

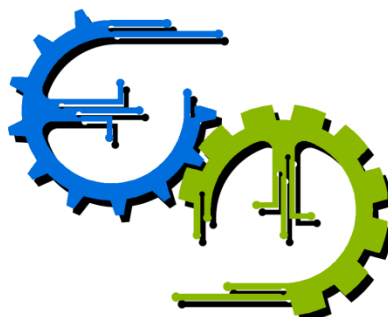


TRABALHO DE GRADUAÇÃO

CONTROLE ADAPTATIVO DA CLIMATIZAÇÃO PREDIAL PARA EFICIÊNCIA ENERGÉTICA

Heyder Antonio Silva de Araújo
Marcella Cortat Campos Melo

Brasília, julho de 2013.



**ENGENHARIA
MECATRÔNICA**
UNIVERSIDADE DE BRASÍLIA

UNIVERSIDADE DE BRASÍLIA

Faculdade de Tecnologia

Curso de Graduação em Engenharia de Controle e Automação

TRABALHO DE GRADUAÇÃO

CONTROLE ADAPTATIVO DA CLIMATIZAÇÃO PREDIAL PARA EFICIÊNCIA ENERGÉTICA

Heyder Antonio Silva de Araújo

Marcella Cortat Campos Melo

Relatório submetido como requisito parcial para obtenção
do grau de Engenheiro de Controle e Automação.

Banca Examinadora

Prof. Adolfo Bauchspiess, UnB/ ENE

Orientador

Prof. Antônio Padilha Lanari Bó, UnB/ ENE

Examinador

Prof. Lélío Ribeiro Soares Júnior, UnB/ ENE

Examinador

Brasília, julho de 2013.

FICHA CATALOGRÁFICA

ARAÚJO, H. A. S., MELO, M.C.C.

CONTROLE ADAPTATIVO da CLIMATIZAÇÃO PREDIAL PARA EFICIÊNCIA ENERGÉTICA

[Distrito Federal, 2013]

xiii, 130p, 297 mm (FT/UnB, Engenheiro, Controle e Automação, 2013). Trabalho de Graduação – Universidade de Brasília. Faculdade de Tecnologia.

1. Automação

2. Conforto térmico

I. Mecatrônica/FT/UnB

REFERÊNCIA BIBLIOGRÁFICA

H. A. S. ARAÚJO, M. C. C. MELO (2013). Estudo do Controle Adaptativo na Eficiência Energética de Climatização Utilizando Ambiente Predial. Trabalho de Graduação em Engenharia de Controle e Automação, Publicação FT.TG-nº XX, Faculdade de Tecnologia, Universidade de Brasília, Brasília, DF, 130p.

CESSÃO DE DIREITOS

AUTORES: H. A. S. ARAÚJO, M. C. C. MELO.

TÍTULO DO TRABALHO DE GRADUAÇÃO: Controle Adaptativo da Climatização Predial Para Eficiência Energética.

GRAU: Engenheiro de Controle e Automação

ANO: 2013

É concedida à Universidade de Brasília permissão para reproduzir cópias deste Trabalho de Graduação e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte desse Trabalho de Graduação pode ser reproduzida sem autorização por escrito do autor.

AGRADECIMENTOS

....

Heyder Antonio Silva de Araújo

Agradeço à minha família, em especial à minha mãe, Rosangela.

Agradeço ao professor Adolfo, por ter nos ajudado com a sua experiência e a sua tranquilidade típicas. Agradeço ao meu colega de trabalho e grande amigo, Heyder, sempre paciente e prestativo. Agradeço também aos companheiros de curso, George e Jefferson, e de semestre, Filipe e Leonardo, todos de grande valia pessoal e profissionalmente.

Marcella Cortat Campos Melo

RESUMO

Existe uma grande preocupação no que diz respeito ao consumo excessivo de energia elétrica. Dentro deste contexto, estudos têm sido feitos no intuito de tornar mais eficiente o uso deste tipo de energia. Geralmente, sistemas térmicos são os que representam a parcela maior do consumo energético. Condicionadores e aquecedores de ar, chuveiros entre outros aparelhos eletro/eletrônicos são responsáveis por grande parte do consumo de energia elétrica em edifícios, empresas e residências. É proposta, deste projeto, otimizar o uso da energia em sistemas térmicos. Este trabalho é parte de uma linha de pesquisa que visa tornar mais eficiente o consumo de energia elétrica em sistemas condicionadores de ar, considerando o conforto. O presente estudo está focado no consumo de energia de um ambiente predial: a primeira etapa, em que são realizados testes, será executada em uma maquete de madeira, cuja dinâmica térmica é mais rápida; a segunda etapa, de validação, será feita na sala de reuniões do Laboratório de Automação e Robótica.

Quanto ao controle, a comunicação será feita pelo uso de módulos XBee associados a arduínos. A medição da temperatura é tomada por sensores do tipo SHT71, e o processamento de dados, com o auxílio do MatLab. A identificação da função de transferência será feita de forma recursiva, com um algoritmo de Mínimos Quadrados adaptado. A abertura/fechamento de portas e/ou janelas, bem como a variação da temperatura externa (envolvendo as salas) permite estudar a eficiência energética do controlador adaptativo, em comparação aos tradicionais Liga-Desliga e PI.

Palavras-chave: conforto térmico, racionalização de energia, controle PID, controle adaptativo, XBee, SHT71, identificação de sistemas.

ABSTRACT

There is been a great concern regarding the use of energy. Within this context, studies have been made in order to make the use of such energy more efficient. Thermal systems are usually the ones which represent the largest amount of energy expenditure in building environments. Air Conditioners, heaters and showers among other electronics are responsible for a great part of the energy consumption in buildings, companies and residences. The propose of this project is to optimize the use of energy in thermal systems. This project is part of a research line that aims a more efficient use of electrical energy in air conditioning systems, considering comfort. Such study is focused on the consumption of energy in building environments: the first stage, mainly for tests, was performed on a wooden model , whose thermal dynamics is faster; the second step, focused on validation, was done on the conference room of the Automation and Robotics Laboratory.

The control was made using XBee modules; the temperature was measured by SHT71 sensors, and the computational environment used was MatLab. The identification of the transfer function was identified in a recursive way, using an adapted Least Squares algorithm. The opening/closing of the door and/or windows, as well as the external temperature variation (involving the adjacent rooms) allowed the study of energy efficiency of the adaptive controller, in comparison with the traditional ON-OFF and PI controllers.

Key-words: thermal comfort, energy saving, PID control, adaptive control, xbee, SHT71, system identification.

SUMÁRIO

1	Introdução	1
1.1	Motivação	1
1.2	Conforto Térmico	2
1.2.1	Termorregulação humana.....	2
1.2.2	Índice PMV	3
1.3	Objetivos.....	4
1.4	Apresentação do Manuscrito	5
2	Revisão Bibliográfica.....	6
2.1	Ambientes de Estudo.....	6
2.1.1	Sala de Reuniões do LARA	6
2.1.2	Maquete – Protótipo da Sala.....	7
2.2	Modelagem Matemática.....	9
2.2.1	Técnicas de Modelagem.....	9
2.2.2	Técnicas de Algoritmo	10
2.3	Fundamentação Teórica	11
2.3.1	Modelo da Parede	11
2.3.2	Identificação Recursiva.....	14
2.3.3	Controle Adaptativo	17
3	Materiais.....	21
3.1	Sensor de temperatura e umidade SHT71.....	21
3.1.1	Protocolo de Transferência de Dados.....	22
3.1.2	Especificações Técnicas.....	23
3.2	XBee.....	24
3.2.1	Modos de Comunicação	25

3.2.2	Adaptador CON-USBBEE ROGERCOM.....	26
3.3	Arduino	26
3.3.1	Módulo de Expansão	27
3.4	Relé de Estado Sólido - T2405Z-M	28
3.4.1	PWM – Acionamento do Relé	29
4	Procedimentos	30
4.1	Rede XBee e Transmissão de Dados	30
4.2	Abordagens de Controle	31
4.2.1	Controle Liga-desliga	32
4.2.2	Controle Proporcional-Integrativo.....	32
4.2.3	Controle Adaptativo	33
5	Resultados	35
5.1	Experimento na Maquete	35
5.1.1	Identificação	35
5.1.2	Controle com Carga Térmica Constante	37
5.2	Experimento na Sala de Reuniões.....	39
5.2.1	Identificação	39
5.2.2	Controle com Carga Térmica Constante	41
5.2.3	Controle com Carga Térmica Variável	45
6	Conclusão	51
7	Trabalhos Futuros	53
	Bibliografia.....	54
	Anexos.....	55

LISTA DE FIGURAS

Figura 1-1. Fisiologia humana e suas trocas térmicas (Lamberts, 2011).....	3
Figura 1-2. Gráfico de relação do PMV com o PPD (Oliveira, 2009).	4
Figura 2-1. Ambiente de estudo – LARA (adaptada de Oliveira, 2010), destacando a sala de reuniões.	6
Figura 2-2. Vista da sala de reuniões do LARA (horário noturno, sem incidência solar).....	6
Figura 2-3. Ar condicionado da sala de reunião (painel de controle, unidade evaporativa).	7
Figura 2-4. Visão do sistema de automação do ar condicionado.....	7
Figura 2-5. Gráfico da temperatura da sala de reuniões com acionamento ON-OFF. Em vermelho, perturbações devido ao trânsito de pessoas.....	8
Figura 2-6. Maquete de madeira de ambiente predial.	8
Figura 2-7. Esquemático das trocas de calor por uma parede (adaptada de Incropera e DeWitt, 2006).	11
Figura 2-8. Malha representativa (modelo) da parede (adaptada de Incropera e DeWitt, 2006).	13
Figura 2-9. Modelo 2R1C da parede (adaptada de Bauchspiess et al., 2006).....	14
Figura 2-10. Esquemático geral de controle adaptativo.....	17
Figura 2-11. Esquemático de controle adaptativo direto.....	18
Figura 3-1. Sensor SHT71 (Sensirion A. G. ,2009).	21
Figura 3-2. Típico circuito de aplicação, incluindo o resistor de pull-up (adaptada de Sensirion A. G. ,2009).	21
Figura 3-3. Diagrama de blocos do sensor SHT71 (Sensirion A. G. ,2009).....	22
Figura 3-4. Início da transmissão (Sensirion A. G.,2009).	22
Figura 3-5. Medição e transmissão de dados – comando 00101, umidade relativa (Sensirion A. G. , 2009).....	23
Figura 3-6. Módulos de comunicação XBee.....	25
Figura 3-7. Esquemático da comunicação entre módulos (pyroelectro.com).....	25
Figura 3-8. Fluxo de dados no XBee.....	26
Figura 3-9. Adaptador COM-USB BEE (rogercom.com.br).....	26
Figura 3-10. Hardware do Arduino (arduino.cc).....	27
Figura 3-11. Placa <i>shield</i> associada ao módulo de expansão XBee (arduino.cc).....	27
Figura 3-12. Relé T2405Z-M (teletronic.ind.br).....	28
Figura 3-13. Circuito interno ao relé T2405Z-M (teletronic.ind.br).	29

Figura 4-1. Esquemático da mini rede XBee para identificação e controle.	31
Figura 4-2. Sistemas de controle aplicados para o processo térmico.	31
Figura 4-3. Esquemático de etapas do controle adaptativo.	33
Figura 5-1. Medição da temperatura no ambiente de estudo na maquete.	36
Figura 5-2. Estimação de parâmetros do ambiente de estudo na maquete, conforme a equação (5.4).	36
Figura 5-3. Medição de temperatura usando controle ON-OFF.	37
Figura 5-4. Medição de temperatura usando controle PI.	38
Figura 5-5. Medição de temperatura usando controle RST.	38
Figura 5-6. Estimação da temperatura.	40
Figura 5-7. Estimação dos parâmetros, conforme a equação 5.7.	40
Figura 5-8. Medição de temperatura usando controle ON-OFF.	41
Figura 5-9. Sinal de controle do controlador ON-OFF.	42
Figura 5-10. Medição de temperatura usando controle PI.	42
Figura 5-11. Sinal de controle do controlador PI.	43
Figura 5-12. Medição de temperatura usando controle RST.	43
Figura 5-13. Sinal de controle do controlador RST.	44
Figura 5-14. Medição da temperatura usando controlador ON-OFF, com perturbações.	46
Figura 5-15. Sinal de controle do controlador ON-OFF, com perturbações.	46
Figura 5-16. Medição da temperatura usando controlador PI, com perturbações.	47
Figura 5-17. Sinal de controle do controlador PI, com perturbações.	47
Figura 5-18. Medição da temperatura usando controlador RST, com perturbações.	48
Figura 5-19. Sinal de controle do controlador RST, com perturbações.	48
Figura IV-1. Diagrama simulink do controle adaptativo geral.	72
Figura IV-2. Diagrama simulink do processo de identificação.	72
Figura IV-3. Diagrama de simulink da estimação de parâmetros.	73

LISTA DE TABELAS

Tabela 3-1. Lista de comandos.	23
Tabela 3-2. Coeficiente de conversão de temperatura de acordo com a alimentação no sensor.	24
Tabela 3-3. Coeficiente de conversão de temperatura de acordo com a resolução.	24
Tabela 3-4. Dados típicos do sensor em relação à temperatura.	24
Tabela 3-5. Valores de período do PWM, considerando tipo de controle e ambiente.....	29
Tabela 4-1. Sintonia empírica de controlador P, PI, PID – modelo CHM (Chien, Hrone e Reswick, 1952).	32
Tabela 4-2. Ganho K para cada ambiente predial estudado.....	34
Tabela 5-1. Relação entre tipos de controladores e melhoria em termos de energia.	45
Tabela 5-2. Relação entre tipos de controladores e melhoria em termos de energia, em comparação com o controlador ON-OFF, com perturbações.	49
Tabela 5-3. Tabela resumo de desempenho dos controladores (configuração, erro RMS e consumo energético).....	50

LISTA DE SÍMBOLOS

AC	Alternate Current
A/D	Analógico/Digital
ADC	Analog to Digital Converter
API	Application Programming Interface
ARMAX	Autoregressive Moving Average Model With Exogenous Inputs
ASHARE	American Society of Heating, Refrigerating and Air Conditioning Engineers
AVR	Alf Vegard RISC Processor
CRC	Cyclic Redundancy Check
DC	Direct Current
I ² C	Inter Integrated Circuits
ICSP	In-Circuit Serial Programming
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
I/O	Input/Output
LARA	Laboratório de Automação e Robótica
LED	Light Emitting Diode
MatLab	MATrix LABoratory
MQ	Mínimos Quadrados
P	Proporcional
PI	Proporcional Integrativo
PID	Proporcional Integrativo Derivativo
PMV	Predicted Mean Vote
PPD	Predicted Percentage of Dissastified
PWM	Pulse-Widht Modulation
RAM	Randon Access Memory
RF	Rádio Frequência
RMS	Root Mean Square
RX	Recepção
SCK	Serial Clock
SDA	Serial Data
SISO	Single Input, Single Output
TRIAC	Triode for Alternating Current
TX	Transmissão
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus

1 Introdução

1.1 Motivação

O racionamento do consumo de energia elétrica tem sido uma preocupação. Grande parte do consumo vem de aparelhos que aquecem ambientes ou que os resfriam, como chuveiros elétricos e aparelhos de ar condicionado. Todos estes equipamentos tem em comum o objetivo de propiciar o conforto térmico. Países tropicais demandam o uso de aparelhos de ar condicionado praticamente ao longo de todo ano.

O Laboratório de Automação e Robótica (LARA) está conduzindo um projeto de automação predial no contexto de ambientes inteligentes. Uma das áreas de estudo é o conforto térmico, que envolve o controle de temperatura e umidade em uma sala de reuniões e seus ambientes adjacentes.

A automação predial é um ramo da modernidade, em que o uso da tecnologia e o incentivo a pesquisas têm aumentado com o passar dos anos. Várias são as áreas de atuação, que vão desde segurança preditiva (contra assaltos, incêndios), até automatização de portas, janelas e demais aparelhos eletrônicos, como o ar-condicionado. O uso de computadores é algo corriqueiro neste tipo de trabalho, e é bastante comum organizá-los em uma rede (cabeada ou não).

No que tange ao campo de ambientes prediais, redes sem fio permitem o uso de sensores, equipamentos interligados por redes wireless e unidades de processamento associadas. As escolhas dos módulos e do padrão de comunicação são de fundamental importância. Para este trabalho, escolhemos módulos XBee: da família Zigbee (IEEE 802.15.4), são módulos de comunicação por ondas de rádio. Possuem bom alcance (30 – 100m) em ambientes fechados ou áreas urbanas, com baixo uso de bateria.

Controlar a temperatura de todos os pontos de um prédio é um trabalho excessivamente dispendioso e demorado. Muitas vezes, os controladores usados no campo da automação predial são simples, dimensionados empiricamente. Uma vez que o modelo não é bem conhecido e está exposto a perturbações, a identificação da dinâmica da planta em estudo é uma forma de obter uma caracterização que permite projetar controladores baseados em

modelos. Devemos, então, nos preocupar com a escolha das variáveis, assim como com a quantidade suficiente de sinais para estabelecer uma lógica de controle satisfatória.

O controle adaptativo tem a peculiaridade de, ao estabelecer uma lei de controle de referência, ser capaz de mudar os parâmetros do controlador em tempo real. A identificação, citada anteriormente, tem papel fundamental nesta abordagem, pois, ela observa a variação dos parâmetros da planta e, por meio, de algum tipo de realimentação, repassa essa informação ao controlador, que faz os devidos ajustes [Åström e Wittenmark, 1995]. Nota-se que esta técnica é compatível com ambientes com carga térmica tanto constante quanto em variação.

Nesse trabalho, temos dois processos térmicos em estudo. Em uma primeira etapa, modelaremos adequadamente um ambiente, por meio de identificação recursiva. Este ambiente é uma pequena sala de uma maquete, feita de madeira. O próximo passo é elaborar um controlador adaptativo. O atuador é um secador de cabelo, que aquece o ambiente da maquete. Em seguida, validaremos o controlador desenvolvido na sala de reuniões, citada anteriormente. Por fim, faremos a comparação entre diversas técnicas de controle, incluindo as clássicas, por meio da energia gasta pelo equipamento.

1.2 Conforto Térmico

O conforto térmico é uma grandeza subjetiva, que tenta quantificar a sensação humana de frio ou calor. Existem vários fatores físicos, fisiológicos e psicológicos envolvidos: estrutura corporal, resposta metabólica, diferenças de percepção, resposta a estímulos sensoriais.

Os estudos em conforto térmico visam, principalmente, analisar e estabelecer as condições necessárias para a avaliação e concepção de um ambiente térmico adequado às atividades e à ocupação humana, bem como estabelecer métodos e princípios para uma detalhada análise térmica de um ambiente [Lamberts, 2011]. Segundo a ASHRAE, o conforto térmico é “a condição da mente que expressa satisfação com o ambiente térmico”.

1.2.1 Termorregulação humana

O ser humano é um animal homotérmico, isto é, precisa manter sua temperatura constante, e por isso possui um mecanismo termorregulador que controla as suas variações térmicas. Nesse sentido, o corpo humano é aproximado a uma máquina térmica, e como tal, precisa de um mínimo de calor para operar, além de também produzi-lo. O calor necessário

para desempenharmos nossas atividades se dá pela metabolização dos alimentos ingeridos. Já o calor produzido, uma parte deste é fundamental para o funcionamento fisiológico, enquanto a outra parcela deve ser dissipada para que não ocorra um superaquecimento do corpo humano.

A Figura 1-1 exemplifica quais os tipos de trocas térmicas que podem haver entre o corpo humano e o ambiente.

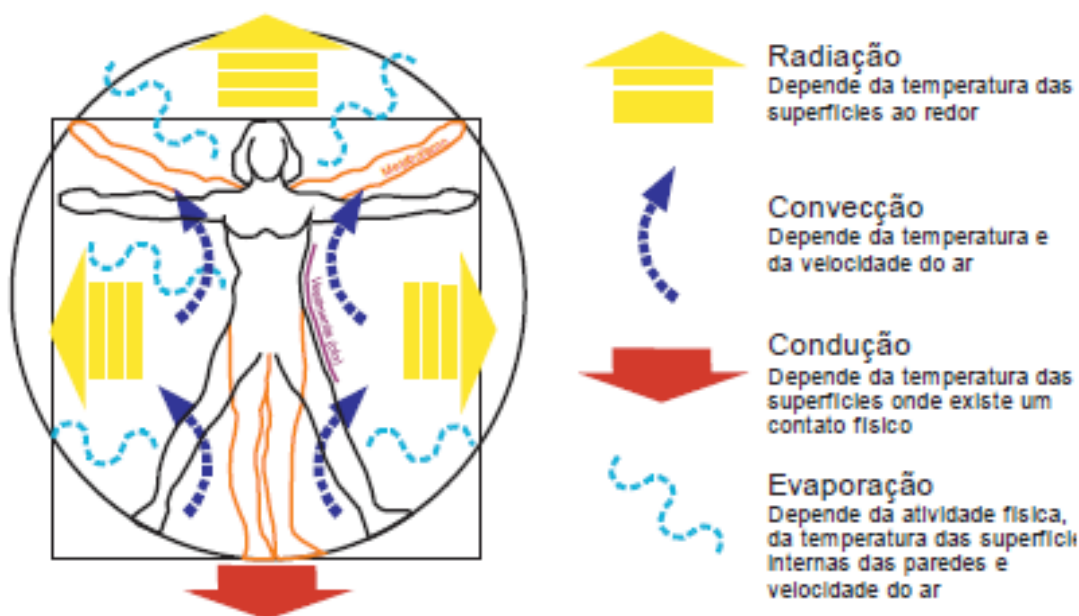


Figura 1-1. Fisiologia humana e suas trocas térmicas (Lamberts, 2011).

1.2.2 Índice PMV

Os estudos oriundos dessa área geraram vários modelos matemáticos e, portanto diversos índices que quantificassem a “agradabilidade” de um ambiente surgiram. Desta forma, criou-se um padrão: o índice PMV (*Predicted Mean Vote*). As normas ISO7730 (1994) e ASHRAE Standard 55 (2005) usam este índice e são as mais aceitas no meio acadêmico.

São considerados quatros parâmetros físicos: umidade, temperatura, temperatura radiante média e velocidade do ar. Além dessas condições ambientais, dois parâmetros individuais são levados em consideração, que são o nível de atividade da pessoa (sentada, andando, conversando) e as vestimentas.

O índice varia, em escala, de 3 a -3, em que o valor 3 equivale a um ambiente muito quente, e o valor -3, a um ambiente frio. O valor 0 corresponde ao equilíbrio ideal em se tratando de conforto térmico.

Associa-se ao PMV um índice de insatisfação com o ambiente, o PPD (*Predicted Percentage of Dissatisfied*) [Oliveira, 2010]. A Figura 1-2 explicita essa relação.

