)\mathbf{\Gamma}\right]^{-1}\mathbf{\Gamma}^T\mathbf{S}(k+1). \quad (2.41)$$

Uma relação de contorno para **S**(k+1) é dada por:

$$\mathbf{S}(N) = \mathbf{Q}_1. \quad (2.42)$$

Uma vez que a condição de contorno é para o último instante de tempo, a equação de Riccati deve ser resolvida recursivamente de trás para frente. Destarte, observando a equação (2.34), a matriz de ganho de realimentação, **K(***k***)**, é dada por:

$$\mathbf{K}(k) = \left[\mathbf{Q_2} + \mathbf{\Gamma}^T\mathbf{S}(k+1)\mathbf{\Gamma}\right]^{-1}\mathbf{\Gamma}^T\mathbf{S}(k+1)\mathbf{\Phi}. \quad (2.43)$$

O ganho calculado deste modo apresenta um valor em regime bem definido. Contudo, seu valor tende a zero com o passar do tempo. Por esta razão, adota-se o valor em regime do ganho e considera-se um problema com tempo infinito. Este tipo de solução é conhecido como Controle Linear Quadrático de Regime Permanente.

Todavia, o algoritmo proposto acima apenas leva a saída do sistema para zero. No caso proposto neste trabalho, é desejável que a saída do sistema acompanhe uma referência. Para tal, faz-se mister mudar a lei de controle dada pela equação (2.28). Como a saída do sistema é levada a zero, uma mudança de variável interessante é definir uma nova lei de controle em que, quando o sistema alcançasse seu estado de regime, o valor da variável de controle também tivesse seu valor de regime, desta forma:

$$\mathbf{u}(k) = -\mathbf{K}(k)\left(\mathbf{x}(k) - \mathbf{x}_{ss}\right) + \mathbf{u}_{ss}, \quad (2.44)$$

em que $\mathbf{x}_{ss}$ e $\mathbf{u}_{ss}$ são os vetores de estado e de controle em estado estacionário, respectivamente. Com esta escolha, quando o vetor de estado alcança seu valor em regime, o vetor de controle também o fará. Estes vetores podem ser calculados a partir das equações em espaço de estado do sistema. As equações de estado para o regime permanente são assim calculadas:

$$\begin{cases} \mathbf{x}_{ss} = \Phi \mathbf{x}_{ss} + \Gamma \mathbf{u}_{ss} \\ \mathbf{y}_{ref} = \mathbf{M} \mathbf{x}_{ss} \end{cases}. \tag{2.45}$$

Isolando as variáveis de interesse em regime permanente, tem-se:

$$\begin{pmatrix} \mathbf{x}_{ss} \\ \mathbf{u}_{ss} \end{pmatrix} = \begin{pmatrix} \Phi - \mathbf{I} & \Gamma \\ \mathbf{M} & 0 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ \mathbf{y}_{ref} \end{pmatrix}, \tag{2.46}$$

de onde se obtêm:

$$\begin{cases} \mathbf{x}_{ss} = \mathbf{L} \mathbf{y}_{ref} \\ \mathbf{u}_{ss} = \mathbf{N} \mathbf{y}_{ref} \end{cases}. \tag{2.47}$$

3 METODOLOGIAS E RESULTADOS

Antes de se implementar no modelo reduzido os algoritmos propostos, os mesmos foram testados anteriormente em ambientes computacionais a fim de se verificar sua adequação ao problema, bem como depurar e ajustar o programa com mais facilidade. Verificada a adequação do algoritmo, tem-se também uma noção da grandeza dos parâmetros a serem escolhidos. Todavia, após a implementação no modelo reduzido, alguns pequenos ajustes foram feitos no algoritmo, visando melhorar ainda mais a resposta do sistema.

Com o intuito de se verificar e validar o funcionamento dos algoritmos propostos, utilizou-se um modelo reduzido, conforme mostrado na Figura 1.1. A temperatura é adquirida por meio de um sensor LM 35. O sinal passa por um pré-processamento, a fim de amplificar o sinal, e depois é convertido para formato digital por meio de um micro-controlador PIC 18F252. Em seguida, por meio de comunicação serial, o valor de temperatura é enviado para um computador. Por se tratar de cálculos complexos para a implementação em micro-controlador, a identificação e o cálculo do valor do sinal de controle são feitos no próprio computador. O computador então calcula o sinal de controle e envia para o micro-controlador via comunicação serial. O micro-controlador recebe o sinal de controle e o envia para o atuador que atua de forma a alterar a temperatura do aquecedor. Este esquema está representado na Figura 3.1 . Como plataforma de estudos, as portas e a janela da sala 01, conforme Figura 1.2 da página 3, foram fechadas com o objetivo de se evitarem maiores trocas com o ambiente.

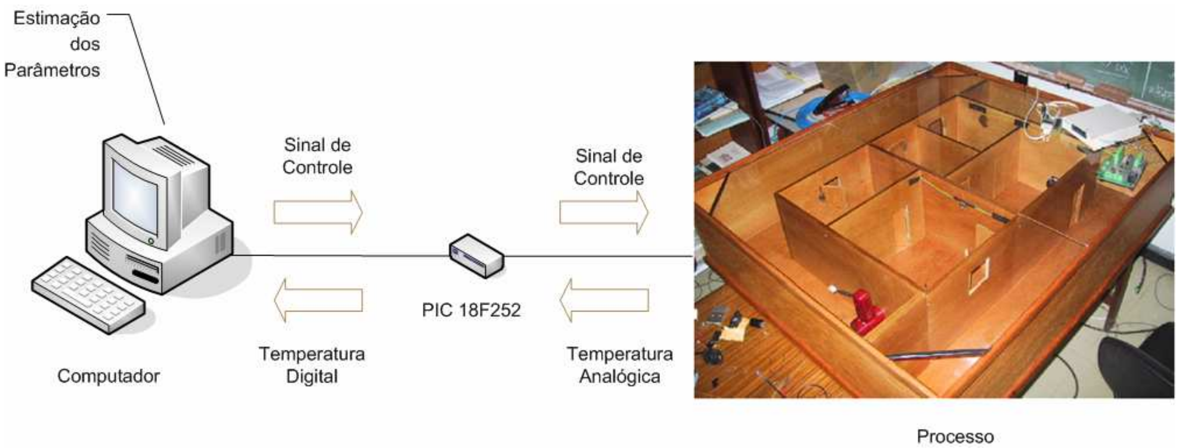


Figura 3.1 – Fluxo de informações no modelo reduzido.

3.1 CONTROLADOR LIGA/DESLIGA

Um dos controladores mais utilizados para o controle de temperatura em ambientes monitorados é o do tipo Liga/Desliga. O diferencial desta técnica é apresentar um algoritmo de controle bastante simples e eficaz. Tratando-se de um método amplamente difundido, optou-se por implementá-lo a fim de se comparar com outros procedimentos sugeridos neste trabalho. A Figura 3.2 mostra o resultado no modelo reduzido do controlador liga/desliga com três níveis de atuações possíveis. A lei de controle foi feita da seguinte forma: se a saída for 0,3ºC maior que a referência, o sinal de controle vai para zero. Se for menor que 0,8ºC, o atuador age próximo ao seu ponto limite de operação. Estando entre os dois limiares, o atuador é definido como um sinal médio entre a saturação máxima e a mínima. Matematicamente:

$$u(k)=\begin{cases}55 & se \quad y \le y_{ref}-0,8 \\ 27,5 & se \quad y_{ref}-0,8<y<y_{ref}+0,3 \\ 0 & se \quad y \ge y_{ref}+0,3\end{cases} \tag{3.1}$$

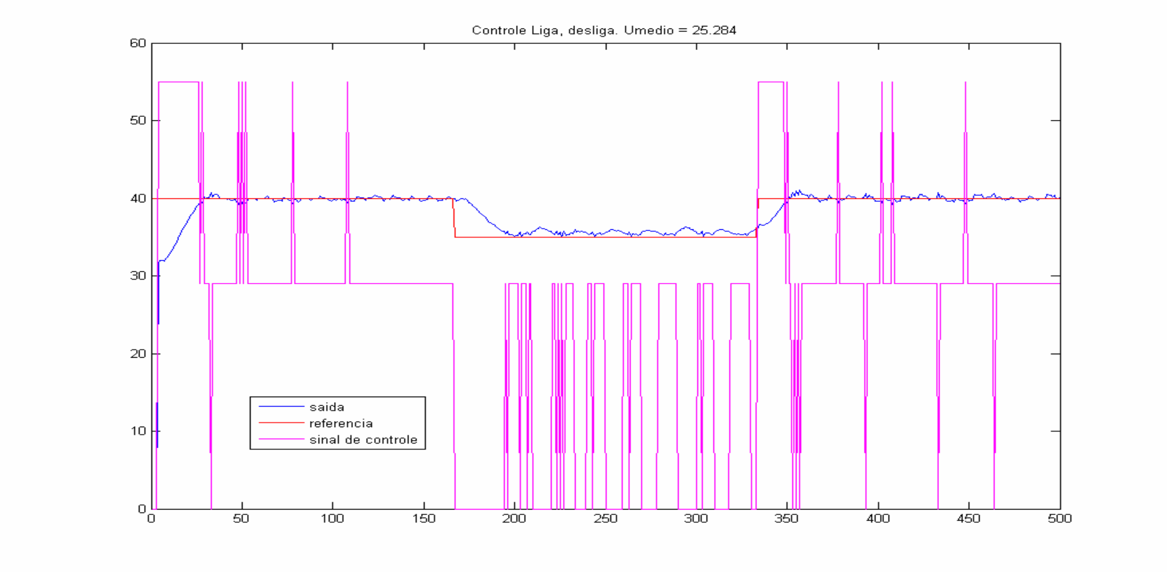


Figura 3.2 – Controlador “liga/desliga”: resposta do sistema.

A resposta do sistema apresentou uma saída satisfatória em relação ao rastreamento da referência. Contudo, percebe-se uma grande oscilação no sinal de controle, alcançando a saturação em diversos momentos, o que não é desejado, tanto em termos de consumo quanto em um maior desgaste do atuador.

3.2 CONTROLADOR COM ESTRUTURA VARIÁVEL

Com o objetivo de melhorar a resposta do sistema, foi proposta a utilização de outra lei de controle na faixa intermediária do controlador “liga/desliga”, conforme mostrado na equação (3.2), em que **x** é o vetor de estados . A Figura 3.3 ilustra o resultado obtido no ambiente reduzido.

A resposta do sistema é bem semelhante. Todavia, observa-se agora uma diminuição das oscilações em que o atuador opera na saturação, o que também resultou numa diminuição do sinal médio de controle.

$$u(k)=\begin{cases}55 & se \quad y \le y_{ref}-0,8 \\ -4\mathbf{x}+0,6\mathbf{y}_{\mathrm{Ref}} & se \quad y_{ref}-0,8<y<y_{ref}+0,3 \\ 0 & se \quad y \ge y_{ref}+0,3\end{cases} \tag{3.2}$$

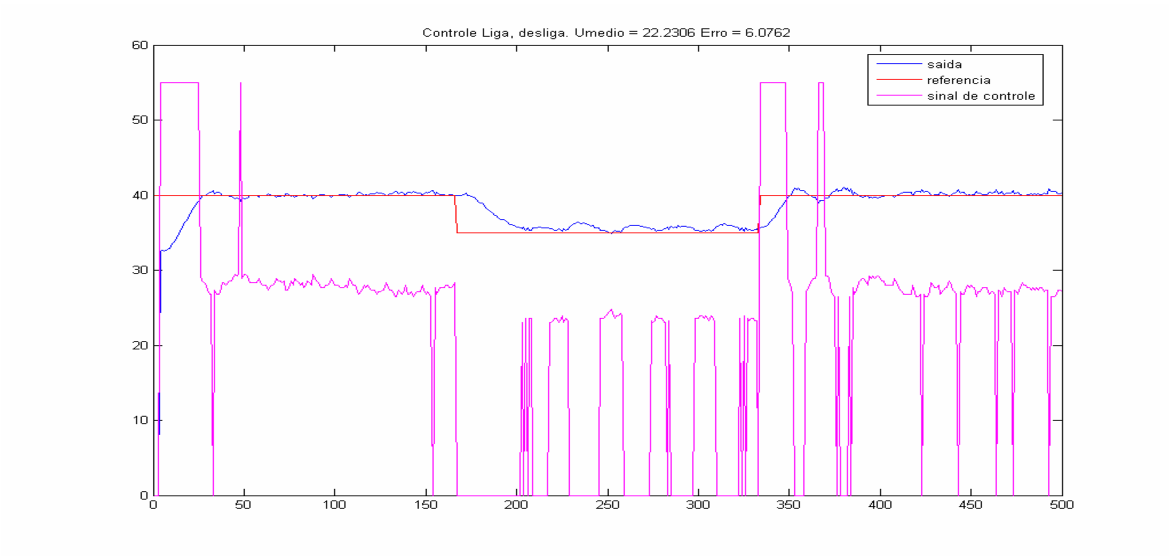


Figura 3.3 – Controlador com estrutura variável: resposta do sistema.

