

DAS Departamento de Automação e Sistemas
CTC Centro Tecnológico
UFSC Universidade Federal de Santa Catarina

Identificação e Emulação de Campos Solares

*Relatório submetido à Universidade Federal de Santa Catarina
como requisito para a aprovação da disciplina:
DAS 5511: Projeto de Fim de Curso*

Lucas Peruzzo

Florianópolis, Julho de 2018

Identificação e Emulação de Campos Solares

Lucas Peruzzo

Esta monografia foi julgada no contexto da disciplina

DAS 5511: Projeto de Fim de Curso

e aprovada na sua forma final pelo

Curso de Engenharia de Controle e Automação

Prof. Julio Elias Normey Rico

Banca Examinadora:

Paulo Renato da Costa Mendes
Orientador na Empresa

Prof. Julio Elias Normey Rico
Orientador no Curso

Prof. Marcelo de Lellis Costa de Oliveira, Avaliador

Amanda Letícia Maran, Debatedor

Lucas Testoni Macedo, Debatedor

Agradecimentos

Agradeço à Universidade Federal de Santa Catarina, em especial ao Departamento de Automação e Sistemas e aos professores pela oportunidade de estudo ao longo desses anos.

Agradeço ao professor Julio Elias Normey Rico pela dedicação como professor e orientador desse trabalho.

Agradeço ao doutor Paulo Renato da Costa Mendes tanto por orientar e direcionar a sequência de execução do trabalho, como por ajudar tecnicamente com ideias de possíveis soluções para os problemas que ocorreram ao longo do projeto. Agradeço também pelo suporte e incentivo para continuar, principalmente no final, onde parecia não haver mais tempo para conclusão do trabalho.

Agradeço ao graduando Henrique Roos pelo amplo auxílio técnico tanto nos problemas hidráulicos como nos problemas lógicos que surgiram.

Agradeço ao professor e doutorando José Dolores Vergara Dietrich por, mesmo estando distante, ajudar nas soluções técnicas relacionadas à planta piloto, sempre respondendo rapidamente.

Agradeço à mestrandia Alice Ferreira Branco pela disponibilidade e ajuda quando solicitado.

Resumo

A crescente demanda energética mundial associada ao ideal cada vez mais sustentável proporciona um cenário de grande desenvolvimento no setor das energias limpas. Para contribuir com esse desenvolvimento, foi criado no Departamento de Automação e Sistemas (DAS) da Universidade Federal de Santa Catarina o Grupo de Pesquisa em Energias Renováveis (GPER), que é o local onde foi desenvolvido o trabalho apresentado neste documento. Esse trabalho em específico é relacionado à energia heliotérmica, que produz baixíssimo impacto ambiental. Essa fonte de energia consiste em transformar energia radiante (irradiação) em energia térmica, a qual posteriormente pode ser utilizada para produção de energia elétrica. O princípio de funcionamento é absorver o máximo da irradiação solar usando equipamentos adequados e converter a irradiação solar em aumento da temperatura do fluido que circula pelo sistema. O GPER atua em pesquisas relacionadas ao controle de temperatura desse fluido, portanto optou-se por construir uma planta piloto nas dependências do DAS para se realizar estudos de controle. Como o espaço para construção é reduzido, foi pensado em uma estratégia para que a planta piloto emulasse um sistema de grande porte. Dessa forma, o trabalho apresentado nesse documento destina-se a desenvolver o sistema de emulação de campos solares de grande porte. Ao longo do documento serão apresentados os procedimentos de identificação e emulação do campo solar, bem como as melhorias realizadas na planta piloto e as adaptações realizadas.

Palavras-chave: Identificação de sistemas, controle de temperatura, energia heliotérmica, emulação de campos solares.

Abstract

The growing global energy demand coupled with the increasingly sustainable ideal provides a major development landscape in the clean energy sector. To contribute to this development, the Renewable Energy Research Group (GPER) was created in the Department of Automation and Systems (DAS) of the Federal University of Santa Catarina, where this work was developed. This work is related to the heliothermic energy, that produces a very low environmental impact. This energy source consists of transforming radiant energy (irradiation) into thermal energy, which can later be used to produce electricity. The principle of operation is to absorb the maximum of the solar irradiation using suitable equipment and to convert that irradiation in increase of the temperature of the fluid that circulates by the system. The GPER acts in research related to the temperature control of this fluid, so it was decided to build a pilot plant in the premises of the DAS to carry out control studies. As the space for construction is reduced, it was thought of a strategy for the pilot plant to emulate a large system. In this way, the work presented in this document is intended to develop the emulation system of large solar fields. Throughout the document will be presented the procedures of solar fields identification and emulation, as well as the proposed improvements in the pilot plant.

Keywords: Identification of systems, temperature control, heliothermic energy, emulation of solar fields.

Lista de ilustrações

Figura 1 – Sistema hélio térmico sem armazenamento de energia. Fonte: [1]	20
Figura 2 – Mecanismo de concentração solar de placa plana. Fonte: [2]	21
Figura 3 – Painel elétrico principal	25
Figura 4 – Imagem do painel elétrico instalado	25
Figura 5 – Interface do sistema supervisorio	28
Figura 6 – Diagrama de blocos do sistema supervisorio	29
Figura 7 – Timer	32
Figura 8 – Timer no estado de interrupção	32
Figura 9 – Leitura dos sinais de entrada obtidos de um arquivo de extensão .txt	33
Figura 10 – Inicialização da saída.txt e acionamento da estrutura que guarda os dados	34
Figura 11 – Função que converte número em string	34
Figura 12 – Implementação das chaves para as vazões	35
Figura 13 – Implementação das chaves para os aquecedores	35
Figura 14 – Condição para a bomba desligada	36
Figura 15 – Acionamento da bomba	37
Figura 16 – Diagrama de blocos para identificação do SCA real.	41
Figura 17 – Exemplo de sinal PRS	44
Figura 18 – Sinais para identificar o atraso de transporte	45
Figura 19 – Sinal da potência de entrada e temperatura de saída do aquecedor	45
Figura 20 – Sinal da potência de entrada e temperatura de saída do aquecedor	46
Figura 21 – Sinal da potência de entrada e vazão	47
Figura 22 – Resposta do sistema à potência	48
Figura 23 – Identificação por aproximação de primeira ordem	49
Figura 24 – Validação do sistema	50
Figura 25 – Erro entre os dados experimentais e a saída do modelo	50
Figura 26 – Sinais obtidos no primeiro ensaio	51
Figura 27 – Irradiação do primeiro ensaio	52
Figura 28 – Sinais obtidos no segundo ensaio	52
Figura 29 – Irradiação do segundo ensaio	53
Figura 30 – Validação do primeiro ensaio	54
Figura 31 – Validação do segundo ensaio	54
Figura 32 – Diagrama de blocos da emulação	57
Figura 33 – Diagrama de blocos do sistema completo	59
Figura 34 – Diagrama de blocos da realimentação linearizante	60
Figura 35 – Malha fechada do sistema com realimentação linearizante	60
Figura 36 – Estrutura de controle interno	62

Figura 37 – Estrutura de predição do Preditor de Smith	62
Figura 38 – Preditor de Smith	63
Figura 39 – Diagrama de blocos no Simulink	65
Figura 40 – Simulação do sistema identificado	66
Figura 41 – Simulação do sistema com realimentação linearizante	66
Figura 42 – Simulação do sistema com controle PI sem atraso	67
Figura 43 – Simulação do sistema com atraso e a estrutura do Preditor de Smith	67
Figura 44 – Simulação do sistema com a estrutura do Preditor de Smith e atraso do modelo diferente do real	68
Figura 45 – Simulação do sistema com 20% de erro nos parâmetros	68
Figura 46 – Variável manipulada sem saturação com o sistema sem atraso	69
Figura 47 – Rastreamento de referência	70
Figura 48 – Variável manipulada	70
Figura 49 – Resposta do sistema com saturação na variável $P(k)$	71
Figura 50 – Exemplo de 2 sinais de temperatura	83
Figura 51 – Exemplo de esquemático OneWire. Retirado de: [3]	84
Figura 52 – Temperatura de saída do painel para diferentes resistores pullup	85
Figura 53 – Sinais sem o circuito de potência elétrica ligado	86
Figura 54 – Mesmos sinais da figura 50 com remoção de valores 127	87
Figura 55 – Verificação da possibilidade de não aquecimento da água	88
Figura 56 – Sinais depois de fixar os endereços dos sensores	90
Figura 57 – Sinais de irradiação	90
Figura 58 – Problemas resolvidos	92
Figura 59 – Exemplo de sinal interpolado usando a rotina criada	94

Lista de tabelas

Tabela 1 – Principais componentes hidráulicos	24
Tabela 2 – Principais componentes da instrumentação	26
Tabela 3 – Variáveis e suas unidades	28
Tabela 4 – Parâmetros do processo	40
Tabela 5 – Variáveis do processo	40
Tabela 6 – Endereços dos sensores	89

Lista de abreviaturas e siglas

UFSC	Universidade Federal de Santa Catarina
DAS	Departamento de Automação e Sistemas
GPÉR	Grupo de Pesquisa em Energias Renováveis
LSSFE	Large Scale Solar Field Emulator
SCA	Solar Collector Assembly
MPC	Model Predictive Control
PRBS	Pseudorandom binary sequence
PRS	Pseudorandom sequence

Sumário

1	INTRODUÇÃO	19
1.1	A energia heliotérmica	19
1.2	Motivação	20
1.3	Objetivo	21
1.4	Estrutura do documento	22
2	LOCAL DE TRABALHO	23
2.1	A LSSFE	23
2.1.1	Instalação hidráulica	23
2.1.2	Instalação elétrica	24
2.1.3	Instrumentação	26
2.1.4	Aquisição e processamento dos sinais	26
2.1.5	Sistema supervisorio	28
2.2	Conclusões	29
3	ADAPTAÇÕES REALIZADAS NO SISTEMA	31
3.1	Modificações para aquisição de dados	31
3.1.1	O problema	31
3.1.2	Timer	31
3.1.3	Leitura de dados	33
3.1.4	Gravação de dados	34
3.1.5	Acionamento dos ramos em malha aberta	35
3.2	Sistema de Segurança	36
3.2.1	Primeira estratégia	36
3.2.2	Segunda estratégia	36
3.2.3	Terceira estratégia	38
3.3	Conclusões	38
4	IDENTIFICAÇÃO DO SISTEMA	39
4.1	Modelo do aquecimento elétrico	39
4.2	Modelo do SCA	39
4.3	Mínimos quadrados	40
4.3.1	Aplicação ao modelo do aquecedor	41
4.3.2	Aplicação ao modelo do SCA	42
4.4	Sinal PRS	43
4.4.1	Implementação	43

4.5	Identificação dos aquecedores elétricos	44
4.5.1	Resultados	50
4.6	Identificação do SCA	51
4.6.1	Resultados	52
4.7	Conclusões	54
5	EMULAÇÃO	57
5.1	Estratégia	57
5.1.1	Projeto do controlador interno	58
5.1.2	Compensação do atraso de transporte	61
5.2	Lei de controle	63
5.3	Simulações	65
5.3.1	Simulação de temperatura experimental	69
5.4	Implementação	69
5.5	Conclusões	70
6	CONSIDERAÇÕES FINAIS E PERSPECTIVAS FUTURAS	73
	REFERÊNCIAS	75
	APÊNDICES	77
	APÊNDICE A – CÁLCULO DA VAZÃO DE SEGURANÇA	79
	APÊNDICE B – DEFINIÇÃO DE T_{SI}	81
	APÊNDICE C – PROBLEMAS NA MEDIÇÃO DAS TEMPERATURAS	83
C.1	Sobre os sensores	84
C.2	Problema de 127C	84
C.2.1	Resistor pullup	85
C.2.2	Circuito de alimentação desligado	86
C.2.3	Resolução do problema	86
C.3	Problema de mudança repentina de valor	86
C.3.1	Análise do código	88
C.3.2	Relação com a irradiação	90
C.4	Conclusões	91
	APÊNDICE D – ROTINA PARA ELIMINAÇÃO DE VALORES INCORRETOS	93

APÊNDICE E – IMPLEMENTAÇÃO DO SINAL PRS	95
---	----

1 Introdução

A demanda energética mundial vem crescendo ao longo dos anos, seja pelo aumento da população, seja pelo desenvolvimento tecnológico. Acompanhado da demanda, vem a produção de energia, que dependendo do tipo da fonte de energia pode ter como consequência algum grau de impacto ambiental. O impacto no meio ambiente começou a ser objeto de preocupação dos líderes globais nas últimas décadas, o que resultou na busca de tecnologias de geração de energia menos poluentes, mais eficientes e sustentáveis. Entre as tecnologias menos poluentes, encontra-se o grupo das fontes renováveis, sendo aquelas cujos recursos (matéria-prima) são reabastecidos naturalmente, como por exemplo a energia solar, eólica, hidráulica, geotérmica, maremotriz e biomassa.

O Brasil encontra-se em uma posição favorável à utilização de vários desses recursos naturais, com grande destaque para a energia solar. Uma vez que o país tem clima tropical, cuja maior parte do seu território se encontra entre a linha do equador e o Trópico de Capricórnio, o que faz com que se tenha excelente incidência de irradiação solar [4].

Por esse motivo, e pela alta demanda por soluções de energias renováveis, foi criado na Universidade Federal de Santa Catarina (UFSC) o Grupo de Pesquisa de Energias Renováveis (GPER), Departamento de Automação e Sistemas (DAS); onde foi desenvolvido o trabalho descrito neste documento (o GPER é melhor apresentado no capítulo 2).

1.1 A energia heliotérmica

A produção de energia elétrica a partir da energia solar (denominada energia heliotérmica) é muito parecida com a energia termelétrica. A grande diferença está na fonte primária da energia. A termelétrica queima carvão, gás ou óleo para produzir calor e assim movimentar a turbina que é acoplada a um gerador.

Na usina heliotérmica o provimento de calor vem da irradiação solar refletida pelos espelhos para um tubo por onde passa o fluido de transferência de calor. O fluido quente é responsável pela produção de vapor que movimenta a turbina. Uma das grandes vantagens da energia heliotérmica é que a energia calorífica pode ser armazenada em, por exemplo, armazenadores de sal fundido, visando a continuidade da geração de energia quando não houver sol [1], bem como manter os níveis de geração constantes quando há nuvens ou baixa incidência solar.