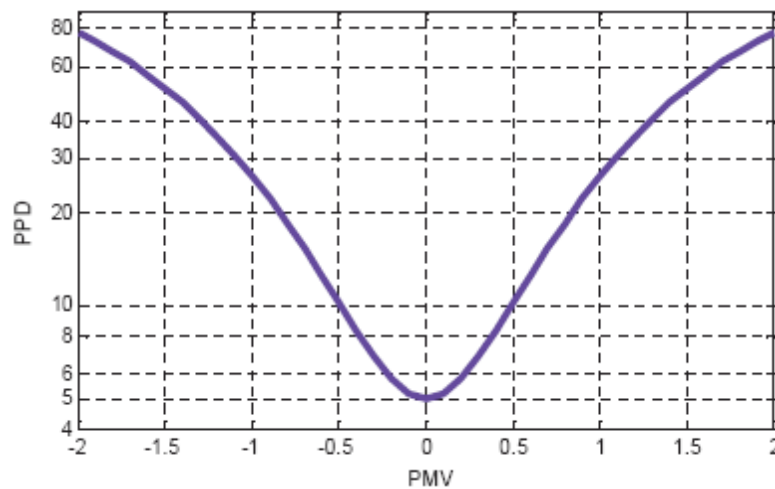


Figura 1-2. Gráfico de relação do PMV com o PPD (Oliveira, 2009).

Nosso trabalho se guiou por este estudo para avaliar quais são as variáveis importantes, e como devemos tratar a presença de pessoas em um ambiente predial.

Oliveira, 2010, mostra que a temperatura é a variável essencial para o conforto térmico. Já o ser humano possui trocas de calor complexas, como podemos ver pela Figura 1-1, e a sua presença é tratada como uma perturbação para a temperatura ambiente.

1.3 Objetivos

O objetivo deste trabalho é desenvolver uma abordagem adaptativa para o controlador de um aparelho de ar condicionado, para um ambiente predial. Como objetivos específicos, temos:

- Instrumentação dos processos (maquete e sala de reuniões do LARA);
- Identificação dos processos térmicos;
- Controle dos processo térmicos;
- Validação nos ambientes propostos, e;
- .Comparação com a abordagem adaptativa com controle PI e Liga-desliga.

1.4 Apresentação do Manuscrito

Este trabalho está organizado da seguinte forma: no capítulo 2 faremos a revisão bibliográfica sobre modelos caixa cinza, modelos de algoritmo, identificação e controle adaptativo. No capítulo 3, serão detalhados os materiais usados na instrumentação do projeto, tais como sensor, os módulos de comunicação e microcontrolador. No capítulo 4, será descrito como procedemos com a comunicação sem fio e o fluxo de dados para fazer o controle adaptativo. A seguir, no capítulo 5, apresentaremos os resultados da identificação de ambos os ambientes de estudo (sala e maquete). A performance de cada tipo de controle empregado (técnicas clássicas e o controle adaptativo) também serão apresentadas neste capítulo, sendo que faremos as devidas comparações de desempenho em termos de economia energética. No capítulo 6, é feita a conclusão a partir dos resultados obtidos, salientando as diferenças entre os ambientes de estudo e outras peculiaridades dos experimentos. Por fim, no capítulo 7, faremos uma lista de sugestões para trabalhos futuros.

2 Revisão Bibliográfica

2.1 Ambientes de Estudo

2.1.1 Sala de Reuniões do LARA

O laboratório LARA pode ser separado em seis ambientes, como pode ser visto na Figura 2-1. Um deles é a sala de reuniões, que também funciona como ambiente de estudo (Figura 2-2). Nela, está o ar condicionado que usaremos como atuador. As Figuras Figura 2-3 e Figura 2-4 mostram a localização do aparelho de ar condicionado na sala de reuniões.

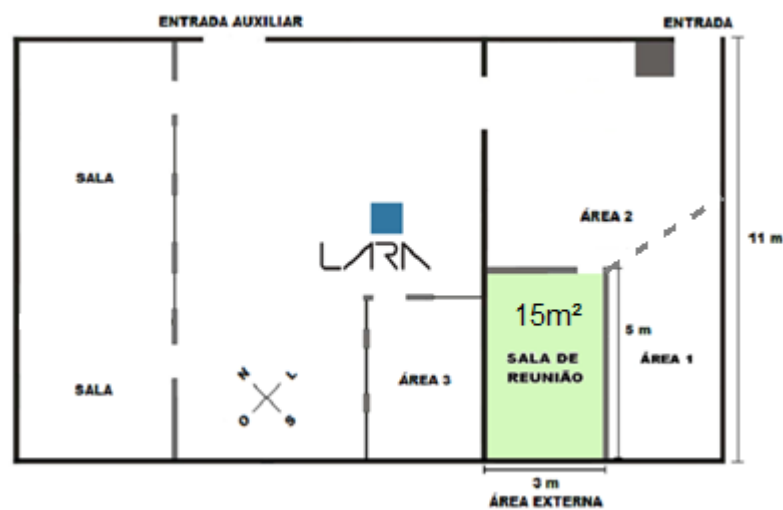


Figura 2-1. Ambiente de estudo – LARA (adaptada de Oliveira, 2010), destacando a sala de reuniões.



Figura 2-2. Vista da sala de reuniões do LARA (horário noturno, sem incidência solar).



Figura 2-3. Ar condicionado da sala de reunião (painel de controle, unidade evaporativa).

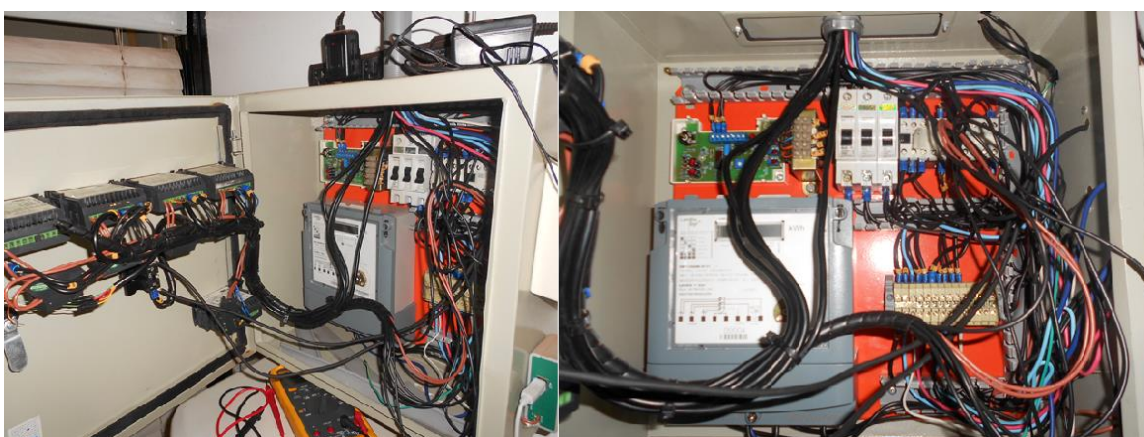


Figura 2-4. Visão do sistema de automação do ar condicionado.

Para facilitar a captação de dados o ambiente ao lado da sala de reunião, foi dividido em duas áreas (áreas 1 e 2). É importante ressaltar que, na sala de reuniões, há um bom isolamento térmico do piso, do teto e da parede que a separa da área 3. O ambiente de estudo é separado das áreas 1 e 2 por divisória simples e vidro.

2.1.2 Maquete – Protótipo da Sala

É muito raro um grupo de pessoas estarem dividindo um ambiente e não interagirem entre si. Muitas pessoas visitam a sala para fazer reunião – o objetivo principal – ou para estudar. Em ambos os casos, fica óbvio que haverá muita movimentação. Isso, ao ser incluído no modelo de sinal, é tratado como uma perturbação. Como exemplo, temos a Figura 2-5, em que a linha em vermelho retrata a perturbação de que falamos.

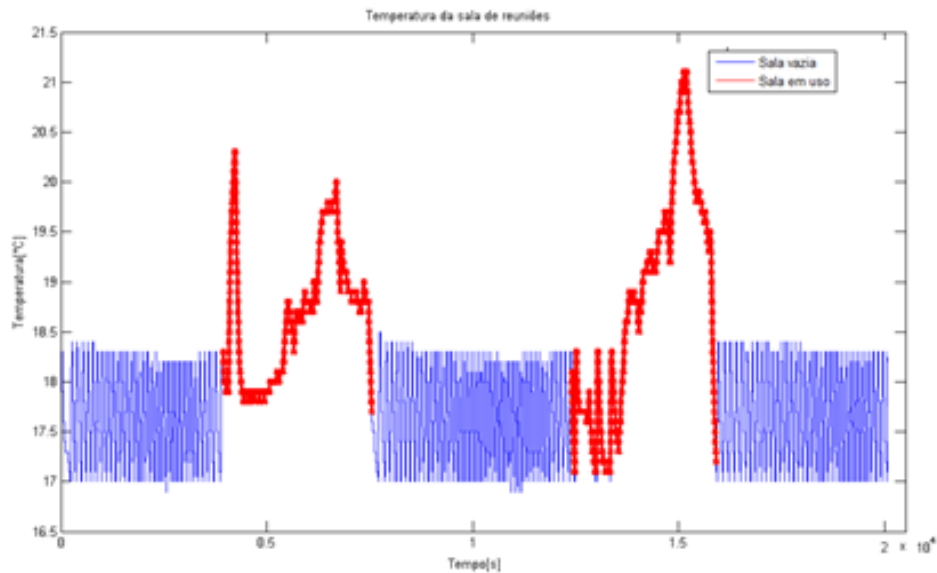


Figura 2-5. Gráfico da temperatura da sala de reuniões com acionamento ON-OFF. Em vermelho, perturbações devido ao trânsito de pessoas.

Como não podemos impedir o uso da sala, usaremos, nos estudos iniciais, uma maquete de madeira, como pode ser visto na Figura 2-6, para testarmos os controladores.



Figura 2-6. Maquete de madeira de ambiente predial.

A temperatura externa será aproximada por um secador de cabelos no duto externo da maquete. Não estamos falando de radiação, uma vez que isso foge do nosso escopo de trabalho, mas sim, do ar aquecido pelo sol em dias quentes. Vale lembrar que a umidade do ar é característica relevante, pois estamos em uma região de clima seco, mas será desprezada neste trabalho em benefício da simplicidade.

Após a fase de testes, a validação será feita no ambiente da sala de reuniões.

2.2 Modelagem Matemática

Existem duas considerações a serem feitas ao se procurar um modelo de algum processo. O modelo a ser desenvolvido é uma representação aproximada, apenas algumas características do sistema real poderão ser descritas, fazendo-se considerações simplificadoras. A outra consideração é a linearidade, ou seja, quando o sistema satisfaz a princípio da superposição. Há sistemas em que essa consideração não é adequada, mesmo para faixas de operação pequenas. Neste trabalho, assumiremos a operação linear.

2.2.1 Técnicas de Modelagem

Há várias formas de classificar técnicas de modelagem. Um delas agrupa os métodos em três categorias denominadas modelagem caixa branca, caixa preta e caixa cinza. Cada uma tem seu propósito.

- *Caixa branca*: é necessário conhecer o sistema em detalhes, bem como as leis físicas que descrevem o sistema a ser modelado. Devido ao conhecimento e ao tempo necessários para modelar um sistema partindo do equacionamento dos fenômenos envolvidos, nem sempre é viável seguir esse procedimento;
- *Caixa preta*: geralmente, possui-se pouco ou nenhum conhecimento prévio do sistema, por isso, esse tipo de modelagem também é denominado modelagem empírica. Aqui, são levantadas as relações de causa e efeito entre entradas e saídas, pois não se conhecem as equações envolvidas no funcionamento de determinado sistema, ou até são conhecidas, mas seria impraticável, por limitações de tempo e demais recursos, levantar tais equações e estimar seus respectivos parâmetros;
- *Caixa cinza*: uma alternativa entre a modelagem puramente física e a puramente empírica, usa informação auxiliar, que não se encontra no conjunto de dados utilizados

durante a identificação. O tipo de informação auxiliar e a forma com que ela é usada varia muito entre diversas técnicas disponíveis, cujo desenvolvimento é um dos grandes desafios na identificação de sistemas [Aguirre, 2007].

Para este trabalho, a modelagem caixa cinza foi escolhida, a priori. A informação auxiliar a qual a descrição deste método se refere será apresentada na sessão Fundamentação Teórica.

2.2.2 Técnicas de Algoritmo

Deve-se definir qual o método (algoritmo) a ser utilizado, que variam entre determinístico e estocástico.

- *Métodos determinísticos*: não tratam o ruído, ainda que se aceite o fato de que os dados estejam contaminados. Esse método só apresenta bons resultados quando a relação sinal/ruído é suficientemente alta.
- *Métodos estocásticos*: utilizam recursos adequados para levar em consideração o ruído e, dessa forma, reduzir seus efeitos sobre o modelo identificado.

Definir como será a captação de dados também influencia o algoritmo a ser usado.

- *Batelada*: quando toda a massa de dados está disponível antes de começar a estimar os parâmetros – o problema numérico será resolvido de uma vez só.
- *Estimação recursiva*: outra situação recorrente é de os dados serem medidos e disponibilizados sequencialmente. É claro que eles poderiam ser armazenados até ter uma quantidade suficiente, mas é possível utilizá-los enquanto a coleta é feita. É útil, principalmente, quando os parâmetros do processo variam lentamente – em função de não-linearidades, desgastes, falhas, dentre outros – ou quando a solução em batelada é dispendiosa, como a execução do algoritmo em tempo real, por exemplo [Aguirre, 2007].

Tendo em vista o contexto do projeto, é interessante fazer um controle em tempo real. Com isso, fica claro que a identificação deve ser em tempo real também. Processos em tempo real não necessariamente precisam de recursividade, supondo um sistema invariante no

tempo. Como se trata de um processo térmico, que é variante no tempo e demasiado lento, escolhemos recorrer a um algoritmo recursivo, com método estocástico.

2.3 Fundamentação Teórica

Um processo térmico tem como características a lentidão, a variação no tempo dos seus parâmetros e a necessidade de se trabalhar com múltiplas entradas e saídas. Desta forma, é interessante trabalhar com um modelo matemático simplificado, tanto para o tratamento de dados, quanto para definir quais variáveis são mais importantes.

2.3.1 Modelo da Parede

Faremos um estudo sobre as trocas de calor por meio da parede.

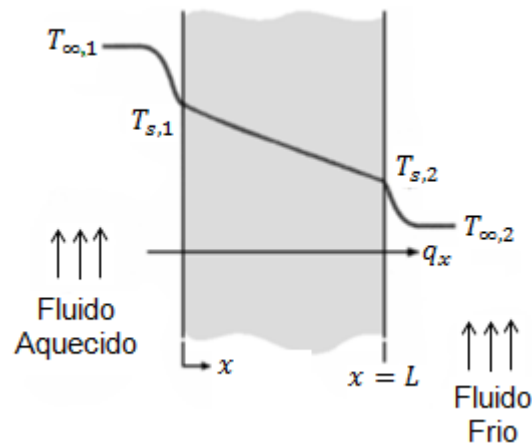


Figura 2-7. Esquemático das trocas de calor por uma parede (adaptada de Incropera e DeWitt, 2006).

Em que,

$T_{\infty,i}$ é a temperatura estacionária no ambiente i ;

$T_{s,i}$ é temperatura na superfície i ;

x é a posição ao longo da parede, e;

q_x é o vetor de transferência de calor pela parede em função da posição x .

A parede de uma sala separa dois fluidos a temperaturas distintas. O processo de trocas de calor se dá em três fases: convecção do calor do fluido à $T_{\infty,1}$ para a superfície da parede à

$T_{s,1}$, condução através da parede pelo comprimento L , e, novamente, convecção da outra superfície da parede à $T_{s,2}$ graus para o fluido de temperatura $T_{\infty,2}$.

Temos, pela lei de Fourier, que a transferência de calor por condução é:

$$q_x = -kA \frac{dT}{dx} = \frac{kA}{L} (T_{s,1} - T_{s,2}) \quad (2.1)$$

Onde,

A é a área da parede;

L é a espessura da parede, e;

k é a constante de condutividade térmica [$W/m.K$].

Já o fluxo de calor é dado por:

$$q_x'' = \frac{q_x}{A} = \frac{k}{L} (T_{s,1} - T_{s,2}) \quad (2.2)$$

As equações (2.1) e (2.2) mostram que tanto a transferência quanto o fluxo de calor são constantes e independentes da posição ao longo da parede.

A transferência de calor pelas paredes deve ser investigada. Iniciaremos com um modelo simples, 2R1C, modelo de primeira ordem, com características lineares. É bom lembrar que, para a consideração de linearidade, definiremos um ponto de operação do sistema. Nesse modelo, a parede é modelada por uma capacitância interna térmica e dois elementos de condutividade térmica R .

Existe um conceito amplamente usado em modelagem de ambientes denominado resistência térmica. Trata-se de recursos matemáticos baseados na analogia entre a difusão do calor e a carga elétrica (potencial elétrico e de transmissão de calor, corrente e transferência de calor), de modo a descrever um sistema térmico como um circuito elétrico.

Da mesma maneira que se associa uma resistência elétrica com a condução de eletricidade, associa-se uma resistência térmica com a condução de calor.

A resistência elétrica, pela lei de Ohm, é definida como a razão entre tensão (diferença de potencial elétrico) e corrente. A resistência térmica *por condução* é, portanto, a razão entre a

diferença de temperatura entre dois pontos e a transferência de calor – veja a equação (2.3), logo abaixo:

$$R_{t,cond} = \frac{T_{s,1} - T_{s,2}}{q_x} = \frac{L}{kA} \quad (2.3)$$

É preciso definir a resistência térmica *por convecção*. Usaremos a lei de resfriamento de Newton, para definir a transferência de calor, ou seja:

$$q = hA(T_s - T_\infty) \quad (2.4)$$

E, assim:

$$R_{t,conv} = \frac{T_s - T_\infty}{q} = \frac{1}{hA} \quad (2.5)$$

Voltando à questão do modelo da parede, há duas considerações a se fazer:

- Pelo princípio da conservação de energia, assim como na lei de nós de *Kirchhoff*, a transferência de calor ao longo da parede é constante, ou seja:

$$q_x = \frac{T_{\infty,1} - T_{s,1}}{1/h_1A} = \frac{T_{s,1} - T_{s,2}}{L/kA} = \frac{T_{s,1} - T_{\infty,2}}{1/h_2A} \quad (2.6)$$

- Em termos de diferença de temperatura, $T_{\infty,1} - T_{\infty,2}$, podemos definir uma resistência total R_{tot} , a qual será a soma de todas as resistências térmicas na malha, uma vez que elas estão em série. A equação (2.7) expressa o valor de R_{tot} :

$$R_{tot} = 1/h_1A + L/kA + 1/h_2A \quad (2.7)$$

A Figura 2-8 a seguir ilustra o conceito de resistência térmica aplicado à modelagem da parede:

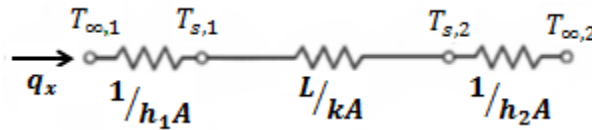


Figura 2-8. Malha representativa (modelo) da parede (adaptada de Incropera e DeWitt, 2006).

A equação (2.8) abaixo descreve a transferência de calor entre as salas vizinhas:

$$\frac{T_o}{T_{iv}} = \frac{1}{sRC_v + 1} \quad (2.8)$$

Vemos que se trata de um sistema dinâmico, cuja função de transferência é de 1º. grau, e a constante de tempo é $\tau = RC_v$, no caso, representa o tempo necessário para atingir o equilíbrio térmico entre duas vizinhanças. O resfriamento promovido pelo ar condicionado também tem a mesma estrutura da equação (2.8), e a constante de tempo associada retrata o efeito de resfriamento da sala.

As características físicas da parede, do piso e do teto são diferentes entre si e relevantes para o problema. Para efeitos de simplificação, considera-se que o modelo 2R1C na Figura 2-9 é suficiente para representar a dinâmica dominante.

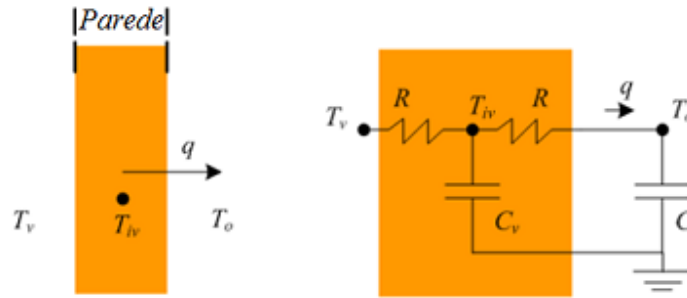


Figura 2-9. Modelo 2R1C da parede (adaptada de Bauchspiess et al., 2006).

O ponto T_{iv} retrata um ponto fictício no meio da parede. A diferença entre T_v e T_o define o vetor q , definindo o fluxo de calor entre duas vizinhanças. Os parâmetros C_v e R são intrínsecos das características da parede, tais como material, suas dimensões e seu formato.

2.3.2 Identificação Recursiva

Um algoritmo recursivo de identificação, por definição, estima os parâmetros de um modelo enquanto os dados são disponibilizados. Essa abordagem é útil quando o processo evolui lentamente.

$$y(k) = \psi_k^T(k-1)\theta_k + \xi(k) \quad (2.9)$$

Na equação (2.9), temos a saída $y(k)$ mensurada no instante k como a relação entre a matriz de regressores ψ_k^T , baseada nas $k-1$ medidas anteriores, o vetor de parâmetros medidos θ_k , e o resíduo $\xi(k)$, parcela do modelo que engloba imprecisões (diferenças entre a estrutura presumida e a real) e ruído.

Usando a implementação recursiva do método dos Mínimos Quadrados (MQ) tradicional, temos o vetor de parâmetros de mínimos quadrados descrito pela equação (2.10), a seguir:

$$\boldsymbol{\theta}_k = \left[\sum_{i=1}^k \psi(i-1)\psi^T(i-1) \right]^{-1} \left[\sum_{i=1}^k \psi(i-1)y(i) \right] \quad (2.10)$$

Para garantir que $\boldsymbol{\theta}_k$ seja uma boa estimativa, existem duas restrições importantes: a covariância de $\boldsymbol{\theta}_k$ deve ser mínima, e $\boldsymbol{\theta}_k$ não pode ser polarizado ($E[\boldsymbol{\theta}_k] - \boldsymbol{\theta} = 0$).

A matriz $P_k = \text{cov}[\boldsymbol{\theta}_k] \in R^{n_\theta \times n_\theta}$ denota a matriz de covariância obtida até a interação k . Vale lembrar que, no algoritmo MQ, é interessante expressar as grandezas em termos de estimações passadas. Desta forma, temos (seguindo a apresentação de Aguirre [2007]):

$$\begin{aligned} P_k &= \left[\sum_{i=1}^k \psi(i-1)\psi^T(i-1) \right]^{-1} \\ P_k^{-1} &= \left[\sum_{i=1}^k \psi(i-1)\psi^T(i-1) \right] + \psi(i-1)\psi^T(i-1) \\ &= P_k^{-1} + \psi(i-1)\psi^T(i-1) \end{aligned} \quad (2.11)$$

Alterando a parcela $\psi(k-1)$ para ψ_k , a equação (2.10) pode ser reescrita como:

$$\boldsymbol{\theta}_k = P_k \left[\sum_{i=1}^{k-1} \psi(i-1)y(i) + \psi(k-1)y(k) \right] \quad (2.12)$$

Ou ainda como:

$$\left[\sum_{i=1}^{k-1} \psi(i-1)\psi^T(i-1) \right] \boldsymbol{\theta}_{k-1} = \left[\sum_{i=1}^{k-1} \psi(i-1)y(i) \right] \quad (2.13)$$

O lado esquerdo da equação (2.13) pode ser representado por $P_{k-1}^{-1}\boldsymbol{\theta}_{k-1}$. Substituindo este resultado na equação (2.12), temos:

$$\begin{aligned} \boldsymbol{\theta}_k &= P_k \left[P_{k-1}^{-1}\boldsymbol{\theta}_{k-1} + \psi(k-1)y(k) \right] \\ &= P_k \left[(P_{k-1}^{-1} - \psi(k-1)\psi^T(k-1))\boldsymbol{\theta}_{k-1} + \psi(k-1)y(k) \right] \\ &= \boldsymbol{\theta}_{k-1} - P_k\psi(k-1)\psi^T(k-1)\boldsymbol{\theta}_{k-1} + P_k\psi(k-1)y(k) \\ &= \boldsymbol{\theta}_{k-1} - P_k\psi(k-1)[y(k) - \psi^T(k-1)\boldsymbol{\theta}_{k-1}] \\ &= \boldsymbol{\theta}_{k-1} - K_k[y(k) - \psi^T(k-1)\boldsymbol{\theta}_{k-1}] \end{aligned} \quad (2.14)$$

A outra premissa para um bom estimador, relativa a não polarização dos dados, é dimensionada pela matriz K_k . Como pudemos ver, $K_k = P_k \psi(k-1)$.