3.3 CONTROLE *SELF TUNING* PARA MODELO DE PRIMEIRA ORDEM

Deseja-se que o sistema responda sem sobre-passo e que não oscile em torno da referência. Uma das maneiras de se obter tal resultado, é fazer com que o polinômio característico desejado seja de primeira ordem. Desta forma:

$$Ac = \frac{b_{m0}}{z - a_{m0}}. \tag{3.3}$$

Na referência utilizada para o controle polinomial (Äström e Wittenmark, 1995), não foi dada grande ênfase como proceder à escolha dos polinômios R, S e T na equação (2.20). Contudo, observou-se certa preferência do autor por determinadas escolhas. Assim, os polinômios foram projetados de maneira empírica, observando-se as condições para que o sistema seja causal no domínio discreto do tempo, conforme (2.25) e (2.26).

$$R = b_0 z, \tag{3.4}$$
$$T = b_{m0}, \tag{3.5}$$
$$S = a_0 - a_{m0}, \tag{3.6}$$

em que a_0 e b_0 são os parâmetros estimados com o rastreamento de Kalman. Com esta escolha, o sinal de controle é dado pela equação (3.7):

$$u(k) = \frac{b_{m0}}{b_0} y_{ref}(k) - \frac{a_0 - a_{m0}}{b_0} y(k). \tag{3.7}$$

Para utilizar o canal integral, escolhem-se apropriadamente os polinômios e, utilizando a equação (2.27) na sua forma discreta, chega-se à seguinte equação para o sinal de controle:

$$u(k) = u(k-1) + \frac{(1 - a_{m0})}{b_0}\left(y_{ref}(k) - y(k)\right) + b_0\left(a_{m0} - a_0\right)\left(y(k) - y(k-1)\right). \tag{3.8}$$

No modelo de referência, nos casos com e sem canal integral, foram escolhidos os seguintes valores:

$$a_{m0} = 0,95, \\ b_{m0} = 0,048. \tag{3.9}$$

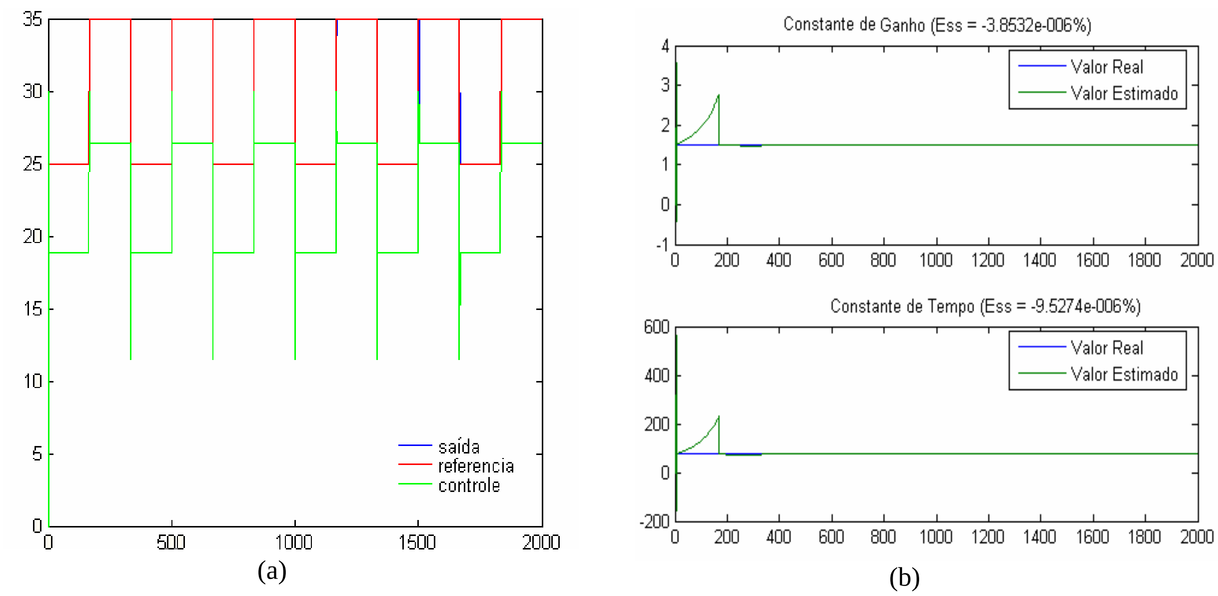
Com o intuito de verificar a eficácia do algoritmo, foram realizadas simulações para diferentes variações dos parâmetros. As simulações foram realizadas de forma crescente de dificuldade, iniciando com parâmetros constantes, seguida por uma variação linear e, finalmente, por uma variação quadrática. As simulações foram realizadas utilizando o controlador polinomial e o rastreamento de parâmetros de Kalman. As matrizes de incertezas foram definidas conforme equações (3.10) e (3.11):

$$\mathbf{Q}(k) = \text{diag}(0,01*[0,1\ 0,1\ 0,001\ 0,001]^2), \tag{3.10}$$
$$\mathbf{R}(k) = [0,0025]. \tag{3.11}$$

As condições iniciais foram escolhidas, como mostradas a seguir:

$$\mathbf{x}(0) = \begin{bmatrix} 0,1 \\ 0,1 \\ 0 \\ 0 \end{bmatrix}, \tag{3.12}$$
$$\mathbf{P}(0) = diag([1 \quad 1 \quad 0,1 \quad 0,1]). \tag{3.13}$$

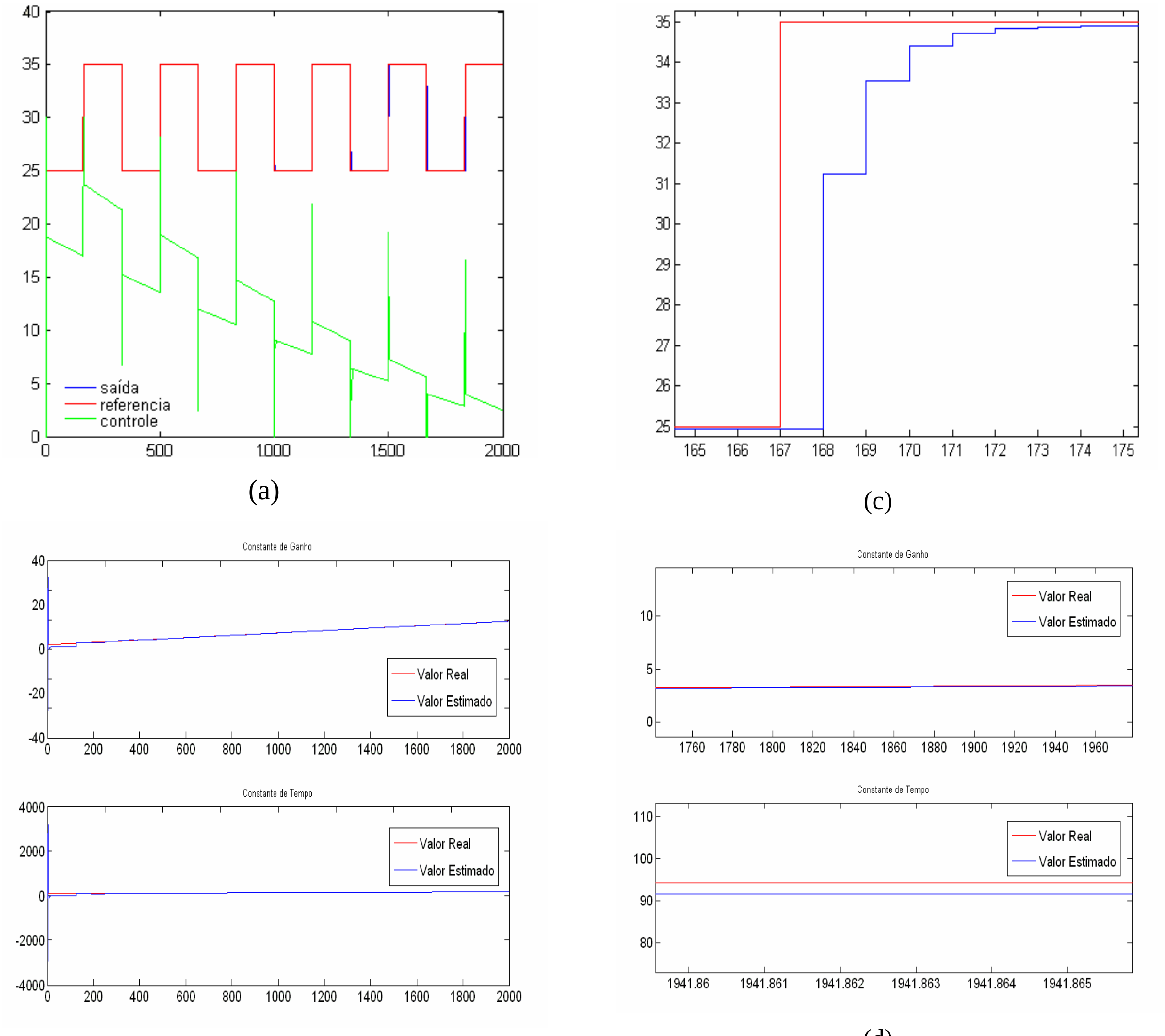
Primeiramente, realizou-se a simulação para parâmetros constantes no tempo. Mostram-se na Figura 3.4 o resultado da estimação e do controle, respectivamente. Vê-se que a saída do sistema acompanhou a referência de maneira satisfatória. A estimação dos parâmetros começou divergindo. Contudo, após um intervalo de tempo, a estimação dos parâmetros convergiu corretamente, sendo que o valor *Ess* corresponde ao erro em regime.



(a) (b)

Figura 3.4 – Controlador polinomial de primeira ordem:
simulação a parâmetros constantes.
(a) Resposta à onda quadrada; (b) Evolução dos parâmetros.

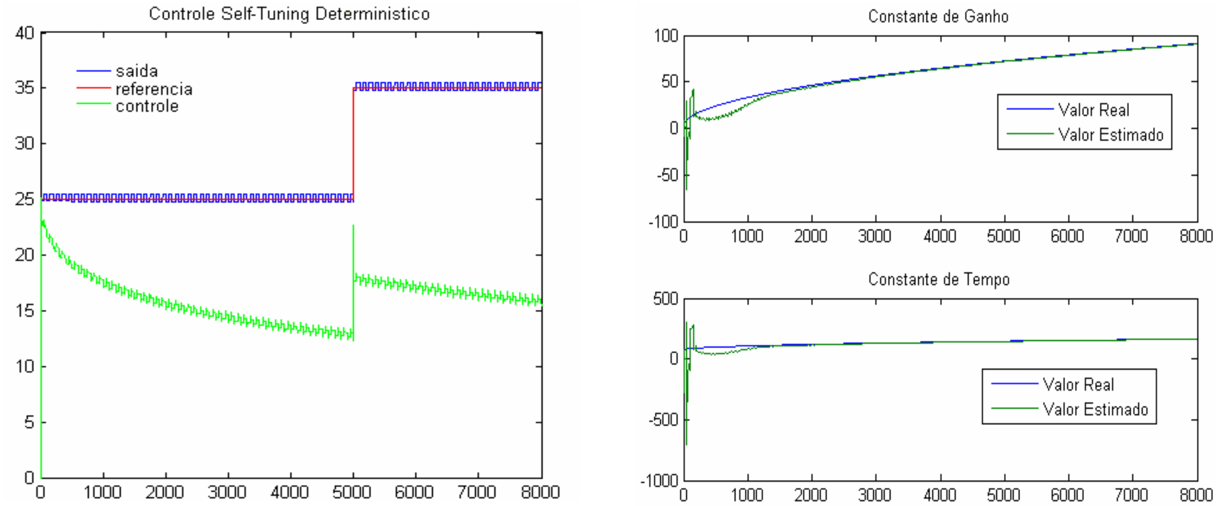
Posteriormente, fez-se com que os parâmetros variassem linearmente no tempo a fim de verificar a robustez da estimação. Os resultados são mostrados a seguir na Figura 3.5. Nota-se, novamente, que o sistema conseguiu acompanhar a referência, mesmo com a variação linear dos parâmetros. A estimação dos parâmetros, por sua vez, apresentou um salto maior no início, mas após algumas iterações conseguiu acompanhar corretamente a evolução dos parâmetros. Em seguida, mostra-se com mais detalhe a resposta do sistema. Observa-se que o sistema não apresenta picos nem oscilações e alcança um pequeno erro em regime, em torno de 3%.



(a) (c) (b) (d)

Figura 3.5 – Controlador polinomial de primeira ordem:
simulação com variação linear dos parâmetros.
(a) Resposta à onda quadrada; (b) Estimação dos parâmetros;
(c) Detalhe da resposta; (d) Detalhe da estimação.

Finalmente, realizou-se a simulação para uma variação quadrática dos parâmetros. Com esta simulação, tem-se uma idéia da robustez da estimação frente a uma variação quadrática dos parâmetros. Inicialmente, devido a esta não-linearidade, o sistema não conseguia estimar corretamente os valores dos parâmetros. Neste caso, a fim de se conseguir realizar a identificação, introduziu-se um distúrbio em forma de pulsos para que o sistema fosse devidamente excitado. Acrescentando-se este distúrbio ao sistema, foi possível realizar tanto a estimação dos parâmetros quanto o controle do sistema de maneira aceitável, conforme visualizado na Figura 3.6.



(a) (b)

Figura 3.6 – Controlador polinomial de primeira ordem:
simulação com variação quadrática dos parâmetros.
(a) Resposta ao degrau; (b) Estimação dos parâmetros.

Os resultados mostram que o controle *self tuning* com controle polinomial consegue lidar com variações suaves dos parâmetros da função transferência proposta, sendo esta uma técnica válida para o controle de temperatura.

O algoritmo de controle *self tuning* para a implementação no modelo reduzido pode ser mais bem visualizado de acordo com o fluxograma a seguir. A lei de controle adotada foi o controle polinomial.