No sistema de geração de energia, o campo solar é o sub-sistema onde acontece a coleta de calor, onde se transforma a irradiação em energia térmica. Ele é formado por vários ramos em paralelo. Cada um desses ramos recebe o fluido frio e devolve fluido

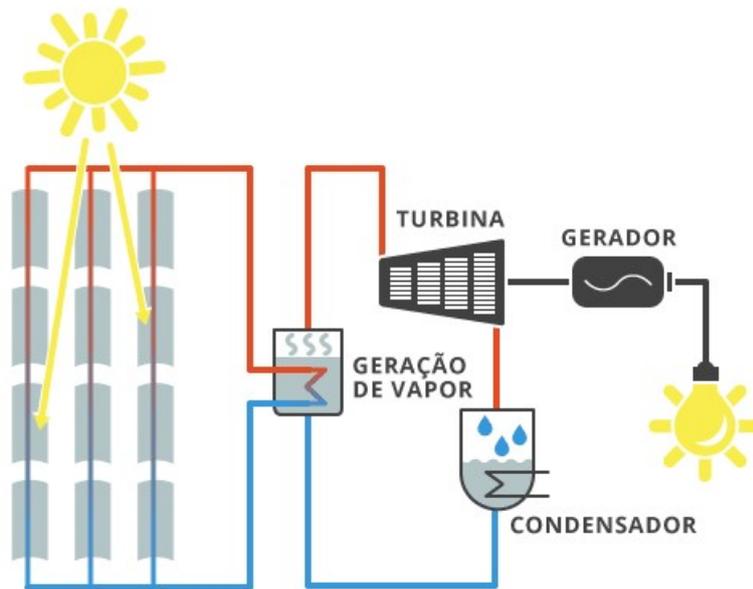


Figura 1 – Sistema hélio térmico sem armazenamento de energia. Fonte: [1]

quente para o sistema. Um ramo pode possuir vários *SCAs* - (*Solar Collector Assemblies*) ligados em série. Um *SCA* é um conjunto de coletores solares agrupados, formando um coletor solar equivalente maior. O coletor solar é a unidade do campo solar responsável por transformar a irradiação em energia térmica. Existem alguns tipos de coletores, cada um com suas particularidades técnicas.

Dessa forma, dependendo da configuração, um sistema de grande porte pode possuir muitos ramos, cada ramo pode possuir muitos *SCA* e cada *SCA* pode ser formado por muitos coletores solares.

Existem diversos tipos de mecanismos de concentração da irradiação como: calha parabólica, fresnel, torre solar, disco parabólico, painel solar de placa plana etc. O trabalho descrito neste documento envolve o mecanismo do tipo painel solar de placa plana ou em inglês *flat plate*. A figura 2 ilustra de forma simplificada o funcionamento desse mecanismo. O número 1 indica a irradiação solar incidente, o 2, a entrada de fluido frio, o 3, a saída de fluido quente, o 4, aponta para o encapsulamento da unidade que deve possuir baixa perda térmica, o 5, indica os tubos internos coletores de calor, e o 6 é o vidro apropriado para alta absorção e baixa reflexão de irradiação.

1.2 Motivação

Para desenvolver pesquisas em controle aplicadas a campos solares de grande porte existe a necessidade de realização de ensaios práticos para validação das técnicas desenvolvidas. Para isso o ideal seria a utilização de um campo de coletores solares com

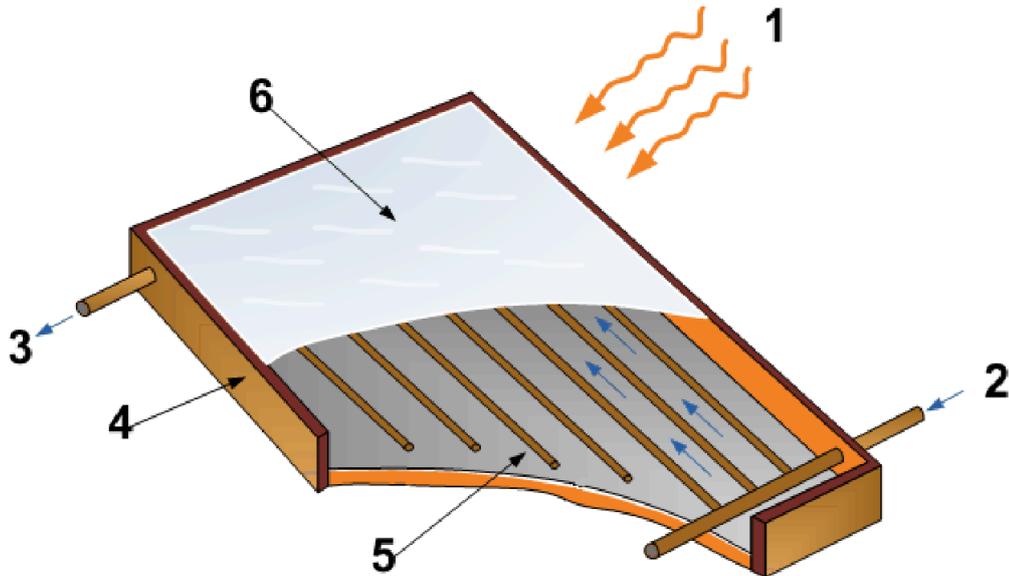


Figura 2 – Mecanismo de concentração solar de placa plana. Fonte: [2]

tamanho industrial, ou pelo menos, suficientemente grande para apresentar características e problemas que não ocorrem em sistemas menores.

Uma das grandes características que influenciam o funcionamento de sistemas heliotérmicos de grande porte e conseqüentemente no projeto de controle é a passagem de nuvens por cima do campo de coletores solares. As nuvens em muitos casos podem não cobrir totalmente o campo, ou ainda mover-se lentamente, cobrindo partes diferentes do campo ao longo do tempo. Esse tipo de perturbação impacta profundamente a temperatura final do fluido, já que ramos com baixa irradiação podem fazer com que a temperatura final seja baixa e insuficiente para geração de energia elétrica.

Para poder reproduzir os fenômenos que acontecem nos grandes campos em laboratório de tamanho reduzido, o GPER optou pela construção de uma planta piloto que emula um sistema de grande porte. O objetivo da planta é validar na prática pesquisas como o artigo [5].

1.3 Objetivo

As técnicas de controle abordadas nos estudos para esse tipo de sistema são do tipo *MPC* (*Model Predictive Control*) ou controle preditivo baseado em modelo. Muitas aplicações de *MPC* têm sido realizadas em plantas heliotérmicas com grande sucesso. O bom desempenho do *MPC* nestas plantas pode ser justificado pela possibilidade de considerar modelos multivariáveis e não lineares com restrições na formulação do *MPC*. Assim, no projeto de pesquisa do GPER um controle *MPC* será projetado para a planta piloto.

Para implementar e sintonizar um *MPC* é necessário construir um modelo do processo. Como a planta piloto foi recentemente construída, no início da realização deste trabalho não havia um modelo do sistema, nem a implementação de uma estratégia para emulação dos painéis solares.

Dessa forma, esse projeto de fim de curso tem como objetivos a identificação do modelo matemático relativo à planta piloto e emulação de partes de um campo solar de grande porte de forma a permitir a emulação de painéis solares para simular a passagem de nuvens em partes do campo.

1.4 Estrutura do documento

O documento é apresentado conforme a lista abaixo:

1. No capítulo 2 são apresentados o local onde ocorreu o trabalho, um breve resumo do problema e as condições em que o sistema se encontrava no início do projeto;
2. No capítulo 3 são apresentadas as adaptações realizadas no sistema para promover a correta obtenção dos dados;
3. No capítulo 4 é apresentada a teoria e implementação da identificação dos dois sistemas envolvidos nesse projeto;
4. No capítulo 5 são apresentadas a estratégia de emulação, o desenvolvimento e implementação em simulação;
5. Por fim, no capítulo 6 são apresentadas as considerações finais e perspectivas.

2 Local de trabalho

Com o objetivo de desenvolver tecnologias nacionais para controle e otimização de sistemas de geração de energia baseados em energia renovável foi criado no Departamento de Automação e Sistemas da UFSC o Grupo de Pesquisa em Energias Renováveis (GPER). Entre os campos de pesquisa do grupo, está a geração de energia heliotérmica (seção 1.1), no qual os objetos de investigação são a utilização de sistemas de controle visando maximizar a energia calorífica extraída pelo campo solar.

Com o objetivo de validar na prática os sistemas de controle desenvolvidos, foi criada uma planta piloto denominada LSSFE (*Large Scale Solar Field Emulator*) como descrito na seção 1.2. Esse LSSFE foi recentemente construído sob o comando do doutorando José Dolores Vergara Dietrich e com auxílio do graduando Henrique Roos. Ela localiza-se ao lado do prédio do Departamento de Automação e Sistemas, mais especificamente em cima do Laboratório Multifásico do DAS-UFSC.

2.1 A LSSFE

Nessa seção será explicado brevemente como o sistema está instalado e suas principais partes.

O processo começa no *SCA* do tipo *flat plate*. Esse é o componente da *LSSFE* que absorve a irradiação e a transforma em energia térmica. O fluido passa pelo *SCA* e é aquecido, depois vai para um tanque de água. Esse tanque de água é o provedor de fluido para todo o sistema (o tanque armazena o fluido que circula pelo sistema), dessa forma o fluido circula pelo sistema e a temperatura média da água cresce lentamente com o acionamento da planta. Além de alimentar o *SCA*, o tanque de água alimenta 3 ramos em paralelo. Cada um dos ramos possui um aquecedor elétrico. Esses aquecedores são os responsáveis por aquecer a água em cada ramo, emulando então o comportamento de um *SCA*. Esse ramos são no final unidos por uma única saída. Essa saída se junta à saída do painel solar, formando a saída final do sistema. Uma representação gráfica desse sistema hidráulico pode ser vista na figura 5.

2.1.1 Instalação hidráulica

A instalação hidráulica é formada pelos itens da tabela 1.

A caixa d'água superior funciona como reservatório rápido de água e também para haver circulação de água no painel quando o sistema está desligado. O tanque de água quente serve para re-circular a água enquanto o sistema opera. Dessa forma, a água

circulante aumenta a sua temperatura gradativamente. Para resfriar essa água existe a caixa d'água inferior. Essa planta possui apenas um ramo com captação de energia solar real e esse ramo possui apenas um SCA, formado por quatro painéis do tipo placa plana acoplados em série.

As válvulas são usada para regular a passagem de água por cada ramo e com isso alterar a temperatura da mesma. Já a bomba centrífuga é a responsável por fazer o fluído circular.

Itens
Caixa d'água superior
Caixa d'água inferior
Tanque de água quente
SCA
Tubos de PPR
Válvulas
Bomba Centrífuga

Tabela 1 – Principais componentes hidráulicos

2.1.2 Instalação elétrica

As figuras 3 e figuras 4 apresentam duas fotos com a indicação dos principais componentes da instalação elétrica e uma breve descrição de suas funções. A instalação elétrica é usada para o acionamento dos componentes ativos do sistema.

Toda a instalação elétrica passa pelo painel elétrico principal. Externo a esse painel, existe um botão de emergência ($n^{\circ}8$) que interrompe a alimentação da planta. Internamente, existem dois circuitos elétricos independentes: o circuito elétrico de potência que alimenta a bomba e os aquecedores, e o circuito elétrico de comando, que alimenta os microcontroladores, sensores e atuadores. Esses circuitos são acionados através dos disjuntores ($n^{\circ}1$).

Um relê é usado para alimentação da bomba e o outro para a alimentação dos motores de passo¹ ($n^{\circ}3$). Os aquecedores elétricos são controlados pelos 3 relês de estado sólido ($n^{\circ}2$), que são comandados pelo Arduíno Mega ($n^{\circ}4$) e alimentados com 220V. Os aquecedores possuem 3 estados de acionamento manual: desligado, mínima potencia e máxima potência. O ($n^{\circ}7$) é a régua que alimenta os Arduínos. O ($n^{\circ}9$) é a placa de aquisição dos sensores de temperatura e irradiação. Ali estão instalados o Arduíno Nano e o sensor de corrente. O ($n^{\circ}5$) é o ponto de encontro do circuito de comando. O ($n^{\circ}6$) é o regulador de tensão para alimentação do Arduíno Nano².

¹ Os motores de passo são usados para abertura das válvulas.

² O Arduíno Nano foi usado pois não haviam pinos suficientes no Arduíno Mega para todos os sensores da planta.

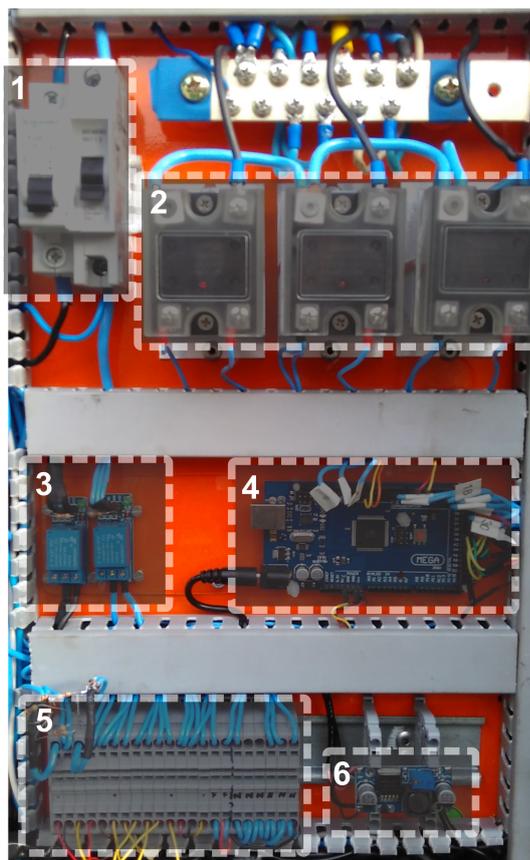


Figura 3 – Painel elétrico principal

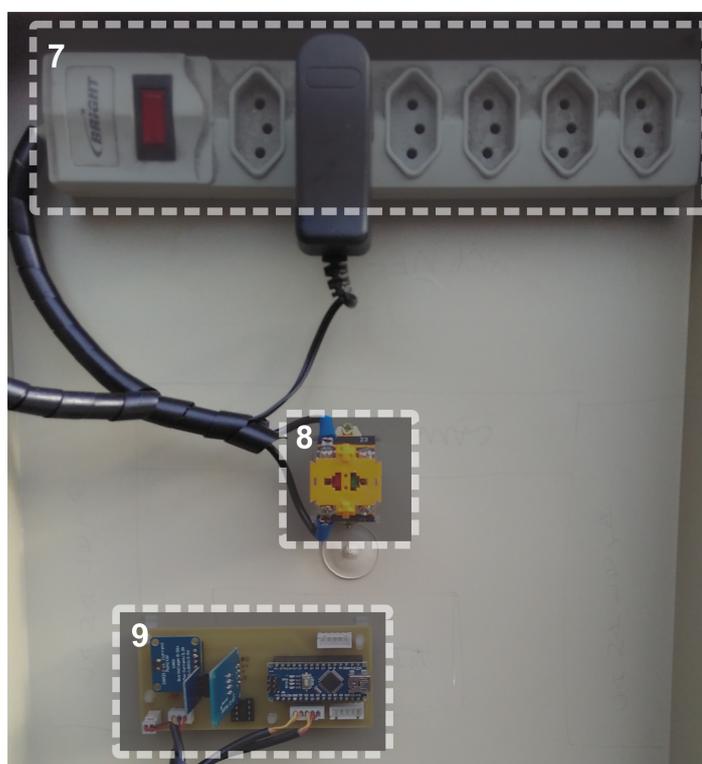


Figura 4 – Imagem do painel elétrico instalado

Além dos dispositivos descritos acima, ainda há os aquecedores elétricos. Cada um tem potência máxima de 2900W quando ligados manualmente na posição 1 (manual [6]). A potência é acionada através do LabView em uma faixa de valores que vai entre 0 e 255, esse valor é convertido em largura de pulsos na corrente que alimenta o aquecedor. Então 255 corresponde a um sinal em 100% do tempo em nível alto e 0 corresponde a um sinal em 100% do tempo em nível baixo. Portanto, sendo x o valor de comando no LabView e y a potência real no aquecedor:

$$y = \frac{2900x}{255}W \quad (2.1)$$

2.1.3 Instrumentação

A instrumentação é formada pelos itens da tabela 2:

Quant.	Item	Nome
4	motor de passo	JK42HS40-1704
4	driver	
8	sensor de temperatura	DS18B20
1	sensor de irradiação	-
1	sensor de corrente	INA219
6	sensor de vazão	FS300A
1	Arduino Mega	-
1	Arduino Nano	-

Tabela 2 – Principais componentes da instrumentação

O ponto central de recebimento das informações é o Arduino Mega, que comunica-se diretamente com o computador através da saída serial. Ele recebe os dados de todos os sensores (direta ou indiretamente) e os envia ao supervisor. Também recebe os sinais de atuação vindos do LabView. O Arduino Nano recebe os dados dos sensores de temperatura e do sensor de irradiação e envia-os ao Arduino Mega através da serial RX/TX. Os drivers são usados para fazer a conversão dos sinais do Arduino para os motores de passo. Os motores de passo são usados para controlar a abertura das válvulas. Os sensores de temperatura são digitais e comunicam-se com o Arduino através do protocolo OneWire. O sensor de corrente INA219 é responsável pela conversão do sinal de irradiação. Os sensores de vazão enviam sinais pulsados ao Arduino que representam quanto o sensor de vazão girou em determinado tempo e com isso calcula-se a vazão.