Para usarmos a recursividade do algoritmo, é preciso calcular P_k . O uso direto da equação (2.11) mostra que é preciso calcular a inversa de uma matriz, o que é dispendioso, mesmo usando um computador. Aplicando o lema da inversão, é possível encontrar uma equação para P_k melhorada:

$$P_k = P_{k-1} - \frac{P_{k-1} \psi(k-1) \psi^T(k-1) P_{k-1}}{\psi^T(k-1) P_{k-1} \psi(k-1) + 1} \quad (2.15)$$

Desta forma, K_k pode ser reescrita como:

$$K_k = \frac{P_{k-1} \psi(k-1)}{\psi^T(k-1) P_{k-1} \psi(k-1) + 1} \quad (2.16)$$

Existe, porém, uma característica a ser levada em conta: o estimador trata todas as medições de forma idêntica, isto é, o peso associado à primeira medida w_1 é o mesmo que o da centésima, w_{100} , e este é o mesmo da milésima medida, w_{1000} . Para um sistema invariante no tempo, isto é razoável, caso contrário, não o é. É preciso, portanto, inserir plasticidade ao algoritmo, ou seja, ponderar de maneira diferenciada as observações. Uma forma de obter isto é utilizando um decaimento exponencial (2.17), a seguir:

$$\begin{cases} w_k(k) = 1 \\ w_i(k) = \lambda w_i(k-1), i < k \end{cases} \quad (2.17)$$

O fator λ representa a razão entre pesos consecutivos para uma mesma massa de dados e é conhecido como fator de esquecimento.

O estimador recursivo de mínimos quadrados com fator de esquecimento λ é mostrado na equação (2.18) [Aguirre, 2007]:[Aguirre, 2007]

$$\begin{cases} K_k = \frac{P_{k-1} \psi_k}{\psi_k^T P_{k-1} \psi_k + \lambda}; \\ \theta_k = \theta_{k-1} + K_k [y(k) - \psi_k^T \theta_{k-1}]; \\ P_k = \frac{1}{\lambda} \left(P_{k-1} - \frac{P_{k-1} \psi_k \psi_k^T P_{k-1}}{\psi_k^T P_{k-1} \psi_k + \lambda} \right). \end{cases} \quad (2.18)$$

Em que:

ψ_k é o vetor de regressores que contém informação até o instante $(k-1)$;

K_k é o ganho de adaptação do vetor estimado, muitas vezes referido como o *ganho de Kalman*, e garante a não polarização do vetor de parâmetros (o filtro de Kalman é, de fato, uma implementação eficiente do MQR [Aguirre, 2007]);

P_k é a matriz de covariância do processo, e;

θ_k é o vetor de parâmetros estimado pelas $(k-1)$ medidas [Aguirre, 2007].

2.3.3 Controle Adaptativo

Um controlador adaptativo é aquele que pode modificar seu comportamento de acordo com a variação da dinâmica do processo. Este tipo de controlador é caracterizado por possuir um mecanismo de ajuste que faz com que a lei de controle se adapte a estas variações, como pode ser visto na Figura 2-10.

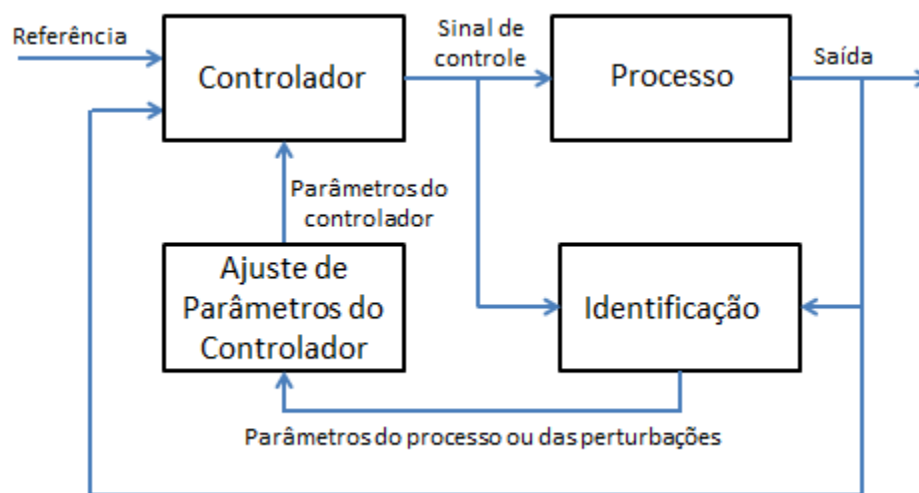


Figura 2-10. Esquemático geral de controle adaptativo.

O diagrama presente na Figura 2-10 possui uma estrutura complexa. Cada ponto pode ser realizado de várias formas distintas. Nota-se que controle adaptativo é mais um conceito do que uma técnica propriamente dita. Para fins de simplificação, pode-se dizer que existem duas abordagens típicas: controle direto e indireto.

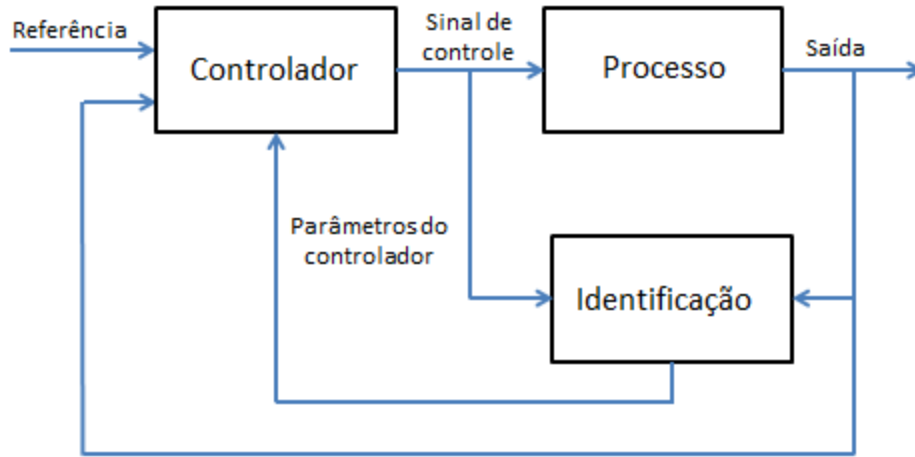


Figura 2-11. Esquemático de controle adaptativo direto.

No controle indireto, primeiramente os parâmetros do processo são estimados de forma recursiva e com estes parâmetros são calculados os parâmetros do controlador, como visto na Figura 2-10. No controle direto, o processo é re-parametrizado, isto é, o estimador é utilizado para obter os parâmetros do controlador a partir das medidas de entrada e saída do sistema, conforme pode ser visto na Figura 2-11. Em ambos os algoritmos, existe a ideia básica de identificar alguns parâmetros que estejam relacionados ou ao processo, ou às especificações para o sistema em malha fechada, e, preferencialmente, aos dois. Vê-se que a identificação assume um papel importante.

Existem várias técnicas para o projeto de um controlador adaptativo: modelo de referência (MRAC), reguladores auto ajustáveis (RST), controle dual, escalonamento de ganho, entre outros. Neste trabalho, foi utilizada a técnica de reguladores auto ajustáveis, detalhada logo a seguir.

Assumindo o modelo do processo como ARMAX SISO (*single-input, single-output*), tem-se:

$$A(q)y(k) = B(q)(u(k) + v(k)) \quad (2.19)$$

$$y(k) = \frac{B(q)}{A(q)}u(k) + \frac{B(q)}{A(q)}v(k) \quad (2.20)$$

Onde A e B são polinômios, q é o operador de atraso, e v é uma perturbação ao sistema.

Deseja-se encontrar um controlador com a seguinte forma:

$$R(q)u(k) = T(q)y_{ref}(k) - S(q)y(k) \quad (2.21)$$

Onde R, S e T são polinômios. Logo, em malha fechada, temos:

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

$$u(k) = \frac{A(q)T(q)}{A(q)R(q)+B(q)S(q)}y_{ref}(k) + \frac{B(q)S(q)}{A(q)R(q)+B(q)S(q)}v(k) \quad (2.23)$$

A equação característica, que neste caso é conhecida como equação diofantina, é:

$$A(q)R(q) + B(q)S(q) = A_c(q) \quad (2.24)$$

Onde A_c é o polinômio característico desejado em malha fechada.

Para obter o polinômio T é necessária mais uma relação. Esta relação é obtida através de um modelo de referência que descreve a dinâmica desejada para o sistema.

$$A_my_m(t) = B_mu_c(t) \quad (2.25)$$

$$\frac{BT}{AR+BS} = \frac{BT}{A_c} = \frac{B_m}{A_m} \quad (2.26)$$

O polinômio B pode ser fatorado como B^+B^- , tal que B^+ é mônico, estável e tão bem condicionado que pode ser cancelado pelo controlador. Já a parcela B^- não pode ser cancelada e, portanto, ela faz parte do polinômio B_m .

$$B_m = B^-B'_m \quad (2.27)$$

Já que B^+ é cancelado, deve ser um fator de A_c . Com isso, também temos que A_m também é fator de A_c . O polinômio característico tem a forma:

$$A_c = A_0A_mB^+ \quad (2.28)$$

Podemos fatorar o polinômio R também, como sugere a equação (2.29) abaixo.

$$R = R'B^+ \quad (2.29)$$

Desta forma, a equação diofantina passa a ser:

$$AR' + B^-S = A_0A_m = A'_c \quad (2.30)$$

E, assim, temos que:

$$T = A_0B'_m \quad (2.31)$$

Existem condições de causalidade, as quais são:

$$\begin{cases} \text{grau de } A_m = \text{grau de } A \\ \text{grau de } B_m = \text{grau de } B \\ \text{grau de } A_0 = \text{grau de } A - \text{grau de } B^+ - 1 \\ \text{grau de } S < \text{grau de } A \end{cases} \quad (2.32)$$

Para eliminar o erro em regime permanente, é proposto em Åström e Wittenmark (1995):

$$R^*(q^{-1})\Delta^*u(k) + S^*(q^{-1})\Delta^*y(k) + A_0(1)A_m(1)y(k) = A_0^*(q^{-1})A_m(1)y_{ref}(k) \quad (2.33)$$

Em que a dedução e o significado de cada termo de (2.33) podem ser encontrados em sua respectiva obra.

3 Materiais

3.1 Sensor de temperatura e umidade SHT71

O SHT71 (Figura 3-1) é um sensor de temperatura e umidade da empresa Sensirion®, que possui comunicação digital, evitando assim que o sinal sofra interferência durante o envio das leituras, feitas por meio de um protocolo de comunicação proprietário. A tensão de alimentação pode variar de 2,4 V a 5,5 V. O sensor possui um módulo de memória de calibração que já vem configurado de fábrica.

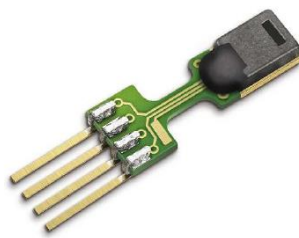


Figura 3-1. Sensor SHT71 (Sensirion A. G. ,2009).

O conversor A/D (Figura 3-3) possui uma resolução configurável, podendo variar de 8 a 14 bits. O sensor possui, ainda, um circuito para a interface serial entre o sensor e o microcontrolador. A Figura 3-2, logo abaixo, ilustra este tipo de circuito.

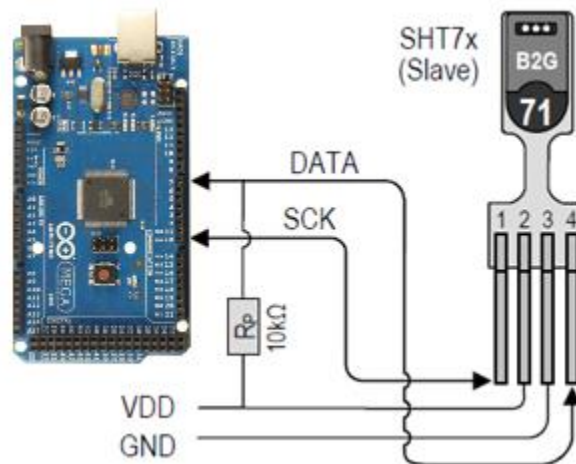


Figura 3-2. Típico circuito de aplicação, incluindo o resistor de pull-up (adaptada de Sensirion A. G. ,2009).

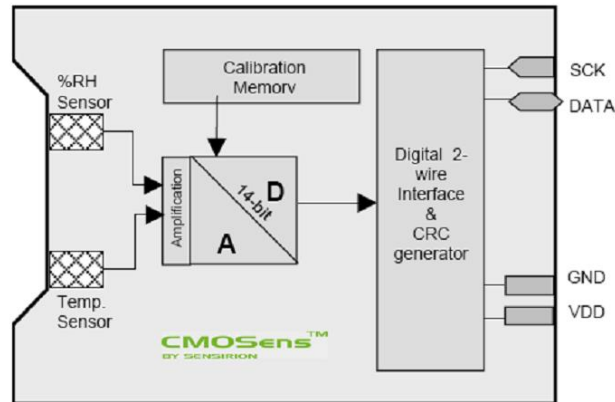


Figura 3-3. Diagrama de blocos do sensor SHT71 (Sensirion A. G., 2009).

3.1.1 Protocolo de Transferência de Dados

O protocolo de comunicação é composto por varias etapas. O sinal SCK é utilizado na sincronização da comunicação entre o sensor e o microcontrolador (Figura 3-4). O inicio da comunicação se dá com o envio do endereço de três bits “000”, que é o único permitido. O comando desejado é formado por cinco bits. Ao receber o sinal de reconhecimento (*acknowledge*), o SHT se prepara para mudar o estado para o valor lógico baixo, o que acontece após a oitava borda de descida do SCK. A linha é liberada após a nona borda de descida de SCK.

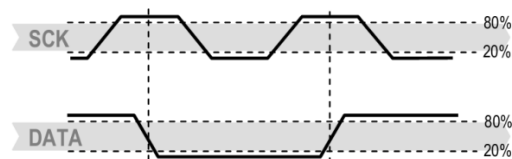


Figura 3-4. Início da transmissão (Sensirion A. G., 2009).

O microcontrolador tem de esperar a medição ser concluída, o que leva no máximo 20/80/320 ms, para uma resolução de 8/12/14 bits no ADC. É feita a leitura, em que são enviados 2 bytes de medida e um de CRC (*checksum*) opcional. O microcontrolador reconhece cada byte mudando DATA para o valor lógico baixo. Com isto, é feito o fim da comunicação, logo após o byte CRC (caso exista) ou após o byte menos significativo da medida.

Um exemplo da rotina de comunicação pode ser visto na Figura 3-5, que trata da medição da umidade relativa pelo sensor.

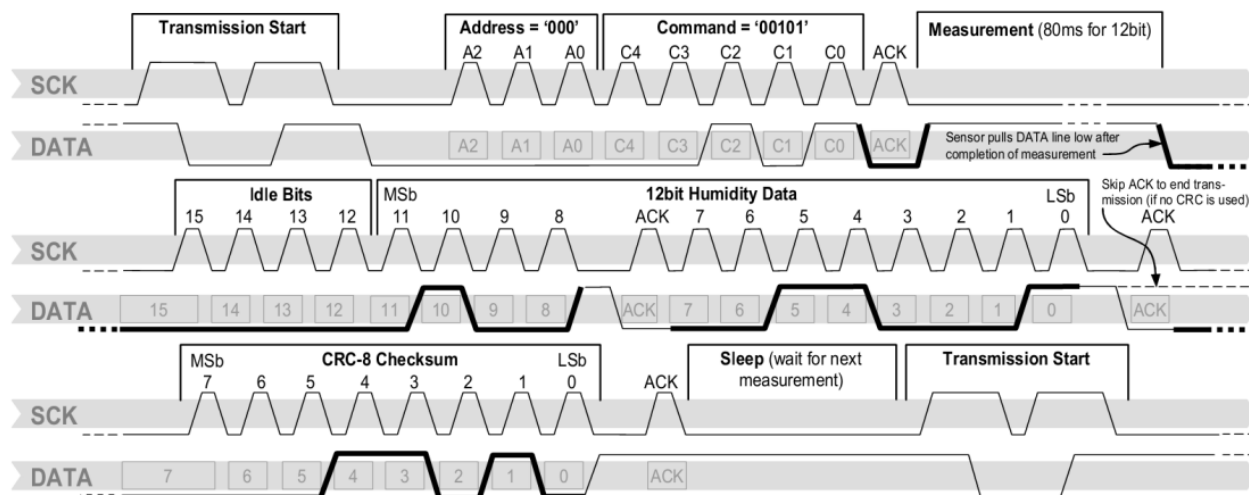


Figura 3-5. Medição e transmissão de dados – comando 00101, umidade relativa (Sensirion A. G. , 2009).

3.1.2 Especificações Técnicas

A Tabela 3-1 mostra uma lista de comandos e códigos para operar o SHT71.

Tabela 3-1. Lista de comandos.

Comando	Código
Medir temperatura	00011
Medir umidade	00101
Ler registrador de status	00111
Escrever registrador de status	00110
Soft reset	11110

Após o dado ser recebido pelo microcontrolador, é realizada a conversão do sinal de saída conforme explicitado na equação (3.1):

$$T = d_1 + d_2 SO_T \quad (3.1)$$

Em que d_1 e d_2 são dados pelas seguintes tabelas:

Tabela 3-2. Coeficiente de conversão de temperatura de acordo com a alimentação no sensor.

<i>VDD</i>	<i>d₁(°C)</i>	<i>d₁(°F)</i>
5 V	-40,1	-40,2
4 V	-39,8	-39,6
3.5 V	-39,7	-39,5
3 V	-39,6	-39,3
2.5 V	-39,4	-38,9

Tabela 3-3. Coeficiente de conversão de temperatura de acordo com a resolução.

<i>SO_T</i>	<i>d₂(°C)</i>	<i>d₂(°F)</i>
14 bits	0.01	0.018
12 bits	0.04	0.072

A Tabela 3-4 resume as principais características do sensor SHT71:

Tabela 3-4. Dados típicos do sensor em relação à temperatura.

<i>Parâmetro</i>	<i>Mínimo</i>	<i>Tipo</i>	<i>Máximo</i>	<i>Unidade</i>
<i>Resolução</i>	0,4 12	0,01 14	0,01 14	°C Bit (ADC)
<i>Precisão</i>	-	±0,4	-	°C
<i>Faixa de operação</i>	-40	-	123	°C

3.2 XBee

XBee (Figura 3-6) são módulos de radiofrequência, desenvolvidos pela Digi International. Possuem como principal característica a implementação de sistemas de baixo custo e baixo consumo de energia para diversas aplicações.

Para a comunicação entre os módulos é utilizado o protocolo IEEE 802.15.4, também conhecido como ZigBee™. A grande parte das famílias de XBee possui alguma forma de controle de fluxo, I/O's, conversores A/D e indicadores de potência de sinal.

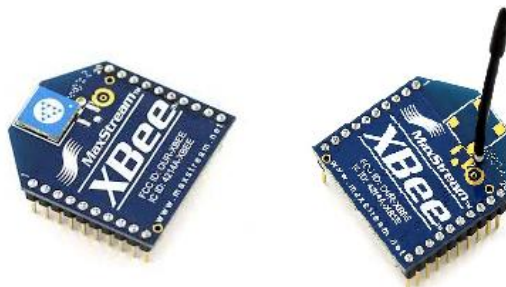


Figura 3-6. Módulos de comunicação XBee.

3.2.1 Modos de Comunicação

A comunicação com o dispositivo controlador ocorre de forma serial assíncrona (Figura 3-7), sendo assim adaptável com qualquer UART compatível a nível lógico e de tensão, ou fazendo o uso de um tradutor de nível lógico.

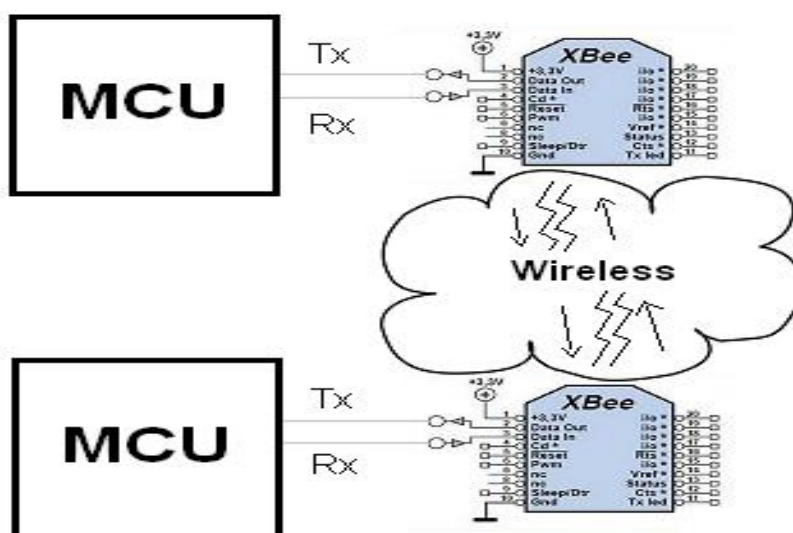


Figura 3-7. Esquemático da comunicação entre módulos (pyroelectro.com).

O XBee opera em modo transparente por padrão. Neste modo, os módulos funcionam como um substituto de uma linha serial, ou seja, todo dado recebido é colocado no *buffer* para a transmissão RF. Quando um dado é recebido por RF, este é colocado no *buffer* de recepção.

A Figura 3-8 mostra essa relação de armazenamento em buffers e transmissão de dados, descrita anteriormente.

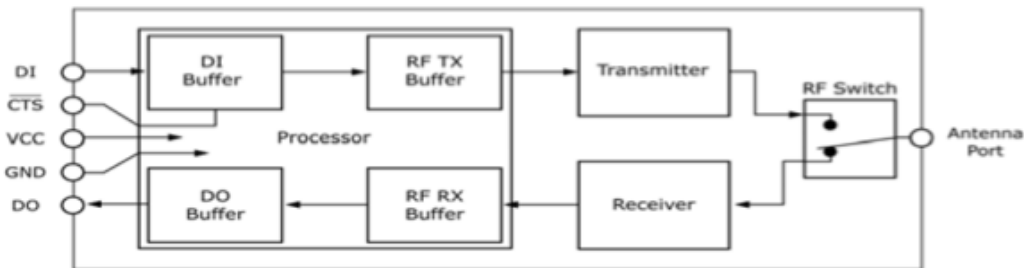


Figura 3-8. Fluxo de dados no XBee.