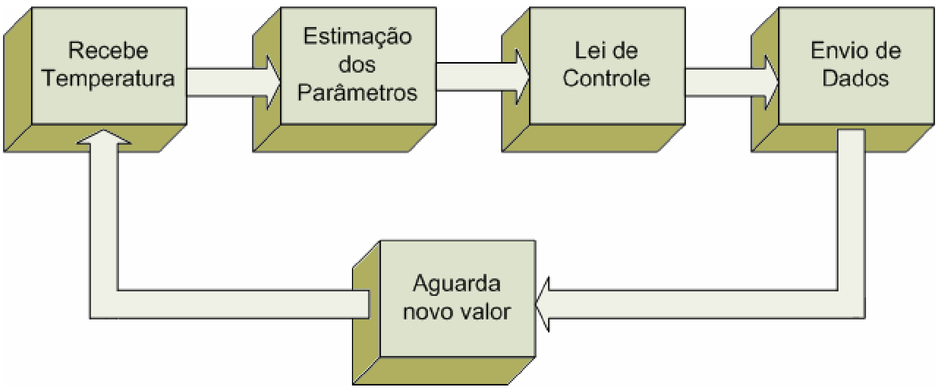


Figura 3.7 – Fluxograma do sistema do controle *self tuning*.

Como primeiro experimento, aplicou-se um degrau de referência para a temperatura de valor de 45ºC. Os resultados do controle e da estimação de parâmetros podem ser vistos na Figura 3.8 e na Figura 3.9, respectivamente. Nota-se que o sistema atingiu a referência com um pequeno erro em regime, após cerca de 350 amostras. Percebe-se também que o sistema respondeu inicialmente com um sobre-passo.

Mantendo-se as mesmas condições iniciais do experimento anterior, mudou-se o valor da referência para 35ºC e 55ºC. Os resultados são mostrados nas Figuras 3.10 a 3.13. Observando os resultados obtidos, percebe-se que o sistema responde de maneira diferente para cada valor de referência, evidenciando sua não-linearidade. Em 35ºC, o sistema apresenta oscilações com maior freqüência e menor amplitude. Em 55ºC, tem-se uma amplitude maior com menor freqüência.

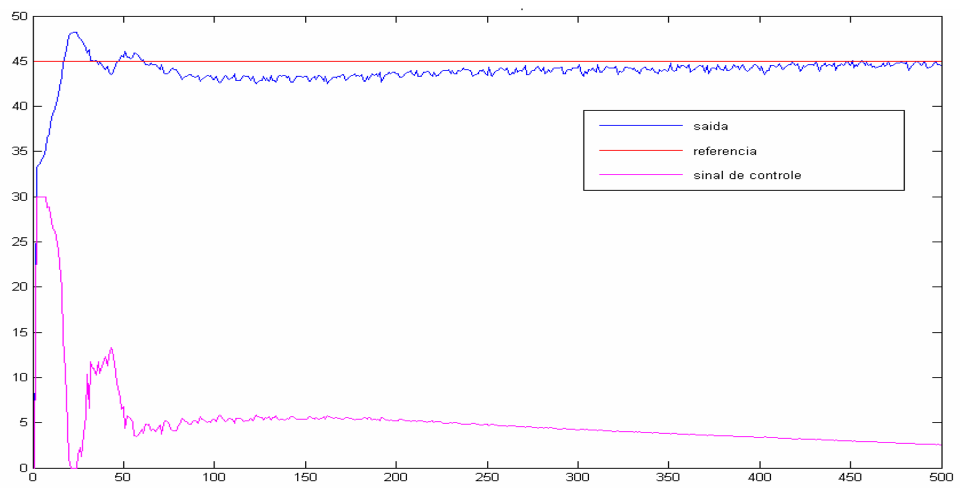


Figura 3.8 – Controlador polinomial de primeira ordem: resposta
do sistema para referência em 45ºC.

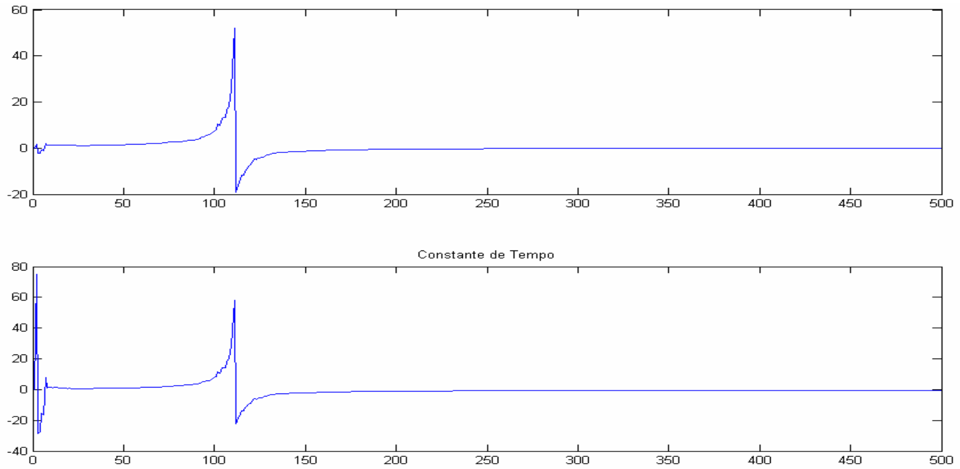


Figura 3.9 – Controlador polinomial de primeira ordem:
resultado da estimação de parâmetros para referência em 45ºC.

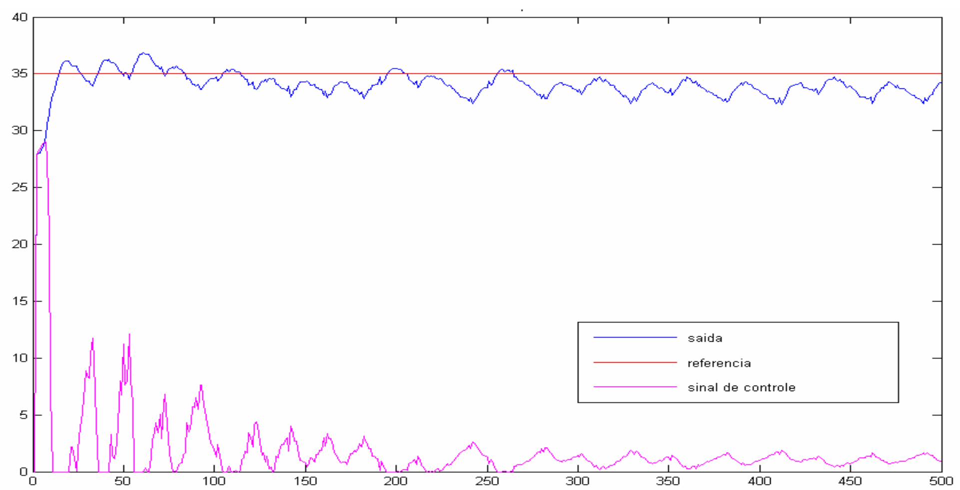


Figura 3.10 – Controlador polinomial de primeira ordem: resposta
do sistema para referência em 35ºC.

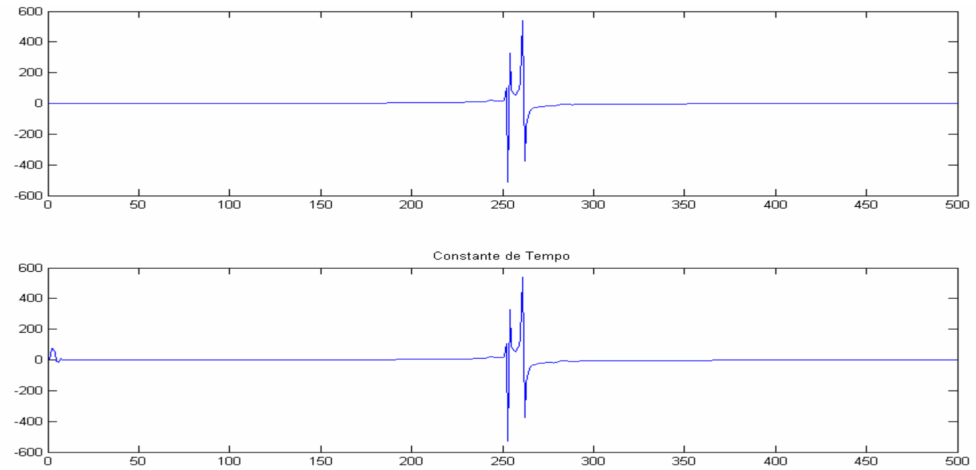


Figura 3.11 – Controlador polinomial de primeira ordem: resultado da estimação de parâmetros para referência em 35ºC.

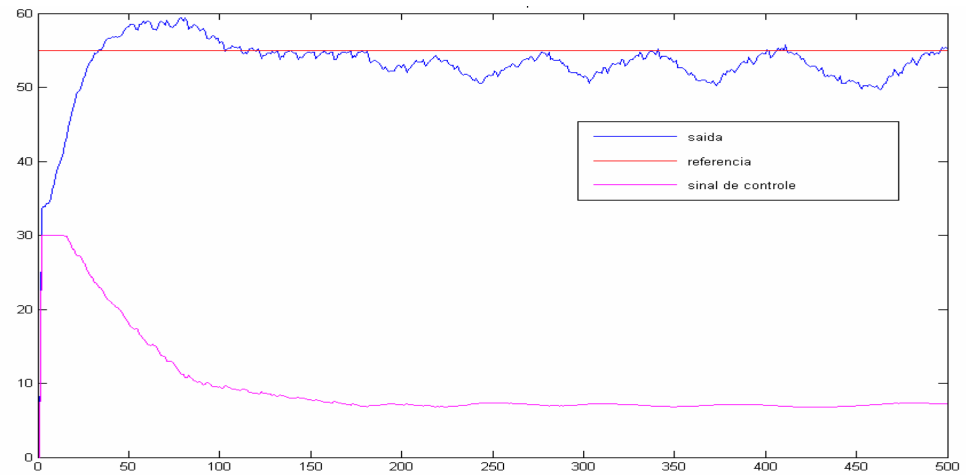


Figura 3.12 – Controlador polinomial de primeira ordem: resposta do sistema para referência em 55ºC.

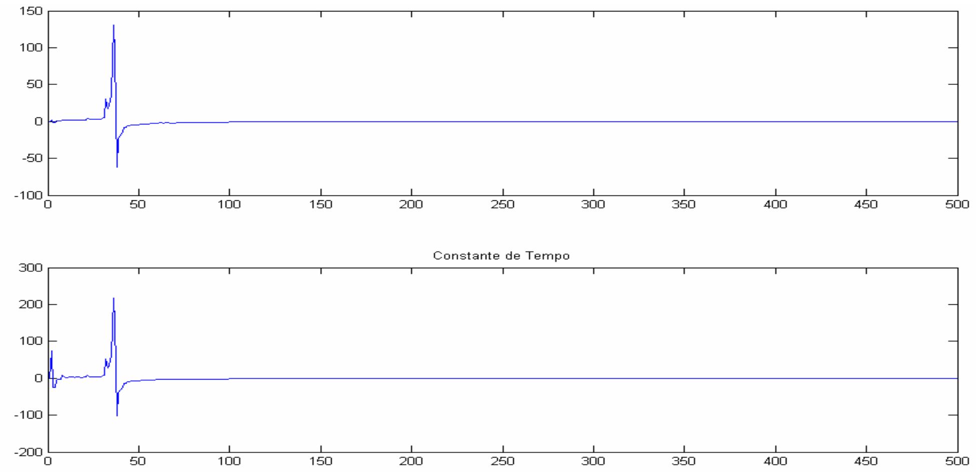


Figura 3.13 – Controlador polinomial de primeira ordem: resultado da estimação de parâmetros para referência em 55ºC.

A fim de avaliar o desempenho do algoritmo frente à alteração de referência, aplicou-se uma onda quadrada a ser seguida. Novamente, foram realizados experimentos em três pontos de operação distintos, em que se pode, outra vez, observar a não-linearidade do sistema. Ainda com relação às não-linearidades, nota-se que a resposta do sistema é diferente para a subida e para a descida O sistema também não conseguiu alcançar um valor em regime satisfatório, o que pode ser minimizado com a implementação de um canal PI ao controlador. Os resultados são mostrados nas figuras 3.14 a 3.16.

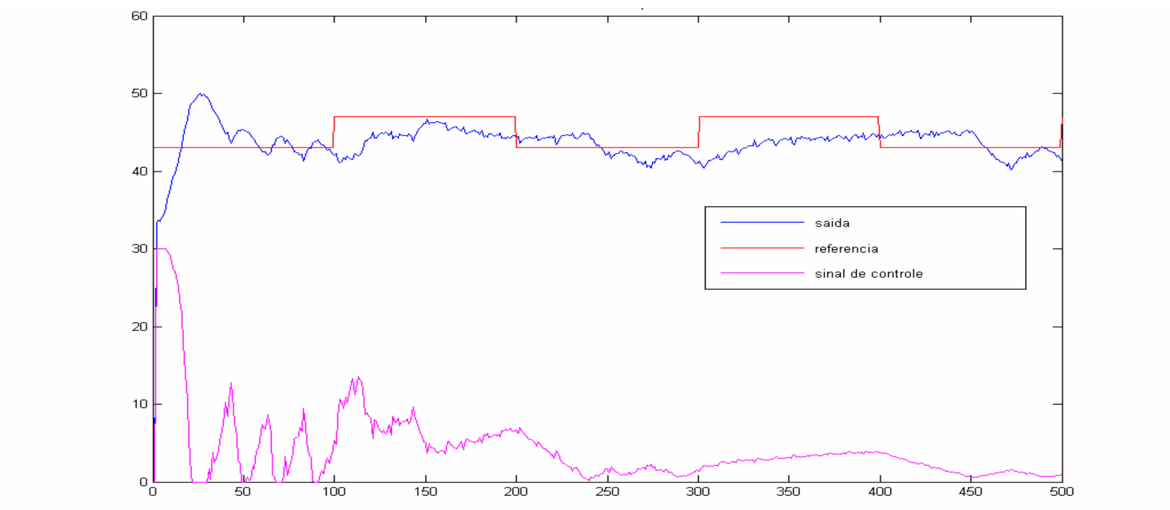


Figura 3.14 – Controlador polinomial de primeira ordem: resposta à onda quadrada de 43ºC a 47ºC.