2.1.4 Aquisição e processamento dos sinais

Os sinais de vazão e irradiação são medidos indiretamente, então as informações lidas devem ser processadas para transformá-las em unidades conhecidas. Para o sinal de vazão, a medição é feita através da contagem de pulsos, que representa a velocidade de

giro do sensor. No caso da irradiação mede-se a corrente e converte-se o valor. Abaixo são mostrados os algoritmos para conversão dessas variáveis.

```

1 // ROTINA SENSOR IRRADIACAO
2 void Read_INA219_Values(void)
3 {shuntvoltage = ina219.getShuntVoltage_mV();
4 busvoltage = ina219.getBusVoltage_V();
5 current_mA = ina219.getCurrent_mA();
6 loadvoltage = busvoltage + (shuntvoltage / 1000);}
```

O cálculo para conversão do valor lido "current_mA" em irradiação (W/m^2) é dado pela fórmula $Irr = current_mA \times 1000/50$.

```

1 attachInterrupt(0, incpulso_1, RISING); //Configura o ...
   pino 2(Interruptao 0) para trabalhar como interrupcao
2 .
3 .
4 .
5 sei(); //Habilita interrupcao
6 delay (1000); //Aguarda 1 segundo
7 cli(); //Desabilita interrupcao
8 vazao_1 = contaPulso_1 / 5.5;
9 .
10 .
11 .
12 void incpulso_1 ()
13 { contaPulso_1++;}
```

O cálculo para conversão da contagem de pulsos em L/min é $vazao_1 = \frac{contaPulso_1}{5.5}$

Para os sensores de temperatura e com o uso das bibliotecas OneWire e DallasTemperature o valor lido pelo Arduino já é em ($^{\circ}C$).

Os dados são enviados do Arduino Nano para o Arduino Mega através de comunicação serial RX/TX, porém como o Nano só possui uma saída serial, essa foi reservada à comunicação com o computador usando a porta USB. Portanto foi usada uma porta digital e a biblioteca SoftwareSerial para criar o sinal serial a ser enviado ao Arduino Mega.

No Arduino Mega há um controle proporcional local de vazão. Assim o sinal a ser enviado pelo sistema supervisorio deve ser um sinal de referência em $\frac{L}{min}$.

```

1  int k_u = 1;
2  int erro_1 = ref_vazao_1*10 - vazao_1*10;
3  int u_1 = k_u * erro_1;

```

Na tabela 3 mostram-se unidades das variáveis do processo.

Tabela 3 – Variáveis e suas unidades

Variável	Unidade
Temperatura	$^{\circ}C$
Irradiação	W/m^2
Vazão	L/min
Potência elétrica	$2900W/255$

2.1.5 Sistema supervisório

O sistema supervisório (figura 5) é implementado em LabView e troca informações com o Arduino Mega, recebe os dados de todos os sensores e envia os sinais de atuação. Nele, pode-se observar as temperaturas, as vazões e a irradiação e atuar nos aquecedores e nas vazões.

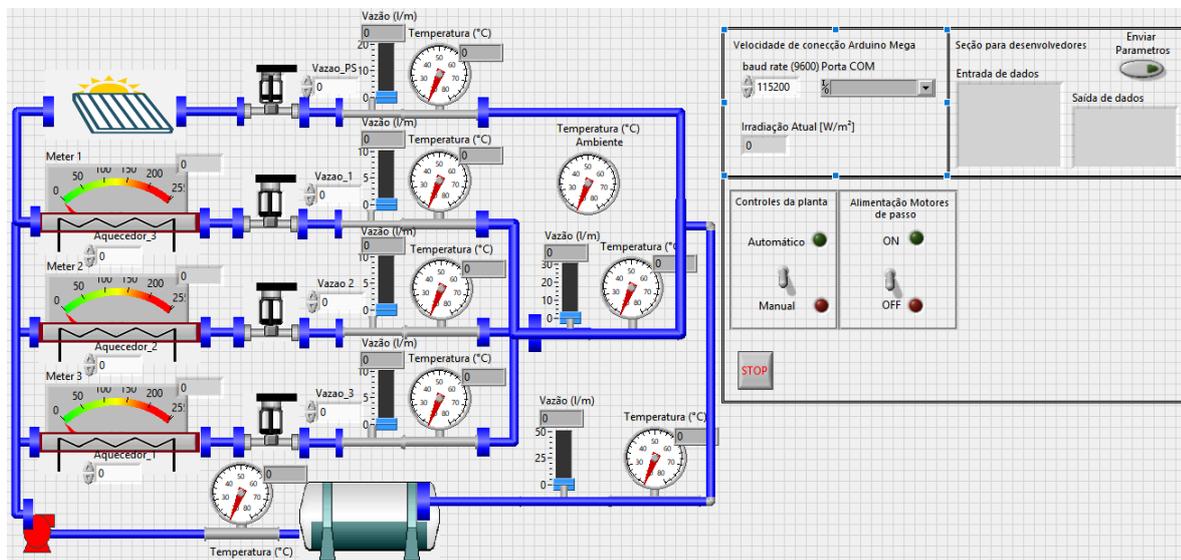


Figura 5 – Interface do sistema supervisório

A figura 6 mostra o diagrama de blocos completo do supervisório implementado em LabView. O importante nesse sistema é notar a implementação da comunicação serial.

A inicialização e finalização da comunicação serial são feitas fora do comando de repetição *"while"*. Dentro do laço, é feito tanto a leitura da *"string"* e a conversão para números (caixa 1), como a concatenação dos valores de atuação em uma única *"string"* (caixa 2) para enviar ao Arduino Mega.

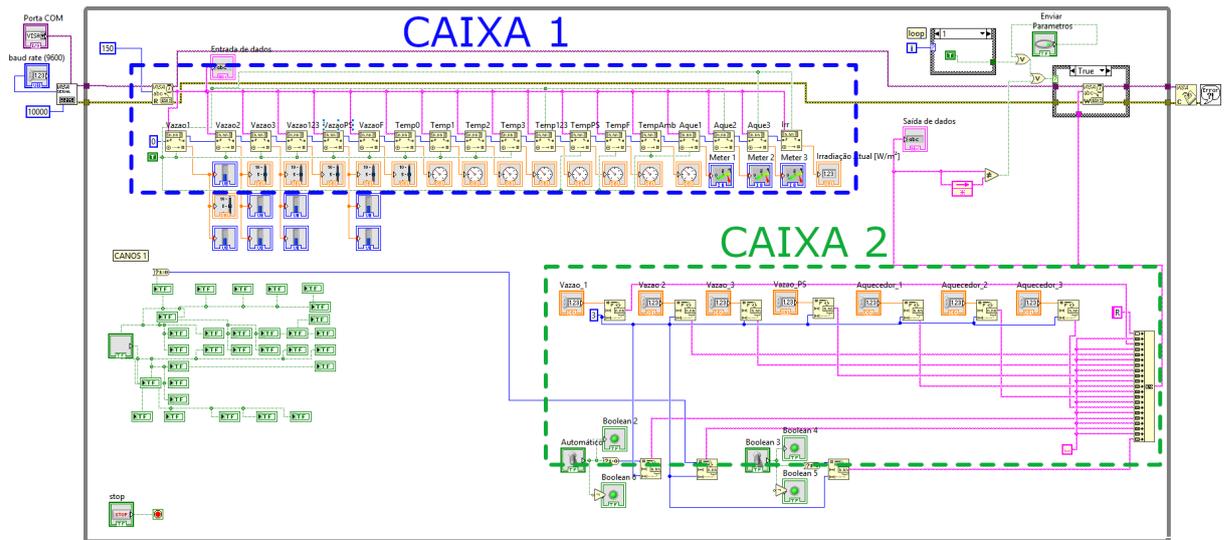


Figura 6 – Diagrama de blocos do sistema supervisório

2.2 Conclusões

Como visto nesse capítulo, o sistema hidráulico, elétrico e a instrumentação já estavam todos prontos e funcionando corretamente. Sendo assim, para fazer a emulação dos SCAs bastaria estudar a estratégia de emulação e fazer a identificação do sistema. Porém para fazer a aquisição dos sinais e posteriormente aplicar a estratégia de emulação foi necessário fazer adaptações no sistema que são demonstradas no capítulo 3.

3 Adaptações realizadas no sistema

Como demonstrado no capítulo 2, a planta piloto foi recentemente construída e por esse motivo o sistema implementado em LabView era simples a ponto de apenas mostrar informações dos sensores lidos e manipulação das variáveis de atuação nos aquecedores e nas vazões.

Para fazer a aquisição dos dados necessários à identificação do sistema foram desenvolvidas mudanças no LabView a fim de garantir a correta obtenção dos mesmos. Além disso, foi implementado o sistema de segurança para não superaquecer os ramos dos aquecedores.

3.1 Modificações para aquisição de dados

3.1.1 O problema

Para se aplicar as teorias de controle em tempo discreto é necessário que haja garantia do período de amostragem. Dessa forma o projeto de controle pode ser considerado confiável em termos de desempenho. Portanto, para resolver a necessidade da troca de dados de forma consistente, o problema foi dividido em partes menores: *timer*, leitura e gravação de dados. As figuras a seguir são partes do algoritmo implementado no software LabView, os números indicam a área da estrutura de condição. O numero 1 refere-se ao "*while*" principal (que faz o programa executar continuamente), os outros números indicam a estrutura "*case*" ou "*if*".

3.1.2 Timer

Dado que a comunicação com o Arduíno é pela porta serial, então ela não é atualizada em janela de tempo constante. Tanto na identificação do sistema quanto no controle, é necessário que o tempo de amostragem seja sempre igual. Para que o tempo de aquisição de cada amostra seja constante isso foi criado um *Timer* que gera uma interrupção sempre que atinge o tempo definido (figura 7).

Esse algoritmo usa o VI¹ *High Resolution Relative Seconds* como base de informação de tempo. De acordo com o manual [7], esse VI retorna o tempo relativo em segundos. A resolução de tempo pode variar conforme o sistema e o processador. Todo o algoritmo é implementado internamente à estrutura 1.

¹ VI em inglês: *Virtual Instrument*

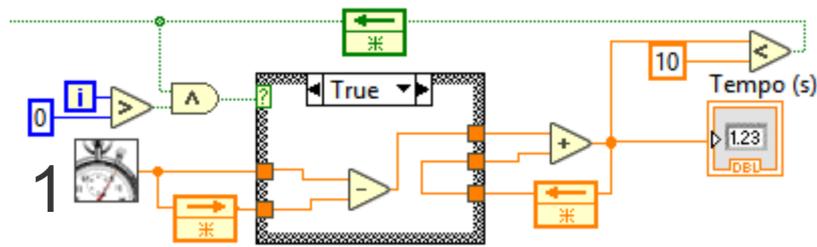


Figura 7 – Timer

O algoritmo basicamente incrementa as janelas de tempo decorridas ao longo de cada laço da estrutura 1. Esse valor incrementado é constantemente comparado com o tempo de amostragem escolhido (na figura 7 o valor é 10s). Quando o valor se torna maior, gera-se a interrupção que ocasiona a mudança da estrutura "case" para o estado falso (Figura 8)

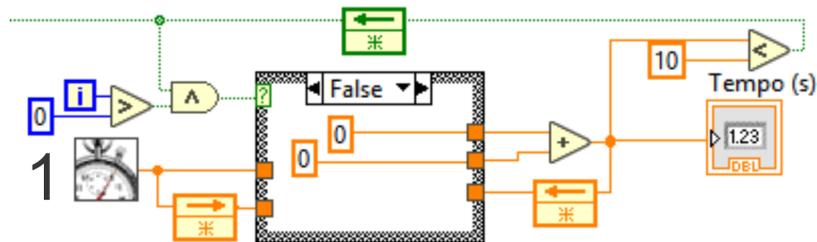


Figura 8 – Timer no estado de interrupção

Como se pode observar na figura 8, ao se executar a interrupção o tempo é zerado e passa a ser incrementado novamente. O mesmo sinal lógico que zera o timer é usado para fazer a aquisição dos dados com a amostragem escolhida.

Com a finalidade de verificar o funcionamento desse algoritmo, foi realizado um teste com duração de 10 minutos e período de amostragem de 5 segundos.

O teste consiste em ler e mostrar na tela um elemento de um vetor de 120 amostras de valor um, após essas amostras os valores do vetor são todos zero. Como o período de amostragem escolhido foi de 5 segundos, $120 \times 5 = 600s = 10min$. Ao iniciar o teste, iniciou-se o cronômetro no celular. A mudança do valor um para zero ocorreu no tempo exato de 10 minutos do cronômetro do celular.

Analisando o resultado acima, é possível concluir que não há diferença temporal aparente em 10 minutos de ensaio.

3.1.3 Leitura de dados

Para a identificação do sistema heliotérmico é necessário atuar na vazão e guardar os dados coletados. Então foi criado um algoritmo para leitura do sinal de entrada de um arquivo externo ao LabView ("entrada.txt") para atuar na vazão e um algoritmo para guardar os dados adquiridos em outro arquivo externo ("saida.txt").