3.2.2 Adaptador CON-USBBEE ROGERCOM

O adaptador COM-USBBEE (Figura 3-9) desenvolvido pela ROGERCOM é utilizado para facilitar a interface entre o módulo XBee e o computador. Este adaptador utiliza um conversor USB/Serial (FTDI Chip) que cria uma porta serial virtual no computador, permitindo assim, a comunicação entre o computador e o XBee.



Figura 3-9. Adaptador COM-USBBEE (rogercom.com.br).

No adaptador, é encontrado um regulador de tensão responsável pela alimentação do XBee, LEDs que indicam a força do sinal RF (RSSI), LEDs que indicam a recepção (RX) ou transmissão (TX) de um dado, um LED que indica a associação a uma rede (ASS), além de um botão de *reset*. Este adaptador foi utilizado para a configuração dos parâmetros dos XBees e para a conexão destes com o MATLAB.

3.3 Arduino

Arduino é uma plataforma de prototipagem de *hardware* baseada em uma placa com um microcontrolador Atmel AVR e uma IDE de desenvolvimento de *software*. O seu *hardware* possui diversas entradas e saídas de fácil acesso, cuja quantidade varia de acordo com o

modelo da placa. A interface de desenvolvimento de *software* é baseada em um compilador de C/C++ e em um *bootloader* – um programa que é carregado no microcontrolador, permitindo que o código criado seja gravado no microcontrolador sem o auxílio de um *hardware* adicional.

Neste trabalho, foi utilizado o modelo Arduino MEGA, como visto na Figura 3-10. Este modelo possui um microcontrolador ATmega1280, que possui 54 pinos de entrada e saída digitais, dos quais 14 deles podem ser usados como saídas de PWM, 16 entradas analógicas, 4 UARTs, um cristal de 16 MHz, 128 kB de memória flash e 8 kB de memória RAM. A placa possui conexão USB, um conector de força, um conector ICSP e um botão de *reset*.



Figura 3-10. Hardware do Arduino (arduino.cc).

3.3.1 Módulo de Expansão

É possível acoplar ao Arduino uma placa chamada de *shield*, cuja função é estender suas funcionalidades. Desta forma, foi utilizado um *shield*, visto na Figura 3-11, para acoplar módulos de comunicação XBee, facilitando a integração entre este e o Arduino.



Figura 3-11. Placa *shield* associada ao módulo de expansão XBee (arduino.cc).

3.4 Relé de Estado Sólido - T2405Z-M

Para fazer acionamento de dispositivos, é comum usar uma chave estática. Elas promovem o chaveamento da potência fornecida para a carga, mas não a modifica em nenhum outro aspecto. O uso de relés é algo corriqueiro neste sentido.

Comumente, existem dois tipos de relés: os eletromecânicos e os de estado sólido. Estes são uma alternativa moderna para o uso de relés eletromecânicos, uma vez que são mais robustos quanto a ruídos e problemas mecânicos. Vale lembrar que o uso de relés de estado sólido é um pouco restrito, pois, não é toda aplicação que é possível usá-los.

O SSR T2405Z-M (Figura 3-12) da Teletrom é um relé de estado sólido, cuja tensão de alimentação, chamada de tensão de comando, pode variar de 3 V a 32 V DC. A tensão de saída, ou tensão de carga, é de 240 V AC, com corrente de saída de até 5 A. Vê-se que um dispositivo de pequeno porte, como o XBee, pode controlar este relé, e o relé pode atuar junto a contadoras ou controladores mais exigentes quanto ao acionamento.



Figura 3-12. Relé T2405Z-M (teletronic.ind.br).

Em um SSR, tem-se um dispositivo semicondutor que ao ser excitado, emite radiação infravermelha. Essa radiação atua sobre o receptor do dispositivo, o qual controla um circuito externo, através de um dispositivo de potência como, por exemplo, um transistor de potência, um SCR, um TRIAC, etc. Para mais detalhes, veja a Figura 3-13.

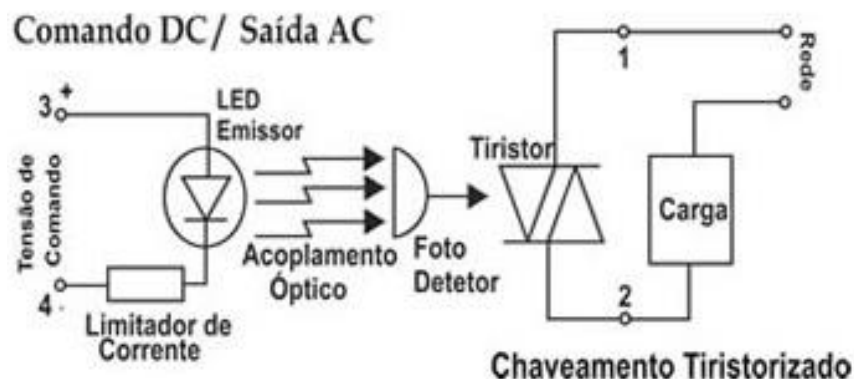


Figura 3-13. Circuito interno ao relé T2405Z-M (teletronic.ind.br).

3.4.1 PWM – Acionamento do Relé

Modulação por largura de pulso, ou mais conhecida como *pulse-width modulation* (PWM), é uma técnica de modulação em que a largura do pulso é adequada ao modulador de sinal de informação. Ela pode ser usada em sistemas de comunicação, mas seu principal uso é controlar o fornecimento de potência a um dispositivo elétrico. Para este uso, sua principal vantagem é a baixa perda de potência durante a transmissão.

O valor da tensão ou corrente a alimentar a carga é controlado por um pulso que varia de 0 (não condução) a 1 (condução) ao longo do tempo. A razão entre o tempo em que o pulso está conduzindo (valor 1) e o tempo total do pulso é denominada ciclo de trabalho (*duty cycle*, em inglês). Esta razão determina a tensão, ou corrente, média e define a potência entregue à carga.

Neste trabalho, foi usado um PWM junto ao sinal de controle, para a modulação do acionamento do relé. O período do PWM foi ajustado empiricamente, e a Tabela 3-5 mostra o seu valor para cada experimento feito ao longo deste trabalho.

Tabela 3-5. Valores de período do PWM, considerando tipo de controle e ambiente.

Controle	Sala de Reuniões do LARA	Maquete
Adaptativo	220 segundos	9 segundos
PI	220 segundos	19 segundos

4 Procedimentos

4.1 Rede XBee e Transmissão de Dados

É preciso confeccionar uma rede de, no mínimo, 3 nós: um nó para transmitir os dados captados pelo sensor, outro para fazer a comunicação com um computador, que faça a identificação e elabore a re-parametrização do controlador, e mais um nó, que atue no relé T2405Z-M. Não há exatamente um tamanho máximo para a rede, mas é importante lembrar que, quanto maior, mais complexa ela é. É importante, portanto, verificar as limitações dos módulos de comunicação (canais disponíveis para transmissão e/ou recepção de dados), definir como a rede deve operar (sincronismo, simultaneidade), definir a sua área de cobertura (uma sala, um andar ou um prédio), considerar modificações no algoritmo (do microcontrolador, de identificação, de controle), entre outras peculiaridades.

Para este trabalho, uma pequena rede foi montada. O primeiro nó é formado pelo SHT71, pelo XBee e pelo Arduino associado a um *shield*. O sensor, à resolução de 14 bits, interage com o XBee por meio do protocolo I²C. O protocolo de comunicação em 2 sinais I²C se destaca por não precisar de interfaces particulares, uma vez que os dispositivos integram as interfaces “on-chip”, o endereçamento e o protocolo de transferência de dados são totalmente definidos via software, entre outros benefícios. Suas principais características são: duas vias de comunicação, serial data (SDA) e serial clock (SCL), ambas bidirecionais, conectadas ao positivo da fonte de alimentação através de um resistor de pull-up; todo dispositivo possui um endereço único no barramento, independente de sua natureza; o número de interfaces conectadas fica dependente da capacitância máxima do barramento, que é de 400 pF.

Voltando à rede, o segundo nó é responsável por repassar as medições de temperatura para o computador, em que há uma instância ativa do MatLab®, software de alta performance voltado para cálculo numérico. O XBee, através de um adaptador USB, passa os dados via porta serial ao computador. Por último, temos o terceiro nó, composto por um Arduino e um *shield*, por outro módulo XBee, e pelo relé de estado sólido, conectado diretamente ao XBee. Como o relé é um dispositivo analógico, só é preciso mandar sinais de nível lógico alto ou baixo para ativá-lo.

A comunicação entre os módulos XBee é feita por radiofrequência.

A Figura 4-1 mostra um esquemático da rede descrita.

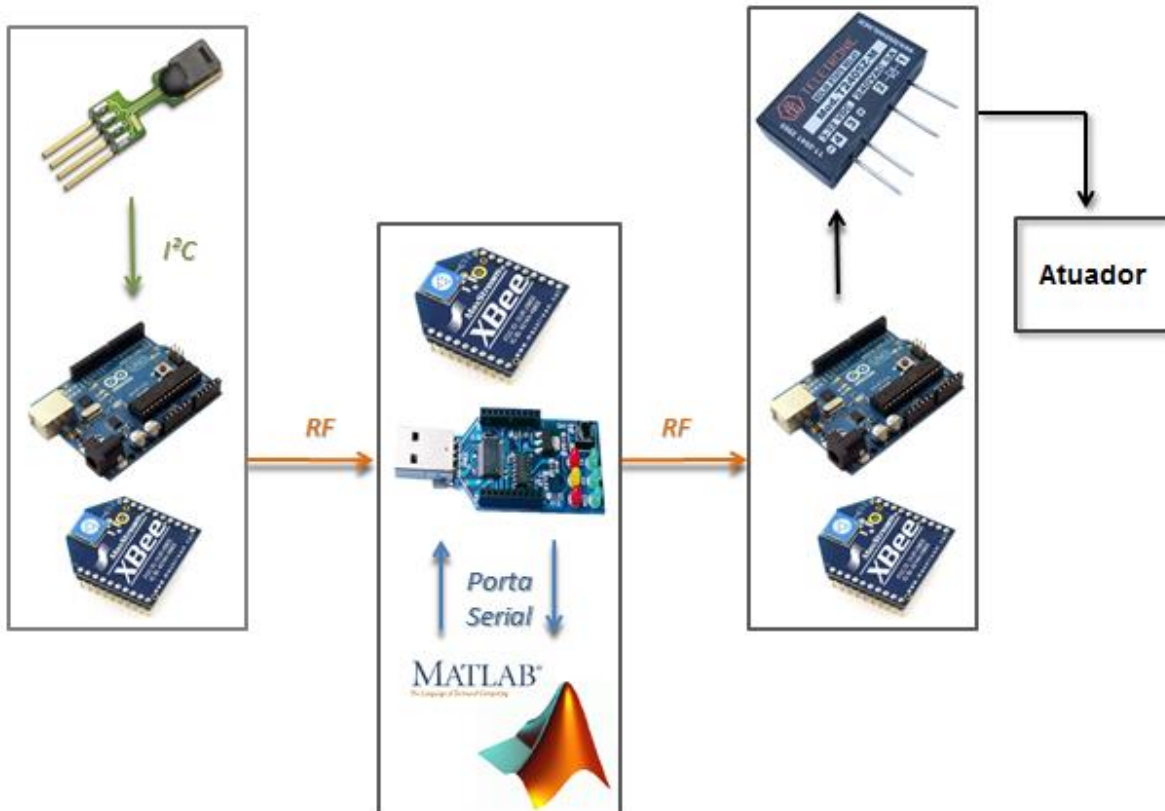


Figura 4-1. Esquemático da mini rede XBee para identificação e controle.

4.2 Abordagens de Controle

Foram utilizadas diferentes técnicas de controle, incluindo as clássicas. Para o controle adaptativo, usamos a técnica de alocação de polos (RST), como explicado no tópico Controle Adaptativo, da Fundamentação Teórica. A Figura 4-2 resume o diagrama de controle que propusemos para o projeto. Cada tipo de controle (ON-OFF, PI e RST) foi executado individualmente.

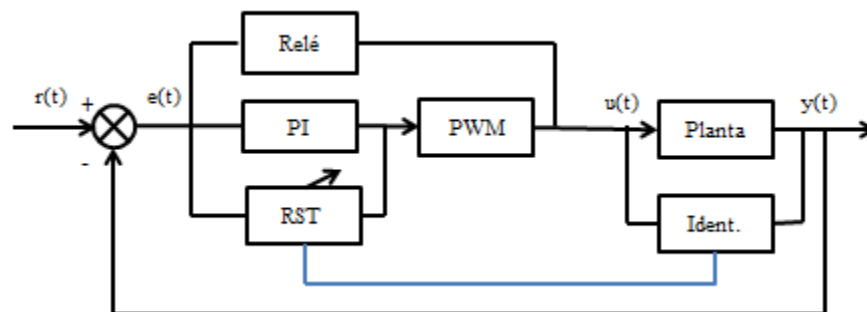


Figura 4-2. Sistemas de controle aplicados para o processo térmico.

4.2.1 Controle Liga-desliga

O controlador liga-desliga é simples e rápido. Leva em conta o erro entre a referência e a temperatura medida, representado na Figura 4-2 pelo relé. A lei de controle é determinada por:

- Para a maquete:

$$\begin{cases} 1 & \text{se } y_{ref} - y < 0 \\ 0 & \text{se } y_{ref} - y > 0 \end{cases} \quad (4.1)$$

- Para a sala de reuniões:

$$\begin{cases} 0 & \text{se } y_{ref} - y < 0 \\ 1 & \text{se } y_{ref} - y > 0 \end{cases} \quad (4.2)$$

4.2.2 Controle Proporcional-Integrativo

A Tabela 4-1 resume os critérios para construir um controle servo, levando em conta dois valores de sobrepasso (*overshoot*).

Tabela 4-1. Sintonia empírica de controlador P, PI, PID – modelo CHM (Chien, Hrone e Reswick, 1952).

Tipo de Controlador	Com <i>overshoot</i> de 0%			Com <i>overshoot</i> de 20%		
	K_p	T_i	T_D	K_p	T_i	T_D
P	$0.3/a$	-	-	$0.7/a$	-	-
PI	$0.35/a$	$1.2 T$	-	$0.6/a$	T	-
PID	$0.6/a$	T	$0.5 L$	$0.95/a$	$1.4 T$	$0.47 T$

O fator integrativo diminui o erro em regime permanente e não é tão sensível a ruídos – ao contrário da parcela derivativa, cuja função é tornar o controle mais rápido.

Para um processo térmico, o tempo para chegar ao valor de referência não é tão crítico, pois a sua dinâmica é lenta. A amplitude do sobressinal, porém, é um ponto relevante do projeto, uma vez que, além da preocupação com saturação, prejudicaria o conforto térmico. Com isso, e considerando o que foi dito no parágrafo anterior, escolhemos fazer um controlador PI, com overshoot de 0%. A estrutura do controlador é dada pela equação (4.3):

$$C = K_p \left(1 + \frac{1}{T_i s} \right) \quad (4.3)$$

4.2.3 Controle Adaptativo

O controle adaptativo vai além de apenas implementar uma lei de controle. A identificação da planta – o objeto de estudo é fator fundamental. Existe, portanto, uma interação constante com os módulos de comunicação e, indiretamente, com o sensor de temperatura. A Figura 4-3 ilustra este processo.

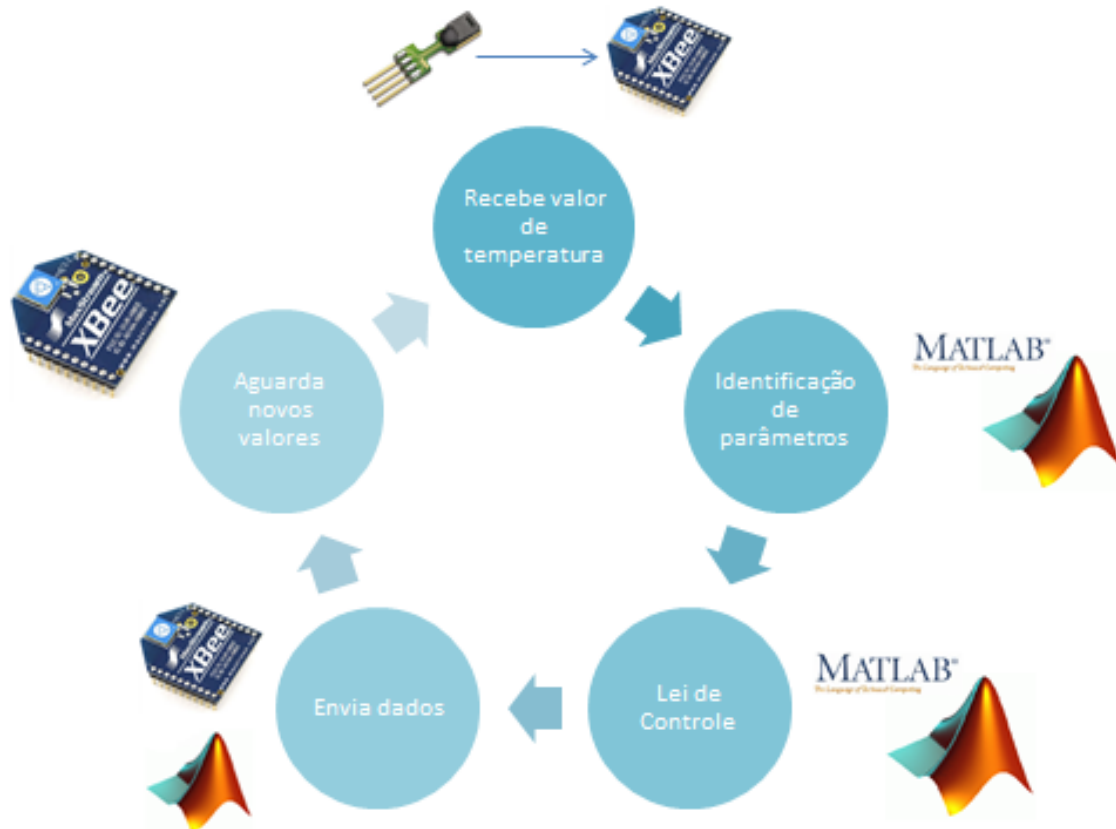


Figura 4-3. Esquemático de etapas do controle adaptativo.

As especificações definidas para os controladores clássicos foram as mesmas para o controlador adaptativo, isto é, sobressinal de 0%, com pouca variação em torno da referência.

Embora existam condições para definir os polinômios R, S e T, não há um algoritmo específico para encontrá-los. Os autores em [Aström e Wittenmark, 1995] mostram certas escolhas que facilitam o design do controlador e, por isso, são mais usuais. As definições de alguns coeficientes e do grau dos polinômios envolvidos foram feitas empiricamente, considerando o domínio do tempo discreto.

De acordo com (Souto, 2006), uma função de segunda ordem com canal integral é uma boa representação para um modelo de referência. Desta forma, temos que:

$$R = b_0 z \quad (4.4)$$

$$T = b_{m0} \quad (4.5)$$

$$S = a_0 - a_{m0} \quad (4.6)$$

Em que a_0 e b_0 são parâmetros da planta, e a_{m0} e b_{m0} são parâmetros do modelo de referência.

Por sua vez, o modelo de referência é dado por:

$$G_m(z) = \frac{b_{m0}z + b_{m1}}{z^2 + a_{m1}z + a_{m2}} = \frac{0.0925z + 0.08738}{z^2 - 0.8998z + 0.07984} \quad (4.7)$$

A lei de controle é expressa pela equação (4.8), a seguir.

$$u(k) = K \left[u(k+1) + \frac{b_1}{b_0} (a_{m0} - a_2)(y(k-2) - y(k-1)) - (a_{m2} - a_0)(y(k-1) - y(k)) + (a_{m0} + a_{m1} + a_{m2})(y_{ref}(k) - y(k)) \right] \quad (4.8)$$

A inclusão do ganho K na equação (4.8) é uma adaptação da lei de controle, pois foi visto que, com esta inclusão, melhorava-se o transiente. Na Tabela 4-2, estão os ganhos escolhidos, empiricamente, para a sala e para a maquete.

Tabela 4-2. Ganho K para cada ambiente predial estudado.

Ambiente	Ganho K
Sala de Reuniões	4
Maquete	33

5 Resultados

5.1 Experimento na Maquete

O experimento completo consiste em realizar a identificação, avaliá-la e obter as performances dos controladores ON-OFF, PI e adaptativo, nesta ordem. No caso da maquete, em que o objetivo era apenas fazer testes, partimos para uma situação de carga térmica constante, com apenas um secador de cabelos, além do atuador, que gerava uma perturbação controlada, no sistema.

5.1.1 Identificação

Como resultado da identificação na maquete, temos a equação (5.2), com referência em um modelo de segunda ordem descrito em (5.1):

$$G(z) = \frac{Y(z)}{U(z)} = \frac{b_0 z + b_1}{z^2 - a_1 z + a_2} \quad (5.1)$$

$$G(z) = \frac{Y(z)}{U(z)} = \frac{0,0234z + 0,0242}{z^2 - 0,5z + 0,5} \quad (5.2)$$

Foi extraído o vetor de regressores:

$$\boldsymbol{\psi}(k) = [y(k-1) \ y(k-2) \ u(k-1) \ u(k-2)] \quad (5.3)$$

E o vetor de parâmetros:

$$\boldsymbol{\theta}(k) = [\theta_1 \ \theta_2 \ \theta_3 \ \theta_4]^T \quad (5.4)$$

Os gráficos contidos nas Figuras Figura 5-1 e Figura 5-2 mostram, respectivamente, a evolução da identificação da maquete e dos seus parâmetros ao longo do tempo.

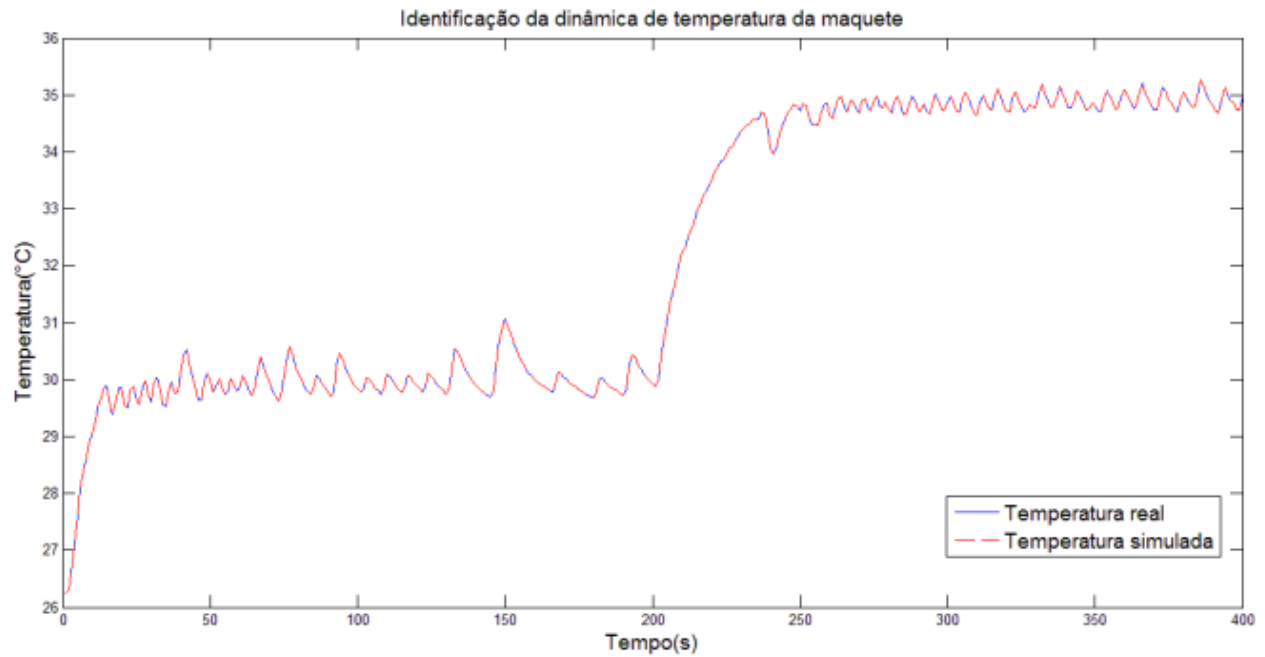


Figura 5-1. Medição da temperatura no ambiente de estudo na maquete.

