

Nathan Luan Dutra Sarmiento

**Identificação e Controle de um Processo
Termoelétrico Multivariável em Escala
Laboratorial**

Campina Grande, Brasil

17 de março de 2016

Nathan Luan Dutra Sarmento

Identificação e Controle de um Processo Termoelétrico Multivariável em Escala Laboratorial

Trabalho de Conclusão de Curso submetido à Coordenação de Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande, Campus Campina Grande, como parte dos requisitos necessários para obtenção do título de Graduado em Engenharia Elétrica.

Univesidade Federal de Campina Grande - UFCG
Centro de Engenharia Elétrica e Informática - CEEI
Departamento de Engenharia Elétrica - DEE

Orientador: George Acioli Jr.

Campina Grande, Brasil

17 de março de 2016

Nathan Luan Dutra Sarmento

Identificação e Controle de um Processo Termoelétrico Multivariável em Escala Laboratorial/ Nathan Luan Dutra Sarmento. – Campina Grande, Brasil, 17 de março de 2016-

42 p. : il. ; 30 cm.

Orientador: George Acioli Jr.

Trabalho de Conclusão de Curso – Univesidade Federal de Campina Grande - UFCG

Centro de Engenharia Elétrica e Informática - CEEI

Departamento de Engenharia Elétrica - DEE , 17 de março de 2016.

Nathan Luan Dutra Sarmento

Identificação e Controle de um Processo Termoelétrico Multivariável em Escala Laboratorial

Trabalho de Conclusão de Curso submetido à Coordenação de Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande, Campus Campina Grande, como parte dos requisitos necessários para obtenção do título de Graduado em Engenharia Elétrica.

Trabalho aprovado. Campina Grande, Brasil, 17 de março de 2016:

George Acioli Jr.
Orientador

Professor
Convidado

Campina Grande, Brasil
17 de março de 2016

Este trabalho é dedicado à minha família.

Agradecimentos

Os agradecimentos deste trabalho são direcionados:

- Ao Pai Celestial, pela força e perseverança, pela capacidade e sabedoria, e por ser Aquele que ouvia minhas preces e lamentações nos momentos mais íntimos;
- Aos professores George Acioli e Péricles Barros, pelos exemplos de pessoa, orientação, confiança, e por estar sempre a disposição para qualquer problema que viesse a ocorrer;
- A Simões, pela grande ajuda ofertada em todos os sentidos, desde montagens realizadas a ideias compartilhadas;
- Aos amigos do LIEC, pelas boas horas de companhia, bom humor, ajuda mútua e companheirismo;
- À minha família, pelo apoio, estímulo e por nunca desistirem de mim, não apenas durante esses cinco anos, mas durante toda a minha vida;
- A todos os amigos e colegas conquistados nestes cinco anos de curso e durante a vida, por me mostrarem que, independente da situação e do momento, sempre é possível obter alegria e risos;
- A todos os professores e a todas as pessoas anônimas que, de forma direta ou indireta, contribuíram para a realização deste trabalho.

"Tenham bom ânimo. O futuro é tão brilhante quanto sua fé."

Thomas S. Monson

Resumo

Módulos Peltier são atuadores que utilizam energia elétrica para gerar uma diferença de temperatura entre as suas faces. A vantagem da utilização de um módulo Peltier para controle de temperatura está na facilidade do seu uso. Visando um uso mais eficiente do Peltier, é necessário implementar uma boa estratégia de controle. Portanto, o objetivo deste trabalho é levantar modelos de um sistema multivariável de módulos Peltier para realização do controle de temperatura destes módulos, usando técnicas de desacoplamento de sistemas multivariáveis.

Palavras-chave: Módulo Peltier; Micro Framework; Controle; Identificação; Desacoplamento

Abstract

Peltier modules are actuators that use electricity to generate a difference temperature between their faces. The advantage of using a Peltier module to control temperature is in the ease of use. in order to achieve a more efficient use of Peltier modules, it is necessary to implement a good control strategy. Therefore, the objective of this work is to raise models of a multivariable system of Peltier modules for performing the temperature control of these modules using decoupling techniques for multivariable systems

Key-words: Peltier Module; Micro Framework; Control; Identification; Decoupling

Lista de ilustrações

Figura 1 – Módulo Termoelétrico baseado no efeito Peltier	1
Figura 2 – Sistema de Controle com Desacoplamento	8
Figura 3 – Sistema de Controle com desacoplador	9
Figura 4 – Sistema de Controle com desacoplador	13
Figura 5 – USBizi	14
Figura 6 – Diagrama de Blocos do Sistema de Controle	15
Figura 7 – Montagem presente no Laboratório de Controle e Otimização	16
Figura 8 – Janela para configurar a porta serial	18
Figura 9 – Janela para trabalhar em malha aberta	18
Figura 10 – Janela para trabalhar em malha aberta	19
Figura 11 – Diagrama de blocos da lógica de programação	20
Figura 12 – Tela inicial do programa criado	21
Figura 13 – Comparação entre sistema real e modelos identificados para o método <i>Time Integral Approach</i>	23
Figura 14 – Comparação entre sistema real e modelos identificados para o método <i>Time Integral Approach</i> com variável instrumental	24
Figura 15 – Comparação entre sistema real e modelos identificados para o método <i>Time Integral Approach</i>	24
Figura 16 – Comparação entre sistema real e modelos identificados para o método <i>Time Integral Approach</i> com variável instrumental	25
Figura 17 – Projeto do Controlador	27
Figura 18 – Diagrama para simulação no Simulink	28
Figura 19 – Temperaturas dos módulos Peltier	29
Figura 20 – Temperaturas dos módulos Peltier	30
Figura 21 – Influência das entradas PWM1 e PWM2 no módulo Peltier 1	32
Figura 22 – Influência das entradas PWM1 e PWM2 no módulo Peltier 2	32
Figura 23 – Influência das entradas PWM1 e PWM2 no módulo Peltier 1	33
Figura 24 – Influência das entradas PWM1 e PWM2 no módulo Peltier 2	34
Figura 25 – Influência das entradas PWM1 e PWM2 no módulo Peltier 1	35
Figura 26 – Influência das entradas PWM1 e PWM2 no módulo Peltier 2	35
Figura 27 – Controle sem desacoplamento	37
Figura 28 – Temperaturas dos módulos Peltier com estratégia de controle sem de- sacoplamento	37
Figura 29 – Influências do desacoplamento nas entradas e saídas cruzadas	38
Figura 30 – Controle sem desacoplamento	39

Figura 31 – Temperaturas dos módulos Peltier com estratégia de controle com desacoplamento	39
--	----

Lista de abreviaturas e siglas

TEM	Módulo termo-elétrico
CPU	Central de processamento único
MIMO	Multiple-input and multiple-output
TITO	Two-input two-output
UFCG	Universidade Federal de Campina Grande
LIEC	Laboratório de Instrumentação Eletrônica e Controle
FOPDT	First Order Plus Dead-Time
LS	Least Square
PWM	Pulse Width Modulation
CAN	Controller Area Network
SNR	Signal-Noise Ratio
PI	Proportional Integral

Lista de símbolos

τ	Constante de tempo da função de transferência
y	Saída do sistema
K	Ganho da função de transferência
u	Entrada do Sistema
θ	Atraso da função de transferência
Z	Variável instrumental
D	Matriz de desacoplamento
H	Matriz resultante após o desacoplamento
G	Matriz de funções de transferência do sistema
C	Matriz de controladores
I	Matriz identidade
τ_l	Constante de tempo maior
τ_s	Constante de tempo menor
k_p	Ganho proporcional
k_i	Ganho integral
T_s	Tempo de amostragem
e	Erro

Sumário

1	INTRODUÇÃO	1
2	FUNDAMENTAÇÃO TEÓRICA	3
2.1	Identificação de Sistemas	3
2.1.1	<i>Time Integral Approach</i>	3
2.1.2	<i>Time Integral Approach</i> utilizando variável instrumental	4
2.1.3	Métodos Robustos diante de condições iniciais não nulas	5
2.1.4	Modelo robusto usando variável instrumental	6
2.2	Desacoplamento de Malhas	6
2.2.1	Revisão Bibliográfica	6
2.3	Aspectos Gerais do Desacoplamento	7
2.3.1	Técnica Nordfelt	10
2.3.2	Técnica Wang	11
2.3.3	Técnica de Desacoplamento Invertido	12
3	MATERIAIS E MÉTODS	14
3.1	Equipamentos e Montagem Física da Planta	14
3.2	A linguagem C#, .NET Micro Framework e Interface de Comunicação com o usuário	15
3.2.1	Lógica de programação	18
4	RESULTADOS EXPERIMENTAIS - IDENTIFICAÇÃO	22
4.1	Relação sinal-ruído alta	22
4.2	Relação sinal-ruído baixa	23
4.3	Modelos identificados	25
5	RESULTADOS EXPERIMENTAIS - CONTROLE COM <i>TOOLBOX</i> DO SIMULINK	27
6	SIMULAÇÃO - DESACOPLAMENTO DAS MALHAS	31
6.1	Desacoplamento por Nordfelt	31
6.2	Desacoplamento por Wang	32
6.3	Desacoplamento Invertido	34
7	SIMULAÇÃO - CONTROLE USANDO TÉCNICA DE SINTONIA E DESACOPLAMENTO	36

8	CONSIDERAÇÕES FINAIS	40
	REFERÊNCIAS	41

1 Introdução

O módulo termoelétrico (TEM) é um atuador que utiliza energia elétrica para gerar uma diferença de temperatura entre as suas faces. Normalmente consiste de um arranjo de pastilhas feitas de um material semiconductor (tipo p e tipo n) que estão arranjadas eletricamente em série e termicamente em paralelo. Na [Figura 1](#) é mostrado o esquema de um módulo termoelétrico baseado no efeito Peltier.

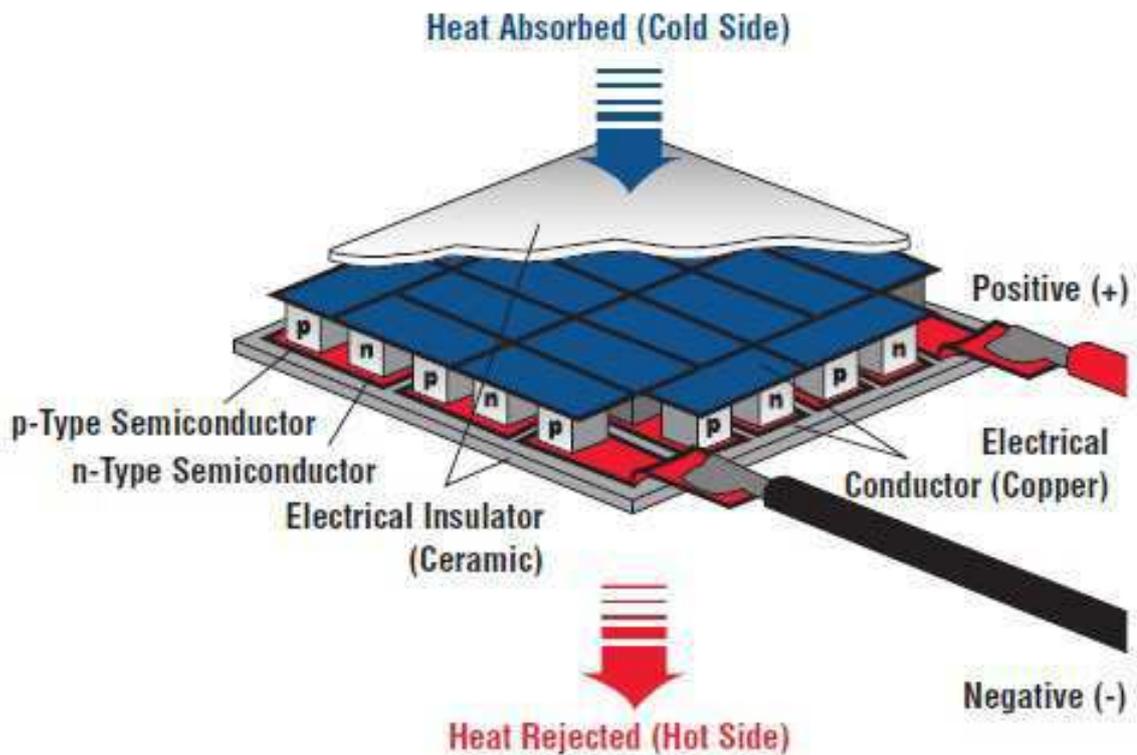


Figura 1 – Módulo Termoelétrico baseado no efeito Peltier

O TEM pode atuar de duas formas: como um resfriador (TEC) ou como um gerador (TEG). Como um resfriador, o TEM pode ser usado para gerenciamento de temperatura e controle de dispositivos microeletrônicos tais como diodo emissor de laser e CPUs. Como um gerador, o TEM pode ser usado para produzir energia em regiões remotas onde gradientes de temperatura estão disponíveis.

A vantagem da utilização de um TEM para controle de temperatura ou geração de energia está na facilidade do seu uso. Dentre os módulos termoelétricos, destacam-se os módulos Peltier. Por não ter partes móveis e não requerer conhecimentos prévios para sua instalação e manutenção, os módulos de Peltier tem se tornado, cada vez mais, fortes candidatos a uso nestas áreas.

Além disso, a maioria dos processos industriais são multivariáveis (MIMO - *multiple-input, multiple-output*) (1). Dentre esses, os que mais prevalecem são os sistemas TITO (*two-input, two-output*), seja porque o processo apresenta essa configuração ou porque um processo complexo foi decomposto em blocos 2x2 (2). Cada par variável manipulada/variável de processo (entrada/saída) define uma malha de controle do sistema.

Devido a importância crescente dos módulos Peltier no controle de temperatura, assim como a ocorrência de processos TITO nos mais diversos sistemas existentes nas indústrias e laboratórios, o Laboratório de Otimização e Controle da UFCG, pertencente ao Laboratório de Instrumentação Eletrônica e Controle (LIEC), possui uma planta montada para o controle de dois módulos peltier ligados por uma superfície metálica. O objetivo principal desta planta é o estudo, a nível acadêmico, de técnicas de modelagem, identificação e sintonia de controladores para esse sistema.

O objetivo geral deste trabalho, portanto, é o estudo e implementação de técnicas de identificação de sistemas, desacoplamento e controle digital de dois módulos Peltier, inclusos na planta acima citada. Além disso, inclui-se neste projeto a criação de uma interface de interação com o usuário, de forma que seja possível alterar, via computador, os valores desejados para temperatura.