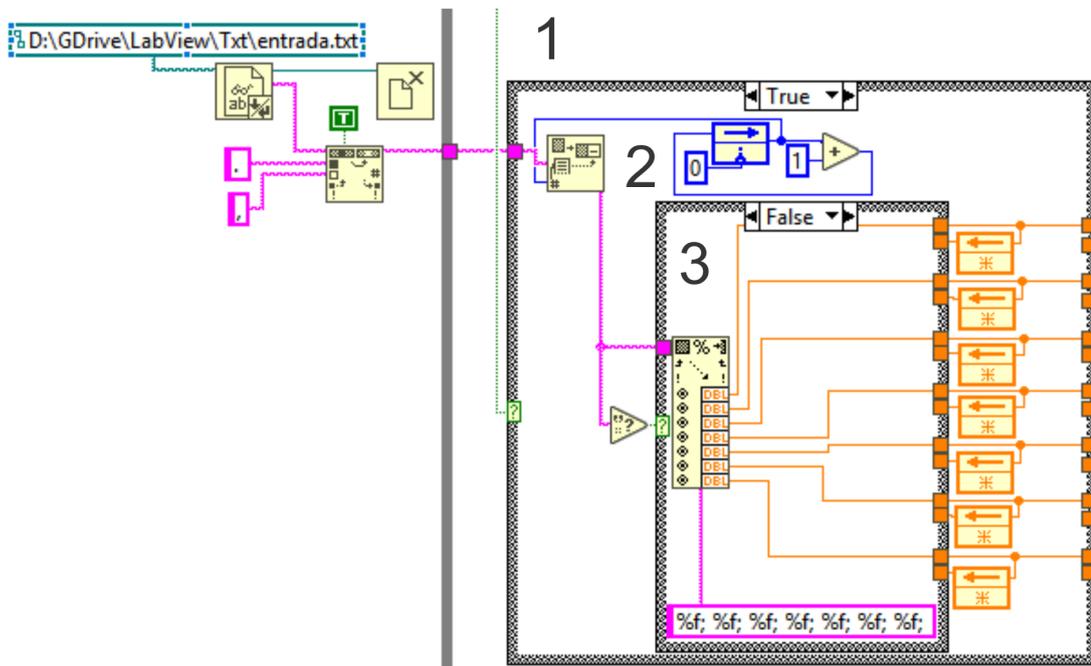


Figura 9 – Leitura dos sinais de entrada obtidos de um arquivo de extensão .txt

O código mostrado na figura 9 se comporta da seguinte forma: fora da estrutura 1 é feita a leitura completa da entrada.txt, então os pontos das casas decimais são convertidos em vírgula². Toda a informação do arquivo é enviada para dentro da estrutura 2 que só é acionada quando há interrupção do *timer* (seção 3.1.2). Toda vez que essa estrutura 2 ocorre (indicada na figura 9), é incrementado uma unidade no contador (blocos em azul da figura 9). Esse contador define a linha de dados que deve ser usada para conversão em dados do tipo *float*. Esses dados saem da estrutura 2 e seguem para o acionamento. A estrutura 3 serve apenas para não dar erro de execução quando o contador excede o tamanho do vetor de tempo do sinal lido no entrada.txt.

Para validar a leitura dos dados foi criado um indicador de *string* no LabView que mostra a *string* lida. Essa *string* é a leitura da linha apontada pelo contador que vinha do arquivo entrada.txt. A mudança dos dados na *string* observada se comportou da maneira correta (apenas quando há interrupção do timer), portanto foi concluído que o algoritmo funcionou corretamente.

² Essa conversão é necessária dependendo da instalação do LabView

3.1.4 Gravação de dados

No caso da gravação de dados ocorre algo semelhante e até mais simples. Já que a estrutura 4 também só é acionada na interrupção do *timer* (figura 10), dentro dela basta reunir todos os dados lidos dos sensores e as variáveis de atuação e concatena-los em uma única *string*, que será gravada em uma linha do arquivo "saida.txt".

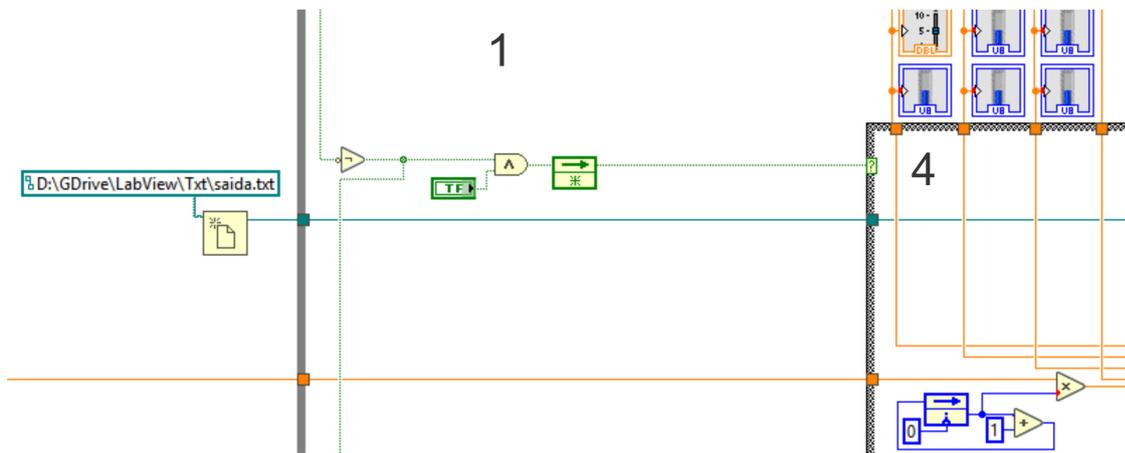


Figura 10 – Inicialização da saída.txt e acionamento da estrutura que guarda os dados

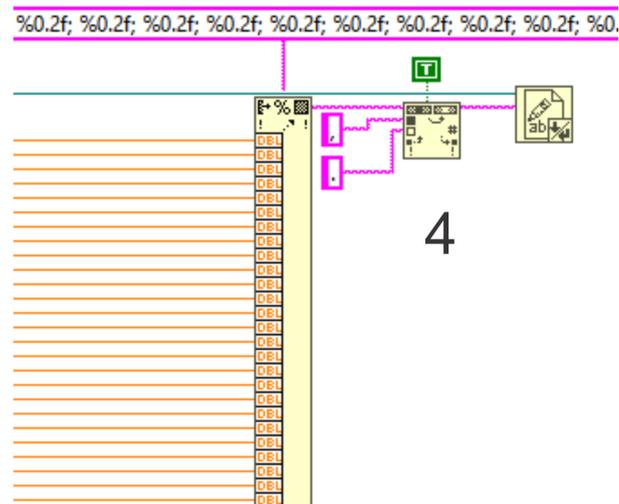


Figura 11 – Função que converte número em string

Existe um contador igual ao da figura 9 mas sua função é criar o vetor de tempo ao se multiplicar o tempo de amostragem pelo valor incrementado. A figura 11 mostra a função que converte os dados em string e envia para a função que guarda os dados em "saida.txt".

Para validar a gravação de dados, foi observado se o arquivo saída.txt continha os dados previamente conhecidos em ordem.

3.1.5 Acionamento dos ramos em malha aberta

Para se ter mais liberdade na operação do sistema, foram criadas chaves que alternam entre malha aberta e malha fechada os sinais de atuação da planta. Ou seja, se estiver em malha aberta, o operador define em tempo real qual é o valor da variável manipulada. Se for em malha fechada, é um sinal que vem da "entrada.txt" ou da malha fechada de controle mesmo. Então cada uma das 4 variáveis para manipulação das vazões e das 3 variáveis para manipulação dos aquecedores possui uma chave exclusiva.

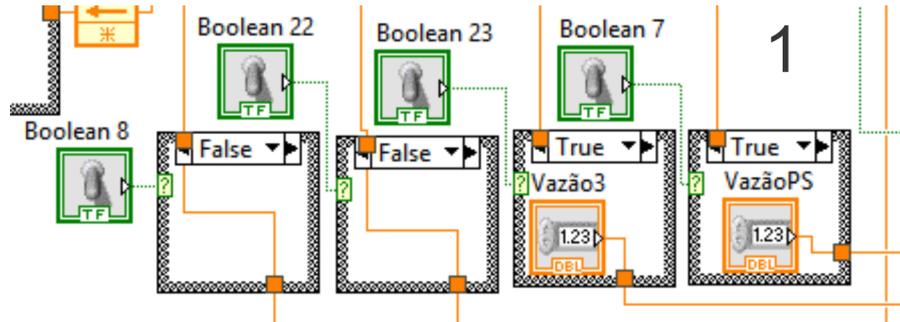


Figura 12 – Implementação das chaves para as vazões

Como visto na figura 12, se verdadeiro, a variável é definida em tempo real pelo operador do sistema, se falso, a informação vem de outra fonte, que nesse exemplo é da leitura da "entrada.txt". O mesmo ocorre com os aquecedores na figura 13.

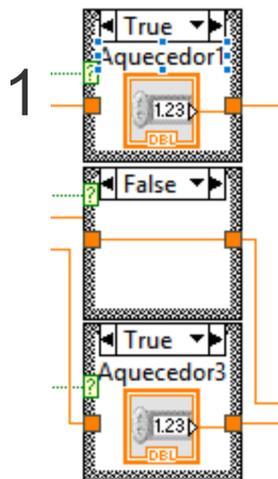


Figura 13 – Implementação das chaves para os aquecedores

Para possíveis alterações nas variáveis de atuação enquanto o sistema esta sendo executado, foram criadas chaves que alternam o estado de cada ramo: se o ramo está em malha aberta ou malha fechada.

Se estiver em malha fechada, o sinal de vazão é provido pelo sistema de controle, se for em malha aberta, o sinal é dado manualmente pelo operador.

3.2 Sistema de Segurança

O sistema de segurança implementado é relacionado ao aquecimento dos campos emulados quando há baixa ou nenhuma vazão. Ao se ligar o circuito de alimentação da bomba, alimenta-se também os aquecedores. Porém o fato de a bomba estar alimentada não garante que ela esteja ligada. Além disso, mesmo estando ligada, um ramo pode simplesmente estar com a sua válvula totalmente fechada.

A baixa vazão pode levar o sistema ao superaquecimento local, resultando em deterioração dos componentes. Portanto foram desenvolvidas algumas estratégias para evitar esse efeito indesejado.

3.2.1 Primeira estratégia

Enviar valores diferentes de zero no acionamento dos aquecedores, apenas quando a bomba estiver ligada. Caso esteja desligada, o LabView envia zero para todos os aquecedores.

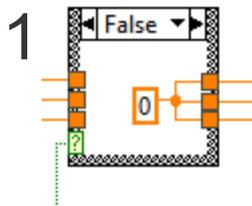


Figura 14 – Condição para a bomba desligada

A figura 14 mostra o estado no qual se envia informação zero para todos os aquecedores. Isso acontece quando o sinal lógico vindo do acionamento da bomba (figura 15) é falso. Quando é verdadeiro, mantém-se os sinais de potência.

A figura 15 mostra a lógica de acionamento da bomba: a estrutura 5 representa o estado da bomba, se está ligada ou desligada. A estrutura 6 representa a função que o botão ON/OFF desempenha dependendo do estado da bomba. O sinal lógico que faz a mudança de estado da bomba é o mesmo que impõe valor zero aos aquecedores.

Essa estratégia foi validada ao verificar a saída serial do LabView: independentemente do valor atribuído aos aquecedores, a saída serial manteve-se em zero com a bomba desligada.

3.2.2 Segunda estratégia

Além desse sistema, existe uma **verificação no Arduino Mega que desliga todos os aquecedores caso ocorra o sinal para desligar a bomba**, como mostrado no código a seguir:

3.2.3 Terceira estratégia

Desligar os aquecedores caso a vazão esteja abaixo de uma vazão determinada de segurança ou a temperatura do ramo esteja acima de uma temperatura de segurança. Essa estratégia pode comprometer a correta emulação dos campos solares, porém essa situação é preferível à destruição dos componentes do sistema. Abaixo o exemplo da implementação dessa estratégia para o ramo 1.

```
1   if(vazao_1 ≤ vminima){
2       digitalWrite(10,0);
3       Alerta_Temp_1 = 0;
4   } else if (temp_1 ≥ temp_seguranca) {
5       digitalWrite(10,0);
6       Alerta_Temp_1 = 1;
7   } else {
8       digitalWrite(10,aquec_1_int); // Aquecedor 1
9       Alerta_Temp_1 = 0;
10  }
```

O procedimento para cálculo da vazão mínima de segurança está descrito no apêndice e foi definida em $1,66L/min$. Já a temperatura máxima de segurança é baseada na temperatura máxima de trabalho dos componentes e é de $65^{\circ}C$.

Paralelo a isso, foi criado um sinal de alerta para indicar no LabView se uma dada temperatura excedeu a temperatura de segurança. No caso do exemplo é a variável "Alerta_Temp_1"³.

Todas estas estratégias de segurança foram validadas experimentalmente na planta piloto : ao se colocar uma vazão menor que a mínima e ao se desligar a bomba.

3.3 Conclusões

Todas as adaptações descritas nesse capítulo foram validadas ao se observar o correto funcionamento durante a utilização das mesmas. Portanto as funções cumprem o papel proposto. Desta forma a planta está em condições de operar para fazer a aquisição de dados. O próximo capítulo dedica-se à identificação do sistema.

³ Um sistema parecido foi criado pelo graduando Henrique Roos para alertar possíveis ramos entupidos.

4 Identificação do sistema

Como será visto no capítulo 5, a emulação de um campo solar é feita simulando o comportamento de um *SCA*. A resposta dessa simulação funciona como referência para a temperatura real (na saída do aquecedor). Para simular, é necessário obter um modelo do *SCA* e para fazer o controle de temperatura local é necessário obter um modelo do sistema de aquecimento elétrico. Então, este capítulo destina-se a estudar esses dois modelos e fazer a identificação dos sistemas para que a emulação seja bem sucedida.

4.1 Modelo do aquecimento elétrico

O modelo do sistema de aquecimento elétrico é apresentado na equação 4.1, onde $T_i(t)$ é a temperatura de entrada da água, $T(t)$ é a temperatura de saída, $F(t)$ é a vazão e $P(t)$ o acionamento de uma potência elétrica. V é o volume interno do aquecedor, ρ é a massa específica da água e c_p é o calor específico.

$$V \frac{dT(t)}{dt} = F(t)[T_i(t) - T(t)] + \frac{P(t)}{\rho c_p} \quad (4.1)$$

Como a identificação dos parâmetros desse sistema será feita a partir de dados amostrados, é necessário o uso de um modelo discretizado (equação 4.2).

$$V \frac{T(k) - T(k-1)}{T_{SI}} = F(k-1)[T_i(k-1) - T(k-1)] + \frac{P(k-1)}{\rho c_p} \quad (4.2)$$

Onde k é o número da amostra e T_{SI} é o período de amostragem do sistema local (ou malha interna).