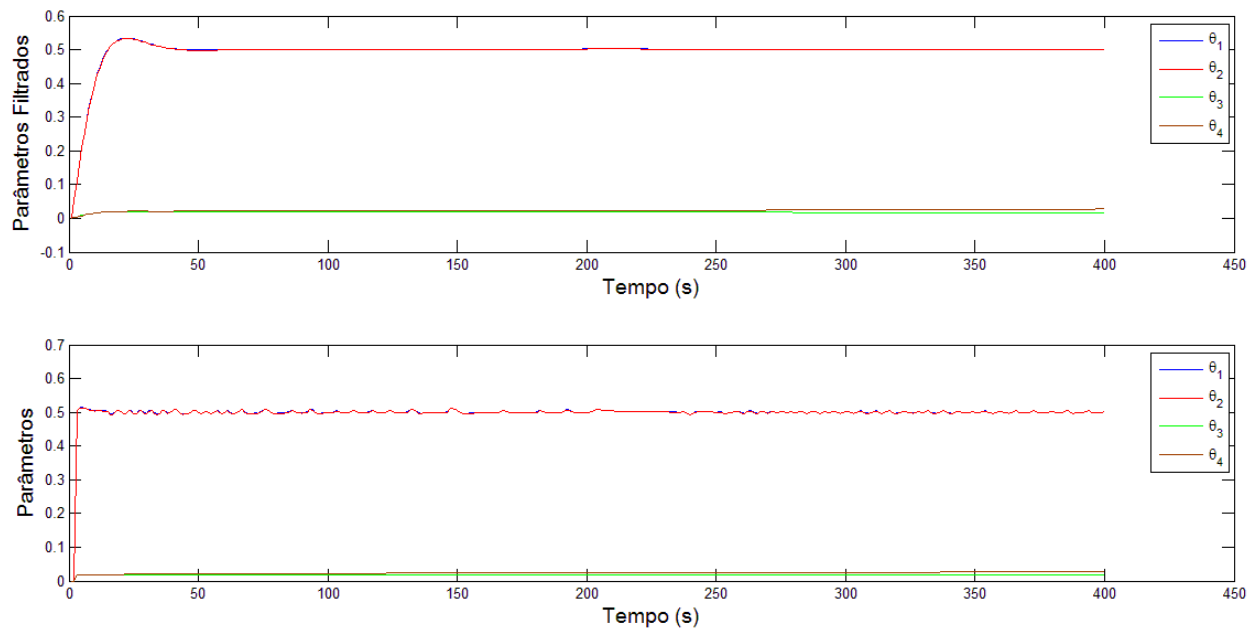


Figura 5-2. Estimação de parâmetros do ambiente de estudo na maquete, conforme a equação (5.4).

Pela Figura 5-1, vê-se que a identificação foi satisfatória, afinal, mal é possível diferenciar o medido do real. O erro quadrático médio foi de $7,69 \times 10^{-4}$.

Filtramos os parâmetros identificados com um filtro passa-baixa Butterworth de segunda ordem, feito em software. Fazer este tratamento é útil para a atuação do controle adaptativo, pois os parâmetros mudarão de forma mais sutil, permitindo que o controlador os acompanhe melhor.

A Figura 5-2 mostra a evolução dos parâmetros da identificação na maquete ao longo do tempo. Como podemos ver, eles variam pouco, estabilizando rapidamente também. O parâmetro mais lento em alcançar o regime permanente (θ_2 dos parâmetros filtrados) o faz em menos de 1 minuto. O parâmetro que mais varia (θ_2 dos parâmetros puros) tem amplitude de sinal menor que 0,05.

5.1.2 Controle com Carga Térmica Constante

Primeiramente, fizemos um controle simples, do tipo ON-OFF. A Figura 5-3, a seguir, mostra o resultado.

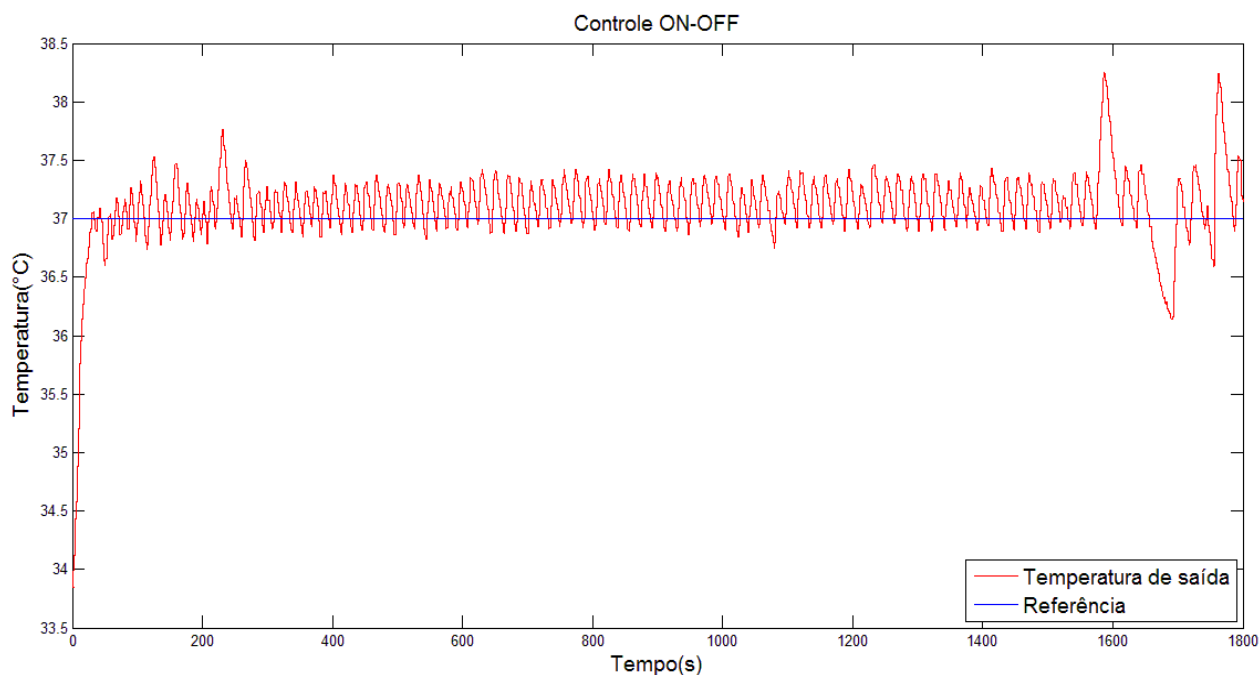


Figura 5-3. Medição de temperatura usando controle ON-OFF.

A histerese foi de [36,8; 37,2 °C].

Vê-se, a partir do instante 1500s na Figura 5-3, que a amplitude de oscilação aumentou em alguns pontos. Isso se deve à perda de sinal da rede sem fio.

Fizemos um controle PI, e o resultado consta na Figura 5-4 abaixo.

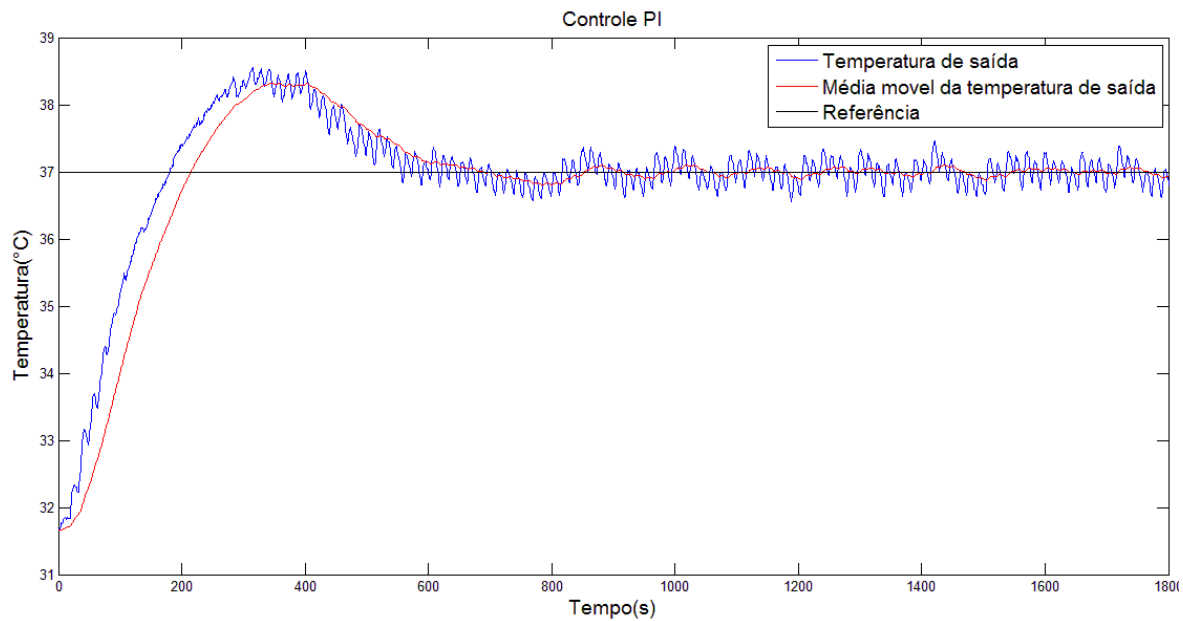


Figura 5-4. Medição de temperatura usando controle PI.

A Figura 5-5 mostra a resposta ao degrau do controle adaptativo proposto.

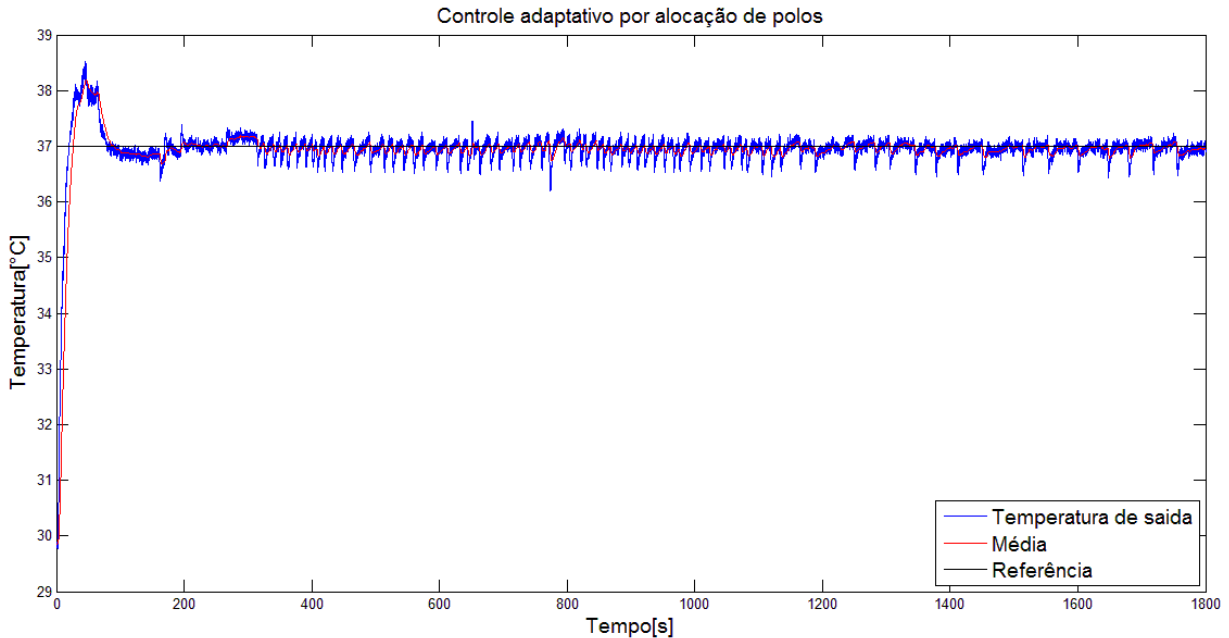


Figura 5-5. Medição de temperatura usando controle RST.

Pela Figura 5-3, vemos que o controle ON-OFF, simbolizado pelo relé na Figura 4-2, é o que gera mais consumo de energia, já que permanece mais tempo. É o mais “instável”, com

variações de sinal com amplitude de quase 1°C. Por causa das oscilações, percebe-se que este tipo de controle exige muito do acionamento do atuador, e, pelo fato de ser uma das técnicas de controle mais simples, não há muitas modificações a se fazer para melhorar o desempenho.

Comparando-se as Figuras 5-4 e 5-5, em termos de tempo de subida e sobressinal, os controladores PI e RST tem desempenho parecido. Quanto a atingir a referência, o controlador PI a atinge mais rápido do que RST, com uma diferença de pouco mais de 3 minutos. Isso era esperado, já que a etapa de identificação foi incorporada ao controlador no caso do RST.

A maquete é um sistema pequeno, de dinâmica rápida, e certas peculiaridades de um ambiente de tamanho real não são visíveis, neste caso.

5.2 Experimento na Sala de Reuniões

Aqui, faremos a validação do que foi desenvolvido na etapa da maquete. Mais uma vez, seguimos com a sequência de fazer a identificação do ambiente, avaliá-la e obter as performances dos controladores. Neste caso, no entanto, há duas configurações de avaliação dos controladores: a primeira, com carga térmica constante, e a segunda, com variação de carga térmica.

5.2.1 Identificação

Como resultado da identificação em batelada dos sinais na Figura 5-6, temos a equação (5.5):

$$\frac{Y(z)}{U(z)} = z^{-3} \frac{-0,0219z+0,0154}{z^2-1,918z+0,9191}. \quad (5.5)$$

O vetor de regressores é:

$$\boldsymbol{\psi}(k) = [y(k-1) \ y(k-2) \ u(k-5) \ u(k-4)], \quad (5.6)$$

O vetor de parâmetros:

$$\boldsymbol{\theta}(k) = [\theta_1 \ \theta_2 \ \theta_3 \ \theta_4]^T. \quad (5.7)$$

Em seguida, foi feita a estimação dos parâmetros do sistema. O resultado é exibido na Figura 5-7.

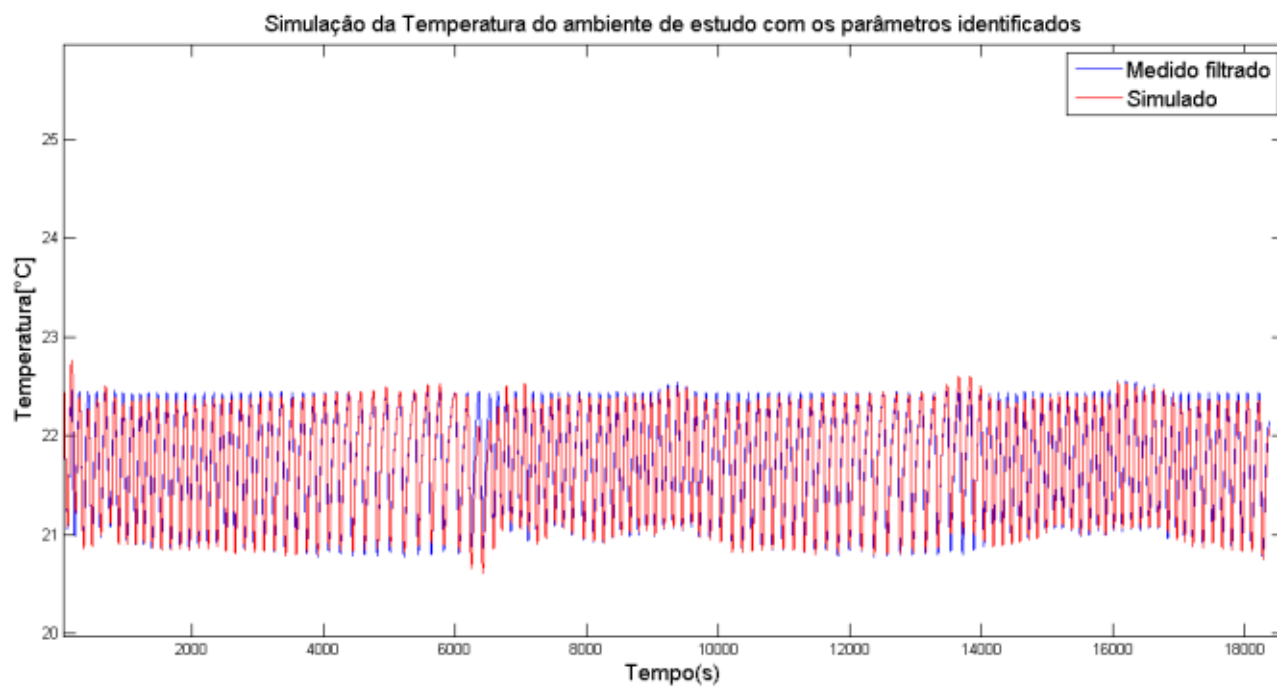


Figura 5-6. Estimação da temperatura.

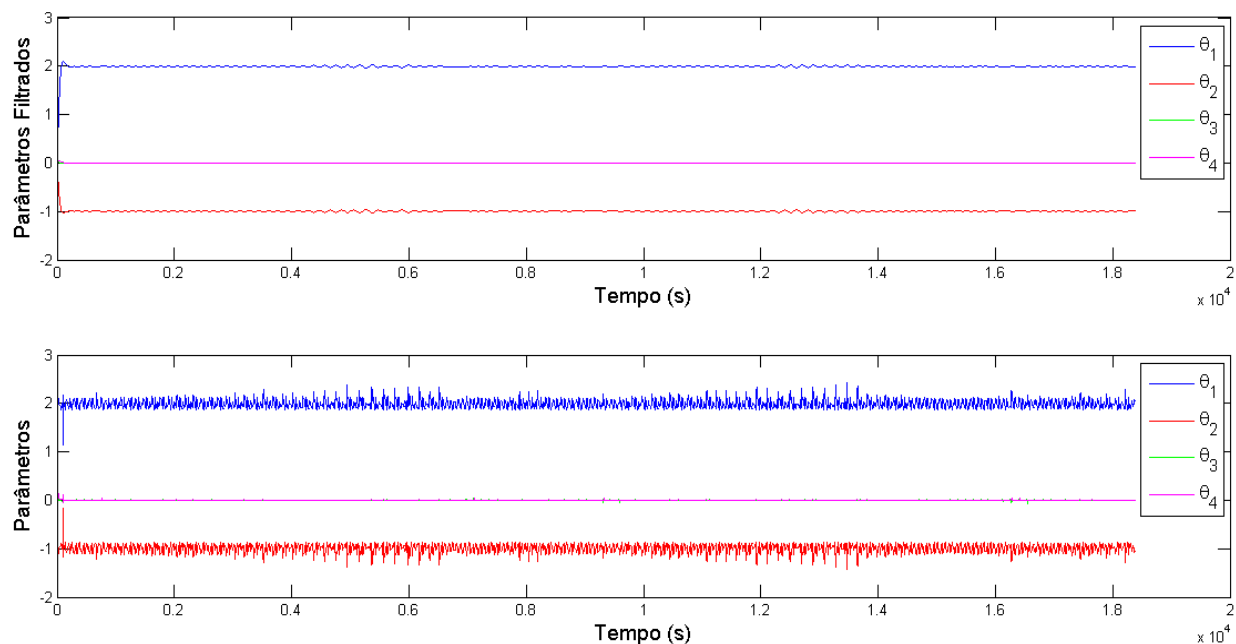


Figura 5-7. Estimação dos parâmetros, conforme a equação 5.7.

Podemos ver pela Figura 5-6, que a identificação foi satisfatória. O erro quadrático médio foi de 0.0156. Os parâmetros da identificação, de acordo com a Figura 5-7, variam bastante quando não filtrados. Reitera-se, portanto, a necessidade de submeter a identificação a um filtro passa-baixa filtro Butterworth de segunda ordem, feito em software, como no caso da maquete.

Comparando-se com os resultados obtidos com a maquete, vê-se que as variações são mais frequentes ao longo do tempo, e estas têm maior amplitude. Isso é refletido no erro quadrático médio, que é um pouco maior do que aquele para a identificação na maquete.

É claro que essas perturbações influenciaram no controle, o que será analisado a seguir.

5.2.2 Controle com Carga Térmica Constante

O primeiro resultado que tivemos foi a saída e o sinal de controle do controlador ON-OFF, representados pelas Figuras Figura 5-8 e Figura 5-9, respectivamente.

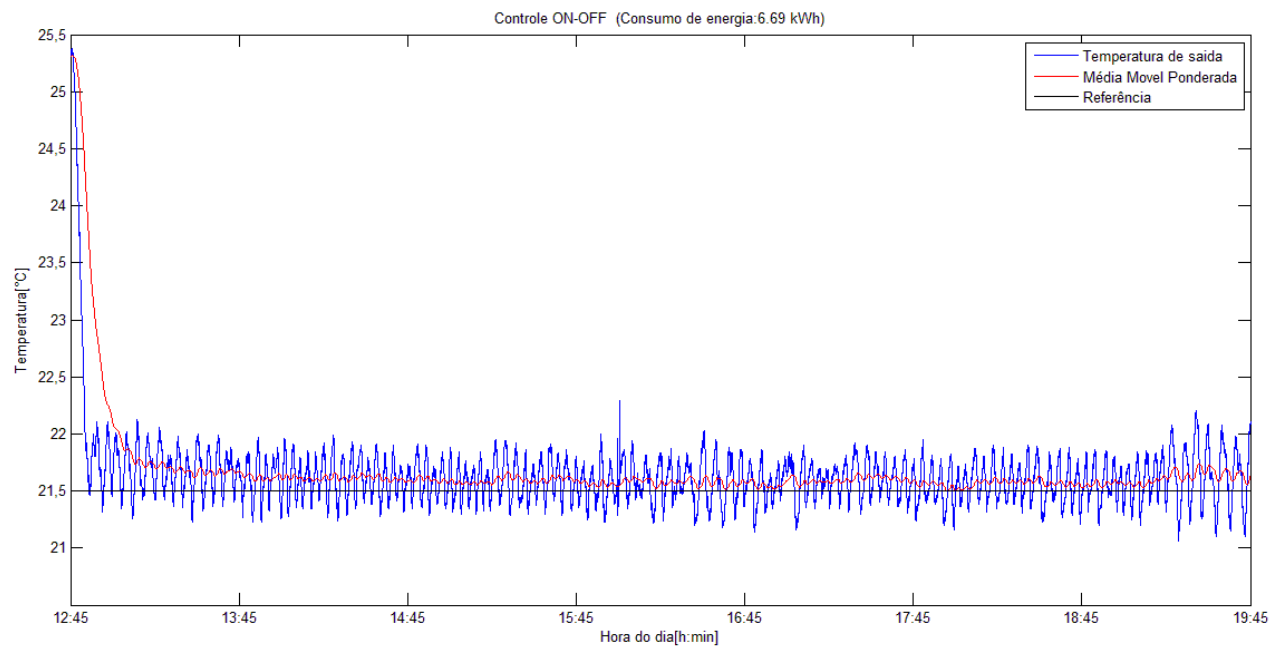


Figura 5-8. Medição de temperatura usando controle ON-OFF.