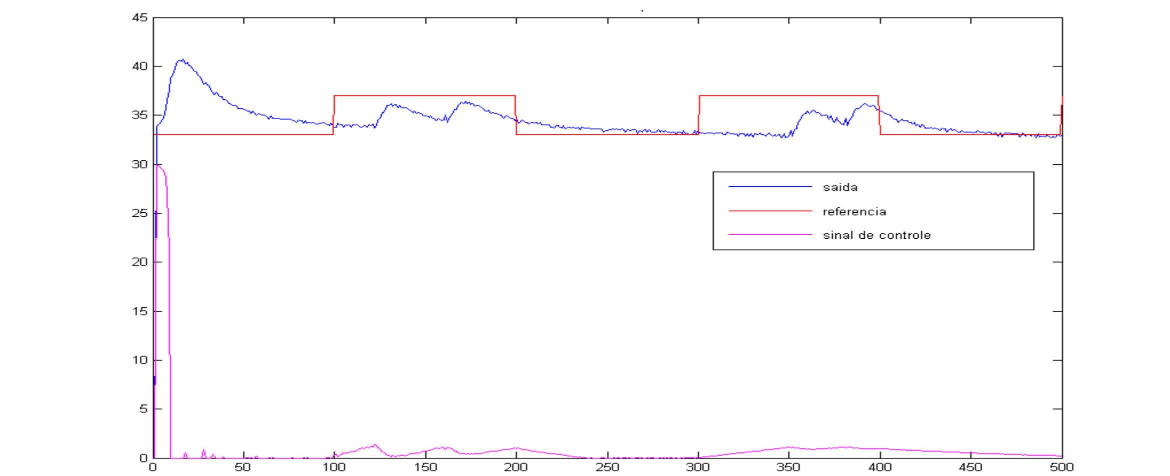


Figura 3.15 – Controlador polinomial de primeira ordem: resposta à onda quadrada de 33ºC a 37ºC.

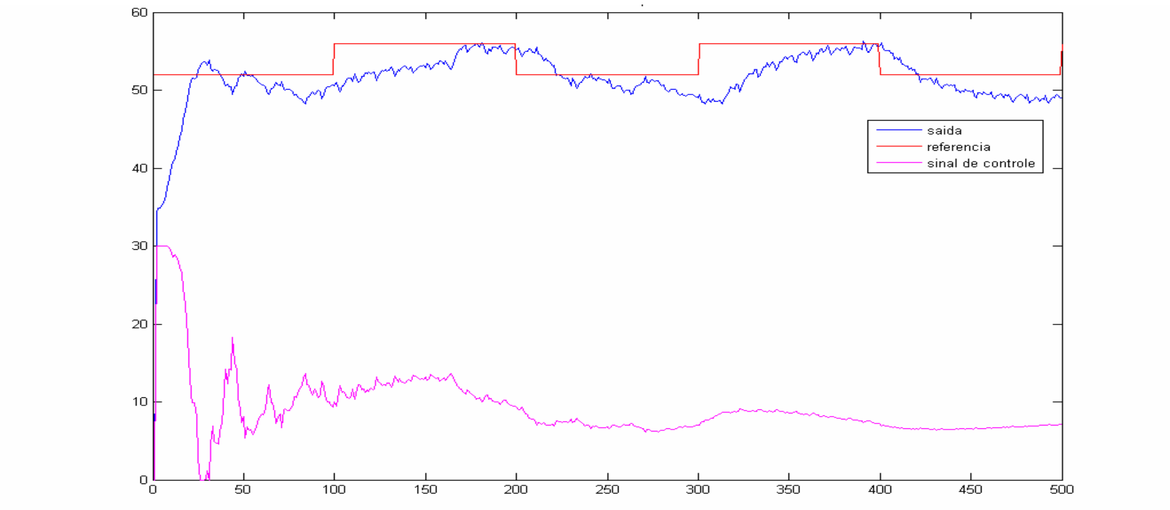


Figura 3.16 – Controlador polinomial de primeira ordem: resposta
à onda quadrada de 52ºC a 56ºC.

3.4 CONTROLADOR *SELF TUNING* COM CANAL INTEGRAL

Observando-se as figuras anteriores, percebe-se que o valor em regime não atinge um valor satisfatório. Com o objetivo de minimizar este problema, implementou-se um canal integral no modelo de 1ª ordem. O resultado é ilustrado na figura 3.17. O modelo de referência é dado pelos parâmetros mostrados em (3.9). Desenvolvendo a equação (2.27) para o canal integral, chegou-se à lei de controle dada pela equação (3.14) .

$$u(k) = u(k-1) + \frac{(1-a_{m0})}{b_0}\left(y_{ref}(k) - y(k)\right) - b_0\left(a_0 - a_{m0}\right)\left(y(k) - y(k-1)\right). \tag{3.14}$$

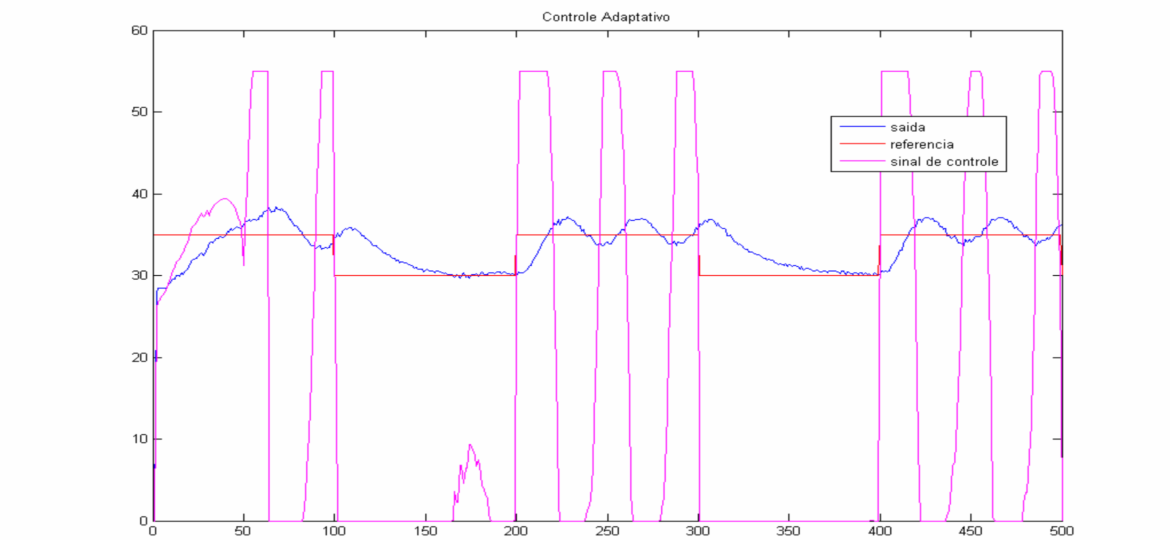


Figura 3.17 – Controlador polinomial de primeira ordem com canal integral:
resposta à onda quadrada.

Nota-se, contudo, que a resposta oscila bastante em torno da referência. A próxima tentativa é então aumentar a complexidade do modelo, conjeturando-se que o sistema pode ser modelado por meio de um sistema de 2ª ordem.

3.5 CONTROLADOR *SELF TUNING* PARA MODELO DE SEGUNDA ORDEM COM CANAL INTEGRAL

Um modelo de primeira ordem é muito simples e pode não refletir satisfatoriamente o comportamento do sistema. Desta forma, adotou-se um modelo de 2ª ordem juntamente com o canal integral. Foram necessárias algumas adaptações no código a fim de poder utilizá-lo em um sistema de segunda ordem. As adaptações são mostradas nas equações (3.15) a (3.18) para a estimação dos parâmetros e na equação (3.19) para o controlador polinomial. A resposta do sistema é mostrada na Figura 3.18. Novamente, embora com menor freqüência, percebem-se as oscilações na resposta do sistema durante a subida.

$$\mathbf{C}(k) = \left[y(k-1) \;\; y(k-2) \;\; y(k-3) \;\; u(k-1) \;\; u(k-2) \; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \right], \tag{3.15}$$

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & T_a & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & T_a & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & T_a & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & T_a & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & T_a \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.16}$$

$$\mathbf{Q}(k) = \text{diag}(0{,}01*[0{,}1 \; 0{,}1 \; 0{,}1 \; 0{,}1 \; 0{,}1 \; 0{,}001 \; 0{,}001 \; 0{,}001 \; 0{,}001 \; 0{,}001]^2), \tag{3.17}$$

$$\mathbf{R}(k) = 0{,}0025, \tag{3.18}$$

$$u(k) = u(k-1) + \frac{b_1}{b_0}(a_{m0} - a_2)\big(y(k-2) - y(k-1)\big) - (a_{m2} - a_0)\big(y(k-1) - y(k)\big) + (a_{m0} + a_{m1} + a_{m2})\big(y_{ref}(k) - y(k)\big). \tag{3.19}$$

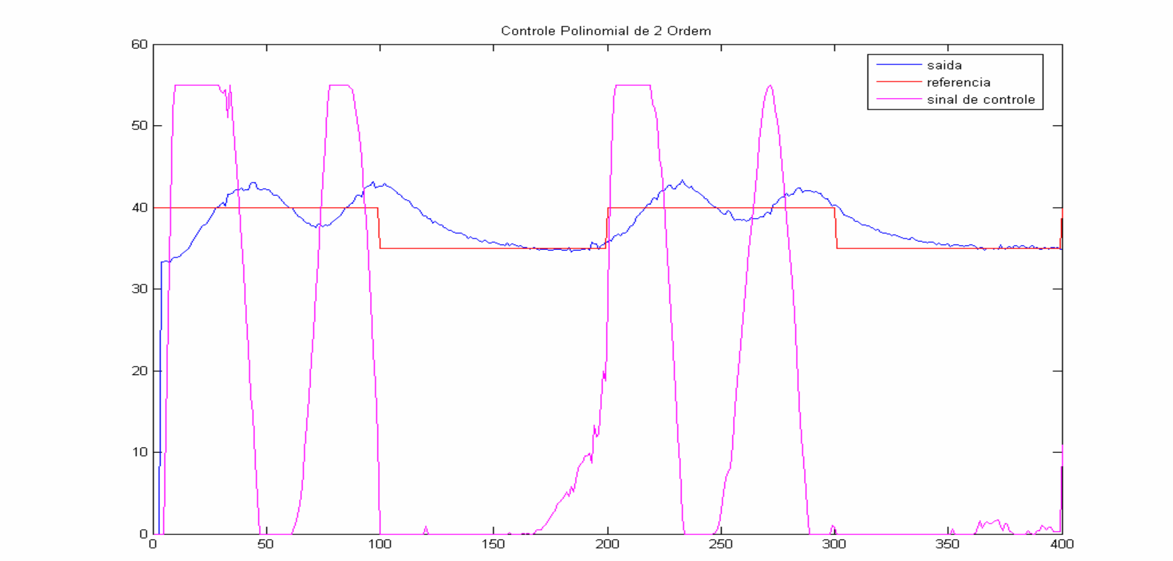


Figura 3.18 – Controlador polinomial de segunda ordem com canal integral: resposta à onda quadrada.

Optou-se, então, por uma redefinição dos parâmetros do projeto. Decidiu-se por tornar mais lentos os pólos desejados, bem como diminuir o ganho do controlador. O modelo de referência fora definido conforme conjunto de equações (3.20). O resultado pode ser visualizado na Figura 3.19. Neste caso, notou-se que as oscilações foram bastante reduzidas, e o sistema passou a apresentar um comportamento satisfatório.

$$\begin{aligned} a_{m0} &= 1, \\ a_{m1} &= -0,7603, \\ a_{m2} &= 0,1445, \\ b_{m0} &= 0,12, \\ b_{m1} &= 0,06. \end{aligned} \tag{3.20}$$

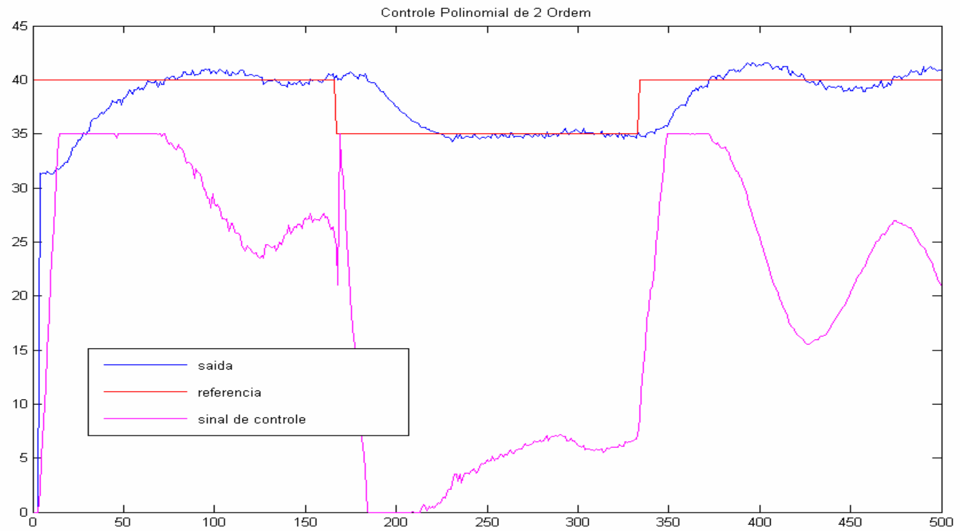


Figura 3.19 – Controlador polinomial de segunda ordem com canal integral: resposta utilizando pólos mais lentos.