Como objetivos específicos deste projeto, destacam-se:

- Estudo da modelagem de módulos termoeletricos e das técnicas de desacoplamento em sistemas MIMO;
- Estudo das técnicas de identificação de sistemas para obtenção de modelos para diversos sistemas termoeletricos;
- Uso da linguagem de programação C# para criação de uma interface de comunicação com o usuário;
- Uso da ferramenta .NET Micro Framework da linguagem C# para programação de um microcontrolador para implementação do controle;

Serão apresentados neste relatório uma breve introdução teórica sobre os temas a serem abordados, os materiais utilizados e os métodos desenvolvidos, assim como alguns resultados experimentais do sistema feito, de forma que através desse Trabalho de Conclusão de Curso, foram aplicados diversos conhecimentos e aprendizados adquiridos ao longo do curso de graduação em Engenharia Elétrica.

2 Fundamentação Teórica

Nesta seção, é feita uma breve revisão bibliográfica sobre as técnicas de identificação, desacoplamento e sintonia de controladores utilizadas como base para o estudo feito.

2.1 Identificação de Sistemas

Identificação de Sistemas é um termo genérico utilizado para descrever as ferramentas matemáticas e os algoritmos que permitem construir modelos dinâmicos a partir de dados medidos. Sistemas térmicos, em geral, possuem uma dinâmica lenta em comparação com outros processos, como por exemplo, os elétricos. Modelos de primeira ordem com atraso (*First Order Plus Deadtime* - FOPDT), nestes casos, conseguem descrever bem seu comportamento, embora nem sempre a modelagem dos mesmos seja de fácil implementação. Assim, o objetivo deste artigo é expor alguns métodos de identificação de sistemas aplicados a uma plataforma de testes presente no Laboratório de Instrumentação Eletrônica e Controle (LIEC) da Universidade Federal de Campina Grande (UFCG). Dentre as diversas técnicas já desenvolvidas para identificação de sistemas FOPDT, foram escolhidas 4 para teste na plataforma experimental. Estas técnicas são descritas em (3), e são mostradas a seguir.

2.1.1 *Time Integral Approach*

Queremos identificar um sistema de primeira ordem:

$$\tau_p \dot{y}(t) + y(t) = k_p u(t - \theta); t \geq \theta, u(t) = h \quad (2.1)$$

Integrando ambos os lados:

$$\int_0^t y(\tau) d\tau = k_p h(t - \theta) - \tau_p y(t) \quad (2.2)$$

Podemos reformular a equação:

$$\psi(t) = \phi^T(t) \gamma \quad (2.3)$$

Onde:

$$\begin{cases} \psi(t) = \int_0^t y(\tau) d\tau \\ \phi(t) = [ht, -h, -y(t)]^T \\ \gamma = [k_p, k_p\theta, \tau_p] \end{cases} \quad (2.4)$$

Tomando-se os tempos de amostragem $\theta < t_1 < t_2 < \dots < t_N$, construímos as matrizes:

$$\Psi = [\psi(t_1), \psi(t_2), \dots, \psi(t_N)]^T \quad (2.5)$$

$$\Phi = [\phi(t_1), \phi(t_2), \dots, \phi(t_N)]^T \quad (2.6)$$

Um algoritmo LS (mínimos quadrados) pode ser estabelecido para estimação de parâmetros:

$$\Psi = \Phi\gamma \quad (2.7)$$

Onde o vetor de parâmetros pode ser resolvido:

$$\gamma = (\Phi^T\Phi)^{-1}\Phi^T\Psi \quad (2.8)$$

E, por fim:

$$\begin{bmatrix} k_p \\ \theta \\ \tau_p \end{bmatrix} = \begin{bmatrix} \gamma(1) \\ \gamma(2)/\gamma(1) \\ \gamma(3) \end{bmatrix} \quad (2.9)$$

2.1.2 *Time Integral Approach* utilizando variável instrumental

Na presença de ruído de medição, uma variável instrumental pode ser usada. Escolhemos uma matriz Z tal que:

1. O inverso de $\lim_{N \rightarrow \infty} (Z^T\Phi)/N$ existe
2. $\lim_{N \rightarrow \infty} (Z^T v)/N = 0$, onde $v = [v(t_1), v(t_2), \dots, v(t_N)]^T$

Uma escolha razoável para Z é:

$$Z = \begin{bmatrix} t_1 & -1 & \frac{1}{t_1} \\ t_2 & -1 & \frac{1}{t_2} \\ \vdots & \vdots & \vdots \\ t_N & -1 & \frac{1}{t_N} \end{bmatrix} \quad (2.10)$$

A estimação de parâmetros, então, é dada por:

$$\gamma = (Z^T \Phi)^{-1} Z^T \Psi \quad (2.11)$$

Lembrando que os parâmetros do modelo podem ser encontrados segundo a equação 2.9.

2.1.3 Métodos Robustos diante de condições iniciais não nulas

Embora os métodos descritos anteriormente sejam de simples aplicação, um método de identificação mais robusto pode ser desejável diante das seguintes questões:

- Dificuldade em saber se o processo está em estado estacionário para realização de um teste de resposta ao degrau
- Os processos são lentos
- As condições iniciais desconhecidas

Sendo assim, inicialmente, definimos uma integral de ordem n da seguinte forma:

$$\int_{[0,t]}^{(m)} f(t) = \int_0^t \int_0^{\tau_{m-1}} \cdots \int_0^{\tau_1} f(\tau_0) d\tau_0 d\tau_1 \cdots d\tau_{m-1}, m \geq 2 \quad (2.12)$$

Tomando o modelo de primeira ordem definido na equação 2.1 e integrando duas vezes, obtemos:

$$\int_{[0,t]}^{(2)} y(t) = k_p h \left(\frac{t^2}{2} - \theta t + \frac{\theta^2}{2} \right) + \tau_p y(0) t - \tau_p \int_0^t y(\tau) d\tau \quad (2.13)$$

Podemos utilizar o algoritmo LS (mínimos quadrados):

$$\begin{cases} \psi(t) = \int_{[0,t]}^{(2)} y(t) \\ \phi(t) = [-\int_0^2 y(\tau) d\tau, ht^2/2, t, h/2]^T \\ \gamma = [\tau_p, k_p, -hk_p\theta + \tau_p y(0), k_p\theta^2] \end{cases} \quad (2.14)$$

Assim, os parâmetros do modelo do sistema podem ser encontrados:

$$\begin{bmatrix} \tau_p \\ k_p \\ \theta \end{bmatrix} = \begin{bmatrix} \gamma(1) \\ \gamma(2) \\ \sqrt{\frac{\gamma(4)}{k_p}} \end{bmatrix} \quad (2.15)$$

Veja que o parâmetro θ deve ser encontrado através de $\gamma(4)$, e não $\gamma(3)$, pois supõe-se que as condições iniciais do sistema são desconhecidas.

2.1.4 Modelo robusto usando variável instrumental

Para lidar com ruído de medição, sugere-se uma variável instrumental para identificação de um processo de ordem n

$$z(t) = \left[-\int_0^t \hat{y}(\tau) d\tau, -\int_{[0,t]}^{(2)} \hat{y}(\tau), \dots, -\int_{[0,t]}^{(n)} \hat{y}(\tau), \right. \\ \left. \frac{ht^n}{n!}, \frac{ht^{n-1}}{(n-1)!}, \dots, t, 1 \right] \quad (2.16)$$

Desta forma, procede-se segundo as equações 2.11 e 2.15 para encontrar os parâmetros do modelo do sistema.

2.2 Desacoplamento de Malhas

2.2.1 Revisão Bibliográfica

Um dos primeiros trabalhos propostos de desacoplamento foi introduzido por Loomis em (4), no qual o autor se dedica ao estudo de técnicas de desacoplamento em sistemas de controle multiloop. Inicialmente os problemas de desacoplamento eram tratados por uma abordagem em matrizes de funções de transferência. Liu e Bergman em (5) e Tripathi em (6) desenvolveram trabalho em relação às condições necessárias para realização do desacoplamento em sistemas multivariáveis. Estas condições estão relacionadas ao posto das matrizes de controle e saída do sistema. Uma condição equivalente para sistemas descritos por matriz de funções de transferência foi descrita por Gilbert e Pivnichny em (7).

Morgan em (8) introduziu o desacoplamento utilizando uma abordagem em espaços de estados. Tripathi em (9) estudou o efeito da variação de parâmetros em sistemas multivariáveis, lineares e invariantes no tempo. Liu, Wang e Ho em (10) estudaram o problema do desacoplamento de blocos de entrada e saída por realimentação de estados em sistemas lineares, onde propôs dois algoritmos para lidar com o problema e constrói, ao fim, uma lei de realimentação que corresponde a um sistema em malha fechada regular, livre de impulsos e não-interativo.

Nazar e Rekasius em (11) começam a trabalhar com desacoplamento de sistemas não-lineares, e fornecem uma condição necessária para o desacoplamento de uma classe

de sistemas não-lineares. Abed e Kuh em (12) mostram que nesta classe de sistemas, os mesmos podem ser desacoplados por pré-compensação dinâmica e realimentação de estados se, e somente se, ele for invertível. Sinha em (13) propõe uma teoria generalizada de controle não-interativo por realimentação de estados de modo que o desacoplamento de sistemas lineares e não lineares possam ser estudados sob a mesma ótica.

Lloyd em (14) começa a discutir sobre desacoplamento em sistemas discretos, visto que até o momento pouco havia sido pesquisado sobre o assunto. Em seu artigo é apresentada uma simples derivação das condições de desacoplamento para sistemas multivariáveis de tempo discreto, assim como mostra resultados experimentais obtidos.

Ao considerar-se o atraso nos sistemas (visto que até então não havia sido feita essa consideração), a análise se torna mais complexa quando o sistema é multivariável, cujas malhas possuem diferentes atrasos e, com isso, uma malha pode ter sua saída afetada por outras malhas em tempos diferentes. Como consequência, o processo deve ser representado por uma matriz de funções de transferência com múltiplos atrasos. Wang em (15), apresenta-se um projeto com desacoplamento utilizando matriz de função de transferência. Neste trabalho, um controlador descentralizado é primeiramente projetado baseado em ‘processos equivalentes’, e os elementos não-diagonais do controlador são determinados com vistas a obter o desacoplamento.

Mais recentemente, trabalhos relacionados à técnicas inteligentes de desacoplamento tem sido propostas. Fu, Chai, Su e Wang em (16) propoem um método inteligente de desacoplamento usando múltiplos modelos e redes neurais. Esse método inclui um conjunto de desacopladores fixados, um controlador de desacoplamento adaptativo reinicializado com redes neurais assim como um controlador de desacoplamento sem as mesmas. Igualmente, Wang, Chai, J. Fu, Zhang e Y. Fu em (17) propoem outras topologias de desacoplamento inteligente baseado em redes neurais e controle adaptativo. Lin e Mon em (18) propoem um método de controle de desacoplamento através do uso de lógica fuzzy hierárquica deslizante, visando então a estabilidade do sistema no desacoplamento de sistemas não-lineares.

2.3 Aspectos Gerais do Desacoplamento

Quando lidamos com sistemas multivariáveis, especialmente em malha fechada, o grau de dificuldade do controle das plantas cresce devido à interação entre as malhas e por estes tipos de sistemas apresentarem maior número de parâmetros a serem ajustados. Assim, a sintonia do controlador de uma malha afeta o desempenho das outras, e vice-versa. Isso pode ser problemático em processos industriais e laboratoriais.

Portanto, é de interesse investigar formas de lidar com a interação entre malhas (19). Uma abordagem alternativa para o projeto de controladores MIMO em sistemas com

forte interação é utilizar uma estrutura de controle com desacoplamento. Essa abordagem decompõe o problema em duas partes: primeiramente, realiza-se o desacoplamento do sistema para minimizar as interações ou tornar o sistema diagonal dominante; daí, então, projeta-se o controlador MIMO de forma descentralizada (20). A ideia do desacoplamento é ilustrada na [Figura 2](#).

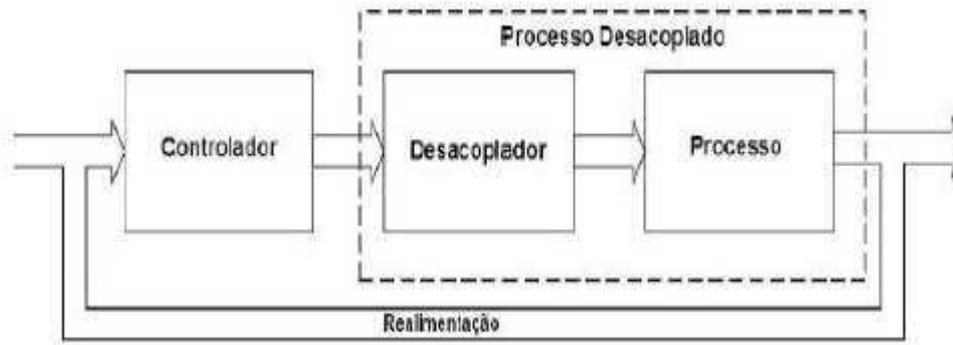


Figura 2 – Sistema de Controle com Desacoplamento

Sendo G a matriz de funções de transferências do sistema, e C a matriz dos controladores a serem projetados, o objetivo do projetista é encontrar uma matriz D , chamada de matriz de desacoplamento, tal que a matriz resultante do sistema com desacoplamento seja uma matriz diagonal. Matematicamente:

Sendo as seguintes matrizes:

$$D(s) = \begin{bmatrix} D_{11}(s) & D_{12}(s) \\ D_{21}(s) & D_{22}(s) \end{bmatrix} \quad (2.17)$$

$$G(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \quad (2.18)$$

Queremos encontrar a matriz diagonal H , tal que:

$$H(s) = G(s)D(s) = \begin{bmatrix} H_{11}(s) & 0(s) \\ 0(s) & H_{22}(s) \end{bmatrix} \quad (2.19)$$

Sendo a matriz de controladores C :

$$C = \begin{bmatrix} C_1(s) & 0(s) \\ 0(s) & C_2(s) \end{bmatrix} \quad (2.20)$$

Esquemáticamente, o sistema em malha fechada é mostrado na [Figura 3](#)

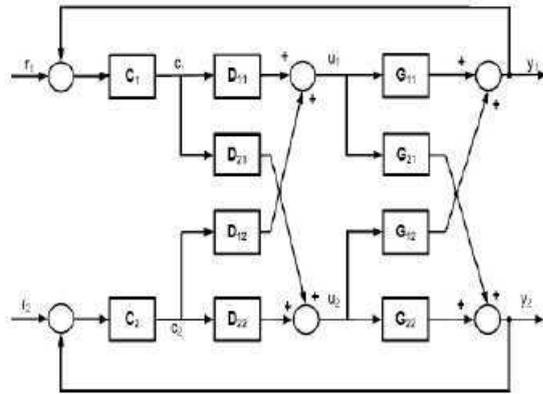


Figura 3 – Sistema de Controle com desacoplador

Assim, de uma forma geral, podemos representar a matriz de desacoplamento da seguinte forma:

$$D(s) = G^{-1}(s)H(s) \tag{2.21}$$

$$D(s) = \frac{adj(G(s))}{det(G(s))}H(s) \tag{2.22}$$

Na literatura, a forma mais comum de classificar o desacoplador é a seguinte: desacoplamento ideal, desacoplamento simplificado, e desacoplamento invertido (1). Cada uma das topologias possui vantagens e desvantagens. O desacoplamento ideal facilita o projeto do controlador, mas é raramente utilizado na prática devido aos complicados elementos do desacoplador, o que dificulta sua realização. O desacoplamento simplificado apresenta um desacoplador simples, mas o processo desacoplado resultante é complexo (20). Com isso, o controlador não pode ser projetado diretamente sem introduzir outras aproximações. O desacoplamento invertido, também chamado de desacoplamento *feed-forward*, evita as desvantagens do simplificado e alcança o propósito do ideal. Além disso, outros benefícios do desacoplamento invertido são: (i) é mais fácil de lidar com o problema de *bumpless transfer* do controlador; (ii) a saturação das variáveis manipuladas é considerada diretamente na estrutura de controle com desacoplamento invertido (1).