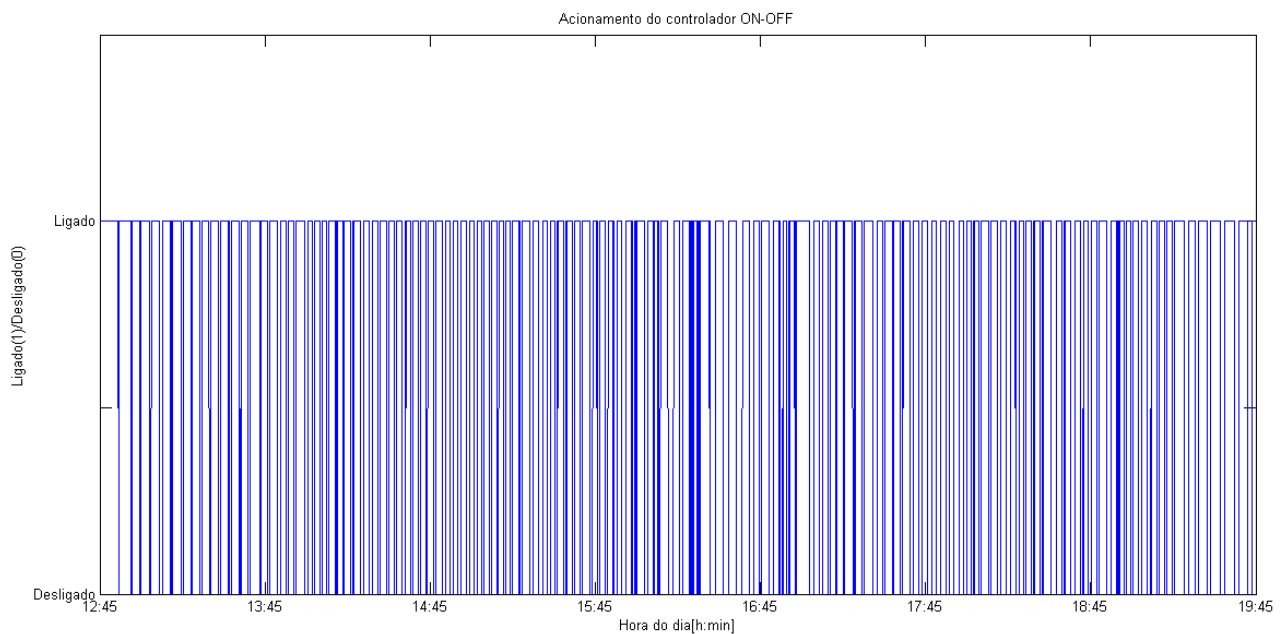


Figura 5-9. Sinal de controle do controlador ON-OFF.

A Figura 5-10 mostra o controle da temperatura quando o atuador está associado a um controlador PI. A Figura 5-11 ilustra o sinal de controle deste controlador.

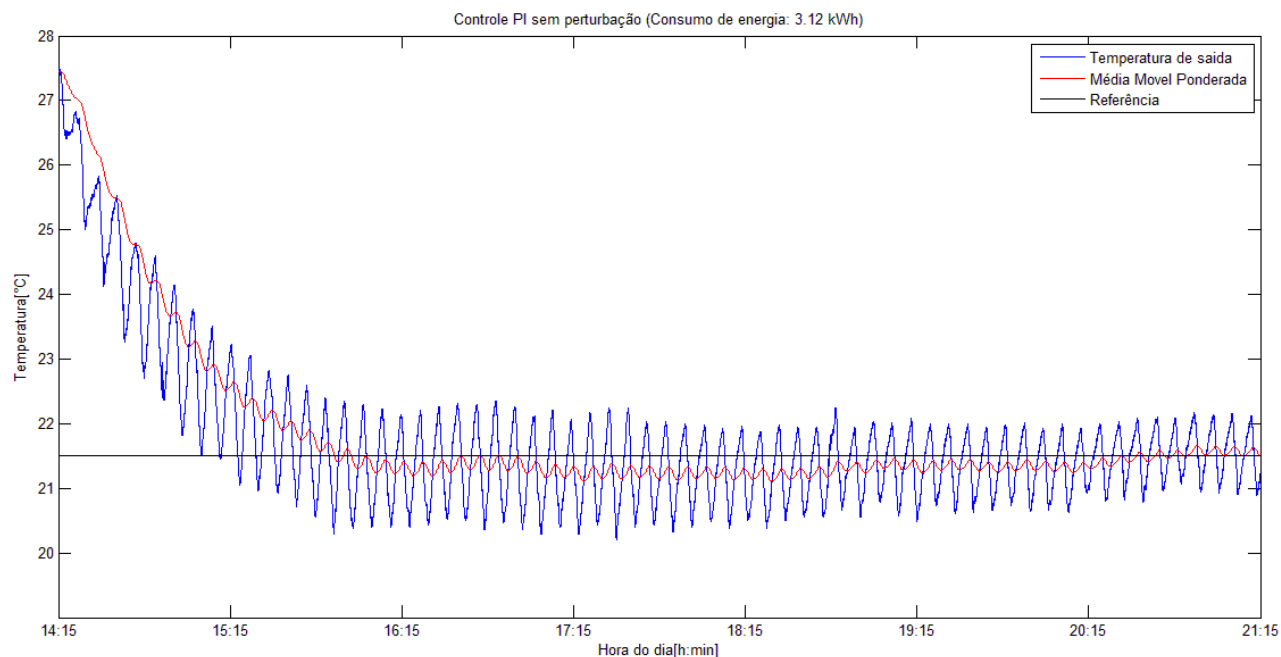


Figura 5-10. Medição de temperatura usando controle PI.

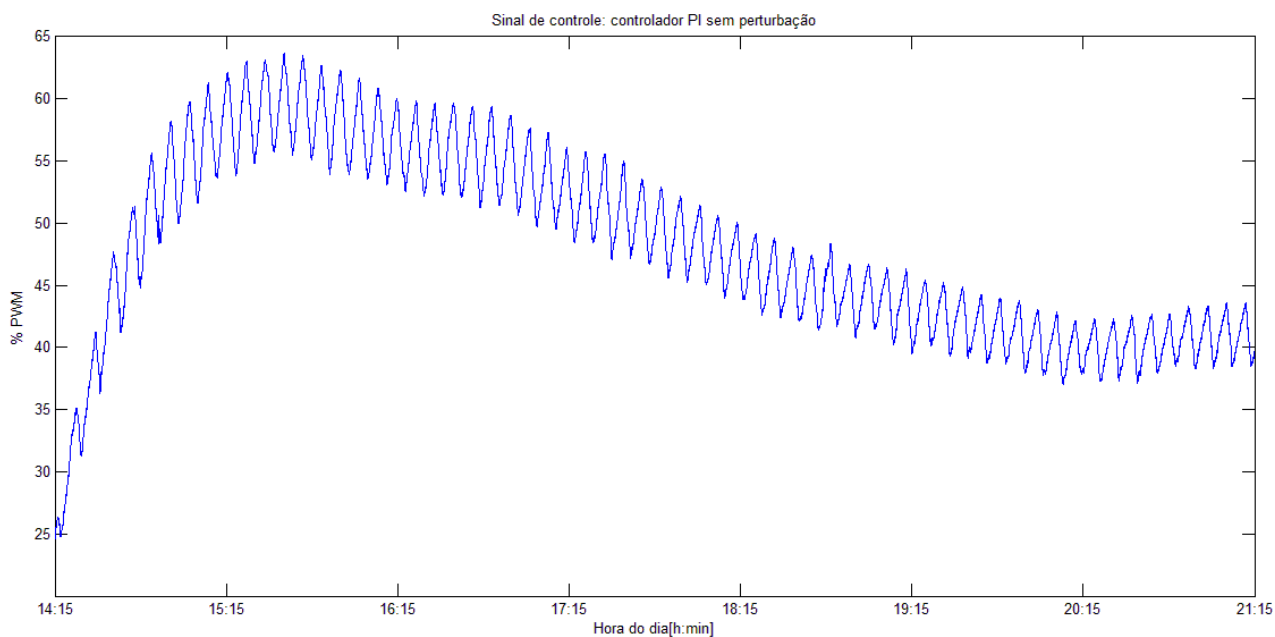


Figura 5-11. Sinal de controle do controlador PI.

A seguir, tem-se o desempenho do controlador adaptativo. A Figura 5-12 mostra a temperatura do ambiente controlada por um controlador RST. A Figura 5-13 mostra o seu sinal de controle.

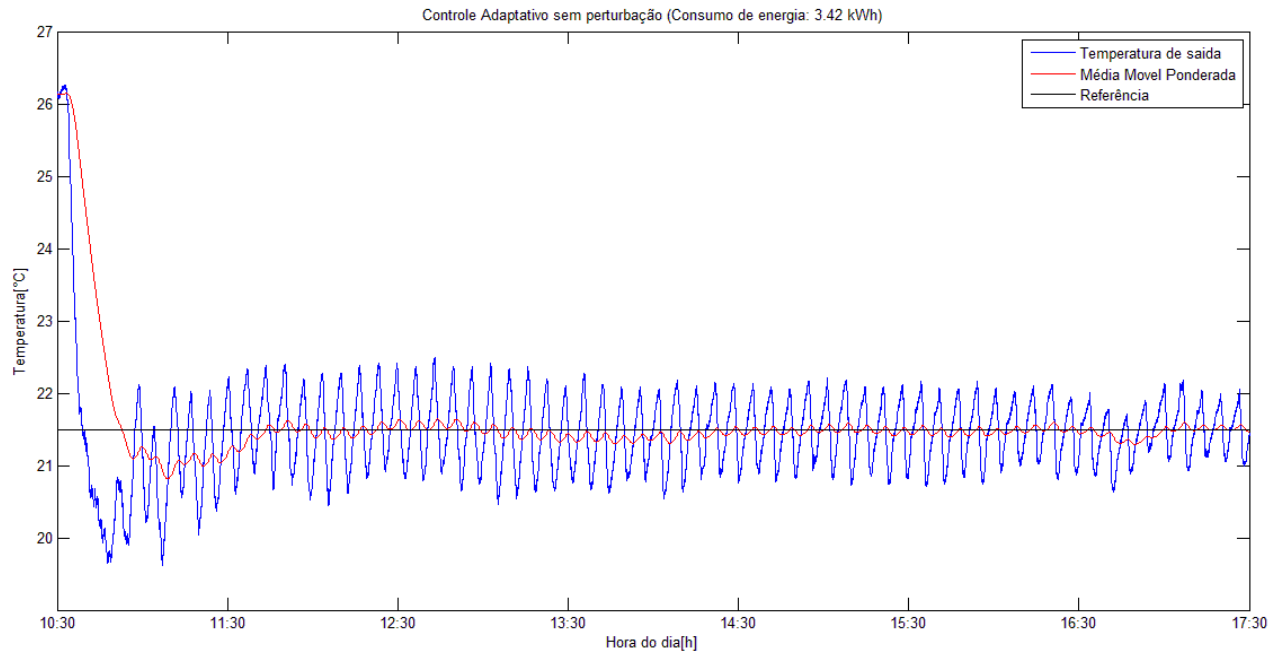


Figura 5-12. Medição de temperatura usando controle RST.

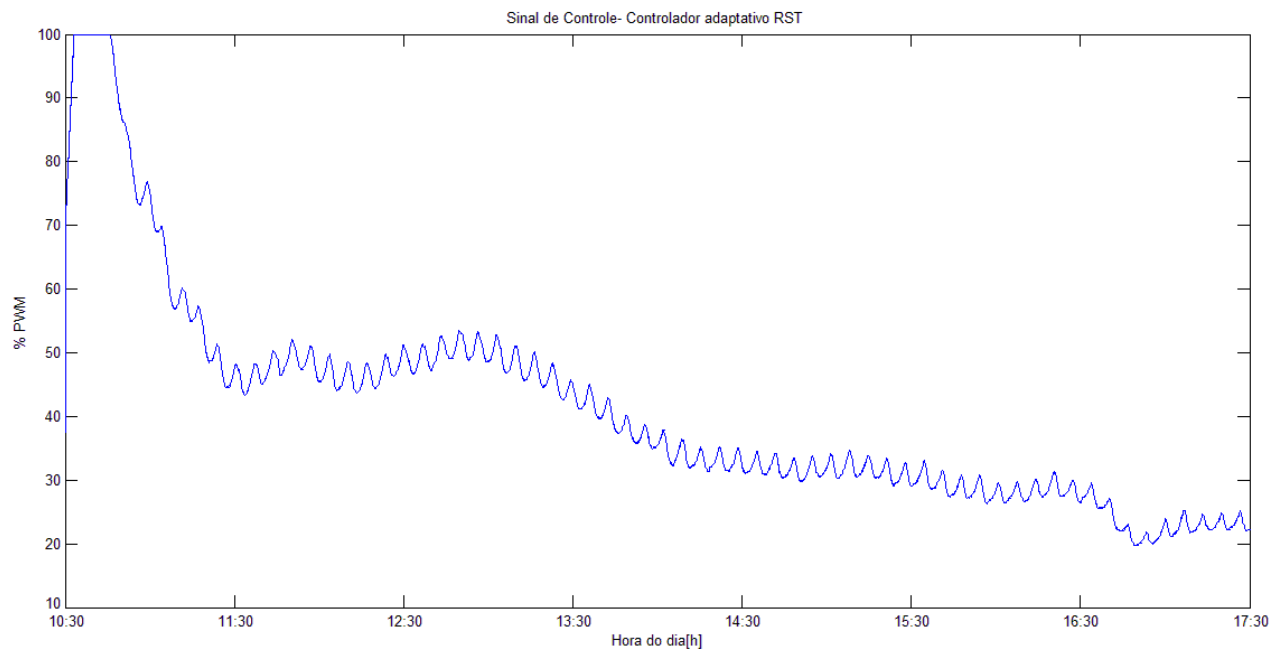


Figura 5-13. Sinal de controle do controlador RST.

Com todos estes resultados, faremos a comparação de desempenho, levando em conta o tempo para atingir a referência e a exigência de acionamento.

Assim como na maquete, vemos que ON-OFF é o mais rápido, de acordo com a Figura 5-8. Comparando-se com o controlador RST, no entanto, a diferença de tempo para alcançar a referência não é exatamente uma vantagem. Este controlador é o que mais gasta energia, como podemos ver pela Figura 5-9.

O controle original do ar condicionado da sala era do tipo ON-OFF diferencial, pois baseava-se apenas em um valor referencial. Desta forma, é possível dimensionar a melhora que os outros controladores adicionariam ao sistema. A equação (5.8) define essa melhora em termos matemáticos, em que E_{ON-OFF} é a energia, em kWh, gasta pelo controlador original, E_{NOVO} é a energia gasta pelo controlador novo e $M\%$ é o percentual de melhora.

$$\frac{E_{ON-OFF} - E_{NOVO}}{E_{NOVO}} \times 100 = M\% \quad (5.8)$$

Em termos de percentual, vemos que a melhora é de:

Tabela 5-1. Relação entre tipos de controladores e melhora em termos de energia.

Controlador	Percentual de melhora
PI	53,36%
RST	48,88%

Não foi possível fazer o mesmo para a maquete, pois não tínhamos como medir a energia gasta pelo secador de cabelos.

Vê-se, pela Tabela 5-1, que o controlador PI, em relação ao RST, é o que menos gasta energia, principalmente no começo do acionamento, nos primeiros 15 minutos transcorridos, como é possível verificar pelas Figuras Figura 5-11 e Figura 5-13. Ainda analisando estas figuras, após 30 minutos aproximadamente, o desempenho dos dois, em termos de sinal de controle, passa a ser parecido. Quanto ao tempo para alcançar a referência, o RST é mais rápido que o PI, com uma diferença de 20 minutos.

5.2.3 Controle com Carga Térmica Variável

Nesta etapa do projeto, expomos o ar condicionado da sala a perturbações cotidianas, como pessoas em reunião, ou apenas estudando na sala, e a porta sendo aberta várias vezes. Este é a configuração mais importante na definição de um bom controlador, que é quando o ambiente está em uso. Para fazermos as devidas comparações, o experimento foi feito para o controlador ON-OFF, o controle original do ar condicionado. As Figuras Figura 5-14 e Figura 5-15 mostram, respectivamente, a sua saída e o seu sinal de controle.

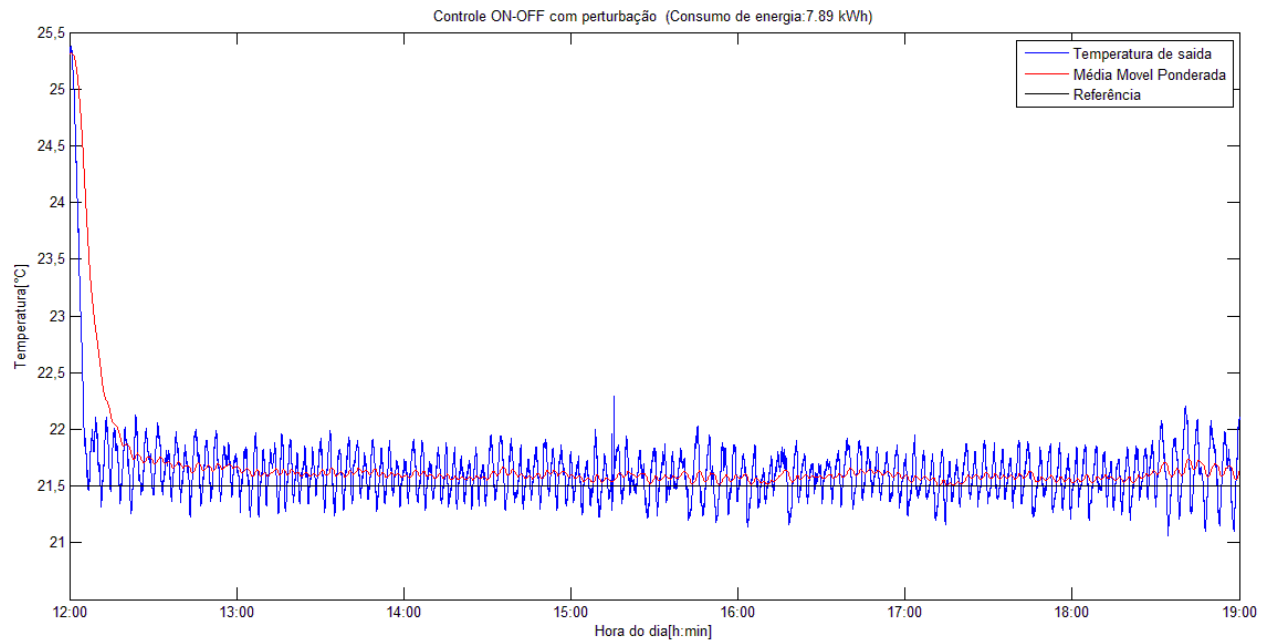


Figura 5-14. Medição da temperatura usando controlador ON-OFF, com perturbações.

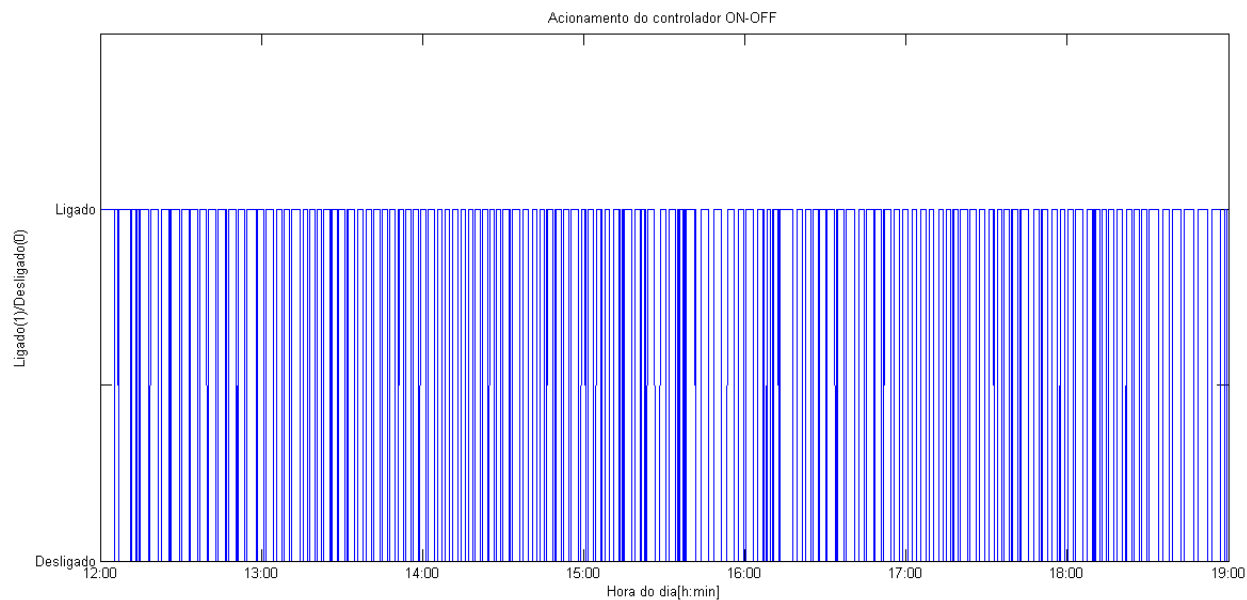


Figura 5-15. Sinal de controle do controlador ON-OFF, com perturbações.

Para o controle PI, chegamos às Figuras Figura 5-16 e Figura 5-17, que ilustram a saída em termos de temperatura e o sinal de controle, respectivamente.