4.2 Modelo do SCA

O modelo do campo usado para identificação neste trabalho tem sido aplicado com bons resultados para concentradores parabólicos e captadores planos [8]. Portanto para os testes iniciais de identificação foi usado o seguinte modelo da equação 4.3 (onde T_{SE} é o período de amostragem do sistema discreto) com as informações nas tabelas 4 e 5.

$$T_{oF}(k) = T_{oF}(k-1) + \frac{\beta_I T_{SE}}{\rho C_p A_a} I(k-1) - \frac{HT_{SE}}{L_{eq} \rho C_p A_a} (\bar{T}(k-1) - T_a(k-1)) - \frac{T_{SE}}{\rho A_a \eta_{eq}} \dot{m}_F(k-1) \frac{T_{oF}(k-1) - T_{iF}(k-1)}{L_{eq}} \quad (4.3)$$

Tabela 4 – Parâmetros do processo

Parâmetro	Descrição
A_a	Seção reta do absorvedor
C_p	Calor específico do fluido
H	Coefficiente de perdas térmicas
L_{eq}	Comprimento equivalente do tubo absorvedor
β_I	Irradiância do modelo
ρ	Densidade do fluido
η_{eq}	Parâmetro do processo

O sistema a ser identificado nesse trabalho não utiliza sensores de vazão mássica, mas sensores de vazão volumétrica (tabela 2) com a unidade de medida em L/min (tabela 3). Porém, considerando que a massa específica da água se altera pouco entre 0 e 100 °C, pode-se assumir que a leitura do sensor em litros equivale a quilogramas de água, já que um litro desse fluido pesa aproximadamente um quilograma. Então a vazão mássica fica de acordo com a equação 4.4.

$$\dot{m}_F = \frac{dm}{dt} \left[\frac{kg}{min} \right] \quad (4.4)$$

Tabela 5 – Variáveis do processo

Var.	Descrição	Unidade
T_{oF}	Temperatura de saída do campo	°C
T_{iF}	Temperatura de entrada do campo	°C
\bar{T}	Temperatura média entre T_{oF} e T_{iF}	°C
T_a	Temperatura Ambiente	°C
\dot{m}_F	Vazão mássica no campo	kg/min
I	Irradiação solar	W/m^2

Como não existe informação sobre a posição da válvula para mudança de vazão, existe um controle proporcional¹ (seção 2.1.3), de ganho $Kp = 10$, para injetar a vazão de referência no sistema. A figura 16 mostra um diagrama de blocos com o controle local de vazão dentro da caixa tracejada em azul e os sinais usados para identificar o sistema SCA em vermelho pontilhado.

4.3 Mínimos quadrados

A origem da ideia básica sobre mínimos quadrados pode ser encontrada nos trabalhos de Gauss sobre estudos astronômicos. Em seus estudos, menciona alguns aspectos básicos à aplicação do método dos mínimos quadrados. Como por exemplo: a necessidade de se

¹ Variando a referência de vazão (V_{ref}) entre 3 e 6L/min observou-se que $0,96 \leq V/V_{ref} \leq 1,04$, portanto esse controlador proporcional não foi alterado.

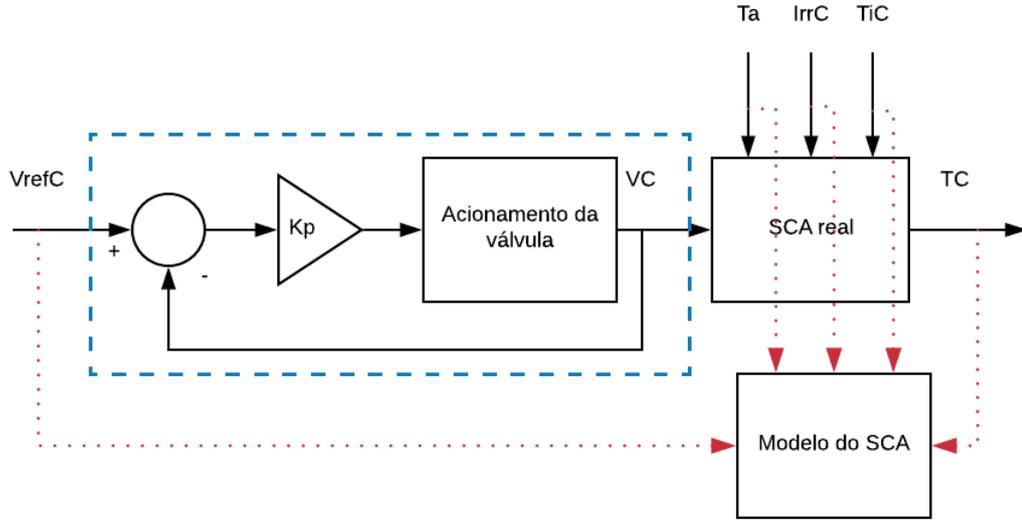


Figura 16 – Diagrama de blocos para identificação do SCA real.

ter no mínimo, um número de observações igual ao número de parâmetros a se determinar. Por causa dos erros de medição, são necessárias mais observações do que o número de parâmetros, já que essas observações extras resultam na redução dos efeitos dos erros. Assim, a diferença entre observação e cálculo deve ser mínima ao se usar os parâmetros estimados.

Como demonstrado no livro [9], o método dos mínimos quadrados minimiza o erro médio quadrático ao encontrar os parâmetros da equação.

$$\hat{\Theta} = [X^T X]^{-1} X^T y \quad (4.5)$$

A equação 4.5 minimiza o erro para funções lineares nos parâmetros, onde $\hat{\Theta}$ é o vetor de parâmetros, X é a matriz de regressão composta pelos dados coletados e y é o vetor de observações da saída.

4.3.1 Aplicação ao modelo do aquecedor

Apesar de o modelo do aquecedor elétrico ser não-linear (equação 4.2), é possível reorganizar a equação para que seja linear nos parâmetros θ_1 e θ_2 (equação 4.7).

$$F(k)[T_i(k-1) - T(k-1)] = V \frac{T(k) - T(k-1)}{T_{SI}} - \frac{P(k-1)}{\rho c_p} \quad (4.6)$$

$$F(k-1)[T_i(k-1) - T(k-1)] = \theta_1[T(k) - T(k-1)] - \theta_2 P(k-1) \quad (4.7)$$

Onde:

$$\theta_1 = \frac{V}{T_{SI}}$$

$$\theta_2 = \frac{1}{\rho c_p}$$

$$X = \begin{bmatrix} T(2) - T(1) & -P(1) \\ T(3) - T(2) & -P(2) \\ \vdots & \vdots \\ T(n) - T(n-1) & -P(n-1) \end{bmatrix}$$

$$y = F(k-1)[T_i(k-1) - T(k-1)]$$

A partir da estimação de $\hat{\Theta}$ é possível calcular a saída usando os parâmetros estimados para comparar com os dados reais. Ao isolar $T(k)$ da equação 4.7 obtém-se a equação 4.8

$$T(k) = \frac{F(k-1)[T_i(k-1) - T(k-1)] - \theta_2 P(k-1)}{\theta_1} + T(k-1) \quad (4.8)$$

O parâmetro θ_1 do modelo discretizado depende do tempo de amostragem (T_{SI}). Dessa forma, a captura dos dados para identificação desse modelo deve ser feita com o mesmo período de amostragem que será utilizado para controle posteriormente. O T_{SI} foi definido em 1,5 segundos e a demonstração do procedimento para obtenção desse valor está no apêndice B.

4.3.2 Aplicação ao modelo do SCA

Assim como na subseção 4.3.1, o modelo do SCA (equação 4.3) é não linear, porém é linear nos parâmetros, então é possível usar a equação 4.5 para estimar os parâmetros. Como a equação 4.3 tem parâmetro fixo igual a um em $T_{oF}(k-1)$, esse parâmetro não deve ser estimado.

$$T_{oF}(k) - T_{oF}(k-1) = \frac{\beta_I T_{SE}}{\rho C_p A_a} I(k-1) - \frac{HT_{SE}}{L_{eq} \rho C_p A_a} (\bar{T}(k-1) - T_a(k-1)) - \frac{T_{SE}}{\rho A_a \eta_{eq}} \dot{m}_F(k-1) \frac{T_{oF}(k-1) - T_{iF}(k-1)}{L_{eq}} \quad (4.9)$$

$$T_{oF}(k) - T_{oF}(k-1) = \theta_1 I(k-1) + \theta_2 (\bar{T}(k-1) - T_a(k-1)) + \theta_3 \dot{m}_F(k-1) (T_{oF}(k-1) - T_{iF}(k-1)) \quad (4.10)$$

Onde:

$$\theta_1 = \frac{\beta_I T_{SE}}{\rho C_p A_a}$$

$$\theta_2 = \frac{HT_{SE}}{L_{eq}\rho C_p A_a}$$

$$\theta_3 = \frac{T_{SE}}{\rho A_a \eta_{eq} L_{eq}}$$

$$X = \begin{bmatrix} I(1) & -(\bar{T}(1) - T_a(1)) & -\dot{m}_F(1)(T_{oF}(1) - T_{iF}(1)) \\ I(2) & -(\bar{T}(2) - T_a(2)) & -\dot{m}_F(2)(T_{oF}(2) - T_{iF}(2)) \\ \vdots & \vdots & \vdots \\ I(n) & -(\bar{T}(n) - T_a(n)) & -\dot{m}_F(n)(T_{oF}(n) - T_{iF}(n)) \end{bmatrix}$$

$$y = T_{oF}(k) - T_{oF}(k - 1)$$

Com a equação 4.11 para calcular a saída usando os parâmetros estimados.

$$T_{oF}(k) = X\hat{\Theta} + T_{oF}(k - 1) \quad (4.11)$$

4.4 Sinal PRS

Para que o ocorra a correta captura do comportamento de um sistema, idealmente ele deve ser excitado em toda a faixa de frequência. Porém, na prática é impossível implementar um sinal com a faixa espectral infinita. Sinais PRBS(Pseudo-Random Binary Sequence) são bons candidatos à excitação do sistema por possuírem uma grande faixa espectral. São sinais periódicos e a mudança entre os valores é de forma determinística, dessa forma o mesmo sinal pode ser gerado repetidas vezes [9].

Como tanto o sistema de aquecimento elétrico como os SCAs são não lineares, usar um sinal PRBS não basta para determinar os parâmetros corretamente, já que o fato de ser binário implica em mudança de amplitude sempre entre os mesmo valores (+Ve - V). Então o correto é implementar um sinal PRS (Pseudo-Random Sequence), assim, não só a faixa de frequência varia, mas também a faixa de amplitude é aleatoriamente variável.

4.4.1 Implementação

Para implementar o sinal PRS foi usado como base a geração do sinal PRBS disponível no livro [9]. O código modificado é apresentado no apêndice E deste documento.

Observando o apêndice E, a partir da linha 37 começa a implementação da variação de amplitude. O código determina que toda vez que ocorrer uma mudança de sinal, a amplitude é gerada aleatoriamente segundo uma distribuição normal, com media 0,5 e desvio padrão de 0,25. Com essa distribuição, o sinal gerado fica compreendido entre 0 e 1, em 95% das vezes. Esse valor normalizado então é convertido para o valor de interesse que

fica entre o valor mínimo e o valor máximos escolhidos (linha 39). A figura 17 mostra um exemplo do sinal.

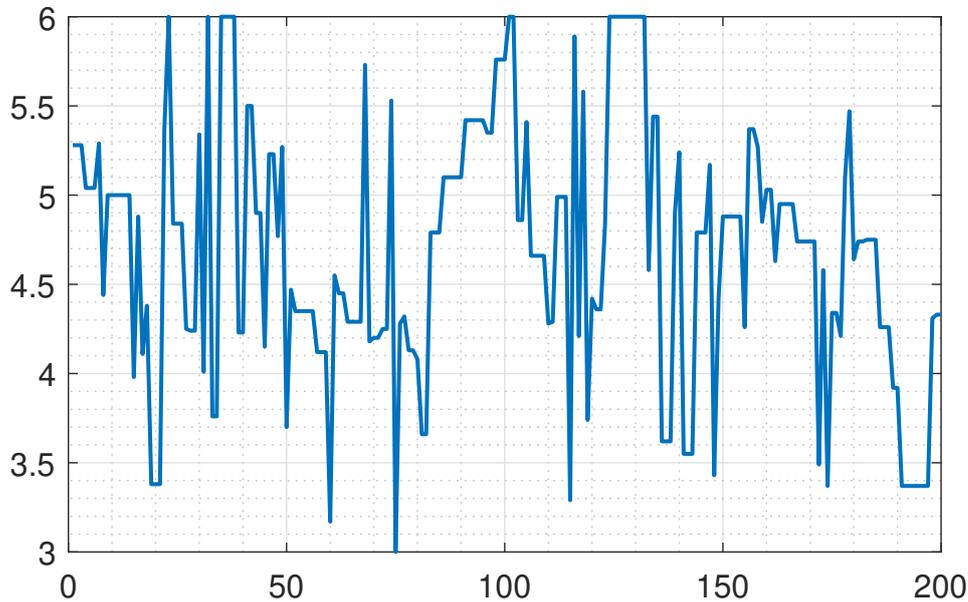


Figura 17 – Exemplo de sinal PRS

4.5 Identificação dos aquecedores elétricos

Foi constatado que o sensor de temperatura em cada ramo de emulação possui uma distância considerável em relação ao aquecedor elétrico. Sabendo que a dinâmica dessas malhas devem ser rápidas na emulação e que a distância pode ser considerável, isso pode gerar atraso de transporte significativo. Se o atraso for significativo ele deve ser modelado e levado em conta no projeto de controle. Além disso, por haver vazão variável, o atraso de transporte também acaba sendo variável. Para descobrir se há atraso de transporte e se sua variação é grande foi aplicado os sinais da figura 18.

O sinal de vazão varia dentro da faixa de operação de 3 e 6L/min, já a potência² é uma onda quadrada com amplitude 100³. O tempo em que ocorre o nível alto da onda quadrada é justamente o centro da duração de cada degrau de vazão. O resultado da aplicação desses sinais para o aquecedor 2 é encontrado na figura 19.

Como se pode ver na figura 19, apareceu um novo problema na leitura dos sensores de temperatura⁴. Então, para melhor visualizar o gráfico, foram eliminadas as leituras

² Como visto na equação 2.1 da seção 2.1.2, o sinal enviado pelo LabView tem unidade 2900W/255, portanto em todas as figuras onde aparecer a potência, leia-se com essa unidade apresentada.

³ Na figura 18 foi usado amplitude 1 apenas para melhor visualização

⁴ Acredita-se que este novo problema possa ter origem no hardware, uma vez que os problemas descritos no Apêndice C já foram solucionados. Por não haver tempo hábil até a data final de entrega deste PFC, os ensaios de identificação foram realizados na presença destes erros e posteriormente processando os dados experimentais.