De posse desta fundamentação inicial, e conhecendo estes tipos principais de desacopladores, foram estudadas três diferentes métodos de desacoplamento de malhas para sistemas TITO, um considerado simples ou ideal, outro simplificado e o último invertido. Assim, deve ser verificada a funcionalidade dos diferentes tipos de desacopladores em um projeto de controle.

2.3.1 Técnica Nordfelt

O método a ser desenvolvido está descrito em (21). Propõe-se uma técnica de projeto para o desacoplador baseado em um modelo de matriz de funções de transferência TITO do sistema. Assume-se que o sistema TITO é linear, estável e não-singular. O foco principal da técnica é projetar o desacoplador adicionando o mínimo de dinâmica e atraso para o sistema. É importante também que o desacoplador não tenha característica passa alta. Além disso, propõe-se que a técnica seja simples para que possa ser facilmente automatizada. O projeto do desacoplador baseia-se na proposição de que todas as matrizes D que tornam o sistema GD diagonal podem ser obtidas da seguinte equação:

$$D(s) = \frac{\text{adj}(G(s))}{K} \quad (2.23)$$

Os diferentes desacopladores possíveis baseiam-se em diferentes matrizes K na Equação 2.23. Com isso, K deve ser definido para que o projeto do desacoplador atenda ao requisito de adicionar minimamente dinâmica e atraso ao sistema desacoplado.

Como as funções de transferência são de primeira ordem, temos que:

$$G_{fopdt}(s) = \begin{bmatrix} \frac{k_{11}}{1+s\tau_{11}}e^{-s\theta_{11}} & \frac{k_{12}}{1+s\tau_{12}}e^{-s\theta_{12}} \\ \frac{k_{21}}{1+s\tau_{21}}e^{-s\theta_{21}} & \frac{k_{22}}{1+s\tau_{22}}e^{-s\theta_{22}}(s) \end{bmatrix} \quad (2.24)$$

$$\text{adj}(G_{fopdt}(s)) = \begin{bmatrix} \frac{k_{22}}{1+s\tau_{22}}e^{-s\theta_{22}} & -\frac{k_{12}}{1+s\tau_{12}}e^{-s\theta_{12}} \\ -\frac{k_{21}}{1+s\tau_{21}}e^{-s\theta_{21}} & \frac{k_{11}}{1+s\tau_{11}}e^{-s\theta_{11}}(s) \end{bmatrix} \quad (2.25)$$

O projeto do desacoplador segue o seguinte algoritmo:

1. Comece com $K = I$, logo $D = \text{adj}(G_{fopdt}(s))$;
2. Remova os maiores atrasos comuns de cada coluna do desacoplador através da multiplicação do elemento da diagonal principal de K correspondente à coluna pelo inverso do atraso;
3. Nos elementos de cada coluna do desacoplador, observe o que tem maior constante de tempo e utilize a seguinte aproximação:

$$\frac{1}{\tau_l s + 1} \approx \frac{1}{(\tau_l s + 1)((\tau_l - \tau_s)s + 1)} \quad (2.26)$$

onde τ_l é a maior constante de tempo da coluna e τ_s é a menor constante de tempo da coluna;

4. Remova os polos comuns de cada coluna do desacoplador através da multiplicação do elemento da diagonal principal de K correspondente à coluna pelo inverso desse polo;

5. simplifique os ganhos de cada coluna do desacoplador, no sentido de adicionar minimamente dinâmica ao processo.

2.3.2 Técnica Wang

O método a ser desenvolvido está descrito em (22). Propõe-se o projeto de um desacoplador simplificado baseado em um modelo FOPTD do sistema. Nesse projeto, a estrutura do desacoplador é escolhida da seguinte forma:

$$D = \begin{bmatrix} 1 & d_{12} \\ d_{21} & 1 \end{bmatrix} \quad (2.27)$$

Com isso, o sistema resultante é dado por:

$$G_{fopdt}(s)D(s) = \begin{bmatrix} \frac{k_{11}}{1+s\tau_{11}}e^{-s\theta_{11}} & \frac{k_{12}}{1+s\tau_{12}}e^{-s\theta_{12}} \\ \frac{k_{21}}{1+s\tau_{21}}e^{-s\theta_{21}} & \frac{k_{22}}{1+s\tau_{22}}e^{-s\theta_{22}}(s) \end{bmatrix} \begin{bmatrix} 1 & d_{12} \\ d_{21} & 1 \end{bmatrix} \quad (2.28)$$

$$G_{fopdt}(s)D(s) = \begin{bmatrix} \frac{k_{11}}{1+s\tau_{11}}e^{-s\theta_{11}} + \frac{k_{12}}{1+s\tau_{12}}e^{-s\theta_{12}}d_{21}(s) & \frac{k_{12}}{1+s\tau_{12}}e^{-s\theta_{12}} + \frac{k_{11}}{1+s\tau_{11}}e^{-s\theta_{11}}d_{12}(s) \\ \frac{k_{21}}{1+s\tau_{21}}e^{-s\theta_{21}} + \frac{k_{22}}{1+s\tau_{22}}e^{-s\theta_{22}}(s)d_{21}(s) & \frac{k_{22}}{1+s\tau_{22}}e^{-s\theta_{22}}(s) + \frac{k_{21}}{1+s\tau_{21}}e^{-s\theta_{21}}d_{12}(s) \end{bmatrix} \quad (2.29)$$

Para que o sistema resultante seja considerado desacoplado, este deve ser diagonal, logo:

$$\frac{k_{12}}{1+s\tau_{12}}e^{-s\theta_{12}} + \frac{k_{11}}{1+s\tau_{11}}e^{-s\theta_{11}}d_{12}(s) \quad (2.30)$$

$$\frac{k_{21}}{1+s\tau_{21}}e^{-s\theta_{21}} + \frac{k_{22}}{1+s\tau_{22}}e^{-s\theta_{22}}(s)d_{21}(s) \quad (2.31)$$

Dessa forma, temos:

$$d_{12}(s) = -\frac{k_{12}(\tau_{11}s + 1)}{k_{11}(\tau_{12}s + 1)}e^{-(\theta_{12}-\theta_{11})s} \quad (2.32)$$

$$d_{21}(s) = -\frac{k_{21}(\tau_{22}s + 1)}{k_{22}(\tau_{21}s + 1)}e^{-(\theta_{21}-\theta_{22})s} \quad (2.33)$$

Observe que se $(\theta_{12} - \theta_{11}) < 0$ ou $(\theta_{21} - \theta_{22}) < 0$ leva a $D(s)$ ser não causal e precisa ser modificado. No caso de $(\theta_{21} - \theta_{22}) < 0$, a primeira coluna do desacoplador pode ser multiplicada pelo termo $e^{(\theta_{21}-\theta_{22})s}$ tornando o termo d_{21} realizável. De forma semelhante, a segunda coluna pode ser multiplicada por $e^{(\theta_{12}-\theta_{11})s}$, para tornar o termo

d_{12} realizável no caso de $(\theta_{12} - \theta_{11}) < 0$. Com isso, o projeto do desacoplador geral pode ser escrito, como segue:

$$D(s) = \begin{bmatrix} e^{-v(\theta_{22}-\theta_{21})s} & -\frac{k_{12}(\tau_{11}s+1)}{k_{11}(\tau_{12}s+1)}e^{-v(\theta_{12}-\theta_{11})s} \\ -\frac{k_{21}(\tau_{22}s+1)}{k_{22}(\tau_{21}s+1)}e^{-(\theta_{21}-\theta_{22})s} & e^{-v(\theta_{11}-\theta_{12})s} \end{bmatrix} \quad (2.34)$$

Onde

$$v = \begin{cases} 1 & \text{se } L \geq 0 \\ 0 & \text{se } L \leq 0 \end{cases} \quad (2.35)$$

2.3.3 Técnica de Desacoplamento Invertido

O desacoplamento invertido proposto em (1) será detalhado aqui. De acordo com a [Figura 3](#) e a equação do problema geral de desacoplamento [Equação 2.22](#), quando da escolha de $H_{11}(s) = G_{11}(s)$ e $H_{22}(s) = G_{22}(s)$ (desacoplamento ideal), as variáveis manipuladas são dadas por:

$$u_1(s) \left[\frac{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)} \right] = c_1(s) - c_2(s) \frac{G_{12}(s)}{G_{11}(s)} \quad (2.36)$$

$$u_2(s) \left[\frac{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)} \right] = c_2(s) - c_1(s) \frac{G_{21}(s)}{G_{22}(s)} \quad (2.37)$$

0

E posteriormente:

$$c_1(s) = u_1(s) \left[\frac{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)} \right] + c_2(s) \frac{G_{12}(s)}{G_{11}(s)} \quad (2.38)$$

$$c_2(s) = u_2(s) \left[\frac{G_{11}(s)G_{22}(s) - G_{12}(s)G_{21}(s)}{G_{11}(s)G_{22}(s)} \right] + c_1(s) \frac{G_{21}(s)}{G_{22}(s)} \quad (2.39)$$

Realizando as devidas substituições, temos:

$$u_1(s) = c_1(s) - u_2(s) \frac{G_{12}(s)}{G_{11}(s)} \quad (2.40)$$

$$u_2(s) = c_2(s) - u_1(s) \frac{G_{21}(s)}{G_{22}(s)} \quad (2.41)$$

Na [Figura 4](#), é mostrado o sistema de controle com a estrutura de desacoplamento invertido definido através das equações [Equação 2.40](#) e [Equação 2.41](#). Observe-se que as funções de transferência do desacoplador são as mesmas utilizadas no desacoplamento simplificado. Com isso, o desacoplamento invertido apresenta, ao mesmo tempo,

a facilidade de implementação dos elementos do desacoplador simplificado e uma função de transferência do sistema desacoplado diagonal $H(s)$ mais simples como no caso do desacoplamento ideal.

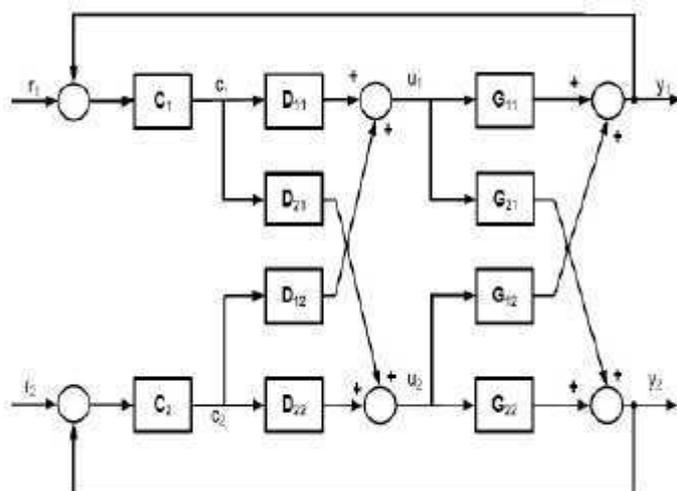


Figura 4 – Sistema de Controle com desacoplador

3 Materiais e Métodos

3.1 Equipamentos e Montagem Física da Planta

A implementação de controladores digitais é fundamental para diversos tipos de equipamentos industriais. Em geral, necessita-se de um dispositivo que realize a leitura das variáveis de estado do sistema e comande os atuadores de forma a obter as respostas desejáveis. Dentre as diversas opções de equipamentos eletrônicos, optou-se pelo uso de um microcontrolador fabricado pela GHIElectronics e que usa o .NET Micro Framework como princípio de operação, o USBizi. A [Figura 5](#) mostra o microcontrolador usado.



Figura 5 – USBizi

O USBizi é um sistema baseado no chip LPC2388/LPC2387, que utiliza o .NET *Micro Framework*. Basicamente, é um subsistema de Mestre Embarcado, que provê muitas das funcionalidades que esse tipo de dispositivos dispõe, com tudo implementado em um único chip. Desenvolvedores podem usar .NET *Micro Framework* e escrever códigos em C# sem ter que lidar com muita complexidade. O USBizi possui total acesso ao .NET *Micro Framework*. Além disso, o módulo possui muitas características adicionais que não são fornecidas em outras plataformas .NET *Micro Framework*, tais como USB Hosting, entradas e saídas analógicas, PWM, CAN, entre outras (23).

Também, o Laboratório de Controle e Otimização da UFCG dispõe de uma planta montada para o controle de dois módulos peltier ligados por uma superfície metálica. Os circuitos eletrônicos presentes permitem o uso de PWM bidirecional nos módulos através do uso da ponte H, além da leitura de temperatura feita pelo LM35. O sinal de PWM do microcontrolador ativa os atuadores, que por sua vez aplicam uma tensão sobre os módulos Peltier. Dessa forma, pode-se controlar a temperatura nas faces dos módulos Peltier através do *Duty Cycle* do PWM do USBizi. Na Figura 6 é mostrado o diagrama de blocos do sistema de controle de temperatura (incluindo o microcontrolador, atuadores e sensores). Na Figura 7 é mostrada a montagem presente no laboratório.