3.6 CONTROLADOR ÓTIMO E IDENTIFICAÇÃO *OFFLINE*

A fim de se verificar o comportamento de um sistema, utilizando o algoritmo de controle ótimo, realizou-se uma simulação de um motor de corrente contínua, somente para validar o algoritmo, com o seguinte modelo de segunda ordem proposto por Ästrom (1990), em que Ta é o período de amostragem:

$$\mathbf{x}(k+1) = \begin{pmatrix} e^{-Ta} & 0 \\ 1-e^{-Ta} & 1 \end{pmatrix} \mathbf{x}(k) + \begin{pmatrix} 1-e^{-Ta} \\ Ta-1+e^{-Ta} \end{pmatrix} \mathbf{u}(k). \tag{3.21}$$

As matrizes de custo, $\mathbf{Q_1}$ e $\mathbf{Q}_2$, foram assim definidas para esta simulação:

$$\mathbf{Q}_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \tag{3.22}$$

$$\mathbf{Q}_2 = 0,01. \tag{3.23}$$

A matriz $\mathbf{Q_1}$ escolhida desta forma assegura, no controle ótimo, que apenas os estados que têm influência na saída serão considerados.

O resultado do controle, mostrado na Figura 3.20, ilustra a referência e a saída do modelo. A partir do resultado obtido, percebe-se que o controle ótimo pode ser uma técnica possível para o controle do sistema térmico levando-se em consideração uma variável representativa do consumo de energia.

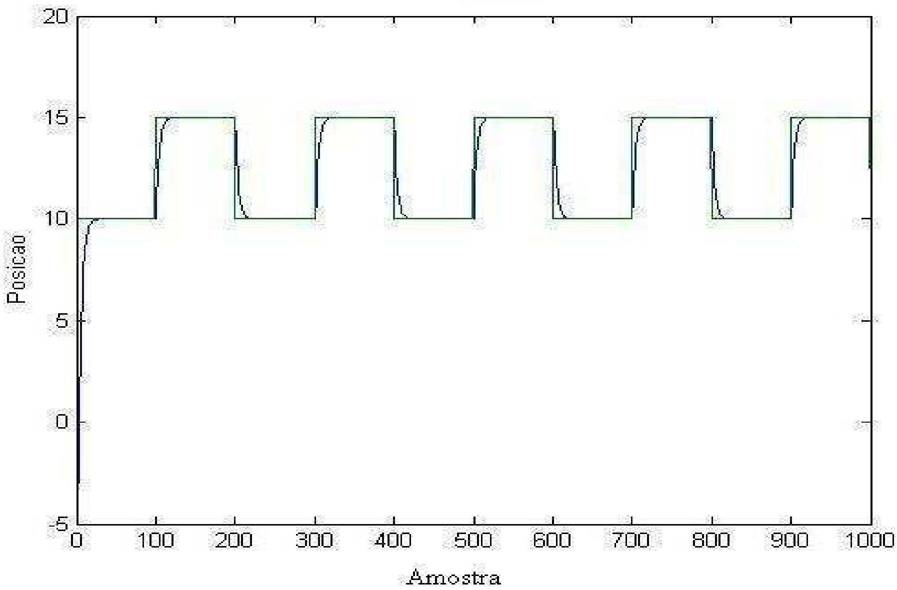


Figura 3.20 – Controlador ótimo: resultado da simulação.

Todavia, este algoritmo pressupõe o acesso a todas as variáveis de estado do sistema. No caso do modelo reduzido, só se tem acesso a uma variável de estado. Desta forma, as variáveis de estado devem ser estimadas. O algoritmo de estimação é dado pelo conjunto de equações em (2.14).

A Figura 3.21 ilustra como foi implementado o controle ótimo para a aplicação no ambiente reduzido. Percebe-se que parte do algoritmo é feita *offline*, isto é, antes que a aplicação comece a rodar no micro-controlador.

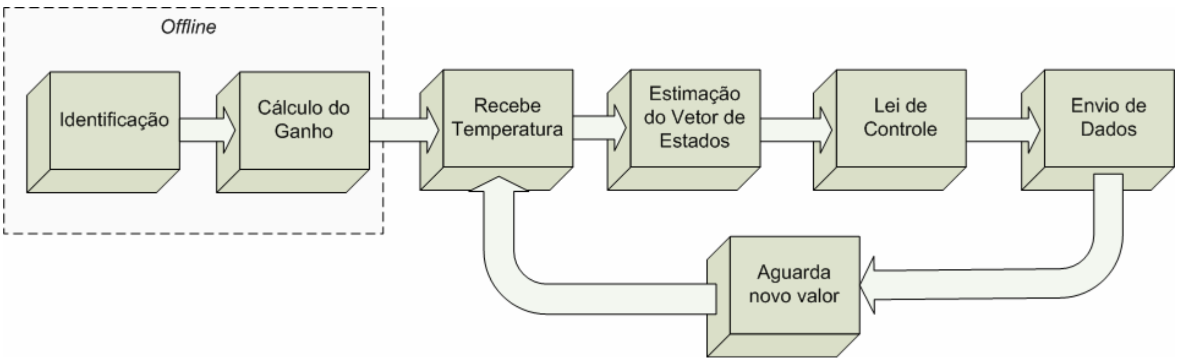


Figura 3.21 – Fluxograma do controle ótimo.

Para a implementação do controle ótimo no ambiente reduzido, faz-se necessário um modelo prévio do sistema. Com este intuito, gerou-se um sinal pseudo-aleatório e se mediu a saída do sistema para se proceder à identificação do modelo, conforme pode ser visto na Figura 3.22.

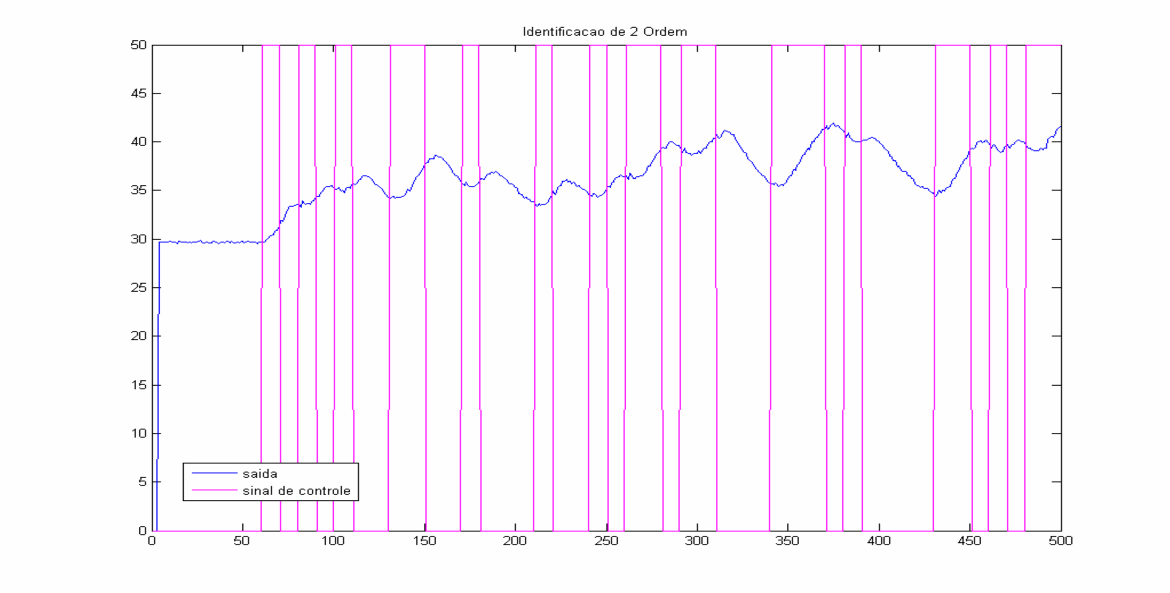


Figura 3.22 – Resposta a um sinal pseudo-aleatório.

A partir desta resposta, foi estimado um modelo *auto-regressivo com entrada exógena* (ARX), da seguinte forma:

$$A\left(q^{-1}\right)y(k) = B\left(q^{-1}\right)u(k), \tag{3.24}$$

em que:

$$A\left(q^{-1}\right) = 1 - 0{,}9995\, q^{-1} + 0{,}0005318\, q^{-2}, \tag{3.25}$$

$$B\left(q^{-1}\right) = 0{,}0006245\, q^{-2}. \tag{3.26}$$

Com base neste modelo, calculou-se recursivamente e *offline* o ganho em regime do sistema, de acordo com a equação (2.43). Em seguida, para a implementação no modelo reduzido, utilizou-se a equação (2.44). Os parâmetros foram então definidos como:

$$\mathbf{K}_\infty = 0{,}0005\,, \tag{3.27}$$

$$\mathbf{N}_\infty = 1{,}7\,, \tag{3.28}$$

$$\mathbf{L}_\infty = 1601\,, \tag{3.29}$$

$$\mathbf{Q_1} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \tag{3.30}$$

$$\mathbf{Q}_2 = 0{,}01. \tag{3.31}$$

Novamente, a escolha da matriz $\mathbf{Q_1}$ garante, no controle ótimo, que apenas os estados que têm influência na saída serão considerados. A Figura 3.23 ilustra o resultado obtido.

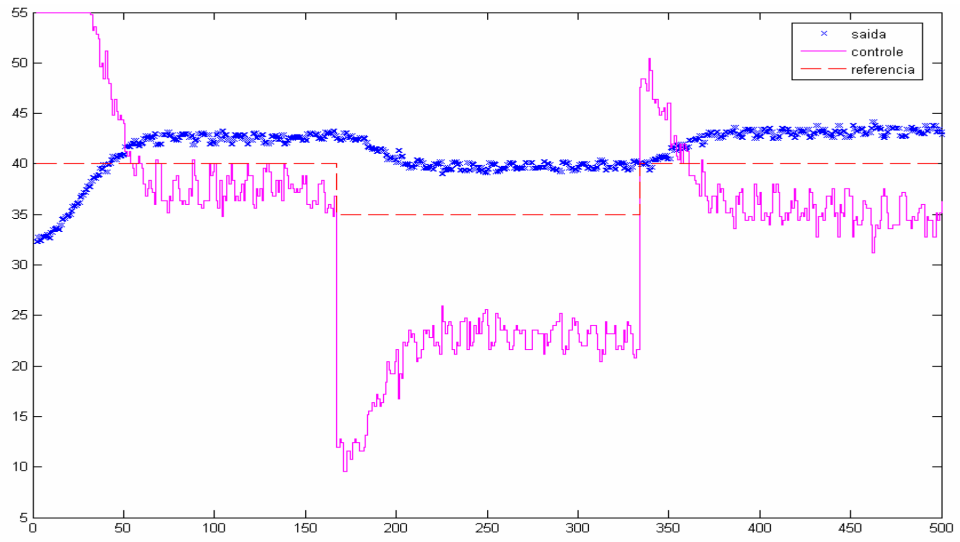


Figura 3.23 – Controle ótimo: resposta do sistema.

Por meio de uma análise visual, infere-se que a resposta do sistema apresenta-se deslocada para cima e com uma amplitude que pouco varia em relação à variação da referência. Por causa destes motivos expostos, os valores de $\mathbf{K}_\infty$ e $\mathbf{N}_\infty$ foram alterados e ajustados manualmente a fim de se melhorar um pouco mais a resposta do sistema em termos do ganho e do erro em regime permanente, como pode ser deduzido observando as equações (2.44) e (2.47). Redefiniram-se os parâmetros da seguinte forma:

$$\mathbf{K}_\infty = 0{,}005\,, \tag{3.32}$$

$$\mathbf{N}_\infty = 0{,}6\,. \tag{3.33}$$

A Figura 3.24 ilustra o resultado obtido. Pode-se observar que o sistema consegue seguir a referência de forma satisfatória, apresentando pequenas oscilações e sem picos no sinal de controle.