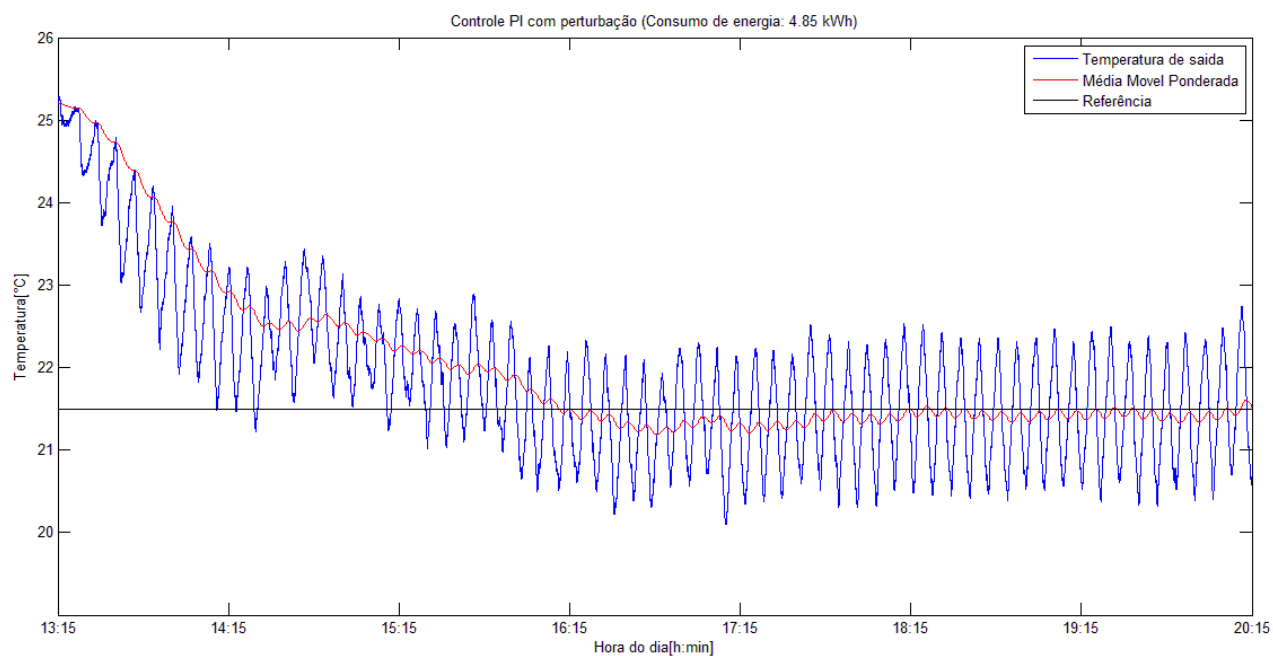


Figura 5-16. Medição da temperatura usando controlador PI, com perturbações.

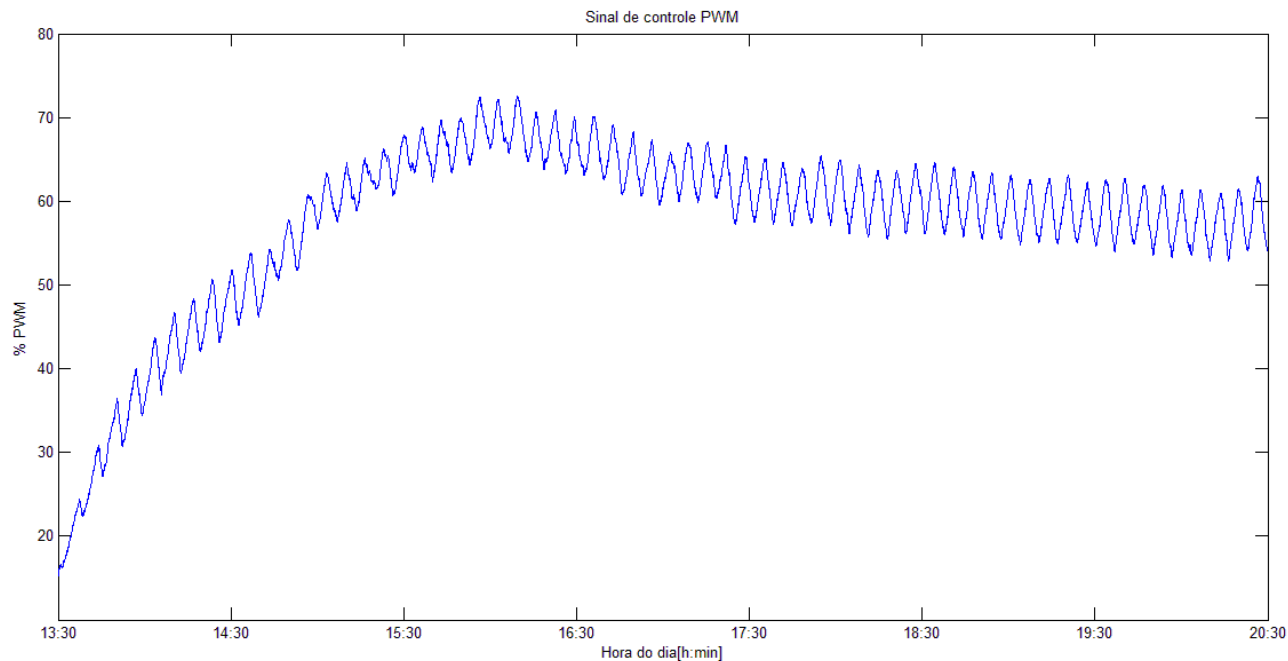


Figura 5-17. Sinal de controle do controlador PI, com perturbações.

Para o controlador RST, podemos ver seu desempenho pelas Figuras Figura 5-18 e Figura 5-19.

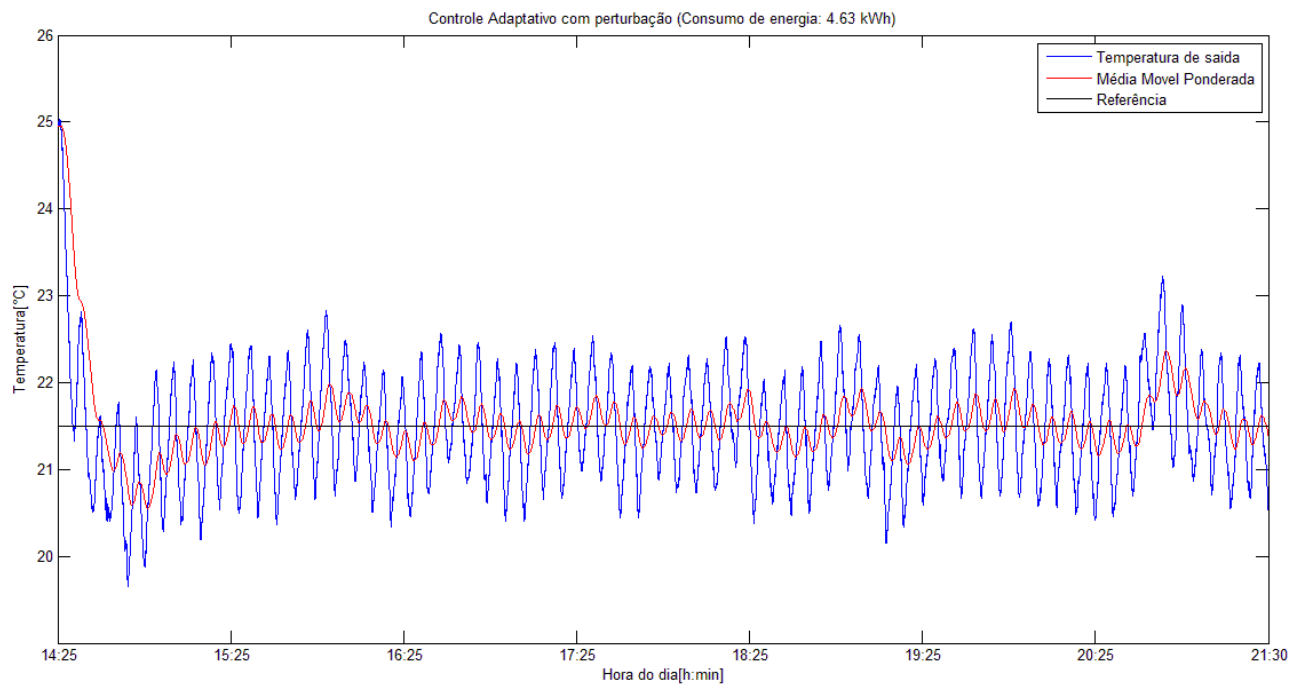


Figura 5-18. Medição da temperatura usando controlador RST, com perturbações.

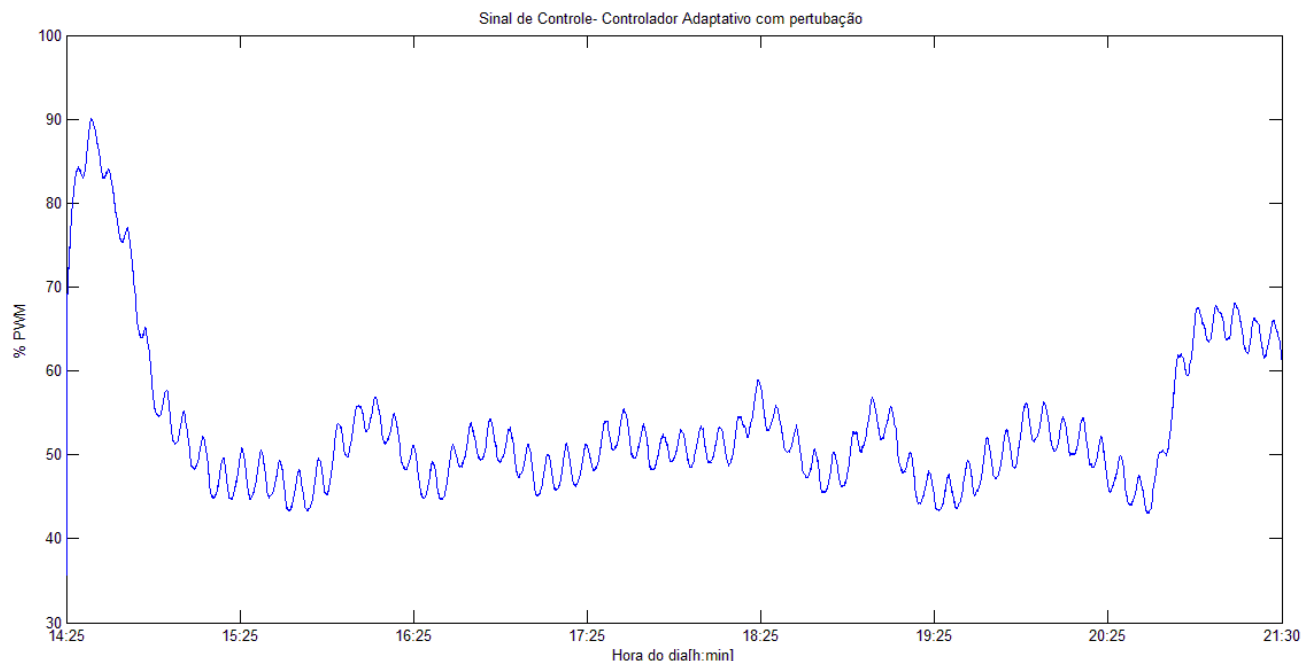


Figura 5-19. Sinal de controle do controlador RST, com perturbações.

Fazendo o mesmo cálculo percentual de melhoria – feito para o caso sem variação da carga térmica –, temos a Tabela 5-2:

Tabela 5-2. Relação entre tipos de controladores e melhoria em termos de energia, em comparação com o controlador ON-OFF, com perturbações.

Controlador	Percentual de melhoria (em comparação ao ON-OFF)
PI	38,53%
RST	41,32%

Em se tratando do caso com perturbações, vemos que o RST é mais econômico e, portanto, a característica adaptativa torna o controlador melhor, como imaginávamos. Analisando as Figuras Figura 5-17 e Figura 5-19, vemos que, quanto ao acionamento, no começo, o controlador PI é menos exigente, mas, ao longo do tempo, este gasta mais energia do que o RST, com uma diferença de 10% a 15% entre os dois. A economia de energia efetiva é de 2,79%, como é possível checar a partir da Tabela 5-2 acima. Seria o caso, também, de utilizar um controlador inicial PI, e fazer a transição para o RST perto do ponto de operação.

Ainda nas Figuras Figura 5-17 e Figura 5-19, em termos de rapidez, vê-se que o PI demora mais que o RST para atingir a referência, com uma diferença de um pouco mais de 1 hora.

Ao se comparar o desempenho temporal do controlador RST com e sem variação térmica, olhando as Figuras Figura 5-13 e Figura 5-19, vemos que este não muda muito, com uma variação em torno de 5 a 6 minutos.

O cálculo do erro RMS de cada sinal de controle é dado pela fórmula (5.9), a seguir:

$$E_{RMS} = \frac{1}{N} \sum_{i=1}^N (y_{ref} - y_i)^2 \quad (5.9)$$

Em que:

E_{RMS} é o erro RMS do sinal de controle relativo à referência;

N é o número de amostras;

y_{ref} é a referência, e;

y_i é a medida de número i na amostra.

Com a equação (5.9), calculamos o erro quadrático médio do sinal de controle visto nesta sessão.

A Tabela 5-3 resume o desempenho de cada controlador, nas duas configurações de carga térmica.

Tabela 5-3. Tabela resumo de desempenho dos controladores (configuração, erro RMS e consumo energético).

Carga Térmica	Controlador ON-OFF		Controlador PI		Controlador Adaptativo	
	E_{RMS}	Consumo [kWh]	E_{RMS}	Consumo [kWh]	E_{RMS}	Consumo [kWh]
Constante	0,1399	6,69	1,5524	3,12	0,5291	3,42
Variante	0,1236	7,89	1,1850	4,85	0,4342	4,63

6 Conclusão

Como era proposta deste trabalho, devíamos elaborar um controlador adaptativo para um sistema de ar condicionado híbrido e comparar o seu desempenho com as técnicas clássicas de controle, ON-OFF e PI. Para tanto, foi necessário fazer uma boa identificação do sistema.

Fizemos uma identificação recursiva de uma sala de reuniões de um laboratório, feita com base no algoritmo de Mínimos Quadrados Recursivo, para a sala e para uma maquete de madeira. O erro quadrático médio obtido para a sala foi de 0,0156, e para a maquete, $7,69 \times 10^{-4}$. A identificação recursiva é rápida, pois há o paralelismo na ação de identificar e de coletar dados, e é mais assertiva, principalmente se adaptarmos o algoritmo de identificação com um fator de esquecimento, cuja função é priorizar as medidas mais recentes.

Posteriormente, a estimação dos parâmetros determinada pela identificação será usada no design de um controlador adaptativo. À priori, foram desenvolvidos controladores ON-OFF, PID e PI, estes últimos associados a um PWM, os quais foram testados em uma maquete de madeira. O uso de uma maquete é justificado por ser um ambiente não sujeito a grandes perturbações térmicas (um grupo de pessoas entrando) e pela sua dinâmica rápida.

Sobre os controles clássicos, chegamos à conclusão que o PI é mais vantajoso para esse tipo de trabalho, uma vez que é o mais econômico em termos de acionamento, apesar de – e pelo fato de – ser o mais lento. O ajuste do PI foi feito segundo o modelo CHR [Chien, Hrone e Reswick, 1952].

A finalidade de se fazer um controle adaptativo é a tornar um controlador simples mais flexível à demanda de carga. Com isso em mente, elaboramos um controlador RST, que estará sujeito aos parâmetros obtidos na identificação, a partir de uma realimentação direta com os parâmetros calculados pela identificação.

Sobre o experimento na maquete, observamos que, em termos de acionamento, vemos que os controladores clássicos gastam mais energia, embora sejam um pouco mais rápidos, característica visível para o controle ON-OFF. Isso mostra que se gasta menos energia com uma abordagem adaptativa. Devemos esclarecer que, embora o controle adaptativo seja mais lento, isto não é exatamente um problema em se tratando de processos térmicos, em que as variações são lentas. Podemos acrescentar ainda um aspecto benéfico dessa lentidão, que é não exigir demais do atuador ao mudar de estado (ligado e desligado). É preciso certo tempo

para ligar e desligar o compressor e, caso este tempo não seja respeitado, a vida útil do aparelho é reduzida sensivelmente.

Ao fazer o experimento na sala de reuniões do laboratório, consideramos duas configurações: uma situação em que a variação térmica é mínima, isto é, há pouca variação térmica, e outra situação, com bastante variação, a qual denominamos, ao longo do trabalho, como perturbações.

Para um ambiente sem variação térmica, vimos uma peculiaridade em relação ao que foi concluído no experimento da maquete: o controlador PI se saiu melhor quanto à economia de energia. Além disso, vimos que este controlador foi um pouco mais lento que o RST, resultado contrário ao comprovado anteriormente. O fato é que um ambiente com pouca variação térmica possui parâmetros mais constantes, e, assim, a característica adaptativa do controlador RST não foi observada.

Quando expostos à variação térmica, tivemos o resultado que esperávamos: o controlador RST mostrou grande economia de energia em relação ao controlador ON-OFF, e relativa economia em relação ao controlador PI. Inclusive, o controle adaptativo foi mais rápido que este último, com uma diferença de tempo um pouco mais de 1 hora. Devemos esclarecer que, quando falamos do aspecto benéfico de uma demora do controlador para atingir a referência, em virtude da grande diferença temporal para este caso específico, essa lentidão deixa de ser benéfica.

Com isso, concluímos que o controlador RST, para ambientes térmicos sujeitos a variações térmicas, é a melhor opção, tanto em desempenho quanto em rapidez, dentro do contexto deste projeto.

7 Trabalhos Futuros

Determinar qual a melhor estratégia é um avanço em pesquisas de economia de energia, mas não é o bastante. Este trabalho espera cumprir a finalidade de auxiliar outros pesquisadores em ambientes prediais inteligentes do Laboratório de Automação e Robótica. Salientamos que é possível otimizar alguns resultados. Seguem abaixo algumas sugestões:

- Usar mais variáveis na estimação do modelo, como humidade, radiação solar, temperatura das salas vizinhas;
- Utilizar módulos de rede sem fio que possuem um menor consumo de energia;
- Testar outros algoritmos de identificação, como o Filtro de Kalman e suas variantes de estimação não-linear;
- Utilizar outras técnicas de controle adaptativo,
- Desenvolver um controlador híbrido, em que duas ou mais técnicas de controle sejam empregadas, de acordo com a necessidade, e;
- Integrar com um sistema de detecção de pessoas, de modo a obter uma melhor estimação da carga térmica presente.

Bibliografia

1. Aguirre, L.A. (2007). Introdução à Identificação de Sistemas – Técnicas Lineares e Não-Lineares Aplicadas a Sistemas Reais. Ed. UFMG, 3ª Ed.
2. Aström, K. L., Wittenm, B., (1995). Adaptive Control. Editora Addison-Weslwy, 2ª Ed.
3. Bauchpiess, A., Ishihara, J. Y., Felgner, F. e Litz L. (2006). “First-Principles Structured Identification for Predictive HVAC Control” CLCA2006, Salvador, 6pp.
4. Fraisse, G., Viardot, C., Lafabrie, O., Archand, G. (2002). Development of a simplified and accurate building model based on electrical analogy. Energy and Buildings 34(2002) 1017-1031.
5. Khalil, H. K. (2002). Nonlinear Systems. Ed. Prentice Hall, 3ª Ed.
6. Lamberts, R., Xavier, A. A., Goulart, S., Vecchi, R., (2011). Conforto e Stress Térmico. LabEEE, UFSC.
7. Ljung, L. (1999). System Identification – Theory for the user. Upper Saddle River, NJ: Pretince Hall.
8. Olmos, C. C. (2009). Sistema híbrido de climatização visando conforto térmico e eficiência energética. Dis. Mestrado, UnB.
9. Oliveira, F.A.R. (2010). Instrumentação e Identificação de um Ambiente Predial Visando Controle Preditivo do Conforto Térmico, Diss. Mestrado, UnB.
10. van Overschee, P., and B. de Moor. Subspace Identification of Linear Systems: Theory, Implementation, Applications. Kluwer, 1996.
11. Souto, R. F., (2006). Avaliação e Implementação de Técnicas de Controle para Condicionamento Térmico de Ambientes Prediais, Trabalho de Graduação, UnB.
12. Xue, D., Chen, Y., Atherton, D. P. (2009). Linear Feedback Control: Analysis and Design with Matlab. Editora SIAM, 1a. Ed.

ANEXOS

I. Código de Lectura de Temperatura

```
#include <math.h>
#include <avr/sleep.h>
#include <avr/power.h>

// comandos para o SHT71

int temperatureCommand = B00000011; // comando para ler
temperatura

int humidityCommand = B00000101; // comando para ler humidade

int comandoEstatus= B00000111; // comando para ler registro de
estatus

// pines i2c

int clockPin = 11; // pino para o clock

int dataPin = 12; // pino de dados

int ack; // erro ack

// para los 2 sensores independientes

int temPin = 0; // pin para el sensor de temperatura

int humPin = 1; // pin para el sensor de humedad

int rojo = 9; // los LEDs van en las salidas PWM 9..

int verde = 10; // ..y 10

int valT = 0; // variable to store the value coming from the
sensor

int valH = 0; // idem

int Nlectura = 0; // esto cuenta el numero de lecturas

int aux = 0; // auxiliar q se resetea cada 100 lecturas

float valTFloat = 0.0; // aqui pongo la transformacion a
flotante de 'val'
```

```

float valHFloat = 0.0; // idem

float milivoltsT = 0.0; // transformo a milivolts

float milivoltsH = 0.0; // idem

float temp = 0.0; // aqui va la temperatura en C

float hum = 0.0; // idem en %

float acumT = 0.0; // aqui voy acumulando hasta 100 lecturas

float acumH = 0.0; // idem

float tempMedia = 0.0; // el promedio de 100 lecturas

float humMedia = 0.0; // idem

float umbralTemperatura = 27.0; // fijo un umbral para iluminar LEDs

float umbralHumedad = 50.0; // fijo un umbral para iluminar LEDs

int espera = 3; // lo que se espera

float diferenciaT = 0.0; // para comparar valores de temperatura

float diferenciaH = 0.0; // para comparar valores de humedad

float RHtrue = 0.0; // humedad real

float Rocio = 0.0; // temperatura de orvalho

int numerobitsT = 14; // especifica o número de bits para a temperatura

int numerobitsH = 12; // especifica o número de bits para a humidade

//sleep mode

volatile int f_timer=0;

ISR(TIMER_OVF_vect) {

    if(f_timer==0) { f_timer=1; } }

```

```

/*void dormir(void) { set_sleep_mode(SLEEP_MODE_IDLE);
sleep_enable(); power_adc_disable(); power_spi_disable();
power_timer0_disable(); power_timer2_disable();
power_twi_disable(); sleep_mode(); sleep_disable();
power_all_enable(); delay(1000); }*/

void setup() {

TCCR1A=0x00;

TCNT1=0x0000;

TCCR1B=0x05;

TIMSK1=0x01;

pinMode(verde, OUTPUT); // sets the pin as output

pinMode(rojo, OUTPUT); // sets the pin as output

Serial.begin(9600); // open serial at 9600 bps }

void loop() { // 1 - inicio das medidas

delay(12); // tem de esperar 12 ms

// ler o estatus

leerEstatus();

// ler a temperatura e converte para celsius

valT = leerValorTemperatura(); // ler a temperatura inteira

temp = calcularTemperatura(valT, numerobitsT); // calcula a
temperatura em

diferenciaT = temp;

// ler a humidade relativa do ar

valH = leerValorHumedad(); // ler o dado em inteiro

hum = calcularHR(valH, numerobitsH); // traduz em %Rh

diferenciaH = hum;

```

```

// calculos adicionais

RHtrue = calcularHumedadVerdadera(temp, valH, hum,
numerobitsH); //calcula a

Rocio = calcularRocio(hum, temp); //calcula a temperatura de
orvalho

// representa o dado

representar(temp, hum, RHtrue, Rocio, valT, valH); //sleep

set_sleep_mode(SLEEP_MODE_IDLE);

sleep_enable();

power_adc_disable();

power_spi_disable();

power_timer0_disable();

power_timer2_disable();

power_twi_disable();

sleep_mode();

sleep_disable();

power_all_enable(); //espera para a proxima leitura para evitar
auto aquecimento

delay(1000); }

// funções de leitura/escrita do SHT7

// ler o valor da temperatura do sensor int
leerValorTemperatura()

{ int valT = 0;

sendCommandSHT(temperatureCommand, dataPin, clockPin); // envia
o comando pa

waitForResultSHT(dataPin); // espera o resu