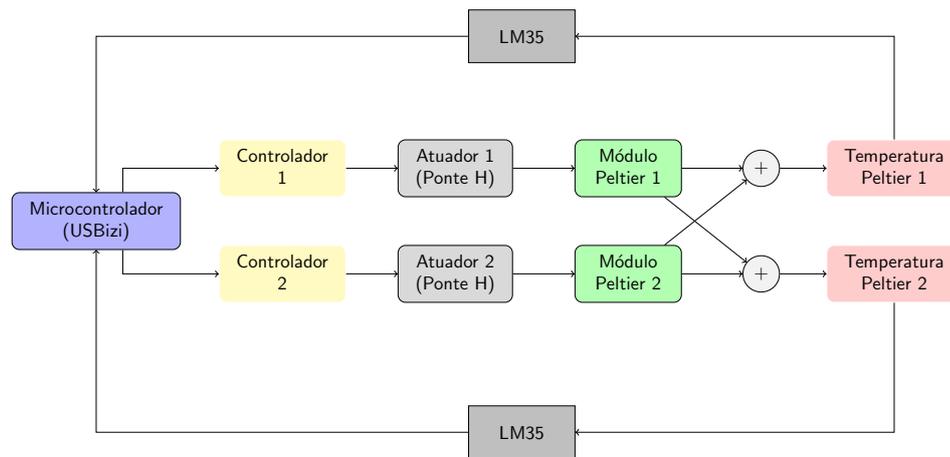


Figura 6 – Diagrama de Blocos do Sistema de Controle

3.2 A linguagem C#, .NET Micro Framework e Interface de Comunicação com o usuário

O envio de comandos através da porta serial para o USBizi deve ser feito através da ferramenta *.NET MicroFramework*, que é compatível com a linguagem C#.

Destacam-se as vantagens do uso desta linguagem de programação:

- Maior interatividade com os usuários;
- Uso de eventos nos controles;
- C# possui o recurso do ambiente do Windows, que, em geral, é mais familiar para a maioria dos usuários;
- Mais fácil implementação de uma interface gráfica adaptada às necessidades requeridas pelo usuário;
- Fácil tratamento de erros;



Figura 7 – Montagem presente no Laboratório de Controle e Otimização

- Compatibilidade com .NET MicroFramework

Por sua vez, o *.NET Micro Framework* da *Microsoft* é um pequeno e eficiente ambiente de desenvolvimento do .NET usado para executar código gerenciado em dispositivos que são pequenos demais e com recursos restritos para *Windows CE* e *.NET Compact Framework*. O *.NET Micro Framework* permite ao usuário escrever aplicações para pequenos dispositivos embarcados com *Visual Studio* e *C#*. (24)

Dentre as vantagens do uso destas ferramentas computacionais, destacam-se (25):

- Roda no *Microsoft Visual C# Express*, que é grátis e tem uma IDE de alta performance;
- *.NET Micro Framework* é gratuito e open-source;
- Capacidade de Debug completo (Breakpoints, execução passo-a-passo, variáveis, etc.);

- Inclui muitos drivers (SPI, UART, I2C, etc.);
- Não há necessidade de datasheets do processador graças ao padrão do Framework;

Foi desenvolvido, então, um aplicativo para computador que realize a comunicação entre usuário e USBizi, a fim da realização do controle da planta estudada. Dessa forma, um usuário leigo pode, através de um software, controlar o sistema mesmo não tendo um perfeito conhecimento sobre seu funcionamento ou protocolos de comunicação. Para que o sistema funcione, é requerido do usuário os seguintes pontos:

1. Configurar a porta serial;
2. Escolher modo de execução;
3. Verificar temperatura para ver se o controle corresponde ao desejado;

Algo importante a ser notado é que enquanto a comunicação serial com o computador não for aberta, não será possível enviar comandos para os dispositivos. Caso essa comunicação não seja aberta, será enviada uma mensagem de erro ao usuário ao tentar realizar qualquer dos passos seguintes.

Na [Figura 12](#) é mostrada a janela principal do programa desenvolvido. Há uma barra de menu para configurar a conexão da porta serial e para escolha do modo de operação. Há dois botões, um que inicia e outro que encerra a leitura de temperatura. Inicialmente, deve-se configurar a conexão da porta serial. Para qualquer comando que tentar ser executado sem antes abrir a porta serial, será enviada uma mensagem de erro que avisa ao usuário que a porta está desconectada. Ao chamar-se a aba de Conexão, abre-se outra janela, que mostrará as entradas seriais disponíveis no computador. Algo de fundamental importância ao ligar-se a comunicação serial do computador é que o usuário tenha certeza de qual das portas estará ligada aos dispositivos. Ao usar-se um adaptador USB-RS232, por exemplo, outra porta será aberta, o que pode causar confusão. Tendo-se certeza da porta escolhida, clica-se o botão OK, a porta será aberta e, automaticamente, definida a *baud rate* de 9600, conforme especificação do projeto. Na [Figura 8](#) é mostrada a janela de configuração da porta serial.

Há no menu a opção para trabalhar em malha aberta. Ao escolher-se esse modo de operação, o usuário deve colocar os valores de *duty cycle* desejados nos *textbox* para serem enviados ao microcontrolador. Na [Figura 9](#) é mostrada a janela para trabalhar em malha aberta.

De maneira análoga, há no menu a opção para trabalhar em malha fechada. Ao escolher esse modo de operação, o usuário deve colocar os valores de referência de temperatura para cada Peltier. São disponibilizadas para o usuário outras *textboxes* nas quais



Figura 8 – Janela para configurar a porta serial

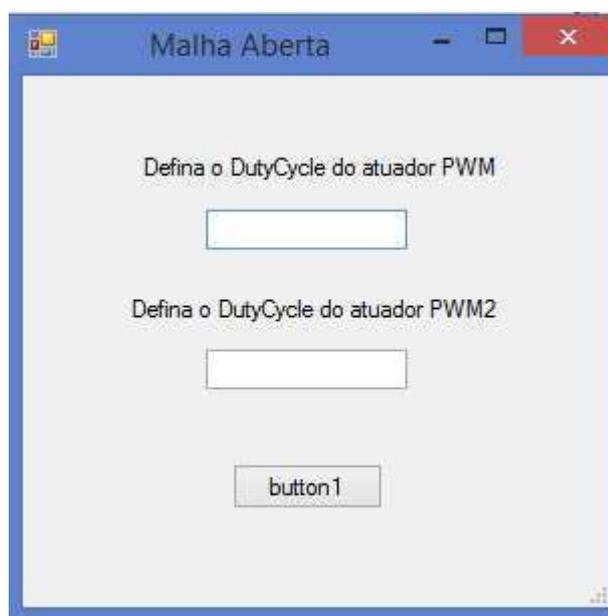


Figura 9 – Janela para trabalhar em malha aberta

o mesmo pode acompanhar a evolução da temperatura dos módulos, assim como o duty cycle que está sendo aplicado a cada circuito de acionamento, o que representa as entradas do sistema. Na [Figura 10](#) é mostrada a tela de trabalho em malha aberta.

3.2.1 Lógica de programação

Como estratégia de ação para desenvolvimento do projeto, deseja-se criar uma aplicação computacional que se comunique respectivamente com o microcontrolador e acompanhe em tempo real a evolução da temperatura dos módulos Peltier para fins de controle e supervisão. Assim, consideram-se algumas questões:

1. Criar um protocolo de comunicação entre o computador e o microcontrolador, além

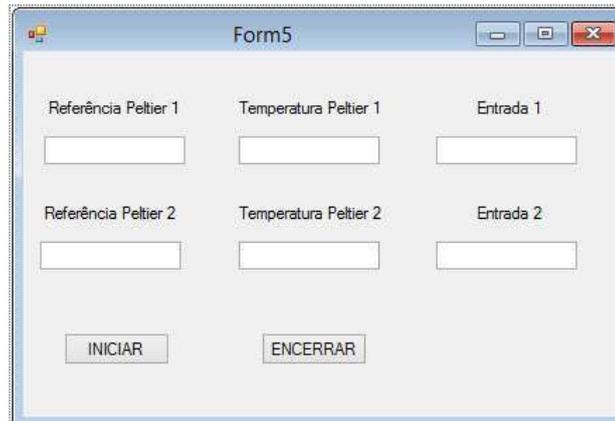


Figura 10 – Janela para trabalhar em malha aberta

de uma interface gráfica que permita ao usuário ter domínio sobre o processo;

2. Levantar o modelo dos módulos para projeção de um controlador;
3. Realizar a leitura de dados da temperatura em tempo real;
4. Projetar um controlador de acordo;
5. Fazer o devido tratamento de erros, para que o manuseio do software seja eficiente.

A programação feita, portanto, deve agir em duas frentes: a programação do microcontrolador, que receberá comandos do computador e agirá nos atuadores, e a programação da aplicação no CPU, que fará leitura dos dados enviados pelo microcontrolador e agirá de acordo com o estado atual das variáveis do sistema. Na [Figura 11](#) é mostrado o diagrama de blocos da lógica de programação.

Tendo em vista os pontos analisados acima, propõem-se as seguintes soluções:

1. Criou-se um protocolo de comunicação entre a CPU e o microcontrolador. Cada mensagem escrita do computador para o micro é um array de 5 bytes. Fazemos isso porque o *.NET Micro Framework* não possui eventos para lidar com a chegada de dados na porta serial. Portanto, ao definir mensagens de tamanho fixo, a leitura de dados no controlador será mais fácil de ser feita. A [Tabela 1](#) mostra os comandos usados no protocolo.

Malha Aberta	Malha Fechada	Ler Temperatura
AXXXX	FXXXX	T0000

Tabela 1 – Mensagens do protocolo de comunicação

Por outro lado, a aplicação no CPU pode trabalhar com eventos de comunicação serial, o que torna o recebimento e envio de comandos mais fácil.

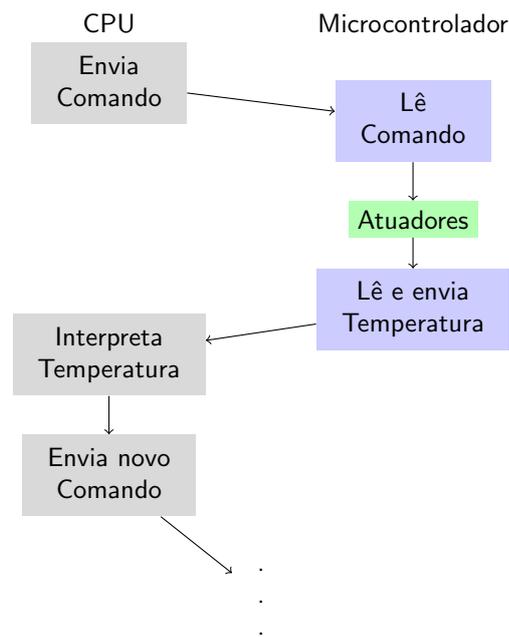


Figura 11 – Diagrama de blocos da lógica de programação

Uma vez que um comando é recebido, o microcontrolador agirá de acordo com a solicitação. Por exemplo, abaixo é mostrado um trecho de código que mostra como o controlador age quando é enviado um comando do tipo AXXYY (XX representa o *duty cycle* para um dos módulos peltier, e YY representa o *duty cycle* para o outro módulo peltier), que caracteriza a malha aberta:

```

porta.Read(buffer, 0, buffer.Length);
str = Encoding.UTF8.GetChars(buffer);

if (str[0] == 'A')
{
value1 = Functions.CharToInt(str[4]) + 10*Functions.CharToInt(str[3]);
// Pega os dois ultimos valores
value2 = Functions.CharToInt(str[2]) + 10 * Functions.CharToInt(str[1]);
// Pega os valores do meio

// Setando o PWM1...
porcentagem = periodo * value1 / 100;
pwm_MalhaAberta1.Set(true);
pwm_MalhaAberta1.SetPulse(periodo, porcentagem);

// Setando o PWM2...
porcentagem = periodo * value2 / 100;

```

```
pwm_MalhaAberta2.Set(true);  
pwm_MalhaAberta2.SetPulse(periodo, porcentagem);  
}
```

A função `CharToInt` foi criada a parte no programa para ajudar na conversão de tipos de variáveis. Uma vez recebidos os valores `XX` e `YY`, faz-se o tratamento da informação e são setados os pinos PWM da USBizi designados inicialmente para atuar nos módulos Peltier.

- Foram criadas abas na tela principal do programa que permitem ao usuário conectar-se com o USBizi e trabalhar em malha aberta ou fechada, além de acompanhar graficamente em tempo real a evolução da temperatura. A janela principal possui dois espaços onde as temperaturas lidas nos dois módulos Peltier são amostradas. Os botões `INICIAR` e `ENCERRAR` iniciam e terminam a leitura da temperatura. A tela principal deste programa é mostrado na [Figura 12](#).



Figura 12 – Tela inicial do programa criado

Uma dificuldade ao se executar o código é fazer a leitura da temperatura de forma constante a cada certo intervalo de tempo, uma vez que um loop infinito travaria a execução de qualquer outra função do programa. Desta forma, criou-se uma segunda *Thread* onde o *loop* pode acontecer sem influenciar o que está sendo executado na *Thread* principal do programa. Assim, os botões `INICIAR` e `ENCERRAR` apenas mudam o valor de certo flag, que habilitará, ou não, a amostragem de valores nos textboxes presentes no executável.

4 Resultados Experimentais - Identificação

Foram testados os métodos de identificação de sistemas anteriormente descritos. As entradas PWM1 e PWM2 do sistema podem variar de 0 a 100 (em porcentagem). Os módulos Peltier são não-lineares, de modo que foram feitos diversos testes de resposta ao degrau, em intervalos igualmente espaçados de operação, correspondendo a uma variação de 10% em *Duty Cycle*. Por exemplo, a função de transferência identificada para um degrau de 30 a 40 em *DutyCycle* é diferente da função de transferência identificada para um degrau de 60 a 70. Assim, um controlador que seja projetado para esse sistema deve levar em consideração esta característica do equipamento. Em outras palavras, os controladores devem ser projetados para determinada faixa de operação, onde o sistema pode ser considerado como linear.

Foi desenvolvido um programa em C# que faz o microcontrolador aplicar um degrau no sistema e realizar várias leituras da temperatura nos dois módulos Peltier. Essas leituras amostradas são usadas em algoritmos desenvolvidos no MATLAB para cada técnica de identificação estudada anteriormente. Mostram-se, então, dois diferentes casos de operação nos quais os métodos de identificação foram aplicados, e são explicados quando são mais eficazes ou não.

4.1 Relação sinal-ruído alta

Relação sinal-ruído ou razão sinal-ruído (frequentemente abreviada por S/N ou SNR, do inglês, signal-to-noise ratio e RSR em português) é definida como a razão da potência de um sinal e a potência do ruído sobreposto ao sinal, geralmente expressa em Decibel. Em outras palavras, a relação sinal-ruído compara o nível de um sinal desejado com o nível do ruído de fundo. Quanto mais alto for a relação sinal-ruído, menor é o efeito do ruído de fundo sobre a detecção ou medição do sinal.