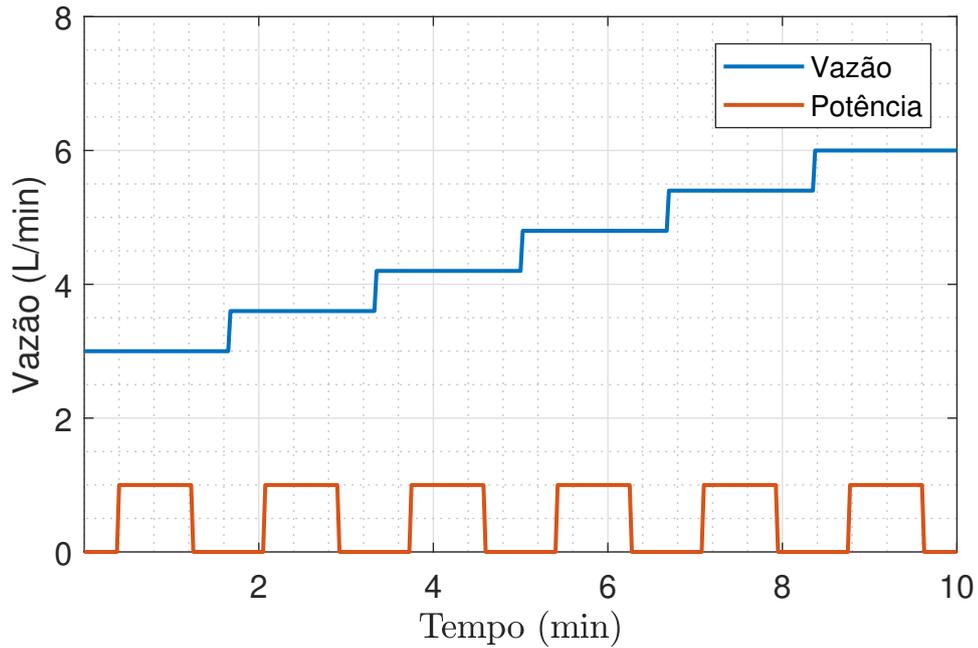


Figura 18 – Sinais para identificar o atraso de transporte

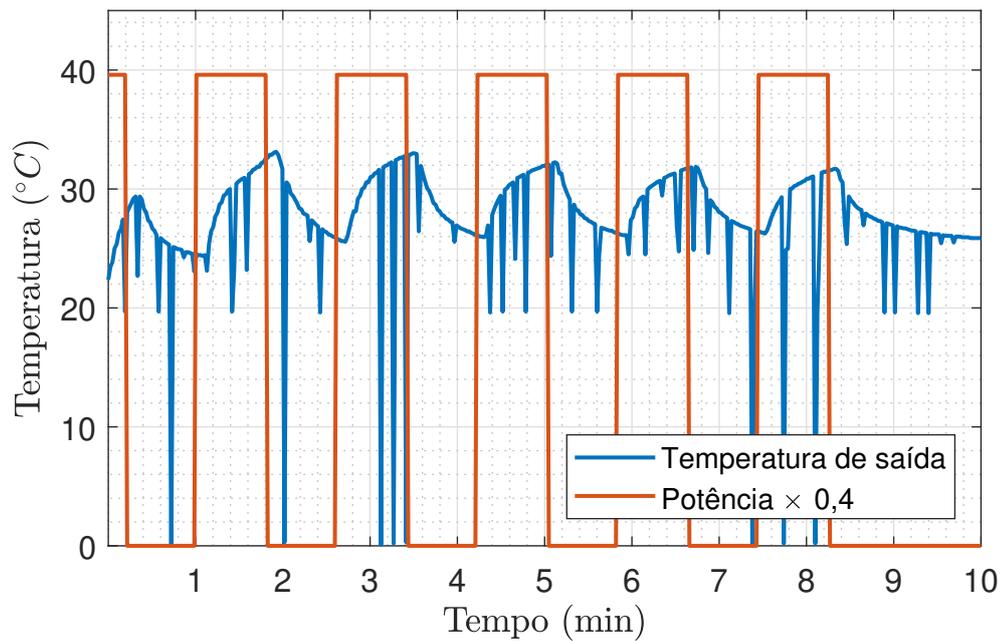


Figura 19 – Sinal da potência de entrada e temperatura de saída do aquecedor

ruins, substituindo-as por pontos de interpolação entre valores corretos (figura 20). Para a identificação dos aquecedores 1 e 3 aconteceram outros dois problemas: a vazão do ramo de aquecimento 1 satura antes de chegar ao valor de 6L/min, já o aquecedor 3 não está aquecendo. Por esses motivos a identificação e emulação será feita apenas para o ramo de aquecimento 2.

Observa-se na figura 19 que existe um atraso considerável entre o acionamento do aquecedor e a resposta da temperatura (considerando que a escala de tempo do

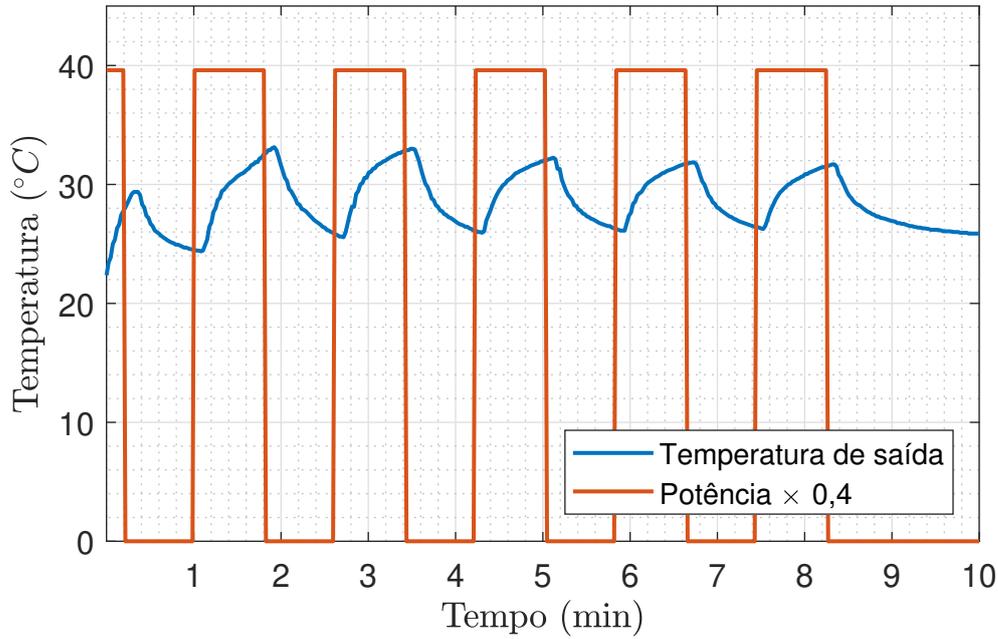


Figura 20 – Sinal da potência de entrada e temperatura de saída do aquecedor

gráfico está em minutos). Ao se mudar a escala para segundos e aproximar o gráfico nas regiões do atraso, percebe-se que para a vazão de 3 L/min, o atraso de transporte dura aproximadamente 7,5 segundos, já para 6 L/min o atraso de transporte dura em torno de 6 segundos, o que dá uma diferença de uma amostra ($T_{SI} = 1,5s$) entre o atraso com maior vazão e o atraso com menor vazão. Dessa forma, considera-se que o atraso não varia e que dura 6 segundos, ou 4 amostras. Com o atraso $d = 4$ então muda-se o modelo para a identificação do sistema de acordo com a equação 4.12.

$$F(k - 1 - d)[T_i(k - 1 - d) - T(k - 1)] = \theta_1[T(k) - T(k - 1)] - \theta_2 P(k - 1 - d) \quad (4.12)$$

Onde 'd' é o número de amostras de atraso. Além do atraso 'd' na saída, o sensor de temperatura da entrada é consideravelmente longe do local de aquecimento. Porém, como não se tem controle sobre a temperatura de entrada da água, não é possível fazer um ensaio para verificar o quanto adiantado está o sinal da temperatura de entrada. Para contornar esse problema, mediu-se a distância do sensor de temperatura da saída em relação ao aquecedor (30cm) e a distância do sensor da entrada ao aquecedor (25cm) e fez-se uma regra de 3 utilizando o atraso de 6 segundos para a temperatura de saída (equação 4.13).

$$tempoA = \frac{25 * 6}{30} = 5 \text{ segundos} \approx 3 \text{ amostras} \quad (4.13)$$

Então, como a medição acontece antes de chegar ao local de aquecimento, esse tempo é adiantado. Converte-se o valor de 5 segundos para 3 amostras. Então, considerando

esse adiantamento o modelo fica de acordo com a equação 4.14, onde 'a' é o número de amostras adiantadas.

$$F(k-1-d)[T_i(k-1-d+a) - T(k-1)] = \theta_1[T(k) - T(k-1)] - \theta_2 P(k-1-d) \quad (4.14)$$

Com o atraso de transporte já identificado, foi criado o sinal (figura 21) para identificação através de mínimos quadrados.

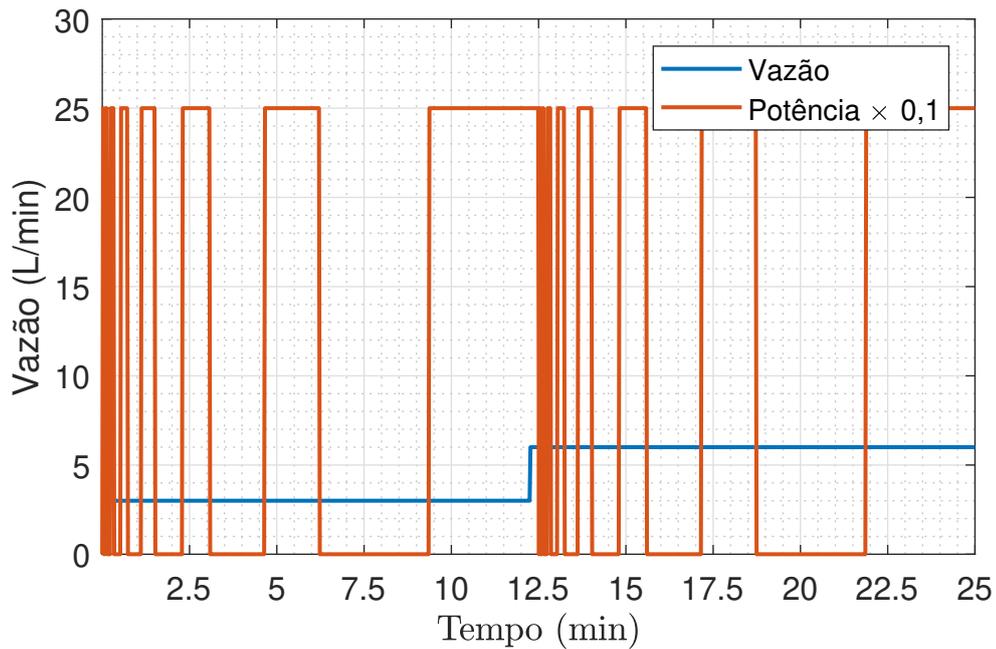


Figura 21 – Sinal da potência de entrada e vazão

Observando a figura 21, percebe-se que o sinal da vazão foi mudado apenas uma vez entre 3L/min e 6L/min. O ideal seria aplicar um sinal PRS (descrito na seção 4.4), porém o motor de passo está respondendo de forma lenta, e portanto não consegue acompanhar um sinal que pode variar com grande amplitude em espaço de tempo curto como o $T_{SI} = 1,5s$. Portanto, para que houvesse o mesmo peso na minimização do erro que relaciona o efeito potência dependendo da vazão, foi criado o sinal de potência repetido para vazão de 3L/min e 6L/min. O sinal de potência⁵ tem largura do pulso variada para excitar tanto as altas frequências quanto as baixas frequências do sistema. Não é necessário variar a amplitude da potência pois a relação com a temperatura de saída da água é linear. O resultado da aplicação desses sinais é visto na figura 22, já com a eliminação e interpolação das medições ruins.

A aplicação do método dos mínimos quadrados usando os dados colhidos não resultou em boa identificação do sistema. Para verificar se havia problemas no algoritmo de

⁵ O gráfico mostra a amplitude da potência em 25, porém a real amplitude aplicada é de 250, muito próximo da amplitude máxima (255)

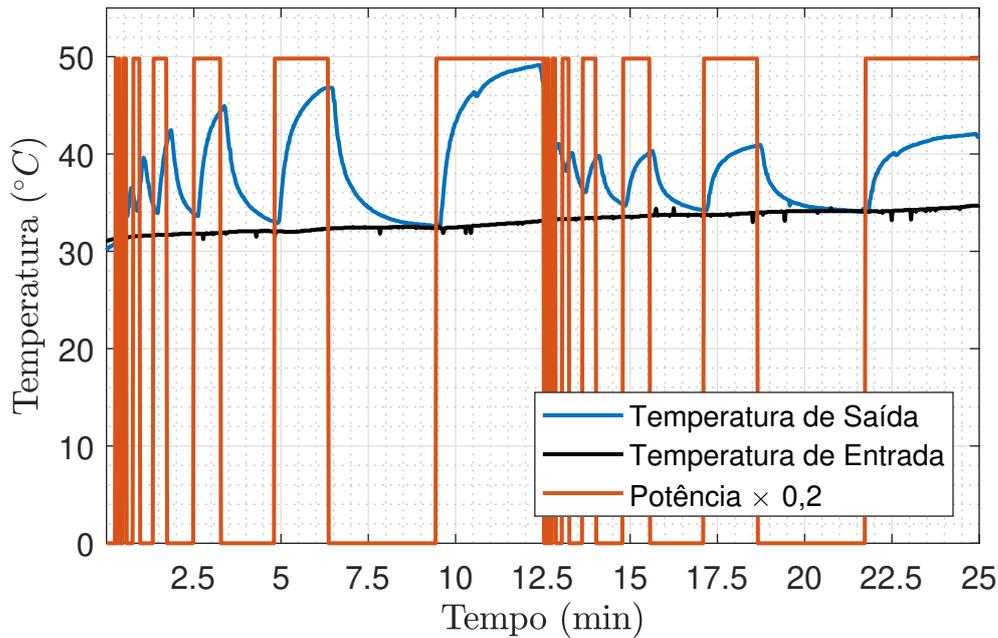


Figura 22 – Resposta do sistema à potência

identificação, foi criado um sistema simulado com o mesmo modelo do balanço de energia proposto usando parâmetros fictícios. Foi colhido os dados desse sistema simulado com o mesmo período de amostragem e mesmos sinais usados para excitar a planta piloto. Os parâmetros foram identificados corretamente, portanto o algoritmo para a identificação está correto e o problema pode ser outro como por exemplo, talvez ser pela baixa excitação do sinal de vazão ou pela quantidade de dados. Então, optou-se por fazer a identificação usando a resposta ao degrau, isso foi feito com o ultimo pulso do sinal da figura 22.