```

```

valT = getData16SHT(dataPin, clockPin); // captura o valo
skipCrcSHT(dataPin, clockPin); // pula o CRC

return valT; }

// ler o valor de humidade

int leerValorHumedad() { int valH = 0;

sendCommandSHT(humidityCommand, dataPin, clockPin); // envia o
cmoando

waitForResultSHT(dataPin); // espera o resu

valH = getData16SHT(dataPin, clockPin); // captura o valo
skipCrcSHT(dataPin, clockPin); // pula o CRC

return valH; }

// envia os comandos para o sensor de acordo com o protocolo do
data sheet

void sendCommandSHT(int command, int dataPin, int clockPin) {

int ack;

pinMode(dataPin, OUTPUT);

pinMode(clockPin, OUTPUT);

digitalWrite(dataPin, HIGH);

digitalWrite(clockPin, HIGH);

delayMicroseconds(1);

digitalWrite(dataPin, LOW);

digitalWrite(clockPin, LOW);

delayMicroseconds(1);

digitalWrite(clockPin, HIGH);

digitalWrite(dataPin, HIGH);

```

```

delayMicroseconds(1);

digitalWrite(clockPin, LOW);

shiftOut(dataPin, clockPin, MSBFIRST, command);

// verifica se aconteceu erro

digitalWrite(clockPin, HIGH);

pinMode(dataPin, INPUT);

ack = digitalRead(dataPin);

if (ack != LOW) Serial.println("ACK error 0");

digitalWrite(clockPin, LOW);

ack = digitalRead(dataPin);

if (ack != HIGH) delay(1); }

// espera o resultado

void waitForResultSHT(int dataPin) {

int ack;

pinMode(dataPin, INPUT);

ack = digitalRead(dataPin);

while (ack == HIGH) {

ack = digitalRead(dataPin);

}

}

// ler o dado

int shiftIn(int dataPin, int clockPin, int numBits) {

int ret = 0;

```

```

for (int i=0; i<numBits; i++) {
    digitalWrite(clockPin, HIGH);
    delayMicroseconds(1000);
    ret = ret * 2 + digitalRead(dataPin);
    digitalWrite(clockPin, LOW);
}

return ret; }

// obtem o dado do sensor

int getData16SHT(int dataPin, int clockPin) {
    int val, aux;

    // MSB

    pinMode(dataPin, INPUT);
    pinMode(clockPin, OUTPUT);

    val = shiftIn(dataPin, clockPin, 8);
    val *= 256;

    aux = digitalRead(dataPin);

    while (aux != 1) {
        aux = digitalRead(dataPin);}

    // envia o ACK

    pinMode(dataPin, OUTPUT);
    digitalWrite(clockPin, HIGH);
    digitalWrite(dataPin, HIGH);
    delayMicroseconds(10);

```

```

digitalWrite(dataPin, LOW);

digitalWrite(clockPin, LOW);

// LSB

pinMode(dataPin, INPUT);

pinMode(clockPin, OUTPUT);

val |= shiftIn(dataPin, clockPin, 8);

return val;

}

// obtem o estatus

int getDataEstatus(int dataPin, int clockPin) {

int val, aux;

// obtem os 8 bits

pinMode(dataPin, INPUT);

pinMode(clockPin, OUTPUT);

val = shiftIn(dataPin, clockPin, 8);

aux = digitalRead(dataPin);

while (aux != 1) {

aux = digitalRead(dataPin);

} // enviar el ACK, MUCHO CUIDADO

pinMode(dataPin, OUTPUT);

digitalWrite(clockPin, HIGH);

digitalWrite(dataPin, HIGH);

delayMicroseconds(10);

```

```

digitalWrite(dataPin, LOW);

digitalWrite(clockPin, LOW); }

// Pula o CRC

void skipCrcSHT(int dataPin, int clockPin) {

pinMode(dataPin, OUTPUT);

pinMode(clockPin, OUTPUT);

digitalWrite(dataPin, HIGH);

digitalWrite(clockPin, HIGH);

digitalWrite(clockPin, LOW); }

// converte a temperatura em celsius float

calcularTemperatura(int valT, int numerobitsT) {

float d1, d2 = 0.0;

float temperatura = 0.0;

d1 = -40.1; // d1 = -39.7; // alimentação = 3.5 volts // d1 = -
39.6; // alimentação = 3.0 volts

if (numerobitsT == 14) {

d2 = 0.01; } else { d2 = 0.04;

}

temperatura = d1 + d2 * valT; return temperatura; }

// alimentação = 5 volts

// calcula a humidade relativa

float calcularHR(int valH, int numerobitsH) {

float c1 = -4.0;

float c2, c3 = 0.0;

```

```

float HR = 0.0;

if (numerobitsH == 12) {

c2 = 0.0405; c3 = -0.0000028;

}

else { c2 = 0.6480; c3 = -0.00072;

}

HR = c1 + c2 * valH + c3 * valH * valH;

return HR; }

//T_dew

float calcularRocio(float RH, float T) {

float Tn1 = 243.12; // de 0 a 50 graus

float Tn2 = 272.62; // de -40 a 0 graus

float m1 = 17.62; // de 0 a 50 graus

float m2 = 22.46; // de -40 a 0 graus

float Ro = 0.0;

float Tn = 0.0;

float m = 0.0;

if (T < 0.0) { Tn = Tn1; m = m1;

}

else {

Tn = Tn2;

m = m2;

}

```

```

Ro = Tn * (log(RH/100) + ((m*T) / (Tn+T))) / (m - log(RH/100) -
((m*T) / (Tn));

return Ro;

}

// coloca os dados na serial, só foi utilizado a temperatura

void representar(float T, float HR, float RHtrue, float Rocio,
int valT, int v)

//Serial.println("*****"); // escribo los datos binarios q
//Serial.println("*****"); // escribo los datos binarios q
he leído (opcional) //Serial.print("val temp = ");
//Serial.print(valT); //Serial.print(" = ");
//Serial.print(valT, BIN); //Serial.print(", val hum = "); //
Serial.print(valH); // Serial.print(" = "); // Serial.print(valH,
BIN);

// datos adicionales (opcional) // Serial.print(" HR verdadera =
"); // Serial.print(RHtrue); // Serial.print(" %, T de rocío = ");
// Serial.print(Rocio); // Serial.println(" C"); // datos
importantes // Serial.print("Sensirion (T, HR) : ");

Serial.println(temp);

// Serial.print(" C "); // Serial.print(hum); // Serial.println("
%");

}

// calcula a humidade verdadeira

float calcularHumedadVerdadera(float T, int valH, float HR, int
numerobitsH) {

float t1 = 0.01; float t2 = 0.0; float HRtrue = 0.0;

if (numerobitsH == 12) {

// el numero de bits para medir la humedad son 8 o

t2 = 0.00008; }

else { t2 = 0.00128; }

```

```

HRtrue = (T - 25.0) * (t1 + t2 * valH) + HR;

return HRtrue;

}

// ler o estatus

void leerEstatus() {

int val= 0;

sendCommandSHT(comandoEstatus, dataPin, clockPin);

waitForResultSHT(dataPin);

val = getDataEstatus(dataPin, clockPin);

skipCrcSHT(dataPin, clockPin);

// Serial.print("estatus = "); // Serial.println(val, BIN);

}

```

II. Código do Módulo de Acionamento

```
int s=0;

int pin=22;

byte state=LOW;

void setup() {

  Serial.begin(9600);

  pinMode(pin,OUTPUT);

  pinMode(8,OUTPUT); }

void loop() {

  while(Serial.available()) {

    Serial.flush();

    s=Serial.read();

    if(s==1) { state=HIGH;

    }

    else { state=LOW; }

  }

  digitalWrite(pin,state);

}
```

III. Código do Algoritmo de Identificação (MatLab)

```
%algoritmo de identificação recursiva

clc

% in= Sinal;

in=acio;

out=T;

% out=lopass_butterworth(T,0.2,2,1);

% out=T_o; % out=t_o;

% fator de decimação a ser usado

fd=1;

% período de amostragem em segundos

Ts=fd*1;

% referir os dados ao ponto de operação em que foi feito o teste

[a b]=size(in);

[c d]=size(out);

u=in;

y=out;

% condições iniciais

P=eye(4)*10^6;

ini=3; % teta99(:,ini-1)=[1.918;-0.9191;-0.0219;0.0154];

teta99(:,ini-1)=[0.01;0.01;0.01;0.01];

xi(1,ini-1)=0;% vetor de resíduos

mse99=0;

yhat99(ini-1)=y(ini-1);% aqui a saída ainda não foi estimada

yhat99(ini-2)=y(ini-2);% então são usados valores medidos
```

```

% fator de esquecimento 0.835 e fd=1 %fator de esquecimento 0.8502 e fd=1 para
o sinal completo % lambda=0.8502;

lambda=0.9502;

erro99=mse99/(length(y));

%calcula o erro médio quadrático

% Algoritmo recursivo

For k=ini:length(y);

psi_k=[y(k-1);y(k-2);u(k-1);u(k-2)];

K_k=(P*psi_k)/((psi_k'*P*psi_k)+1);

teta99(:,k)=teta99(:,k-1)+K_k*(y(k)-psi_k'*teta99(:,k-1));

P=(1/lambda)*(P-(psi_k*psi_k'*P)/(psi_k'*P*psi_k+lambda));
yhat99(k)=[yhat99(k-1);yhat99(k-2);u(k);u(k-1)]'*teta99(:,k); utilizando dados
passados

xi(k)=y(k)-yhat99(k);% resíduo

mse99=(xi(k))^2+mse99;% esse somatório é utilizado para calcular o MSE

end;

teta=zeros(4,length(out));

for h=1:4 teta(h,:)=lopass_butterworth(teta99(h,:),0.05,Ts,2);%filtra os
parametros utilizando um filtro passa-baixas

end

% valor predito(estimado)

figure(1)

subplot(2,1,1);

plot(teta(1,:), 'b');

hold on;

plot(teta(2,:), 'r');

plot(teta(3,:), 'g');

plot(teta(4,:), 'y');

```

```

xlabel('Tempo (s)');

ylabel('Parâmetros Filtrados');

legend('\theta_{1}', '\theta_{2}', '\theta_{3}', '\theta_{4}');

subplot(2,1,2);

plot(teta99(1,:), 'b');

hold on

plot(teta99(2,:), 'r');

plot(teta99(3,:), 'g');

plot(teta99(4,:), 'y');

xlabel('Tempo (s)');

ylabel('Parâmetros');

legend('\theta_{1}', '\theta_{2}', '\theta_{3}', '\theta_{4}');

hold off

figure(2)

plot(y, 'b');

hold on

plot(yhat99, 'r')

hold off

xlabel('Tempo (s)')

ylabel('Temperatura')

legend('Medido filtrado', 'Simulado')

```

IV. Diagramas do Controle Adaptativo

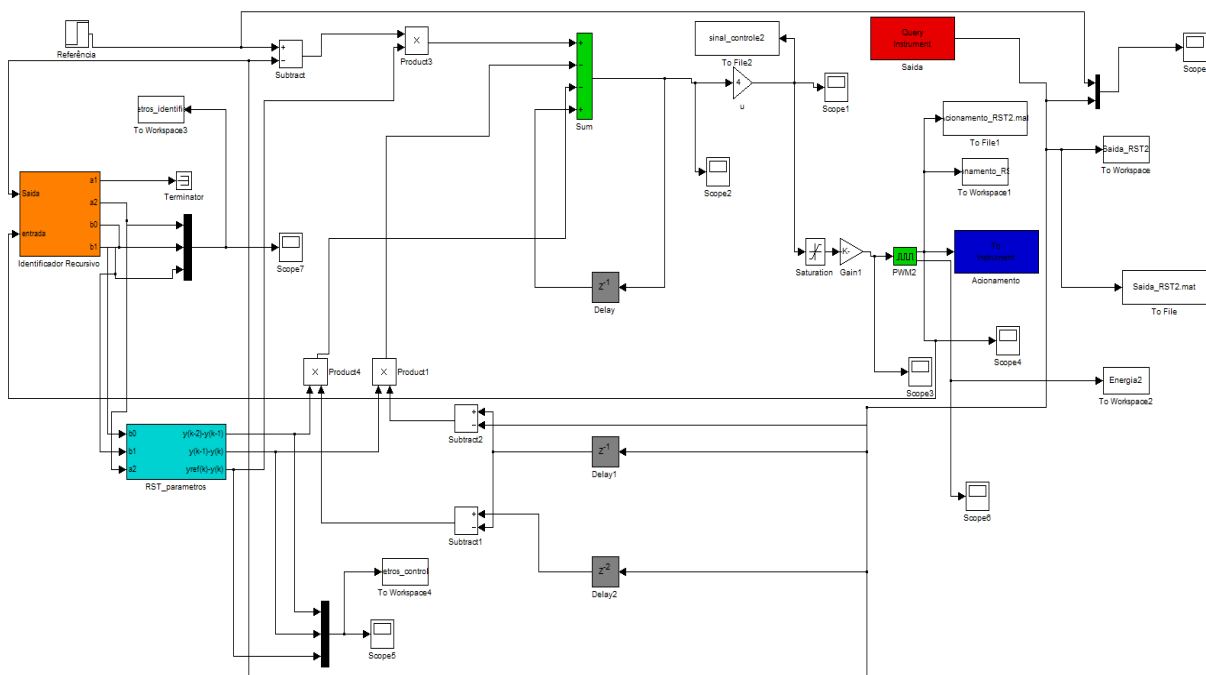


Figura IV-1. Diagrama simulink do controle adaptativo geral.

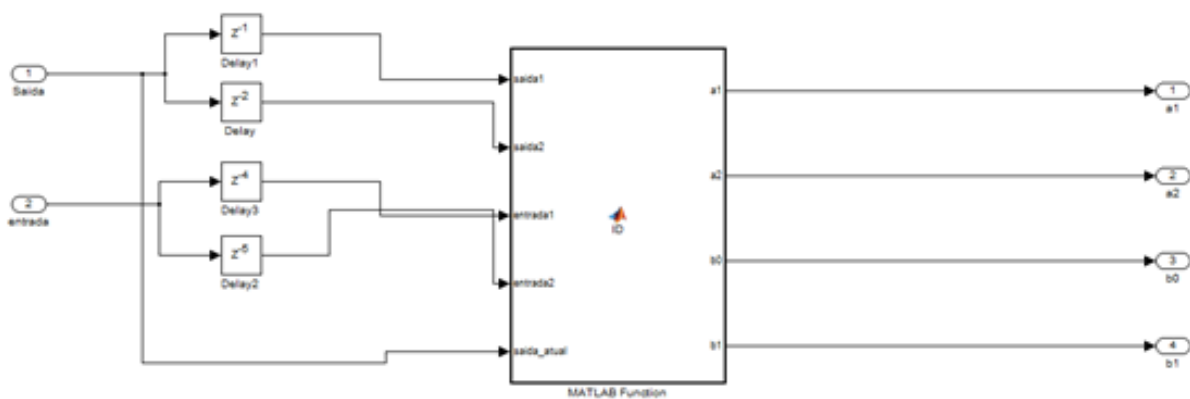


Figura IV-2. Diagrama simulink do processo de identificação.

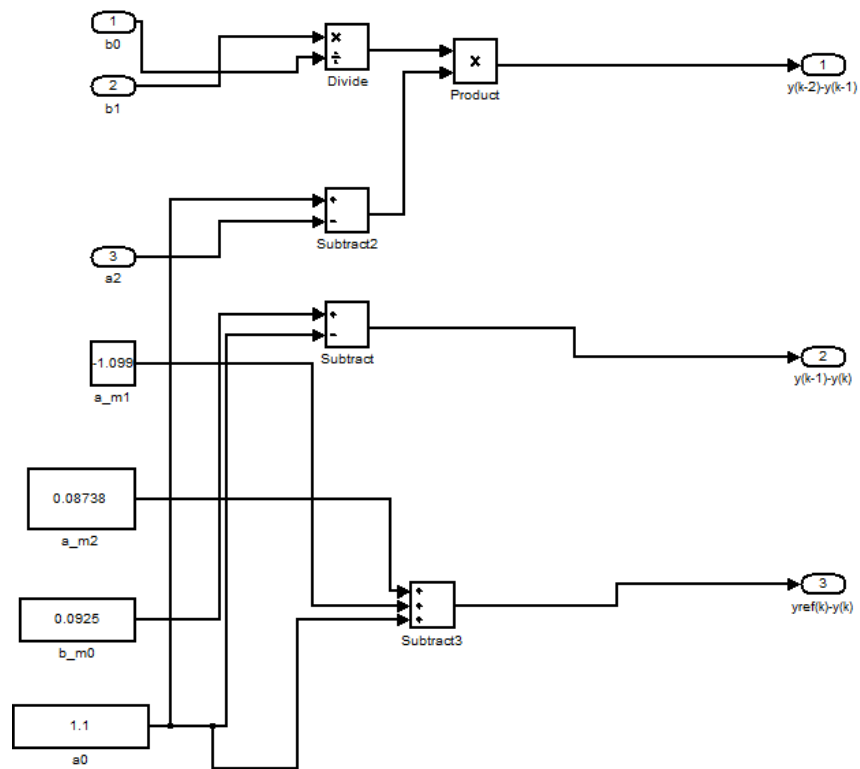


Figura IV-3. Diagrama de simulink da estimação de parâmetros.

V. Diagrama do Controle ON-OFF

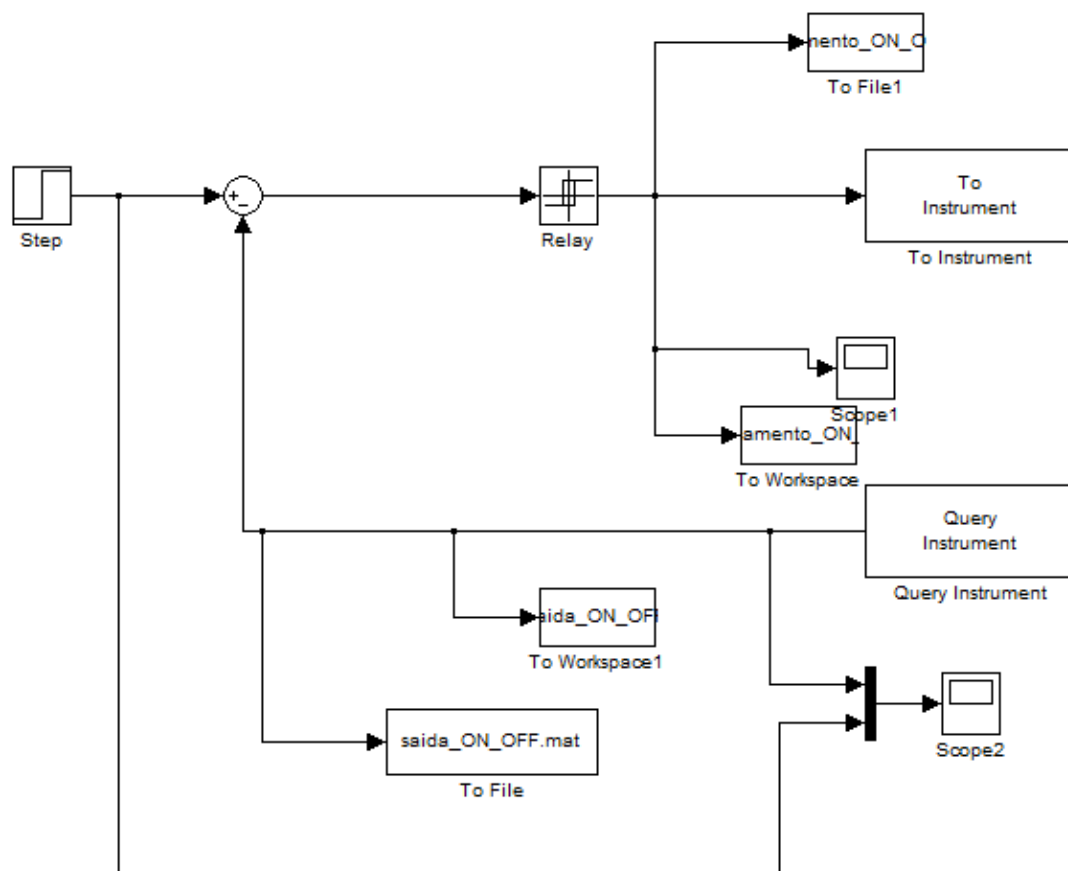


Figura V-1. Diagrama simulink do controle ON-OFF.

VI. Diagrama do Controle Proporcional Integrativo

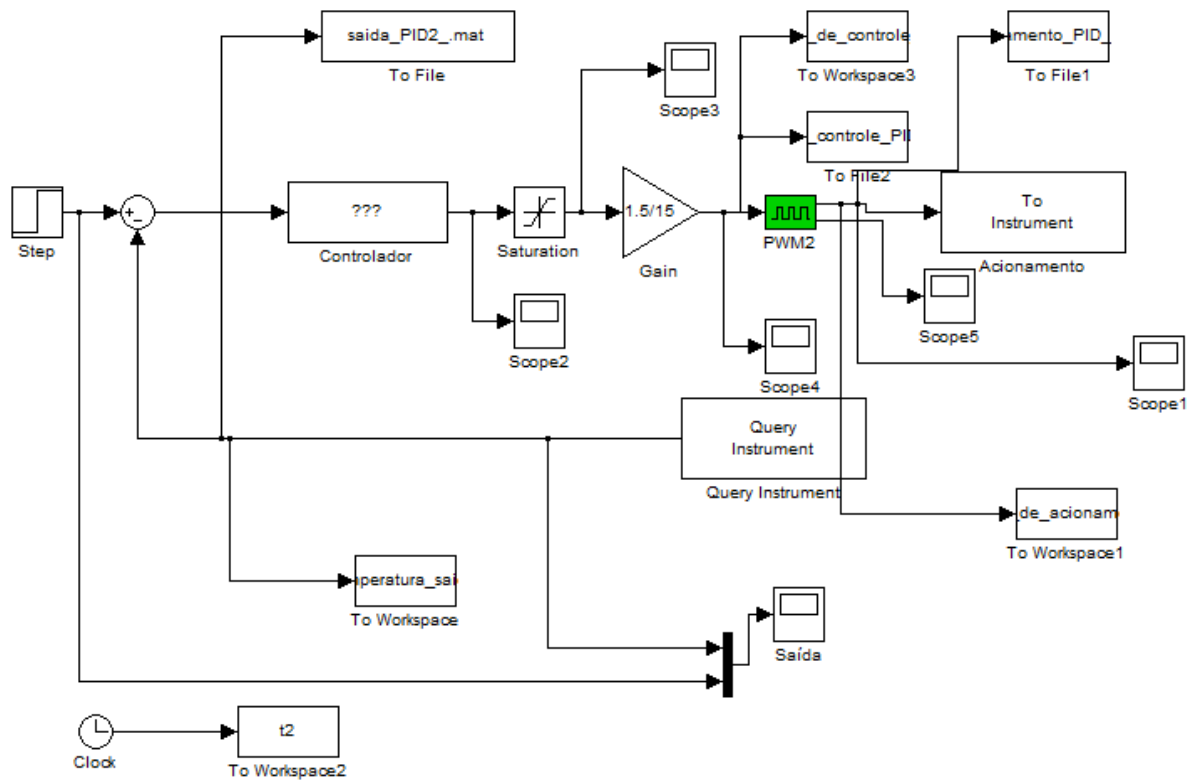


Figura VI-1. Diagrama simulink do controle PI.