Inicialmente, analisou-se a função de transferência G_{22} (isto é, a função de transferência que relaciona a temperatura do módulo Peltier 2 em função de um degrau aplicado na entrada PWM2) para um faixa de operação do sinal de entrada entre 20 e 30. Para o método *Time Integral Approach* sem e com variável instrumental, respectivamente, foram obtidas as seguintes funções de transferência:

$$G_{22} = \frac{0.5953}{54.3602s + 1} e^{-13.2176t} \quad (4.1)$$

$$G_{22} = \frac{0.5949}{54.0916s + 1} e^{-13.3424t} \quad (4.2)$$

Os gráficos do sistema real, e dos modelos identificados sem e com tratamento de ruído por esse método, respectivamente, são mostrados na figura 13

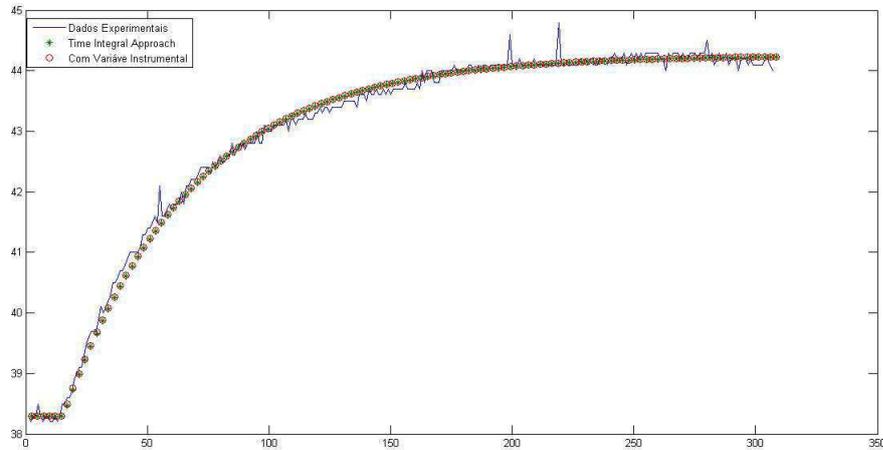


Figura 13 – Comparação entre sistema real e modelos identificados para o método *Time Integral Approach*

Por sua vez, ao ser usada a técnica robusta de identificação, foram obtidas as seguintes funções de transferência, para o modelo sem variável instrumental e com variável instrumental, respectivamente:

$$G_{22} = \frac{0.6019}{59.3212s + 1} e^{-7.0683t} \quad (4.3)$$

$$G_{22} = \frac{0.6020}{59.3629s + 1} e^{-7.0683t} \quad (4.4)$$

Os gráficos do sistema real, e dos modelos identificados sem e com tratamento de ruído por esse método, respectivamente, são mostrados na figura 14

Para um casos como este, onde a relação sinal-ruído é alta, por experimentação, o método mais eficaz observado foi *Time Integral Approach* com variável instrumental, resultado confirmado em casos semelhantes a este.

4.2 Relação sinal-ruído baixa

De forma análoga, quanto mais baixa for a relação sinal-ruído, maior é o efeito do ruído de fundo sobre a detecção ou medição do sinal. Agora, analisou-se a função de transferência G_{21} (isto é, a função de transferência que relaciona a temperatura do módulo Peltier 2 em função de um degrau aplicado na entrada PWM1) para um faixa de operação do sinal de entrada entre 80 e 90. Para o método *Time Integral Approach*

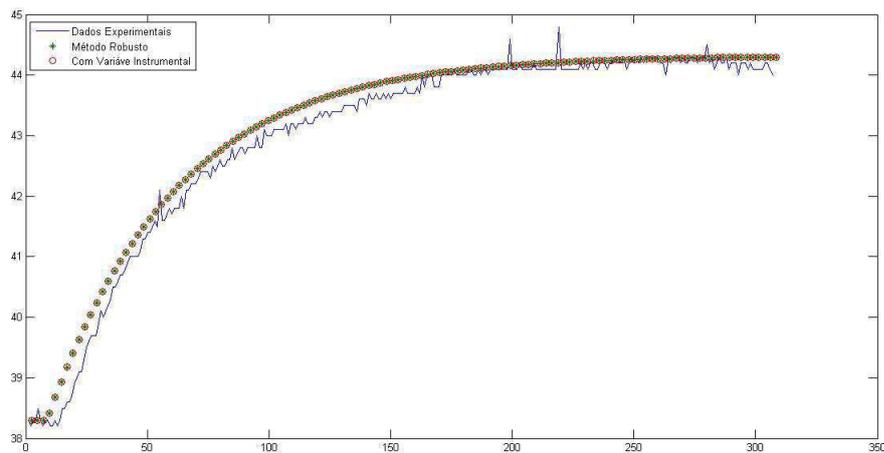


Figura 14 – Comparação entre sistema real e modelos identificados para o método *Time Integral Approach* com variável instrumental

sem e com variável instrumental, respectivamente, foram obtidas as seguintes funções de transferência:

$$G_{12} = \frac{0.0352}{32.2813s + 1} e^{-166.0386t} \quad (4.5)$$

$$G_{12} = \frac{0.0898}{181.5732s + 1} e^{-155.9785t} \quad (4.6)$$

Os gráficos do sistema real, e dos modelos identificados sem e com tratamento de ruído por esse método, respectivamente, são mostrados na [Figura 15](#)

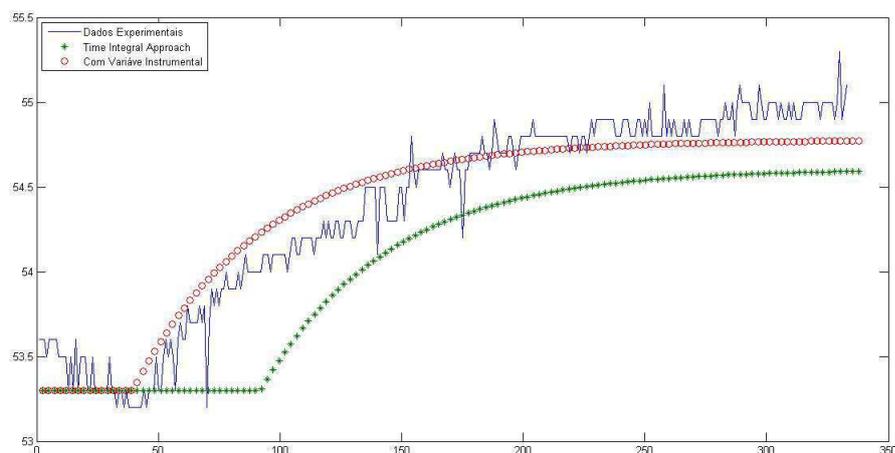


Figura 15 – Comparação entre sistema real e modelos identificados para o método *Time Integral Approach*

Por sua vez, ao ser usada a técnica robusta de identificação, foram obtidas as

seguintes funções de transferência, para o modelo sem variável instrumental e com variável instrumental, respectivamente:

$$G_{12} = \frac{0.0915}{194.1239s + 1} e^{-112.9364t} \quad (4.7)$$

$$G_{12} = \frac{0.1093}{155.9785s + 1} e^{-112.9364t} \quad (4.8)$$

Os gráficos do sistema real, e dos modelos identificados sem e com tratamento de ruído por esse método, respectivamente, são mostrados na [Figura 16](#)

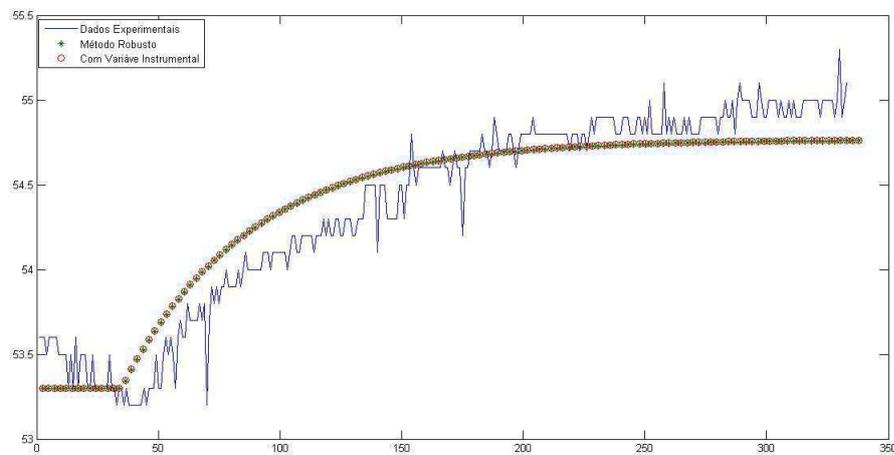


Figura 16 – Comparação entre sistema real e modelos identificados para o método *Time Integral Approach* com variável instrumental

Para um casos como este, onde a relação sinal-ruído é baixa, por experimentação, o método mais eficaz observado foi o *Time Integral Approach* com variável instrumental, resultado confirmado em casos semelhantes a este, apesar de mesmo este método não reproduzir um modelo muito próximo ao real. Em casos como este, onde a precisão do modelo não é adequada, pode-se também recorrer a outros métodos de identificação, como o método gráfico, por exemplo.

4.3 Modelos identificados

Dependendo da relação sinal-ruído do sistema, escolheu-se o método de identificação que mais se mostrou fiel ao processo real. Conforme já exposto anteriormente, os modelos dos módulos são não-lineares para diferentes pontos de operação. Nas [Tabela 2](#) e [Tabela 3](#) são mostradas as funções de transferência identificadas para cada intervalo de operação.

Temperatura	G_{11}	G_{12}
24-31	$G_{11} = \frac{0.6891}{52.8475s+1} e^{-23.4815s}$	$G_{12} = \frac{0.249}{99.8766s+1} e^{-41.2939s}$
31-36	$G_{11} = \frac{0.571}{76.85s+1} e^{-6.07s}$	$G_{12} = \frac{0.175}{128.57s+1} e^{-30.86s}$
36-39	$G_{11} = \frac{0.8287}{26.2073s+1} e^{-25.4837s}$	$G_{12} = \frac{0.085}{2.3202s+1} e^{-49.8854s}$
39-43	$G_{11} = \frac{0.2534}{32.2437s+1} e^{-45.5662s}$	$\frac{0.0383}{34.5424s+1} e^{-57.5055s}$
43-46	$G_{11} = \frac{0.2570}{85.9250s+1} e^{-30.0526s}$	$G_{12} = \frac{0.0156}{79.67s+1} e^{-12.9s}$

Tabela 2 – Funções de transferência identificadas - Influência da entradas PWM1 e PWM2 na saída 1

Temp	G_{12}	G_{22}
24-34	$G_{21} = \frac{0.274}{83.17s+1} e^{-15.84s}$	$G_{22} = \frac{0.821}{40.79s+1} e^{-3.44s}$
34-42	$G_{21} = \frac{0.284}{85.5683s+1} e^{-51.2089s}$	$G_{22} = \frac{0.8}{41.4852s+1} e^{-35.884s}$
42-49	$G_{21} = \frac{0.2358}{83.5732s+1} e^{-40.3853s}$	$G_{22} = \frac{0.6617}{38.2859s+1} e^{-26.4396s}$
49-55	$G_{21} = \frac{0.1995}{69.1935s+1} e^{-54.276s}$	$G_{22} = \frac{0.5685}{38.645s+1} e^{-30.7327s}$
55-60	$G_{21} = \frac{0.1724}{71.1834s+1} e^{-55.3497s}$	$G_{22} = \frac{0.4715}{33.0195s+1} e^{-41.6365s}$

Tabela 3 – Funções de transferência identificadas - Influência da entradas PWM1 e PWM2 na saída 2

5 Resultados Experimentais - Controle com *toolbox* do Simulink

Para testar os modelos feitos e projetar os controladores, desenvolvemos uma simulação usando a ferramenta Simulink, do MATLAB. Foram escolhidos controladores PI digitais por tratar-se de um controlador de mais fácil implementação e que fornece erro nulo ao sistema em regime permanente. Este controlador caracteriza-se pela função de transferência no domínio Z mostrada na [Equação 5.1](#).

$$K_p + K_i T_s \frac{1}{z - 1} \quad (5.1)$$

Onde K_p é o coeficiente proporcional do controlador, K_i é o coeficiente integrativo do controlador e T_s é o intervalo de amostragem, que vale 1s. Dessa forma, no projeto do controlador, temos o esquema mosrado na [Figura 17](#)

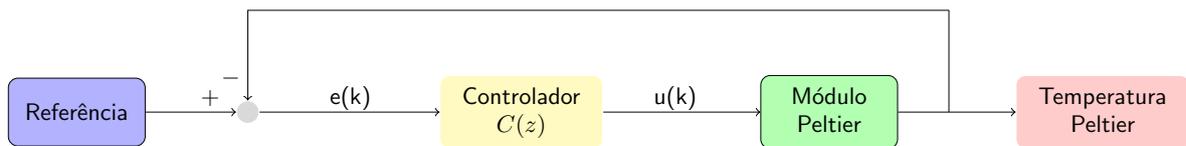


Figura 17 – Projeto do Controlador

Onde $e(k)$ representa o erro encontrado na iteração k, e $u(k)$ representa a entrada (*Duty Cycle*) aplicada ao sistema. Para realizar o controle, assim, procuramos a entrada $u(k)$ baseados na lei de controle a ser utilizada. O seguinte desenvolvimento matemático é realizado:

$$U(z) = C(z)E(z) \quad (5.2)$$

$$U(z) = \left(K_p + K_i \frac{1}{z - 1}\right)E(z), \text{ sendo } T_s = 1 \quad (5.3)$$

$$U(z) = \frac{K_p z - K_p + K_i z}{z - 1} E(z) \quad (5.4)$$

$$U(z)z - U(z) = K_p z E(z) + (K_p - K_i)E(z) \quad (5.5)$$

Multiplicando ambos os lados por z^{-1} , temos:

$$U(z) = z^{-1}U(z) + K_p E(z) - (K_p - K_i)z^{-1}E(z) \quad (5.6)$$

Aplicando a transformada Z inversa à [Equação 5.6](#), temos:

$$u(k - 1) - u(k) = K_p e(k - 1) + (K_i - K_p) e(k) \quad (5.7)$$

$$u(k) = u(k - 1) + (K_p - K_i) e(k) - K_p e(k - 1) \quad (5.8)$$

Dessa forma, de acordo com a [Equação 5.8](#), o valor atual de *Duty Cycle* a ser aplicado ao sistema pode ser encontrado através do valor anterior de entrada aplicado e dos valor atual e anterior do erro. Esta lógica foi implementada computacionalmente de acordo com o seguinte trecho de código.

```
u[0] = u[1] + Kp_P11 * erro_P1[0] + (Ki_P11 - Kp_P11) * erro_P1[1];
```

Os erros e entradas são armazenadas em um array, e atualizados a cada iteração, de forma que o valor de *Duty Cycle* escrito pelo USBizi ao sistema obedece ao sistema de controle.