A figura 23 mostra a resposta do sistema no último pulso apresentado na figura 22. Para fazer a identificação de primeira ordem desse sistema foi necessário levar em consideração a temperatura de entrada do tipo rampa. Para demarcar a inclinação da rampa, usou-se a reta verde inferior. Com ela criou-se outra reta verde (superior) com a mesma inclinação para melhor percepção do regime permanente. Então para calcular o ΔT usou-se os pontos em que as retas verdes cruzam o eixo da temperatura.

$$\Delta T \approx 41,3 - 34 = 7,3^{\circ}C$$

Com o valor da variação de temperatura calcula-se o 95% para determinar o $t_{5\%}$. Então $7,3 \times 0,95 \approx 41$. Depois, posiciona-se uma terceira reta paralela (preta) às retas verdes na temperatura $41^{\circ}C$, o ponto vermelho superior indica a intersecção entre a reta preta e a temperatura e conseqüentemente o tempo em que ela ocorre. A diferença entre o tempo do ponto vermelho superior e o tempo do ponto vermelho inferior resulta no $t_{5\%}$

$$t_{5\%} = 120 - 10 = 110s$$

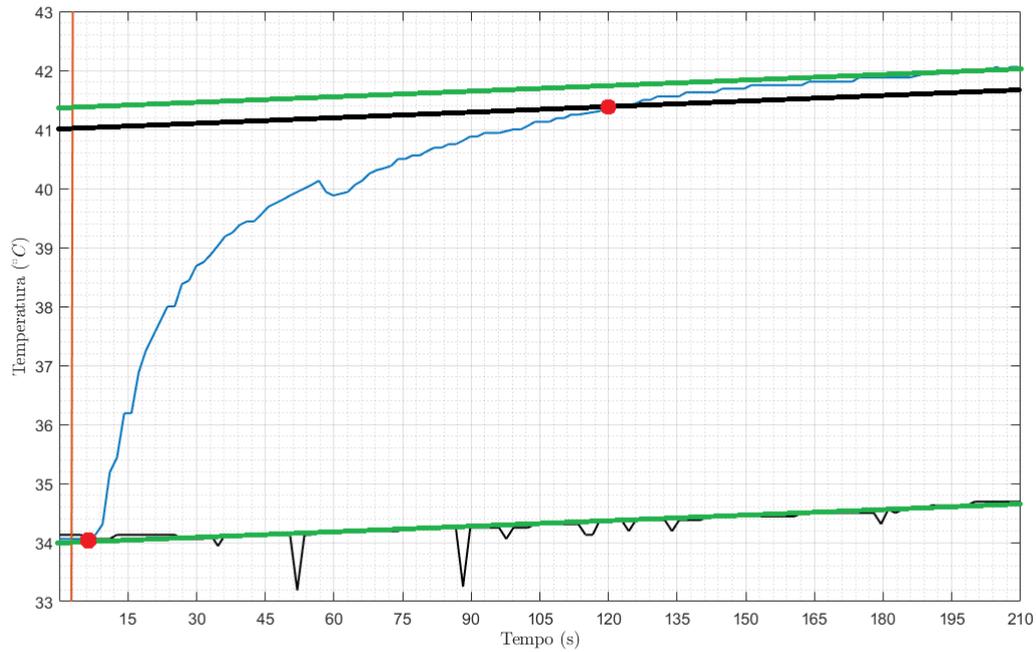


Figura 23 – Identificação por aproximação de primeira ordem

$$\tau = \frac{t_{5\%}}{3} \approx 37s = 0,617min$$

Como a vazão é constante em 6L/min durante o período analisado, a não linearidade do sistema deixa de existir e portanto é possível aplicar a transformada de Laplace na equação 4.1. Aplicando e reorganizando tem-se:

$$T(s) = \frac{T_i(s)}{\frac{V}{F}s + 1} + \frac{1}{F\rho c_p} \frac{P(s)}{\frac{V}{F}s + 1} \quad (4.15)$$

$$\tau = \frac{V}{F} = \frac{V}{6} = 0,617min \quad (4.16)$$

$$V = 3,7L \quad (4.17)$$

$$\theta_1 = \frac{V}{T_{SI}} = \frac{3,7}{0,025} = 148 \quad (4.18)$$

Para calcular θ_2 usa-se a equação 4.1 e sabendo que em regime permanente $\frac{dT(t)}{dt} = 0$ então:

$$0 = F[T_i(t) - T(t)] + \theta_2 P(t) \quad (4.19)$$

$$\theta_2 P(t) = F[T(t) - T_i(t)] = F\Delta T = 7,3F \quad (4.20)$$

$$\theta_2 = \frac{7,3F}{P} = \frac{7,3 \times 6}{250} = 0,1752 \quad (4.21)$$

4.5.1 Resultados

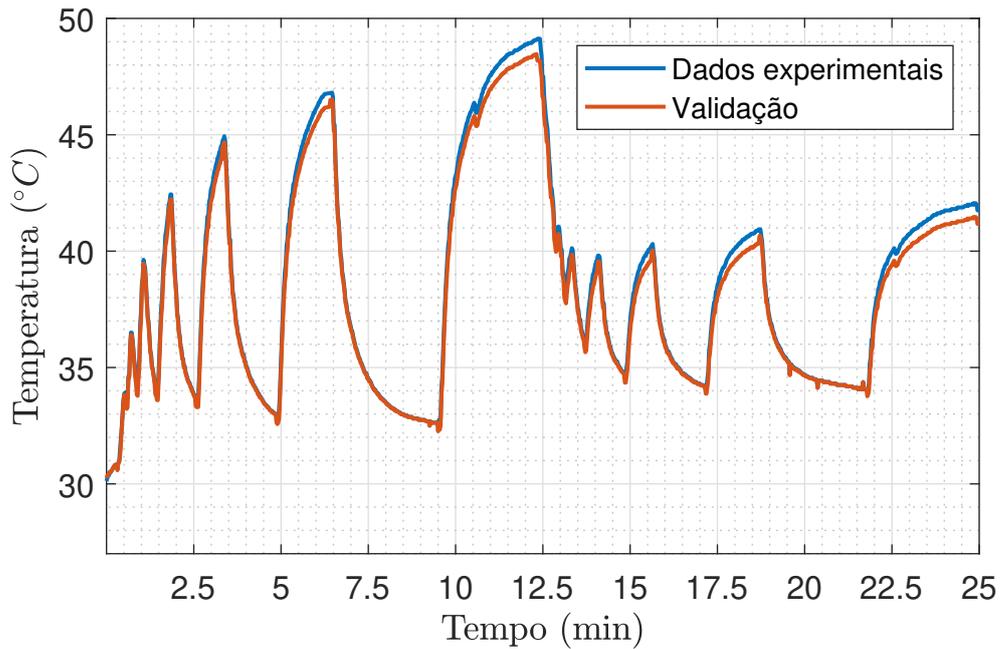


Figura 24 – Validação do sistema

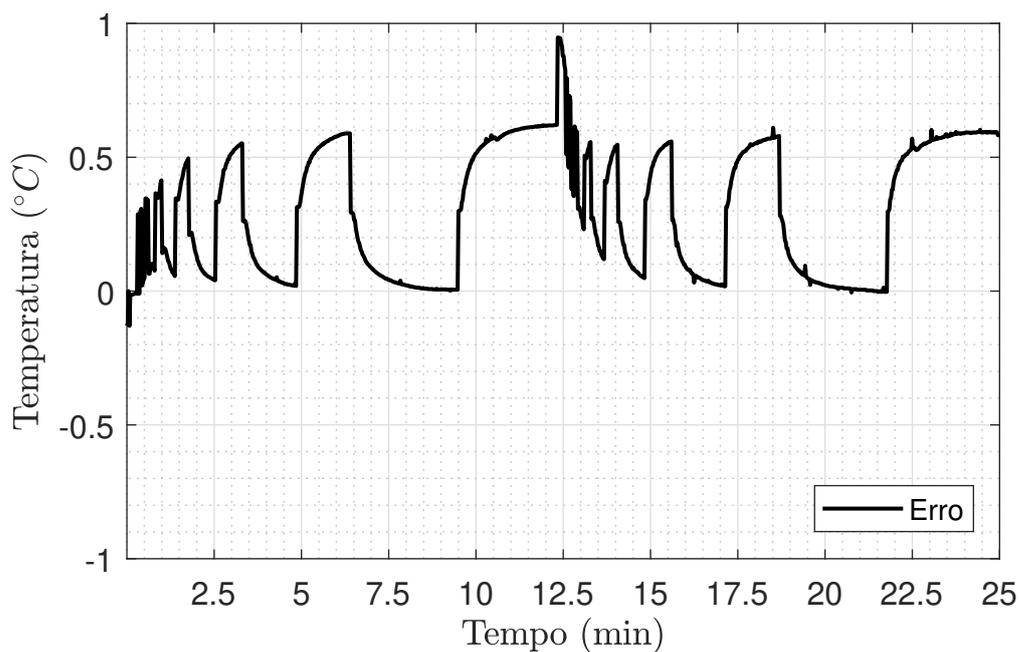


Figura 25 – Erro entre os dados experimentais e a saída do modelo

Comparando os resultados das figuras 24 e 25, observa-se que o erro em regime permanente é um pouco maior que 1%. Além disso, foi criado um sistema simulado com esses parâmetros e com sinais de entrada iguais aos aplicados na figura 21. O resultado da simulação mostrou uma resposta parecida aos dados coletados na figura 22. Então o modelo do sistema de aquecimento elétrico fica de acordo com a equação 4.22

$$T(k) = \frac{F(k-5)[T_i(k-2) - T(k-1)] - 0,1752P(k-5)}{148} + T(k-1) \quad (4.22)$$

4.6 Identificação do SCA

Foram feitos 2 ensaios em dias diferentes: um para identificar os parâmetros do sistema e outro para validar a identificação. Para uma melhor identificação, o ideal é que o ensaio durasse todo o período do dia, porém, por haver sombra no painel no início da manhã e no final da tarde, o período dos ensaios foi das 10:30h às 15:00h, aproximadamente. O período de amostragem dessa coleta de dados foi de $T_{SE} = 30s$. Sendo assim, o sinal PRS aplicado na vazão tem a largura do pulso com duração mínima de 30 segundos. Além disso esse sinal tem variação de amplitude entre 6L/min e 3L/min. As figuras 26, 27, 28 e 29 apresentam os dados coletados durante os dois ensaios realizados e na seção 4.6.1, os resultados da identificação do SCA.

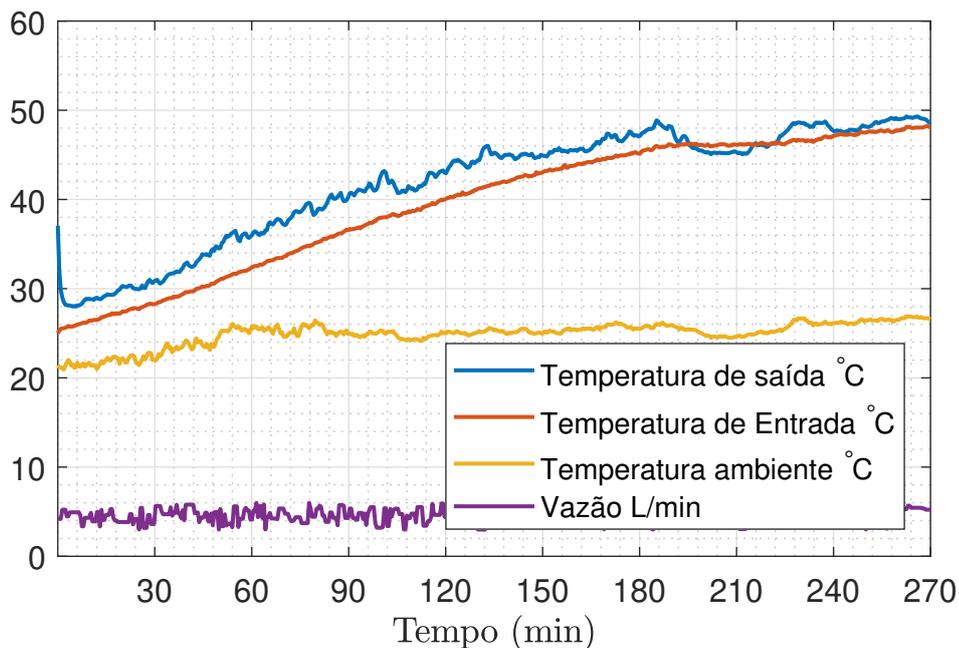


Figura 26 – Sinais obtidos no primeiro ensaio

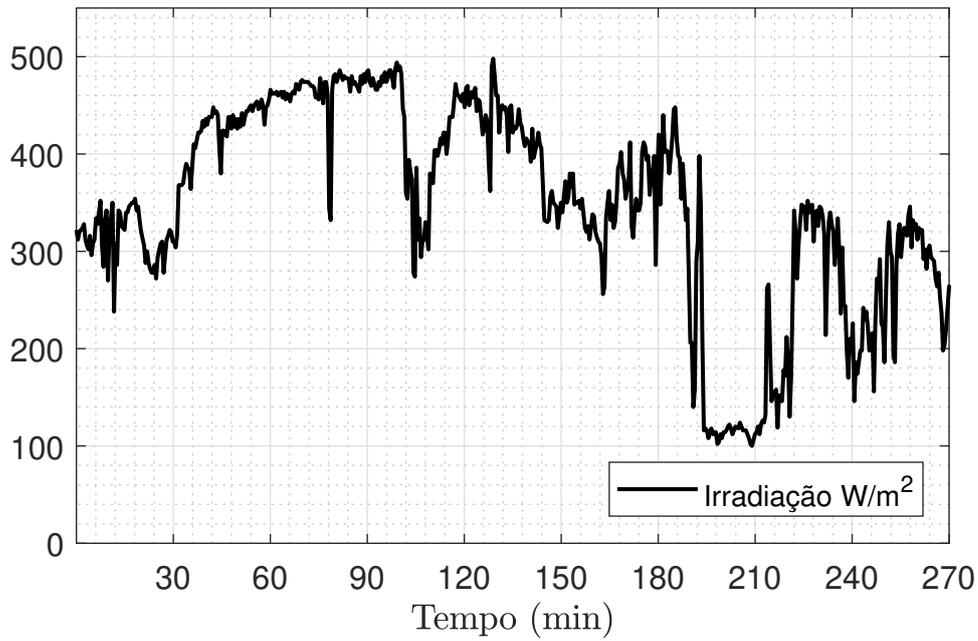


Figura 27 – Irradiação do primeiro ensaio

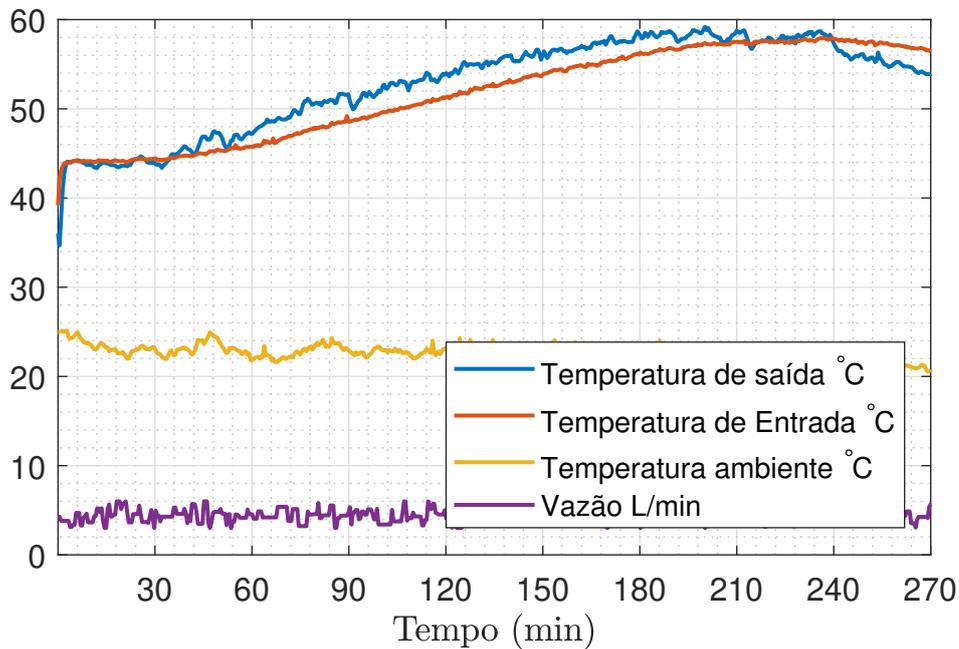


Figura 28 – Sinais obtidos no segundo ensaio

4.6.1 Resultados

Como foram feitos dois ensaios, foi feita uma estimaco para cada um dos ensaios. Para validar cada estimaco, usou-se o conjunto de dados no usado para estimar. A seguir