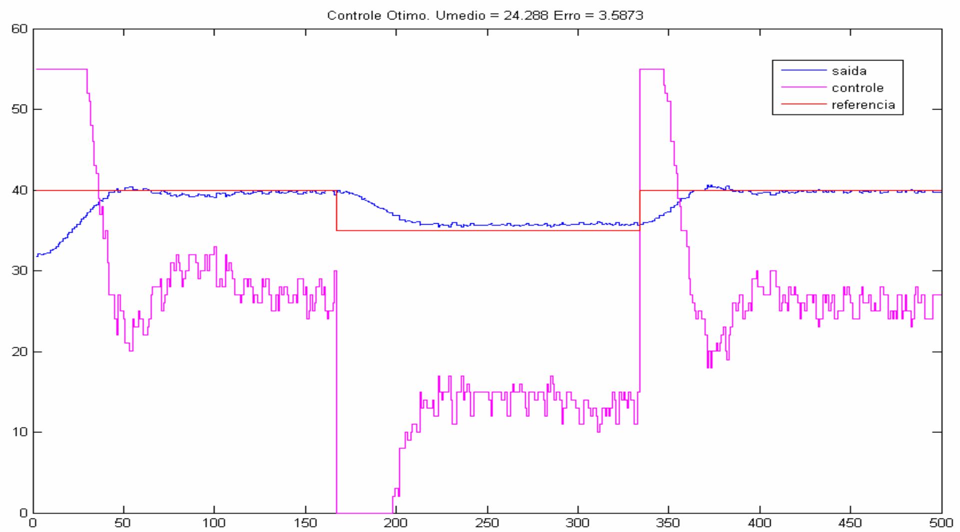


Figura 3.24 – Controle ótimo: resposta do sistema com alteração dos parâmetros.

3.7 COMPARAÇÃO DE DESEMPENHO DOS CONTROLADORES

Os resultados mostrados até agora foram válidos para se ter uma idéia do desempenho dos diversos controladores implementados. Todavia, com o objetivo de avaliar o desempenho dos controladores, necessitam-se de informações quantitativas a respeito de seu comportamento.

Neste sentido, o desempenho de cada controlador será avaliado com base em dois critérios: uma variável ligada ao rastreamento da referência e outra ao consumo de energia.

A fim de se verificar quão bem a saída acompanha a referência, tomou-se a média dos quadrados dos erros da seguinte forma:

$$J = \frac{1}{2N}\sum_{k=1}^{N}\left(y_{ref}(k) - y(k)\right)^2. \tag{3.34}$$

As equações (3.35) e (3.36) mostram as variáveis utilizadas como parâmetro para o consumo de energia. Sendo que estas são respectivamente a média aritmética e geométrica do sinal de controle.

$$U_{medio} = \frac{1}{N}\sum_{k=1}^{N} u(k), \tag{3.35}$$

$$U_{rms} = \sqrt{\frac{1}{N}\sum_{k=1}^{N} u^2(k)}. \tag{3.36}$$

A partir destas variáveis de desempenho, realizou-se o controle de temperatura no modelo reduzido utilizando cada um dos algoritmos propostos por um intervalo de tempo de cerca de 30 minutos. Nas Figuras 3.25 a 3.29, mostram-se os gráficos da resposta do sistema para cada técnica de controle.

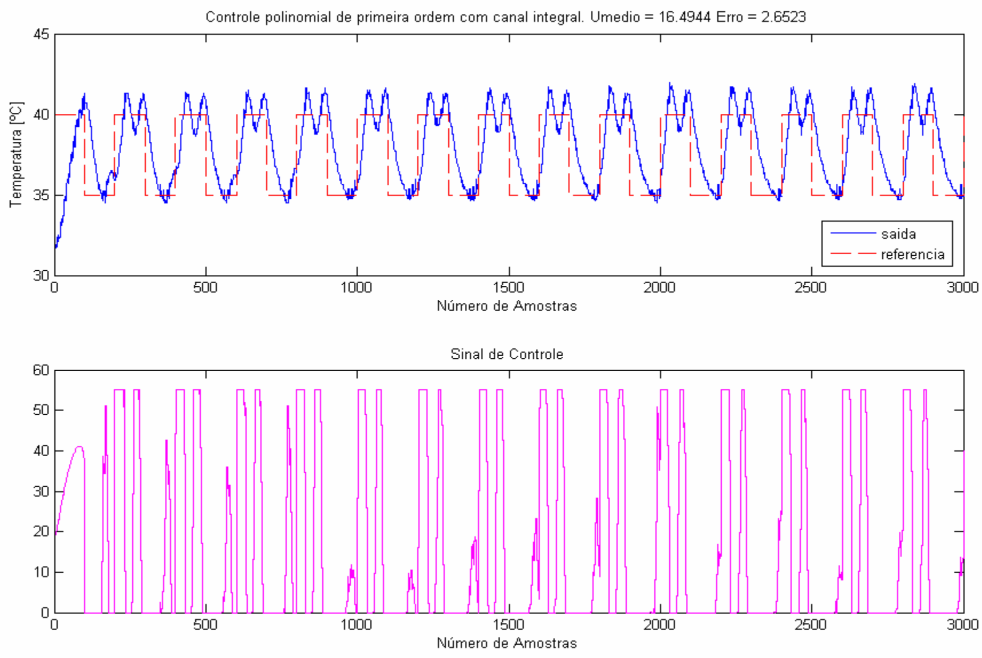


Figura 3.25 – Controle polinomial de primeira ordem com canal integral: resposta do sistema a um longo intervalo de tempo.

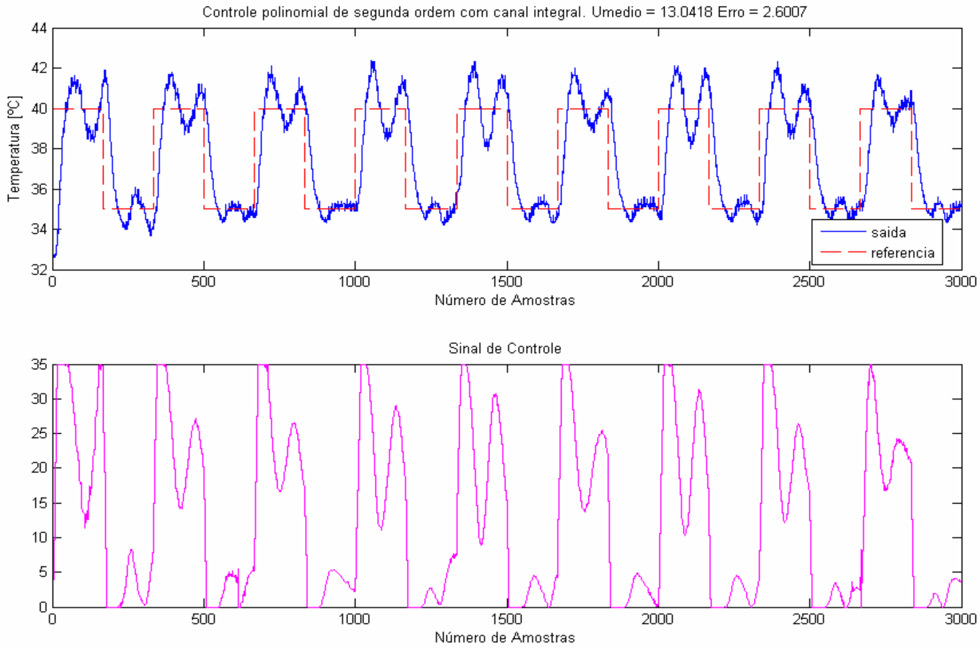


Figura 3.26 – Controle polinomial de segunda ordem com canal integral: resposta do sistema a um longo intervalo de tempo.

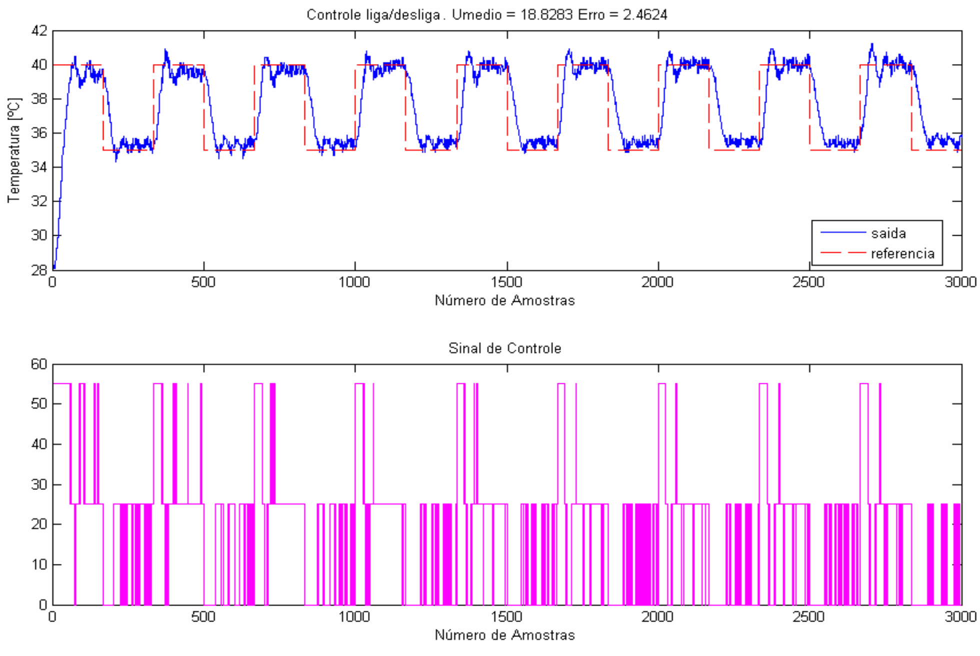


Figura 3.27 – Controle liga/desliga: resposta do sistema a um longo intervalo de tempo.

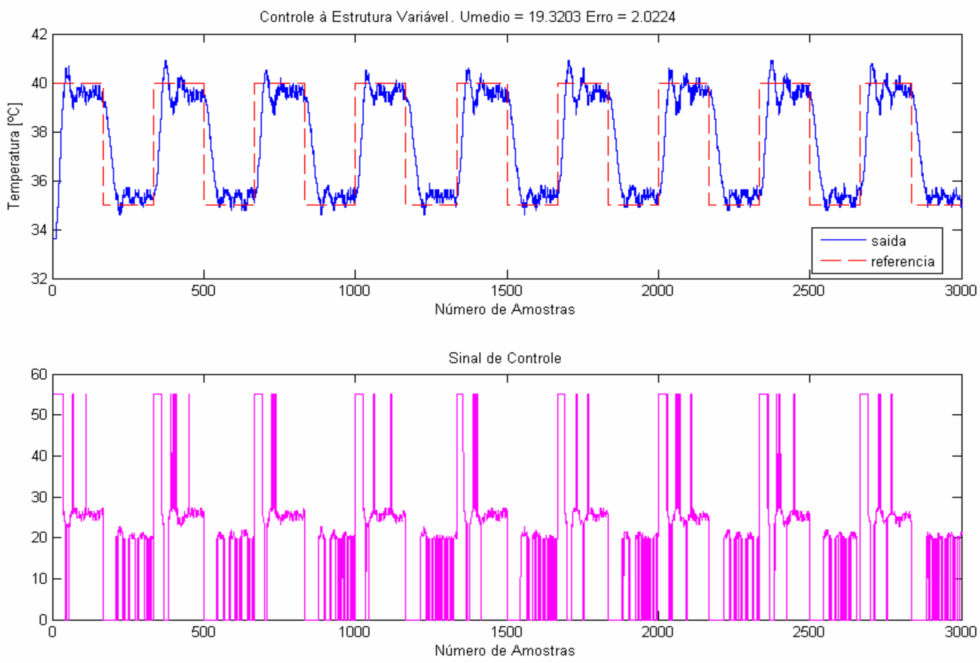


Figura 3.28 – Controle com estrutura variável:
resposta do sistema a um longo intervalo de tempo.

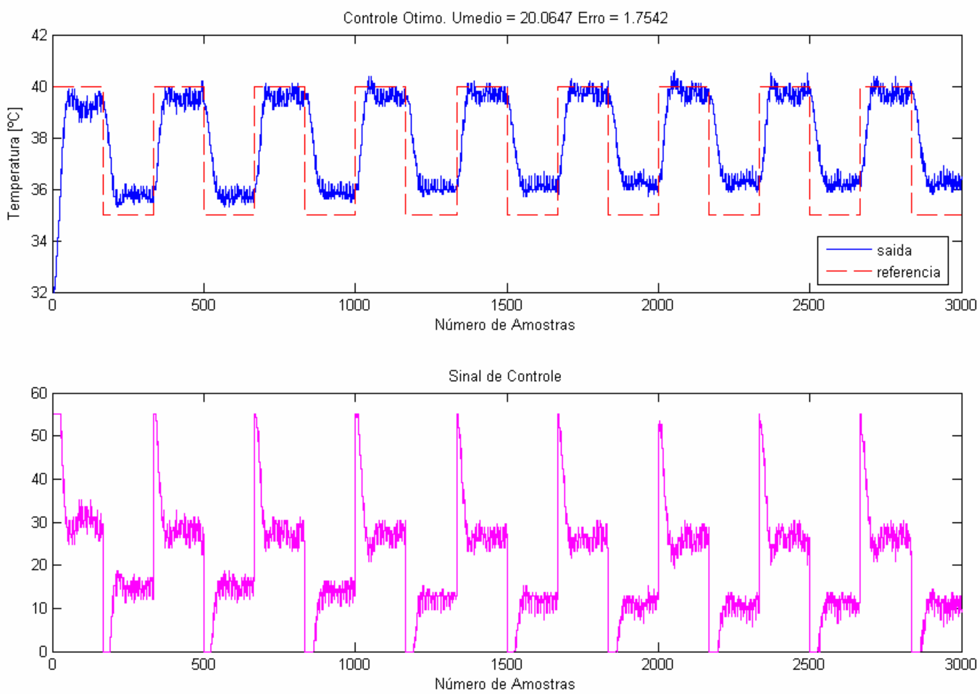


Figura 3.29 – Controle ótimo: resposta do sistema a um longo intervalo de tempo.