Utilizou-se, para a sintonia dos controladores, a ferramenta de sintonia automática do Simulink. Na [Figura 18](#) é mostrado o diagrama de blocos da simulação feita.

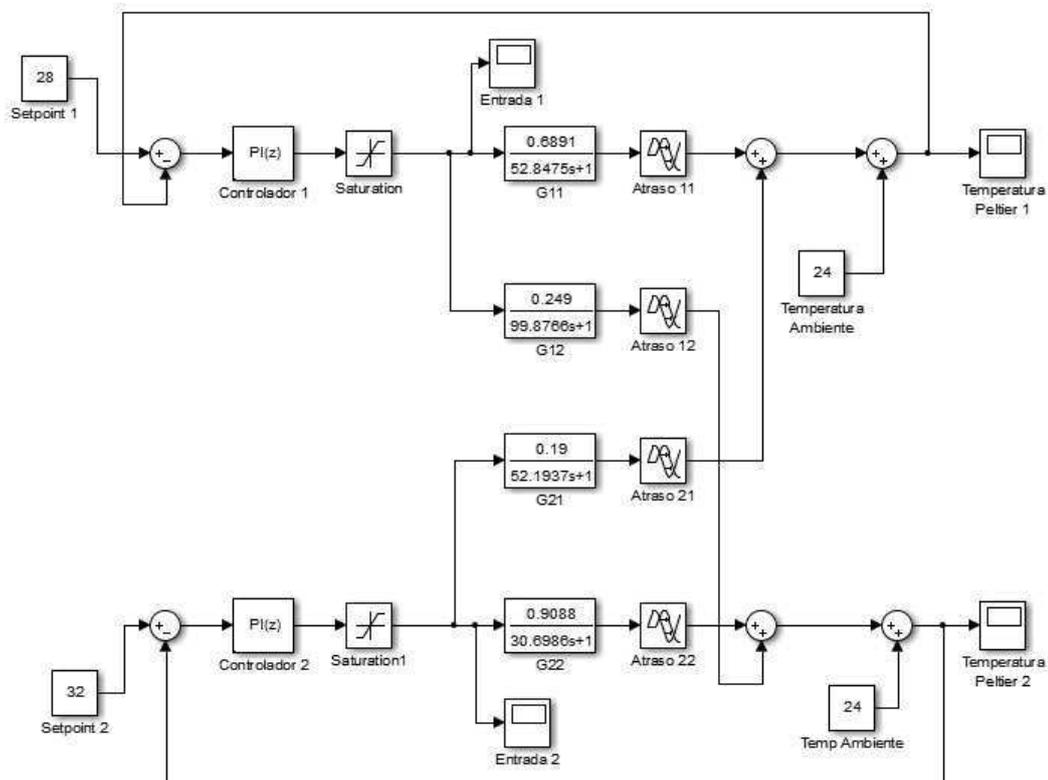


Figura 18 – Diagrama para simulação no Simulink

Como já discutido anteriormente, os módulos Peltier se configuram como sistemas

não lineares, conforme demonstrado pelas diferentes funções de transferência identificadas para diferentes intervalos de operação. Dessa forma, foram calculados vários controladores, para diferentes pontos de operação. Os coeficientes dos controladores sintonizados são mostrados nas [Tabela 4](#) e [Tabela 5](#).

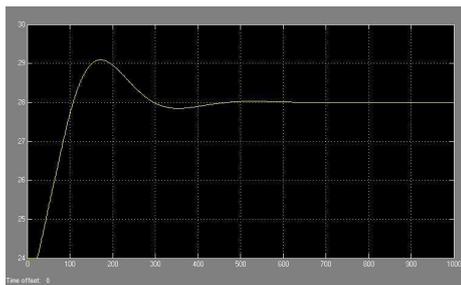
Temperatura de operação	K_{p1}	K_{i1}
24-31	0.8219	0.0255
31-36	2.3131	0.0621
36-39	6.3512	0.2499
39-43	1.0983	0.0436
43-46	6.1537	0.0684

Tabela 4 – Parâmetros para o controlador 1

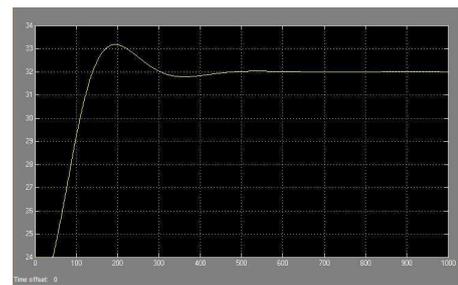
Temperatura de operação	K_{p2}	K_{i2}
24-34	1.2818	0.0647
34-42	0.4569	0.0163
42-49	0.2499	0.0215
49-55	0.7092	0.0251
55-60	0.8079	0.0262

Tabela 5 – Parâmetros para o controlador 2

A título de ilustração, na [Figura 19](#) são mostrados os resultados de simulação feita para o diagrama da [Figura 18](#), levando em conta os devidos controladores com coeficientes mostrados nas [Tabela 4](#) e [Tabela 5](#).



((a)) Temperatura do Peltier 1



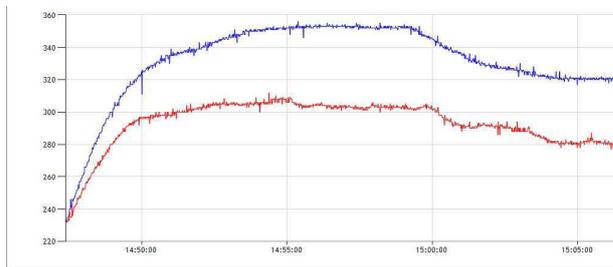
((b)) Temperatura do Peltier 2

Figura 19 – Temperaturas dos módulos Peltier

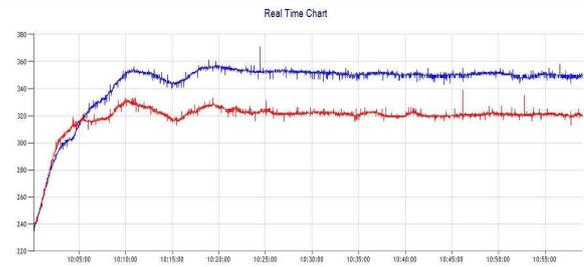
A título de informação, destaca-se que o método de integração usado pelo software para as simulações realizadas é o método de Euler.

Uma vez calculados os modelos, os parâmetros do controlador e realizadas simulações, foram feitos experimentos para verificação da validade da estratégia de controle. Na [Figura 20](#) são mostrados dois experimentos feitos. No primeiro, são estabelecidas como referências os valores de 35 e 30 graus, e em dado momento, as referências são mudadas

para 32 e 28 graus. No segundo gráfico, as referências são levadas diretamente a 35 e 32 graus.



((a)) Temperatura do Peltier 1



((b)) Temperatura do Peltier 2

Figura 20 – Temperaturas dos módulos Peltier

Veja que em ambos os casos, o controlador foi eficiente em levar a temperatura de saída para a referência, embora haja uma pequena oscilação em torno do ponto de oscilação. Isso se dá devido à maior complexidade do modelo real do Peltier, bem como devido a perturbações. Dessa forma, verificamos que os modelos identificados são válidos para realização do controle, uma vez que os controladores projetados por meio do *toolbox* do Simulink foram funcionais em seu objetivo de realizar o controle da temperatura.

6 Simulação - Desacoplamento das malhas

Na [Capítulo 2](#) foram abordadas três técnicas de desacoplamento. Com o auxílio do MATLAB e Simulink, as técnicas previamente explanadas foram aplicadas, sendo calculados, assim, os desacopladores e feitas as devidas simulações. Como, devido ao sistema ser não-linear, existem diversas funções de transferência, e 25 diferentes combinações para as funções de transferência do sistema, foi escolhida uma dessas combinações para aplicação dos métodos estudados.

A matriz de funções de transferência escolhida é mostrada na [Equação 6.1](#)

$$G = \begin{bmatrix} \frac{0,6871}{52,85s+1}e^{-23,5s} & \frac{0,274}{83,17s+1}e^{-15,84s} \\ \frac{0,249}{99,89s+1}e^{-41,3s} & \frac{0,821}{40,79s+1}e^{-3,44s} \end{bmatrix} \quad (6.1)$$

6.1 Desacoplamento por Nordfelt

Foi desenvolvido uma rotina em MATLAB para realização do algoritmo presente na [subseção 2.3.1](#). Nesta rotina, foram calculadas a matriz de desacoplamento, D , e a matriz resultante do sistema com desacoplamento, H . A matriz D é mostrada na [Equação 6.2](#) e a matriz H é mostrada na [Equação 6.3](#).

$$D = \begin{bmatrix} 0,821e^{-23,5s} & \frac{-0,274}{59,09s+1}e^{-15,84s} \\ \frac{-0,249}{30,32s+1}e^{-7,64s} & 0,6891 \end{bmatrix} \quad (6.2)$$

$$H = \begin{bmatrix} \frac{1713s^2+70,38s+0,5038}{160000s^3+9909s^2+183s+1}e^{-23,5s} & \frac{278s^2+1,006s+0,02053}{102900s^3+7151s^2+154,3s+1}e^{-23,5s} \\ \frac{535,6s^2+118,3s+0,01723}{311800s^3+14300s^2+211,8s+1}e^{-23,5s} & \frac{2279s^2+77,41s+0,4907}{200400s^3+10710s^2+183s+1}e^{-23,5s} \end{bmatrix} \quad (6.3)$$

Percebe-se que os elementos da diagonal principal de H são muito próximos de zero. De fato, sua influência face a magnitude dos comportamentos presentes na diagonal principal de H em regime permanente é ínfima, de modo que podemos simplificar H , conforme mostrado na [Equação 6.4](#)

$$H = \begin{bmatrix} \frac{1713s^2+70,38s+0,5038}{160000s^3+9909s^2+183s+1}e^{-23,5s} & 0 \\ 0 & \frac{2279s^2+77,41s+0,4907}{200400s^3+10710s^2+183s+1}e^{-23,5s} \end{bmatrix} \quad (6.4)$$

Assim, após o uso do desacoplador, podemos trabalhar com um sistema no qual praticamente não há interferência da entrada 1 na saída 2, nem da entrada 2 na saída 1.

Além disso, os desacopladores foram testados no Simulink para verificação de sua funcionalidade. Foram feitos dois testes para cada desacoplador: no primeiro coloca-se

um degrau unitário na entrada 1, enquanto a entrada 2 é zerada; no segundo a entrada 2 recebe um degrau unitário enquanto a entrada 1 é zerada. Na [Figura 21](#) são mostradas as influências das entradas PWM no módulo Peltier 1, e na [Figura 22](#) as influências das entradas PWM no módulo Peltier 2

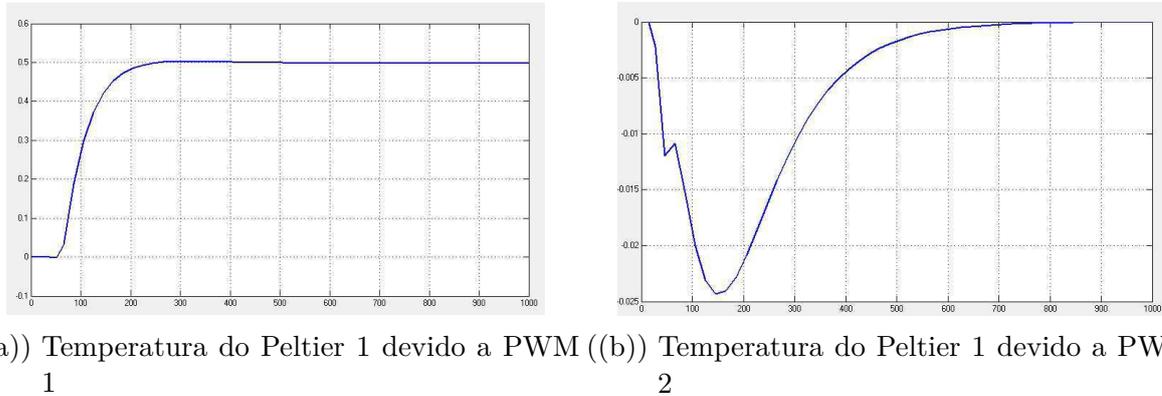


Figura 21 – Influência das entradas PWM1 e PWM2 no módulo Peltier 1

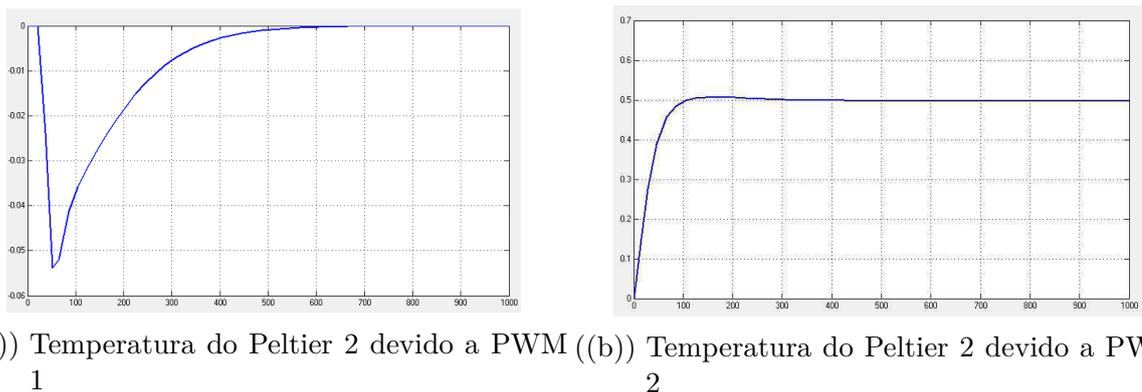


Figura 22 – Influência das entradas PWM1 e PWM2 no módulo Peltier 2

Verifica-se, assim, que o desacoplamento das malhas foi eficiente, uma vez que, para a primeira simulação, a entrada 2 não exerce influência em regime permanente sobre o Peltier 1, e no segundo experimento a entrada 1 não exerce influência em regime permanente sobre o Peltier 2.