Com os dados obtidos, calcularam-se as variáveis de desempenho, conforme tabela 3.1.

Tabela 3.1 – Parâmetros de desempenho das técnicas de controle.

Método	J[ºC^2]	$U_{médio}$	U_{ms}
Self Tuning de Primeira Ordem	2,65	16,50	27,68
Self Tuning de Segunda Ordem	2,60	13,04	17,87
Liga/Desliga	2,46	18,83	25,29
Estrutura Variável	2,02	19,32	25,04
Controle Ótimo	1,75	20,06	23,55

A partir dos resultados da tabela 3.1, percebe-se que a técnica de controle que mais economiza energia, *self tuning* com modelo de segunda ordem, apresenta o segundo maior erro ao tentar seguir a referência. Por outro lado, a técnica com o menor erro é a que apresenta o maior consumo médio de energia. Podem-se classificar os métodos de acordo com o critério utilizado, conforme mostrado nas tabelas 3.2 e 3.3.

Tabela 3.2 – Classificação pelo erro em relação à referência.

Método	Diferença (%)
Controle Ótimo	00,00
Estrutura Variável	15,42
Liga/Desliga	40,57
Self Tuning de Segunda Ordem	48,57
Self Tuning de Primeira Ordem	51,43

Tabela 3.3 – Classificação de acordo com o consumo médio.

Método	Diferença (%)
Self Tuning de Segunda Ordem	00,00
Self Tuning de Primeira Ordem	26,53
Liga/Desliga	44,40
Estrutura Variável	48,16
Controle Ótimo	53,83

Uma observação importante a ser realizada na tabela 3.3 é em relação ao controle *self tuning* de primeira ordem. Apesar de aparecer na segunda colocação, ele apresenta o maior valor do consumo quadrático médio. Em outras palavras, seu sinal de atuação oscila com elevada amplitude, podendo comprometer a vida útil do atuador.

De acordo com as tabelas 3.2 e 3.3, têm-se métodos diferentes para critérios distintos. Há um compromisso entre o consumo de energia e o rastreamento da referência. Se a prioridade for o consumo de energia, o controle *self tuning* de segunda ordem é o mais indicado. Se a preferência for manter a temperatura mais próxima da desejada, indica-se o emprego do controle ótimo.

Todavia, há também uma técnica que apresenta bons resultados em ambos os critérios. O controle com estrutura variável apresenta um resultado tão bom quanto o controle ótimo para seguir uma dada referência e um resultado intermediário para o consumo de energia. A tabela 3.4 ilustra os métodos indicados para cada critério. Neste trabalho, não se modificaram os parâmetros dos controladores de forma exaustiva a fim de encontrar o conjunto de parâmetros com a melhor resposta do sistema. Neste sentido, estes parâmetros poderiam ser otimizados.

Tabela 3.4 – Métodos de controle indicados para cada critério.

Critério	Método
Erro	*Self Tuning* de Segunda Ordem
Consumo	Controle Ótimo
Ambos	Estrutura Variável

4 CONCLUSÕES

Neste trabalho, propôs-se a utilização de diferentes técnicas para o controle de temperatura de uma sala em tamanho reduzido. Dentre elas, têm-se que algumas são capazes de minimizar a variável ligada ao consumo de energia, fator este tão importante em nossa sociedade. Os métodos utilizados para o controle foram o controle com estrutura variável, o *self tuning* com controle polinomial, o liga/desliga e o ótimo.

Alguns métodos aqui descritos utilizam a identificação dos parâmetros ou a estimação dos estados. Neste sentido, o rastreamento de Kalman mostrou-se uma ferramenta bastante válida para ambos os fins.

Um dos controladores que têm mais chamado a atenção da comunidade acadêmica é o baseado em lógica *Fuzzy*. Um dos seus maiores méritos é conseguir lidar com diversas variáveis de entrada, além da temperatura. Exemplos podem ser umidade e qualidade do ar e temperatura externa. Contudo, neste trabalho, tem-se uma abordagem mais analítica do problema, tanto na identificação quanto no controle, com o objetivo de se conhecer melhor o sistema a ser controlado.

Por outro lado, têm-se a simplicidade e funcionalidade do controlador do tipo liga/desliga, sendo este o mais difundido para aplicações envolvendo o controle de temperatura de ambientes.

Os resultados obtidos mostram a existência de um custo de oportunidade entre o consumo de energia e seguir o valor de referência de temperatura. Para um menor consumo, o controle *self tuning* de segunda ordem apresentou o melhor resultado. Já o controlador que melhor seguiu um valor de temperatura foi utilizando técnicas de controle ótimo. Todavia, ambos os controladores apresentam um baixo desempenho no critério em que não se destacam. Uma alternativa é adotar um controle com estrutura variável para um resultado satisfatório tanto em termos de consumo como em termos de atingir certo valor de temperatura. Há de se dizer, no entanto, que se está comparando o resultado de um controlador não-linear com outros lineares, apesar de adaptativos (variantes no tempo).

Neste processo a saturação do atuador é um fator primordial. Dificilmente será obtido o ótimo utilizando controladores lineares. O controle preditivo seria uma abordagem interessante, pois otimiza uma função de custo considerando explicitamente a saturação do processo. Entretanto, o custo computacional é bem maior, pois é necessário resolver em tempo real um problema de programação quadrática.

A partir deste trabalho, foi possível mostrar que o controle de temperatura pode ser alcançado por meio de diferentes técnicas. Trabalhos futuros agora podem ser realizados em aparelhos reais de ar-condicionado em ambientes de tamanho real, verificando novamente uma técnica mais adequada de acordo com os critérios de erro e de consumo de energia. Uma topologia possível seria ligar cada aparelho de ar condicionado a um micro-controlador por meio de um sinal de atuação vindo do micro-controlador. Os micro-controladores, por sua vez, comunicam-se via barramento RS 485, ou por meio de tecnologia sem-fio, a um computador central dotado de um sistema supervisório. Já o computador, poderá comunicar-se com um servidor, tanto para armazenar um histórico com dados relativos ao consumo quanto para futuras expansões para aplicações *web*. Este esquema pode ser vislumbrado na Figura 4.1.

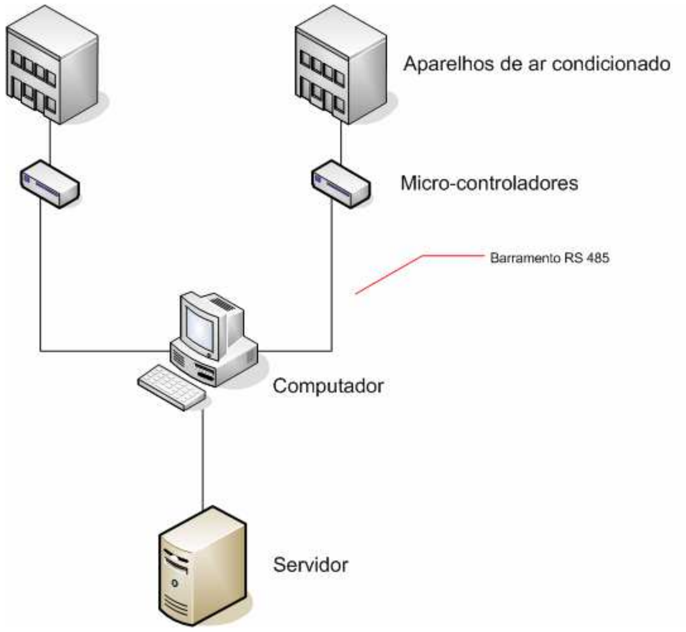


Figura 4.1 – Topologia possível para a aplicação em aparelhos de ar condicionado.

REFERÊNCIAS BIBLIOGRÁFICAS

Aguirre, L. A. (2000). “Introdução à Identificação de Sistemas”. Editora UFMG. Belo Horizonte, Brasil.

Ästrom, K. J., Bjorn W. (1990) “Computer-Controlled Systems: Theory and Design”. Second Edition. Prentice-Hall, Estados Unidos.

Ästrom, K. J., Bjorn W. (1995) “Adaptative Control”. Second Edition. Addison-Wesley Publishing Company, Estados Unidos.

Bauchspiess, A., Souza, A.S., Leite, A.A.C., Ramos,L.M.A. Pereira, E.S., Santos, R.J. (2004). “Fuzzy thermal control with remote access for building automation.” In: INCOM'2004 - 11th IFAC Symposium on Information Control Problems in Manufacuring, Salvador, Brasil.

Franklin G. F., Powell, J. D. Workman, M. L. (1997). “Digital Control of Dynamic Systems”. Terceira Edição. Addison-Wesley Longman, Estados Unidos.

Kirk D. E. (1970). “Optimal Control Theory: An Introduction”. Prentice-Hall, Estados Unidos.

Kolokotsa D., Tsiavos D., Stravrakakis G. S., Kalaizakis K. e Antonidakis E. (2001). “Advanced fuzzy logic controllers design and evaluation for buildings’s occupants thermal-visual comfort and indoor air quality satisfaction”. Energy and Buildings (33), pp. 531-543.

Kwok W. H. M., Wai T. D. C. (2003) “Adaptive comfort temperature model of air-conditioned building in Hong Kong”. Building and Environment (38), pp 837 – 852

Riederer P., Marchio D., Visier J. C., Husaunndee A., Lahrech R.. (2000). “Influence of sensor position in building thermal control: development and validation of an adapted zone model”. Seventh International IBPSA Conference. Rio de Janeiro, Brasil.

Santos, R. de J. (2005). “Controle fuzzy para racionalização de energia elétrica em processo de condicionamento de ar”. Dissertação de Mestrado em Engenharia Elétrica. Universidade de Brasília. Brasília, Brasil.

Signor, R. (1999). “Análise de Regressão do Consumo de Energia Elétrica frente a variáveis arquitetônicas para edifícios comerciais climatizados em 14 capitais brasileiras”. Dissertação de Mestrado em Engenharia Civil. Universidade Federal de Santa Catarina. Florianópolis, Brasil.

Zaheeruddin M., Tudoroiu N. (2004). “Neuro-PID tracking control of a discharge air temperature system”. Energy Conversion and Management (45), pp. 2405–2415

Zupančič, B., Škrjanc, I. (1998). “Advanced Fuzzy Control of the Temperature in the Test Chamber.” In: Faculty of Electrical Engineering, University of Ljubljana. Disponível em: <http://wire0.ises.org/wire/doclibs/SWC1999.nsf/20a4ea381a36ec7fc12569270037c619/2a3a3a14369de848c1256920003d613a!OpenDocument>. Acesso em 28/03/2006

APÊNDICES

APÊNDICE I – CÓDIGO DO MATLAB

I.1 – CONTROLE LIGA/DESLIGA

```
%%% Parâmetros da Simulação
N             = 3000;          % numero de amostras
Ta            = 1;             % taxa de amostragem
refInicial    = 35;            % características da entrada de onda quadrada
refFinal      = 40;
saturacaoMax = 60;             % saturação do controlador
saturacaoMin = 0;

%%% Geracao do Sinal de Referencia
yref = refInicial + (refFinal-refInicial)*(sin(2*pi*0.003*Ta*[1:N]')>0);
%%% Laço principal
uc = zeros(N,1);
y  = zeros(N,1);
try
    s1 = serial('COM2', 'BaudRate', 9600);
    fopen(s1)
    fprintf(s1,'%3.4f\r',Ta);
    pause(3);
    for n = 4 : N
        %%% Processo real
        y(n) = str2num( fscanf(s1) );
        if y(n) > ( yref(n) + 0.3 )
            uc(n) = 0;
        elseif y(n) < ( yref(n) - 0.8 )
            uc(n) = saturacaoMax - 5;
        else
            uc(n) = 4*(yref(n) - y(n) ) + 0.6*yref(n);
            if uc(n) > saturacaoMax
                uc(n) = saturacaoMax;
            elseif uc(n) < saturacaoMin
                uc(n) = saturacaoMin ;
            end
        end
        fprintf(s1,'%3.4f\r',uc(n) );
    end
    fclose(s1);
    clear s1;
    erro     = yref - y;
    J2       = sum( erro.^2 )/(2*N)        % quantificando o erro
    umedio = mean(uc)                      % sinal médio de controle
catch
```

```
    fclose(s1)
    beep()
    error('Erro na porta serial')
end
```