6.2 Desacoplamento por Wang

A matriz de desacoplamento pela técnica de Wang é encontrada segundo a [Equação 2.34](#) e [Equação 2.35](#). Conforme feito para a seção anterior, encontramos, através de simulação feita no MATLAB, as matrizes de desacoplamento, D , e a matriz resultante do

sistema, H . A matriz D é mostrada na [Equação 6.5](#) e a matriz H é mostrada na [Equação 6.6](#).

$$D = \begin{bmatrix} 1 & \frac{-13,64s-0,3337}{83,17s+1}e^{-12,4s} \\ \frac{-19,1s-0,3613}{99,88s+1}e^{-17,8s} & 1 \end{bmatrix} \quad (6.5)$$

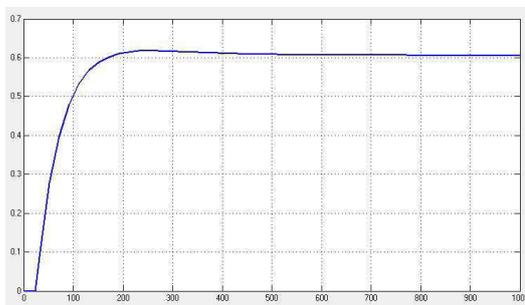
$$H = \begin{bmatrix} \frac{5545s^2+118,3s+0,606}{439000s^3+17980s^2+235,9s+1}e^{-23,5s} & \frac{-1,828s^2+-0,01929s+0,0000323}{282200s^3+13700s^2+207,1s+1}e^{-23,5s} \\ \frac{-0,2332s^2+0,00088s+0,0000282}{527500s^3+20530s^2+252,6s+1}e^{-23,5s} & \frac{6607s^2+141s+0,722}{338800s^3+15770s^2+223,8s+1}e^{-23,5s} \end{bmatrix} \quad (6.6)$$

Percebe-se que os elementos da diagonal principal de H são muito próximos de zero. De fato, sua influência face a magnitude dos comportamentos presentes na diagonal principal de H em regime permanente é ínfima, de modo que podemos simplificar H , conforme mostrado na [Equação 6.7](#)

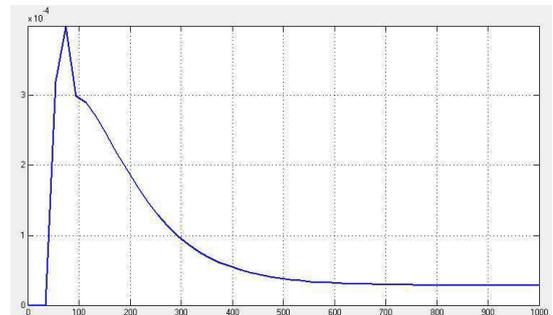
$$H = \begin{bmatrix} \frac{5545s^2+118,3s+0,606}{439000s^3+17980s^2+235,9s+1}e^{-23,5s} & 0 \\ 0 & \frac{6607s^2+141s+0,722}{338800s^3+15770s^2+223,8s+1}e^{-23,5s} \end{bmatrix} \quad (6.7)$$

Assim, após o uso do desacoplador, podemos trabalhar com um sistema no qual praticamente não há interferência da entrada 1 na saída 2, nem da entrada 2 na saída 1.

Além disso, os desacopladores foram testados no Simulink para verificação de sua funcionalidade. Foram feitos dois testes para cada desacoplador: no primeiro coloca-se um degrau unitário na entrada 1, enquanto a entrada 2 é zerada; no segundo a entrada 2 recebe um degrau unitário enquanto a entrada 1 é zerada. Na [Figura 23](#) são mostradas as influências das entradas PWM no módulo Peltier 1, e na [Figura 24](#) as influências das entradas PWM no módulo Peltier 2

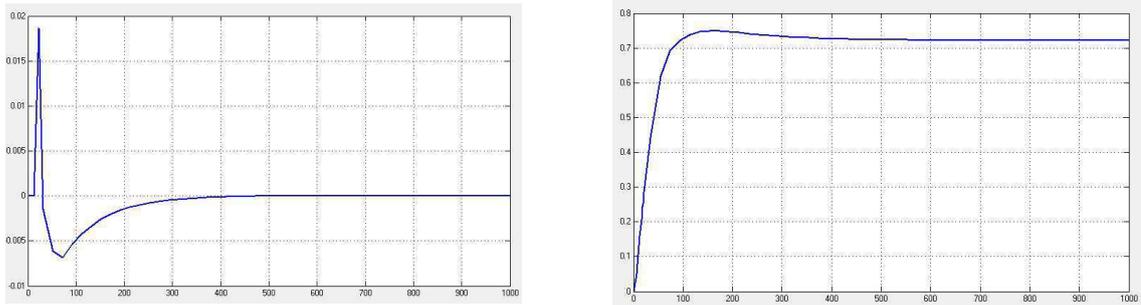


((a)) Temperatura do Peltier 1 devido a PWM 1



((b)) Temperatura do Peltier 1 devido a PWM 2

Figura 23 – Influência das entradas PWM1 e PWM2 no módulo Peltier 1



((a)) Temperatura do Peltier 2 devido a PWM 1 ((b)) Temperatura do Peltier 2 devido a PWM 2

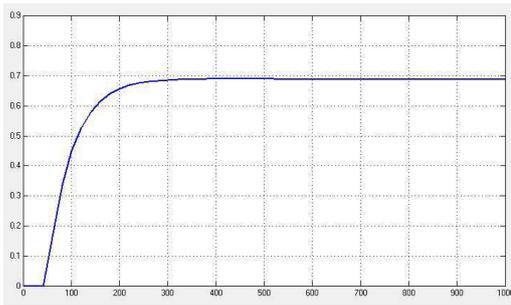
Figura 24 – Influência das entradas PWM1 e PWM2 no módulo Peltier 2

Verifica-se, assim, que o desacoplamento das malhas foi eficiente, uma vez que, para a primeira simulação, a entrada 2 exerce mínima influência em regime permanente sobre o Peltier 1, e no segundo experimento a entrada 1 exerce mínima influência em regime permanente sobre o Peltier 2.

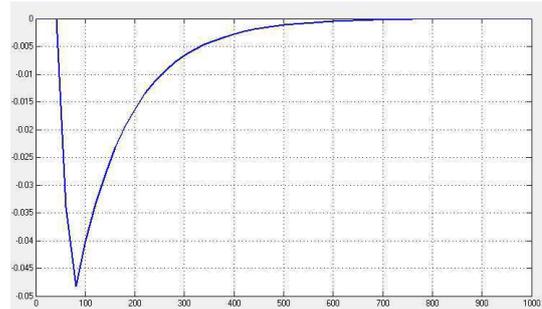
6.3 Desacoplamento Invertido

A técnica de desacoplamento invertido é especificada na [subseção 2.3.3](#). Embora se trate de uma técnica de simples implementação por se tratar de uma realimentação com as funções de transferência do próprio sistema, as funções de transferência da matriz de desacoplamento D e da matriz resultante H são de quarta ordem, possuindo 4 pólos e 4 zeros. Dessa forma, não serão explicitadas aqui as matrizes referentes a esse método de desacoplamento. Porém, as simulações do sistema desacoplado foram feitas no Simulink, devido a simplicidade estrutural do desacoplador.

Procedeu-se, assim, de forma semelhante a feita anteriormente para os outros dois métodos. Foram feitos dois testes para cada desacoplador: no primeiro coloca-se um degrau unitário na entrada 1, enquanto a entrada 2 é zerada; no segundo a entrada 2 recebe um degrau unitário enquanto a entrada 1 é zerada. Na [Figura 25](#) são mostradas as influências das entradas PWM no módulo Peltier 1, e na [Figura 26](#) as influências das entradas PWM no módulo Peltier 2

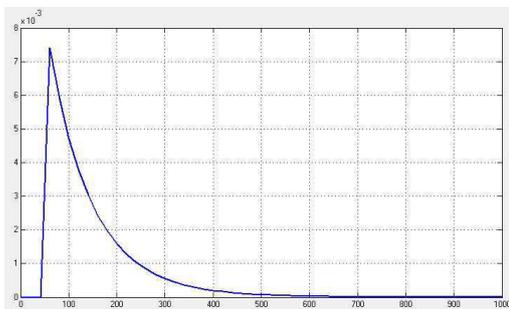


((a)) Temperatura do Peltier 1 devido a PWM 1

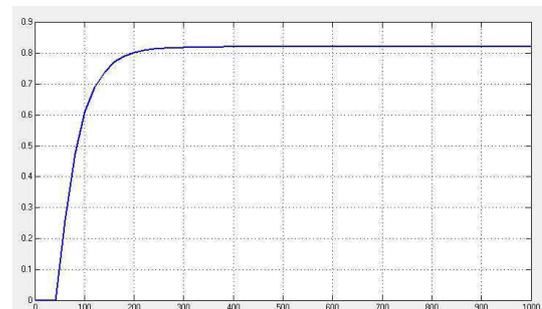


((b)) Temperatura do Peltier 1 devido a PWM 2

Figura 25 – Influência das entradas PWM1 e PWM2 no módulo Peltier 1



((a)) Temperatura do Peltier 2 devido a PWM 1



((b)) Temperatura do Peltier 2 devido a PWM 2

Figura 26 – Influência das entradas PWM1 e PWM2 no módulo Peltier 2

Verifica-se, assim como nos métodos anteriores, que o desacoplamento das malhas foi eficiente, uma vez que, para a primeira simulação, a entrada 2 exerce mínima influência em regime permanente sobre o Peltier 1, e no segundo experimento a entrada 1 exerce mínima influência em regime permanente sobre o Peltier 2.

7 Simulação - Controle usando técnica de sintonia e desacoplamento

Para verificar mais claramente o efeito que o desacoplamento tem sobre o sistema, tomamos o processo estudado em ??, bem como a sua matriz de desacoplamento calculada [Equação 6.2](#) e sua matriz resultante do sistema [Equação 6.4](#), e testaremos uma das diversas técnicas de sintonia de controladores existentes. O objetivo desta simulação é mostrar que, enquanto as malhas não estiverem desacopladas, a realização do controle será dificultada.

Dentre as técnicas de sintonia de controladores existentes, foi escolhida a técnica *Internal Model Control* (IMC), pois tem a vantagem de apresentar uma estrutura simples para lei de controle, em que o projeto do controlador está diretamente relacionado a escolha do filtro IMC, tipicamente um filtro passa-baixas com ordem pré-definida. Dessa forma, o parâmetro de ajuste de interesse é a constante de tempo do filtro, que representa um equilíbrio entre desempenho em malha fechada e robustez a erros do modelo.

Os parâmetros do controlador PI projetado, k_p e k_i são obtidos de acordo com e

$$k_p = \frac{\theta}{k(\theta + \tau)} \quad (7.1)$$

$$k_i = \frac{1}{\theta} \quad (7.2)$$

Inicialmente, tenta-se controlar a planta sem a matriz de desacoplamento. Dessa forma, para o controlador da temperatura do Peltier 1, usa-se apenas a função de transferência G_{11} e para o controlador da temperatura do Peltier 2, usa-se apenas a função de transferência G_{22} . Usando e , temos:

$$k_{p1} = 0,3402 \quad (7.3)$$

$$k_{i1} = 0,0426 \quad (7.4)$$

$$k_{p2} = 0,0789 \quad (7.5)$$

$$k_{i2} = 0,2907 \quad (7.6)$$

Sendo assim, deseja-se setar a temperatura de ambos os módulos Peltier em 28 graus, conforme mostrado na Figura 27. Os resultados obtidos são mostrados na Figura 28.

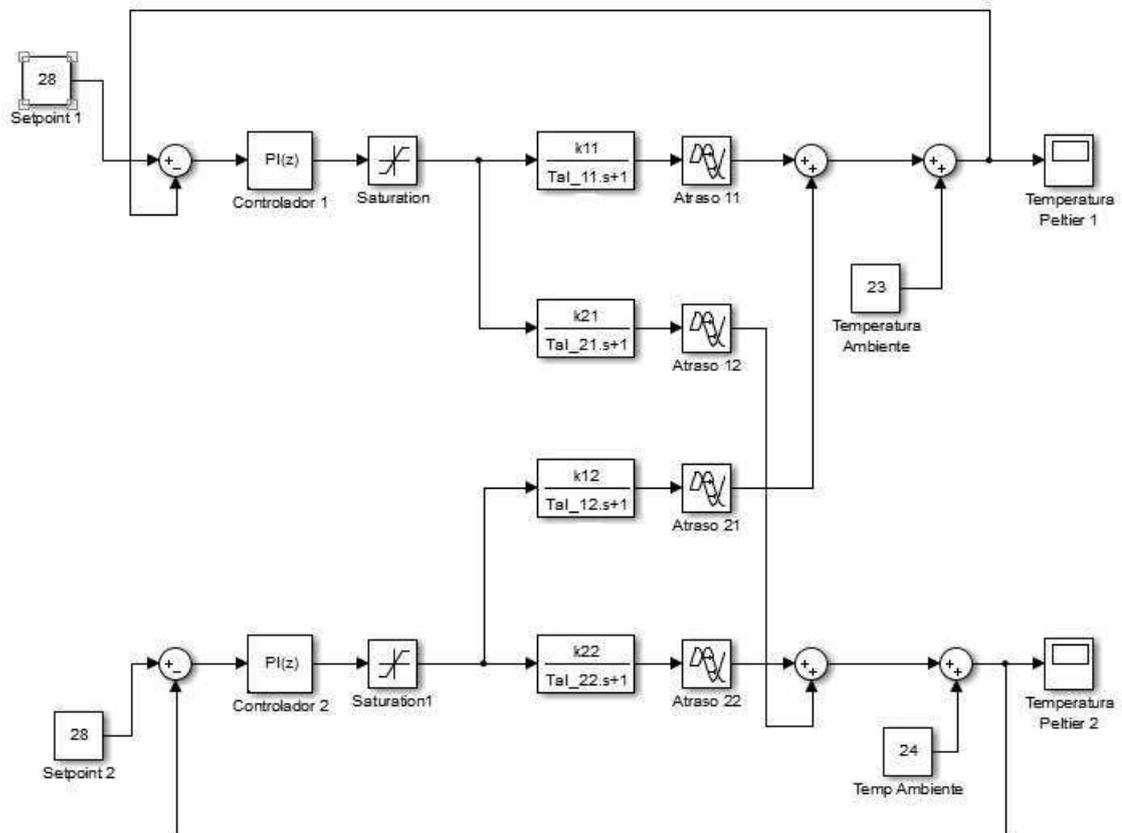
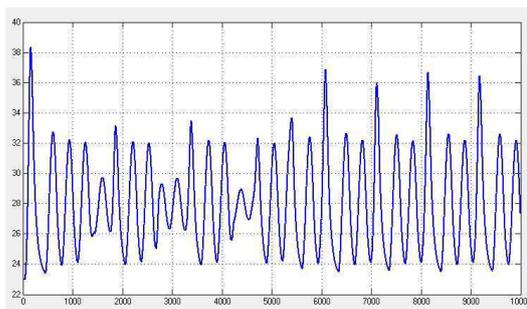
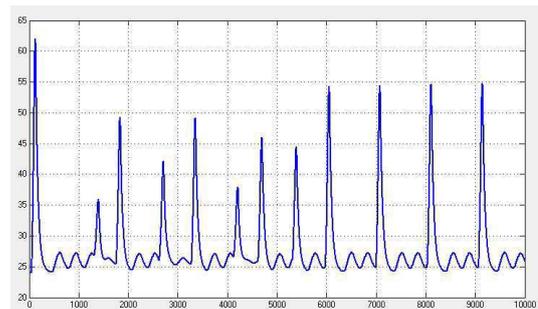


Figura 27 – Controle sem desacoplamento



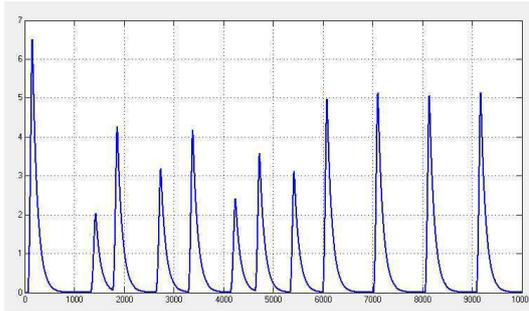
((a)) Temperatura do Peltier 1



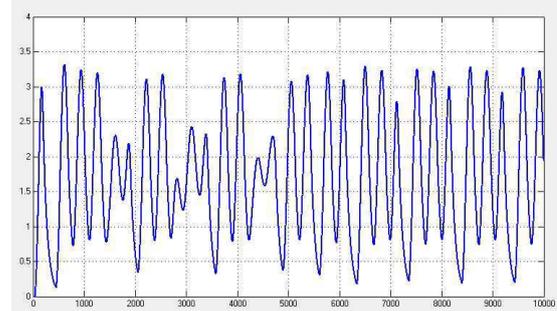
((b)) Temperatura do Peltier 2

Figura 28 – Temperaturas dos módulos Peltier com estratégia de controle sem desacoplamento

Percebe-se que, devido à interação entre as malhas, o controle não é possível usando as técnicas normais de sintonia de controladores. Para ilustrar melhor esse desacoplamento, são mostradas na Figura 29 as influências em separado das entradas cruzadas, o que causa o desacoplamento.



((a)) Influência da entrada 1 no peltier 2



((b)) Influência da entrada 2 no peltier 1

Figura 29 – Influências do desacoplamento nas entradas e saídas cruzadas

Dado esse fato, busca-se agora o desacoplamento e posterior controle. Para o controlador da temperatura do Peltier 1, usa-se a função de transferência resultante H_{11} e para o controlador da temperatura do Peltier 2, usa-se apenas a função de transferência resultante H_{22} . Para a aplicação da técnica de sintonia, podemos aproximar H_{11} e H_{22} por modelos FOPDT, levando em conta que suas raízes relacionadas com as potências de ordem maior são muito próximas de zero. Veja:

$$H = \begin{bmatrix} \frac{1713s^2+70,38s+0,5038}{160000s^3+9909s^2+183s+1}e^{-23,5s} & 0 \\ 0 & \frac{2279s^2+77,41s+0,4907}{200400s^3+10710s^2+183s+1}e^{-23,5s} \end{bmatrix} \quad (7.7)$$

Podemos aproximar o modelo acima de H pelo modelo em , para propósitos de controle:

$$H = \begin{bmatrix} \frac{0,5038}{183s+1}e^{-23,5s} & 0 \\ 0 & \frac{0,4907}{+183s+1}e^{-23,5s} \end{bmatrix} \quad (7.8)$$

Assim, com esse modelo simplificado, achamos os coeficientes dos controladores de acordo com e :

$$k_{p1} = 0,287 \quad (7.9)$$

$$k_{i1} = 0,0163 \quad (7.10)$$

$$k_{p2} = 0,2032 \quad (7.11)$$

$$k_{i2} = 0,0242 \quad (7.12)$$

Sendo assim, deseja-se setar a temperatura de ambos os módulos Peltier em 28 graus, conforme mostrado na Figura 30. Os resultados obtidos são mostrados na Figura 31.

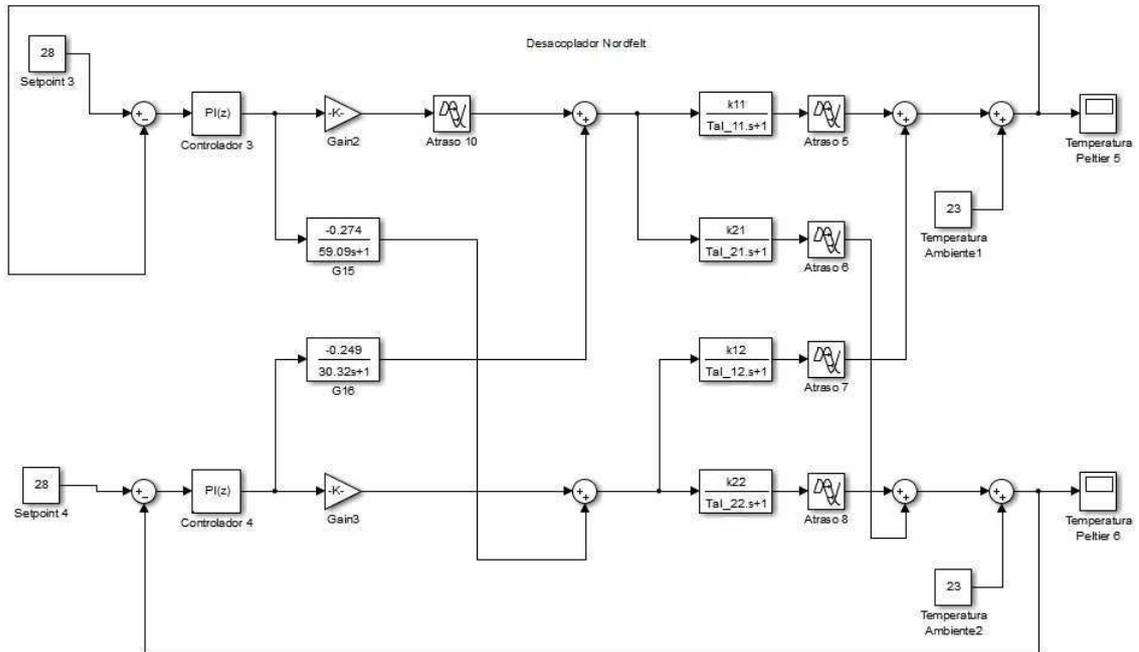
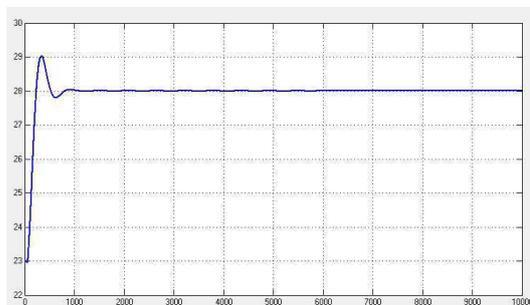
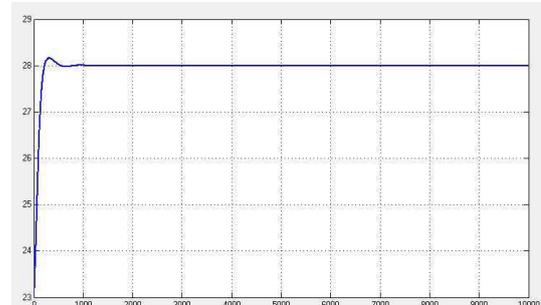


Figura 30 – Controle sem desacoplamento



((a)) Temperatura do Peltier 1



((b)) Temperatura do Peltier 2

Figura 31 – Temperaturas dos módulos Peltier com estratégia de controle com desacoplamento

Percebe-se que o uso do desacoplamento fez com que não houvesse interação entre as malhas, e o controle tornou-se mais fácil, podendo, assim, ser usadas técnicas comuns de sintonia de controladores.

8 Considerações Finais

O estudo de técnicas de identificação de sistemas e de técnicas de desacoplamento são essenciais ao engenheiro electricista, uma vez que a maioria dos processos industriais são de modelagem complexa e possuem interações entre as diversas malhas. Dessa forma, o controle dos processos torna-se de maior complexidade. Tendo em vista a importância destes assuntos, é desejável que estudantes de graduação possam ter acesso a plantas e ferramentas nas quais os mesmos possam aplicar as devidas técnicas relativas a esses assuntos.

Tendo em vista este objetivo, neste trabalho foi desenvolvido um aplicativo para realizar a comunicação entre um computador e um microcontrolador responsável por realizar o controle de dois módulos termoeletrônicos Peltier. O sistema pode trabalhar em malha aberta e realizar a leitura periódica da temperatura dos módulos. Isso possibilita, por exemplo, a identificação do sistema e sintonia de controladores. Além disso, o sistema também pode trabalhar em malha fechada, fazendo com que as temperaturas das faces dos módulos Peltier atinjam um certo valor desejado. Em ambos os casos, é possível supervisionar as temperaturas graficamente para segurança dos equipamentos e usuários. O aplicativo foi testado em uma planta presente no Laboratório de Controle e Otimização da UFCG e verificou-se seu correto funcionamento.

Assim, ficam disponíveis aos laboratórios da UFCG, especialmente ao LIEC, os estudos feitos e aplicativos desenvolvidos neste trabalho para futura aplicação acadêmica, visando o aperfeiçoamento das disciplinas do curso de Engenharia Elétrica que abordem tais assuntos. Espera-se, assim, uma melhor formação acadêmica dos estudantes.

Como futuros trabalhos a serem desenvolvidos, portanto, destaca-se a comunicação entre o computador ligado ao microcontrolador e outros dispositivos via redes industriais, possibilitando o uso do sistema de forma remota e aumentando sua aplicabilidade em indústrias, laboratórios e afins. Também, podem ser estudadas sobre a mesma planta, uma vez que os modelos já estão identificados, outras técnicas de controle de processos multivariáveis.

Referências

- 1 SHINSKEY, F. G. *Process Control Systems: application, design and tuning*. [S.l.]: New York: McGraw-Hill, 1996. Citado 3 vezes nas páginas 2, 9 e 12.
- 2 ASTROM, K. J.; JOHANSSON, K. H.; WANG, Q.-G. Design of decoupled pi controllers for two-by-two systems. *IEEE Proceedings - Control Theory and Applications*, v. 149, p. 74–81, 2002. Citado na página 2.
- 3 LIU, T.; WANG, Q.-G.; HUANG, H.-P. A tutorial review on process identification from step or relay feedback test. *Journal of Process Control*, p. 1597–1623, 2013. Citado na página 3.
- 4 LOOMIS, R. H. Decoupling techniques in multiloop control systems. 1960. Citado na página 6.
- 5 LIU, C. K.; BERGMAN, N. J. On necessary conditions for decoupling multivariable control systems. *IEEE transactions on automatic control*, p. 131–133, 1970. Citado na página 6.
- 6 TRIPATHI, A. N.; NEWTON, R. H. C. Note on necessary and sufficient condition. *IEEE transactions on automatic control*, p. 201–202, 1971. Citado na página 6.
- 7 GILBERT, E. G.; PIVNICHNY, J. H. A computer program for the synthesis of decoupled multivariable feedback systems. *IEEE transactions on automatic control*, v. 7, p. 119–128, 1969. Citado na página 6.
- 8 MORGAN, B. S. The synthesis of linear multivariable systems by state feedback. *Joint Automatic Control Conference*, p. 468–472, 1964. Citado na página 6.
- 9 TRIPATHI, A. N. Decoupling multivariable systems by state feedback: sensitivity considerations. *Proceedings - The Institution of Electrical Engineers*, v. 119, p. 737–742, 1972. Citado na página 6.
- 10 LIU, X.; WANG, X.; HO, D. W. C. Note on necessary and sufficient condition. *IEEE transactions on automatic control*, p. 201–202, 1971. Citado na página 6.
- 11 NAZAR, S.; REKASIUS, Z. V. decoupling of a class od nonlinear systems. *IEEE transactions on automatic control*, p. 257–260, 1971. Citado na página 6.
- 12 ABED, H. A.; KUH, E. invertibility, reproducibility and decoupling of a class of nonlinear systems. p. 61–68, 1971. Citado na página 7.
- 13 SINHA, P. K. State feedback decoupling of nonlinear systems. *Thecnicals notes and correspondence*, p. 487–489, 1977. Citado na página 7.
- 14 LLOYD s. Decoupling a multivariable discrete-time system. *Electronics Letters*, v. 6, p. 831, 1970. Citado na página 7.
- 15 WANG, Q.-G. Auto-tuning of multi-variable pid controllers from decentralized relay feedback. *Automatica*, v. 33, p. 319–330, 1997. Citado na página 7.

- 16 FU, Y. et al. intelligent decoupling control of nonlinear multivariable systems. *Conference on Decision and Control*, p. 1350–1355, 2007. Citado na página 7.
- 17 WANG, Y. et al. Adaptive decoupling switching control based on generalised predictive control. *IET Control Theory and Applications*, p. 1828–1841, 2011. Citado na página 7.
- 18 LIN, C.-M.; MON, Y.-J. Decoupling control by hierarchical fuzzy sliding-mode controller. *IEEE transactions on control systems technology*, p. 593–598, 2005. Citado na página 7.
- 19 LEE, Y.; PARK, S.; LEE, J. H. On interfacing model predictive controllers with low-level loops. *Preprints of 14th World Congress os IFAC*, N, p. 313–318, 1999. Citado na página 7.
- 20 JR, G. A.; BARROS, P. R. Estudo comparativo de técnicas de projeto de desacoplador para sistemas multivariáveis. *Divulgação Científica e Tecnológica do IFPB*, v. 28, p. 130–143, 2015. Citado 2 vezes nas páginas 8 e 9.
- 21 HAGGLUND, P. N. and T. Decoupler and pid controller design of tito systems. *Journal of Process Control*, v. 16, p. 923–936, 2006. Citado na página 10.
- 22 WANG, Q.-G.; HWANG, B.; GUO, X. Auto-tuning of tito decoupling controllers from step tests. *ISA Transactions*, v. 47, p. 103–110, 2000. Citado na página 11.
- 23 ELECTRONICS, G. *USBizi User Manual*. <http://www.mikrocontroller.net/attachment/61823/USBizi_User_Manual.pdf>. <http://www.mikrocontroller.net/attachment/61823/USBizi_User_Manual.pdf>. Citado na página 14.
- 24 KÜHNER, J. *Expert .NET Micro Framework*. second edition. Totowa, New Jersey: Apress, 2009. Citado na página 16.
- 25 ELECTRONICS, G. *Guia para iniciantes em C# e .NET Micro Framework*. 2011. Citado na página 16.