I.2 – CONTROLE *SELF TUNING* DE PRIMEIRA ORDEM

```
% ******************** Parâmetros da Simulação *************************
N            = 500;              % numero de amostras
Ta           = 1;                % taxa de amostragem
refInicial            = 45;    % características da entrada de onda quadrada
refFinal              = 45;
saturacaoMax          = 30;    % saturação do controlador
saturacaoMin          = 0;

% *********************************** Processo *********************************
%             Y(s) =  Kp
%             ----   ------
%             U(s)  sT + 1
%
%  y(k) = a0 * y(k-1) + b0 * u(k-1)

% ****************** Geracao do Sinal de Referencia ***********************
yref = refInicial + (refFinal-refInicial)*(sin(2*pi*0.1*Ta*[1:N]')>0);

% ************************** Filtro de Kalman *******************************
Xe = zeros(4,1);
Pe = zeros(4,4);
Xe = [0.1 0.1 0 0]';                     % condição inicial
Pe = diag([1 1 0.1 0.1].^2);
Q = diag(0.01*[0.1 0.1 0.001 0.001].^2);
R = 0.05^2;
% Considerando modelo de primeira ordem de evolução dos parâmetros
A = [1 0 Ta 0; 0 1 0 Ta; 0 0 1 0; 0 0 0 1];

% ************************** Inicializando Vetores ****************************
uc = zeros(N,1);
y  = zeros(N,1);

% ************************** Polinômio Desejado ******************************
bm0   = 0.048;
am0   = 0.95;
try
    s1 = serial('COM2', 'BaudRate', 9600);
    fopen(s1)
    fprintf(s1,'%3.4f\r',Ta);
```

```
    pause(3);
    % Laço principal
    for n = 2 : N
        % Processo real
        y(n) = str2num( fscanf(s1) );
        % Filtro de Kalman: predição.
        Xp = A*Xe;
        Pp = A*Pe*A' + Q;
        % Filtro de Kalman: correção.
        C = [y(n-1) uc(n-1) 0 0];
        G = Pp*C'*inv(C*Pp*C'+R);
        Xe = Xp + G*(y(n)-C*Xp);
        Pe = (eye(4)-G*C)*Pp;
        % Constantes estimadas do processo
        a0e(n)= Xe(1);
        b0e(n)= Xe(2);
        Tpe(n) = -Ta / log(b0e(n));
        Kpe(n) = a0e(n) / (1 - exp(-Ta/Tpe(n)));
        % Controle Polinomial
        uc(n) = ( bm0/b0e(n) )*yref(n) - ( a0e(n)-am0)/b0e(n)*y(n);
        if uc(n) > saturacaoMax
            uc(n) = saturacaoMax;
        elseif uc(n) < saturacaoMin
            uc(n) = saturacaoMin;
        end
        % Enviar sinal de controle
        fprintf(s1,'%3.4f\r',uc(n) );
    end
    fclose(s1);
    clear s1;
    erro   = yref - y;
    J2     = sum( erro.^2 )/(2*N)              % quantificando o erro
    umedio = mean(uc)                          % sinal médio de controle
catch
    fclose(s1)
    beep()
    error('Erro na porta serial')
end
```

I.3 – CONTROLE *SELF TUNING* DE SEGUNDA ORDEM

```
% ** Controle Self-Tuning Deterministico de um Sistema de Segunda Ordem

%%% Parâmetros da Simulação
N          = 3000;                 % numero de amostras
Ta          = 1;                   % taxa de amostragem
refInicial   = 35;                 % características da entrada de onda quadrada
refFinal     = 40;
saturacaoMax = 55;                 % saturação do controlador
saturacaoMin = 0;

%%% Processo
%%%
%%%  y(k) = a0 * y(k-1) + b0 * u(k-1) + b1 * u(k-2)

%%% Geracao do Sinal de Referencia
yref = refInicial + (refFinal-refInicial)*(sin(2*pi*0.003*Ta*[1:N]')>0);

%%% Filtro de Kalman
Xe = zeros(5,1);
Pe = zeros(5,5);
Xe = [0.1 0.1 0.1 0.1 0.1 0 0 0 0 0]';                 % condição inicial
Pe = diag([1 1 1 1 1 0.1 0.1 0.1 0.1 0.1].^2);
Q = diag(0.01*[0.1 0.1 0.1 0.1 0.1 0.001 0.001 0.001 0.001 0.001].^2);
R = 0.05^2;
A = [1 0 0 0 0 Ta 0 0 0 0;
    0 1 0 0 0 0 Ta 0 0 0;
    0 0 1 0 0 0 0 Ta 0 0;
    0 0 0 1 0 0 0 0 Ta 0;
    0 0 0 0 1 0 0 0 0 Ta;
    0 0 0 0 0 1 0 0 0 0 ;
    0 0 0 0 0 0 1 0 0 0 ;
    0 0 0 0 0 0 0 1 0 0 ;
    0 0 0 0 0 0 0 0 1 0 ;
    0 0 0 0 0 0 0 0 0 1 ; ];
%%% Laço principal
uc = zeros(N,1);
ub = zeros(N,1);
y  = zeros(N,1);
% Modelo de referencia
bm0   = 0.12;
bm1   = 0.06;
am0   = 1;
am1   = - 0.7603;
am2   =  0.1445;
```

```
try
    s1 = serial('COM2', 'BaudRate', 9600);
    fopen(s1)
    fprintf(s1,'%3.4f\r',Ta);
    pause(3);
    for n = 4 : N
        %%% Processo real
        y(n) = str2num( fscanf(s1) );
        %%% Filtro de Kalman: predição.
        Xp = A*Xe;
        Pp = A*Pe*A' + Q;
        %%% Filtro de Kalman: correção.
        C = [y(n-1) y(n-2) y(n-3) uc(n-1) uc(n-2) 0 0 0 0 0];
        G = Pp*C'*inv(C*Pp*C'+R);
        Xe = Xp + G*(y(n)-C*Xp);
        Pe = (eye(10)-G*C)*Pp;
        %%% Constantes estimadas do processo
        a2e(n)= Xe(1);
        a1e(n)= Xe(2);
        a0e(n)= Xe(3);
        b0e(n)= Xe(4);
        b1e(n)= Xe(5);
        % Controle Polinomial
        uc(n) =  uc(n-1) + b1e(n)/b0e(n)*(am0-a2e(n))*(y(n-2)-y(n-1)) +(am2-a0e(n))*(y(n-1)-y(n)) +
                    (am0+am1+am2)*(yref(n)-y(n)) ;
        if uc(n) > saturacaoMax
            uc(n) = saturacaoMax;
        elseif uc(n) < saturacaoMin
            uc(n) = saturacaoMin ;
        end
        fprintf(s1,'%3.4f\r',uc(n) );
    end
    fclose(s1);
    clear s1;
    erro   = yref - y;
    J2     = sum( erro.^2 )/(2*N)                % quantificando o erro
    umedio = mean(uc)                            % sinal medio de controle
catch
    fclose(s1)
    beep()
    error('Erro na porta serial')
end
```

I.4 – CONTROLE ÓTIMO

```
% Controlador ótimo de um Sistema de Segunda Ordem  ***

% Parâmetros da Simulação
N              = 3000;             % numero de amostras
Ta             = 1;                % taxa de amostragem
refInicial     = 35;               % características da entrada de onda quadrada
refFinal       = 40;
saturacaoMax = 55;                 % saturação do controlador
saturacaoMin = 0;

%%% Processo
%%%          y[k] =  B(q^1)  u[k]
%%%                 -------
%%%                  A(q^1)
%%%

%%% Geracao do Sinal de Referencia
ref = refInicial + (refFinal-refInicial)*(sin(2*pi*0.003*Ta*[1:N]')>0);
ref = ref';
ttt = zeros(1,N);
Ref = [ref; ttt];

% -------------------- Constantes --------------------
a1 = -0.9995;
a2 = 0.0005318;
b0 = 0.0006245;
p1 = 1;
p2 = 0.01;
Phi = [-a1 -a2; 1 0];
T   = [1; 0];
Cd  = [0 b0];
Q1  = [p1 0; 0 0];
Q2  = p2;
for n = 1:N
    K(:,:,n) = [0.005; 0];
end
% Filtro de Kalman
Xe = zeros(2,2);
Pe = zeros(2,2);
Xe(:,:,1) = [0.1 0.1; 0 0;]';
Pe(:,:,1) = diag([1 1].^2);
Q = diag(0.01*[0.1 0.1].^2);
R = 0.001^2;
```

```
x(:,:,1)  = [40000; 40000];
xb(:,:,1) = x(:,:,1) - Ref(:,1);
J = zeros(N,1);
e = zeros(N,1);
%%% Laço principal
uc = zeros(N,1);
y  = zeros(N,1);
try
    s1 = serial('COM2', 'BaudRate', 9600);
    fopen(s1)
    fprintf(s1,'%3.4f\r',Ta);
    pause(3);
    for n = 2 : N
        %%% Processo real
        y(n) = str2num( fscanf(s1) );
        %%% Filtro de Kalman: predição.
        Xp = Phi*Xe;
        Pp = Phi*Pe*Phi' + Q;
        %%% Filtro de Kalman: correção.
        C = [y(n-1) uc(n-1)];
        G = Pp*C'*inv(C*Pp*C'+R);
        Xe = Xp + G*(y(n)-C*Xp);
        Pe = (eye(2)-G*C)*Pp;
        %%% Constantes estimadas do processo
        a0e(n)= Xe(1);
        b0e(n)= Xe(2);
        Ee = [a0e(n); b0e(n)];
        xb(:,:,n) = Ee(:,:) - Ref(:,n).*1601;
        uc(n) = -K(:,:,n)'*xb(:,:,n) + 0.6*Ref(1,n);
        if uc(n) > saturacaoMax
            uc(n) = saturacaoMax;
        elseif uc(n) < saturacaoMin
            uc(n) = saturacaoMin ;
        end
        fprintf(s1,'%3.4f\r',uc(n) );
    end
    fclose(s1);
    clear s1;
    erro      = ref' - y;
    J2        = sum( erro.^2 )/(2*N)          % quantificando o erro
    umedio = mean(uc)                         % sinal medio de controle
catch
    fclose(s1)
    beep()
    error('Erro na porta serial')
end
```

APÊNDICE II – CÓDIGO DO MICROCONTROLADOR

```
/****************************************************************************************/
/*                    UNIVERSIDADE DE BRASÍLIA                                          */
/*              DEPARTAMENTO DE ENGENHARIA ELÉTRICA                                     */
/*    LABORATÓRIO DE AUTOMAÇÃO, VISÃO E SISTEMAS INTELIGENTES                           */
/*       ORIENTADOR : ADOLFO BAUCHSPIESS                                                */
/*                   ALUNO: RODRIGO FONTES SOUTO                                        */
/****************************************************************************************/

#include  <18F252.h>
#device   adc=10
#use      delay(clock=10000000)
#fuses    HS, NOWDT, NOLVP, PUT
#use      rs232(baud=9600, xmit=PIN_C6, rcv=PIN_C7, parity=N, bits=8)
#include <stdlib.h>
#include <string.h>

void reset();
void saida (int out);
void ler_temp();
void transmite();
void calcula_par();

/*** DEFININDO VARIÁVEIS E CONSTANTES GLOBAIS DO PROGRAMA ****/
SIGNED INT32        pwm_conta,
                    t0_conta,
                    pid_conta;
INT                 u=0;
INT16               time=0,
                    per=0;
FLOAT               temperatura1 = 0;

/********* RESETANDO VARIÁVEIS (VOLTANDO AO PADRÂO)**********/
void reset()
{
  OUTPUT_LOW(PIN_C1);
  temperatura1 = 0;
  pwm_conta    = 1302;
  pid_conta     = 7812;
  t0_conta      = 78125;                    // então tempo é incrementado a cada 1 segundo
  u             = 0;
}
```

```
/************************ GERANDO PWM DE SAIDA *********************/
void saida (int out)
{
  if (per<out)  {
      OUTPUT_HIGH(PIN_C1);
  }  else  {
      OUTPUT_LOW(PIN_C1);
  }
}

/****************** LÊ OS CANAIS ANALÓGICOS ********************/
void ler_temp()
{
  char temp[10];

  set_adc_channel(0);                    //setando leitura canal 0
  delay_uS(40);
  temperatura1 = (float)read_adc()*0.100459433;
}

/****** ROTINA DE TRANSMISSÃO DE DADOS PARA O SERVIDOR ******/
void transmite()
{
  printf ("%3.1f\n",temperatura1);
}

/****** ROTINA TRATAMENTO DE INTERRUPÇÃO DO TIMER 0 ******/
#INT_TIMER0
void TRATA_T0()                    // como freq. crystal == 4MHz
{
  char aux[10];

  STATIC BOOLEAN LED;
  t0_conta   -= 256;
  pid_conta  -= 256;
  pwm_conta  -= 256;               // freq. interna = 1/4 freq. crystal(1MHz)

  IF (pwm_conta<=0)                // usando CONTA, freq. cai para 1K/125=8Hz
  {
    pwm_conta +=1302;              // então tempo é incrementado a cada 125 mili segundos
    per++;
    IF (per==60)
    {
      per=0;
    }
  }
```

```
  IF (t0_conta<=0)                // usando CONTA, freq. cai para 125/125=1Hz
  {
    t0_conta +=78125;             // então tempo é incrementado a cada 1 segundo
    LED=!LED;
    OUTPUT_BIT(PIN_B0,LED);
    time++;
  }
  IF (pid_conta<=0)
  {
    pid_conta+=7812;
    ler_temp();
    transmite();
    gets(aux);
    u = atof(aux);
    saida(u);
  }
}

/************************* ROTINA PRINCIPAL **************************/
void main()
{
 pwm_conta=1302;
 pid_conta=7812;
 t0_conta =78125;                         // então tempo é incrementado a cada 1 segundo
 // referência do timer ciclo interno dividido por 64
 SETUP_TIMER_0(RTCC_INTERNAL|RTCC_DIV_128|RTCC_8_BIT);
 // habilita interrupção de timer0
 ENABLE_INTERRUPTS(GLOBAL|INT_TIMER0);
 // configurando 8 portas como analógicas
 SETUP_ADC_PORTS(ALL_ANALOG);
 // Tad= 2/10000000 -> 0.2 micro segundos (condição do pic Tad: 20 >Tad>1,6 )
 SETUP_ADC(ADC_CLOCK_INTERNAL);
 OUTPUT_LOW(PIN_C1);
 OUTPUT_LOW(PIN_C4);
 WHILE(TRUE) {   }
}
```