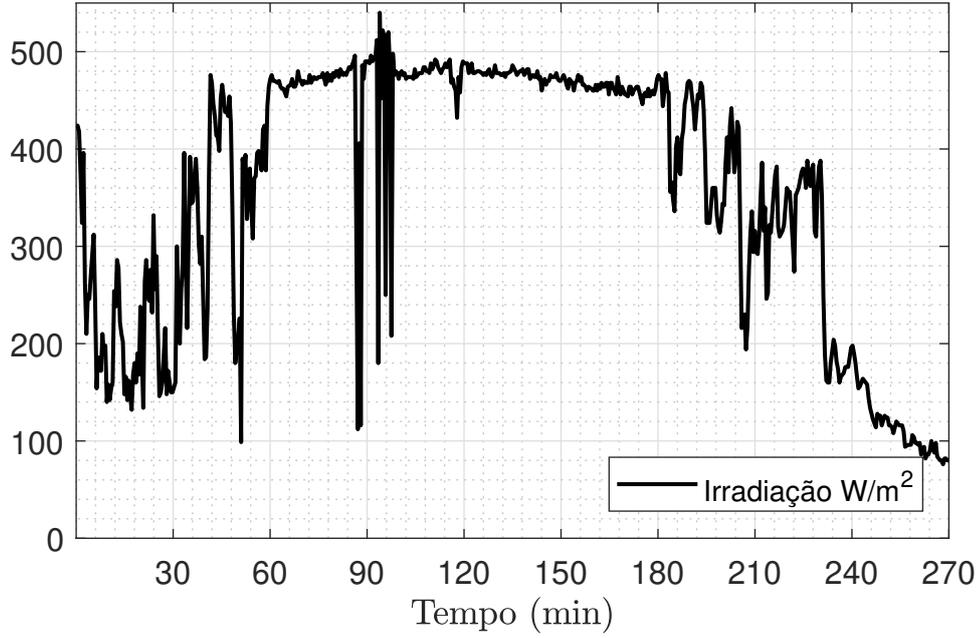


Figura 29 – Irradiação do segundo ensaio

são mostrados os parâmetros estimados, a validação e a comparação entre as estimações.

$$T_{oF}(k) - T_{oF}(k-1) = \theta_1 I(k-1) + \theta_2 (\bar{T}(k-1) - T_a(k-1)) + \theta_3 \dot{m}_F(k-1) (T_{oF}(k-1) - T_{iF}(k-1)) \quad (4.23)$$

Para o primeiro ensaio, foram estimados os seguintes parâmetros: $\theta_1 = 0.00232 \left[\frac{^\circ C m^2}{W} \right]$, $\theta_2 = -0.01675$ e $\theta_3 = -0.04955 \left[\frac{min}{kg} \right]$. Esses parâmetros foram validados usando os dados do segundo ensaio, conforme a figura 30. O desvio padrão dessa validação é $\sigma = 0.2597^\circ C$

Para o segundo ensaio, foram estimados os seguintes parâmetros: $\theta_1 = 0.00217 \left[\frac{^\circ C m^2}{W} \right]$, $\theta_2 = -0.01918$ e $\theta_3 = -0.03972 \left[\frac{min}{kg} \right]$. Esses parâmetros foram validados usando os dados do primeiro ensaio, conforme a figura 31. O desvio padrão dessa validação é $\sigma = 0.2362^\circ C$

Ao analisar os gráficos das validações, percebe-se que as duas estimações são muito boas,. O critério para escolher qual estimaco usar foi o do menor desvio padro, que no caso  a segunda identificaco. Ento o modelo do painel solar identificado fica de acordo com a equaco 4.24.

$$T_{oF}(k) = 0.00217 I(k-1) - 0.01918 (\bar{T}(k-1) - T_a(k-1)) - 0.03972 \dot{m}_F(k-1) (T_{oF}(k-1) - T_{iF}(k-1)) + T_{oF}(k-1) \quad (4.24)$$

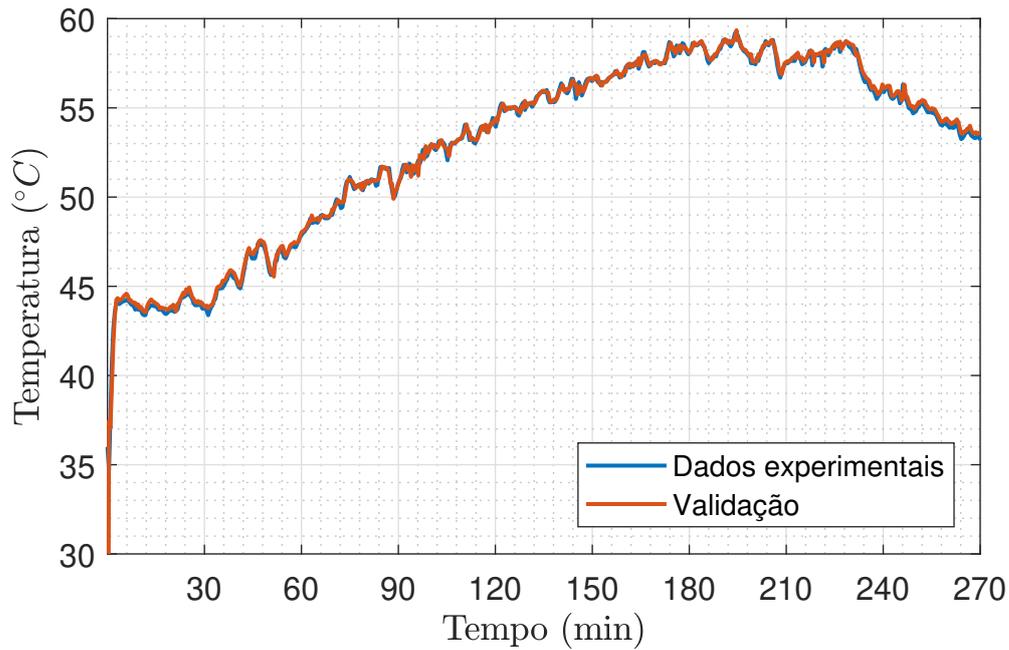


Figura 30 – Validação do primeiro ensaio

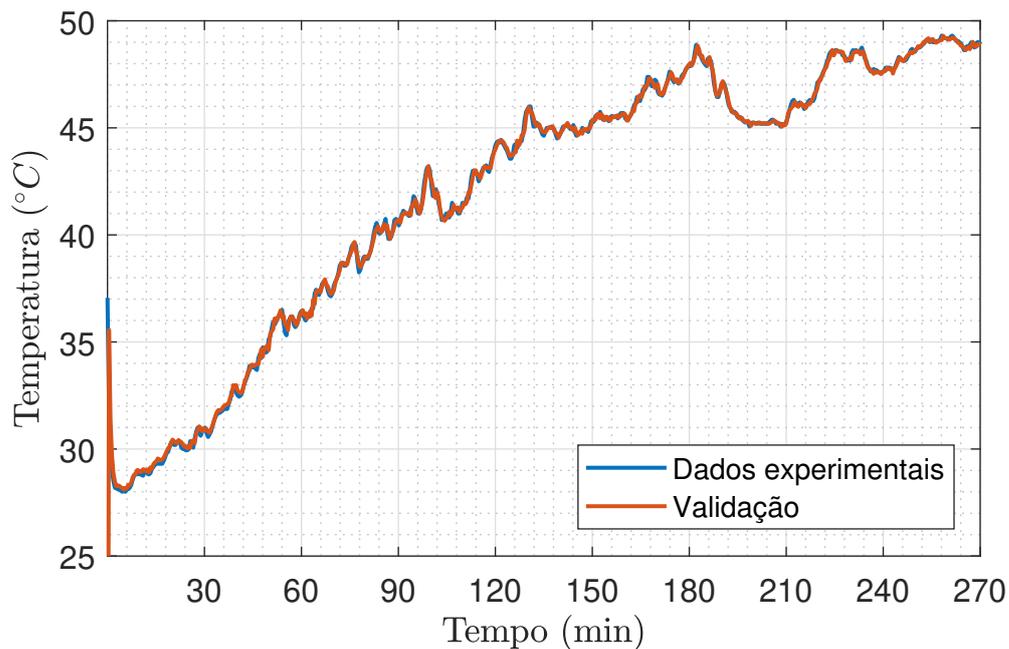


Figura 31 – Validação do segundo ensaio

4.7 Conclusões

Como visto nas seções acima, tanto a identificação do sistema de aquecimento elétrico como a identificação do SCA foram concluídas com sucesso e podem ser usadas para os respectivos propósitos. Com o modelo do aquecimento elétrico será feito o controle local de temperatura para garantir a emulação de um SCA. Já o modelo do SCA será usado emular outros SCAs e futuramente para o projeto de controle preditivo baseado em modelo.

Portanto esses modelos são usados para a estratégia de emulação que é apresentada no capítulo seguinte.

5 Emulação

O objetivo de emular um campo solar de grande porte é basicamente poder estudar os efeitos da diferença de irradiação solar em partes distintas do campo sem a necessidade de realizar experimentos em campos de tamanho real, uma vez que ainda não existe no Brasil uma unidade deste porte destinada a pesquisa e desenvolvimento.

A diferença de irradiação se dá principalmente durante a passagem de nuvens que cobrem parcialmente o conjunto de coletores solares. Os coletores que não recebem irradiação solar direta podem funcionar como radiadores durante a passagem de fluido, dissipando calor para o meio ambiente, que é um efeito indesejado em um sistema de aquecimento. Portanto, para um campo solar de grande porte, a necessidade de controle não está só em manter a saída na temperatura desejada, mas também em evitar que coletores funcionem como radiadores¹.

A emulação de um SCA se dará através do acionamento de um aquecedor elétrico. O sinal de acionamento do aquecedor é influenciado por um sinal de irradiação simulado², de tal forma que se cada aquecedor recebe um sinal de irradiação diferente os aquecedores desempenham o papel de coletores localizados distantes o suficiente como para serem afetados por diferentes conjuntos de nuvens, como em um sistema de grande porte real.

5.1 Estratégia

A estratégia para emular um SCA é apresentada na figura 32.

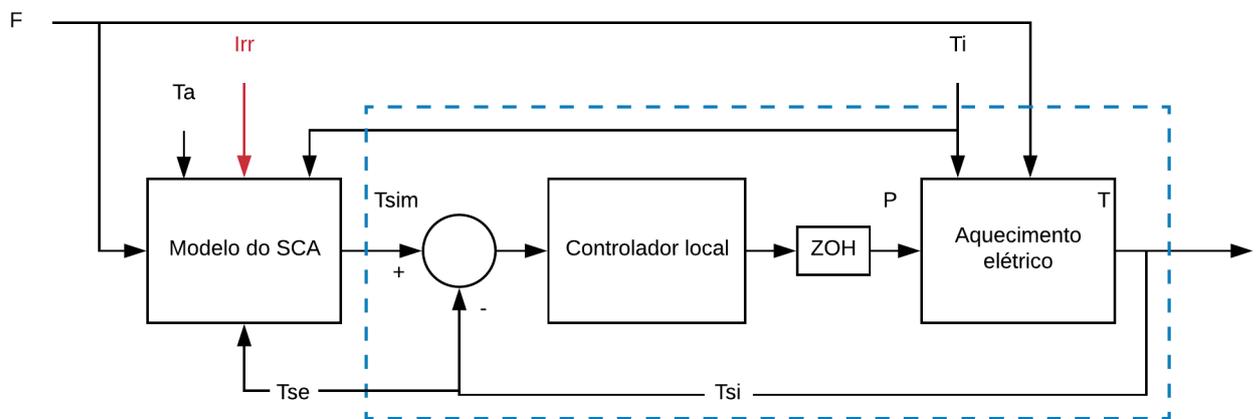


Figura 32 – Diagrama de blocos da emulação

¹ Nesses casos corta-se a vazão do sistema para não resfriar o fluido

² Sinal obtido com o sensor de irradiação em experimentos passados.

A área tracejada da figura 32 contém o controle local de temperatura, se o tempo para atingir regime permanente dessa malha for menor que o T_{SE} (Tempo de amostragem externo, usado para simular o campo solar controlado pelo MPC multivariável), então a temperatura do fluido segue a temperatura simulada T_{SIM} para cada amostra do controlador externo MPC. Em regime permanente $T/T_{SIM} = 1$. Nessa condição de seguimento de referência dentro do tempo desejado, o diagrama da figura 32 cumpre a função de emular um SCA.

O modelo usado para simular o campo solar é dado por:

$$T_{oF}(k) = T_{oF}(k-1) + \frac{\beta_I T_{SE}}{\rho C_p A_a} I(k-1) - \frac{HT_{SE}}{L_{eq} \rho C_p A_a} (\bar{T}(k-1) - T_a(k-1)) - \frac{T_{SE}}{\rho A_a \eta_{eq}} \dot{m}_F(k-1) \frac{T_{oF}(k-1) - T_{iF}(k-1)}{L_{eq}} \quad (5.1)$$

Já o modelo que relaciona a temperatura de saída do aquecedor com a vazão e temperatura de entrada da água e a potência de aquecimento é:

$$V \frac{dT(t)}{dt} = F(t)[T_i(t) - T(t)] + \frac{P(t)}{\rho c_p} \quad (5.2)$$

Onde T_{SIM} na figura 32 é a temperatura simulada pelo modelo do SCA, T é a temperatura real, medida na saída do aquecedor, P é o sinal de acionamento do aquecedor, que é convertida em Watts, T_i é a temperatura de entrada da água, F é a vazão³, T_a é a temperatura ambiente, I_{rr} é a irradiação (em vermelho, pois é o único sinal fictício), T_{SE} é o tempo de amostragem do controle externo e T_{SI} é o tempo de amostragem do controle interno. A figura 33 mostra o diagrama de blocos completo do sistema, com o MPC, a emulação e o SCA real.

5.1.1 Projeto do controlador interno

Para projetar o controle local de temperatura é usado o modelo da equação 4.7 com os parâmetros θ_1 e θ_2 do modelo discreto com aproximação de Euler do modelo contínuo apresentado na equação 5.2. Neste capítulo se considera que θ_1 e θ_2 são conhecidos. O procedimento de identificação destes parâmetros foi apresentado no capítulo 4.

$$\theta_1 [T(k) - T(k-1)] = F(k-1) [T_i(k-1) - T(k-1)] + \theta_2 P(k-1) \quad (5.3)$$

³ No diagrama, o mesmo sinal de vazão está sendo usado como entrada do modelo SCA e do sistema real de aquecimento elétrico, porém na prática o sinal de entrada do modelo é a vazão de referência e no aquecedor elétrico é a vazão real

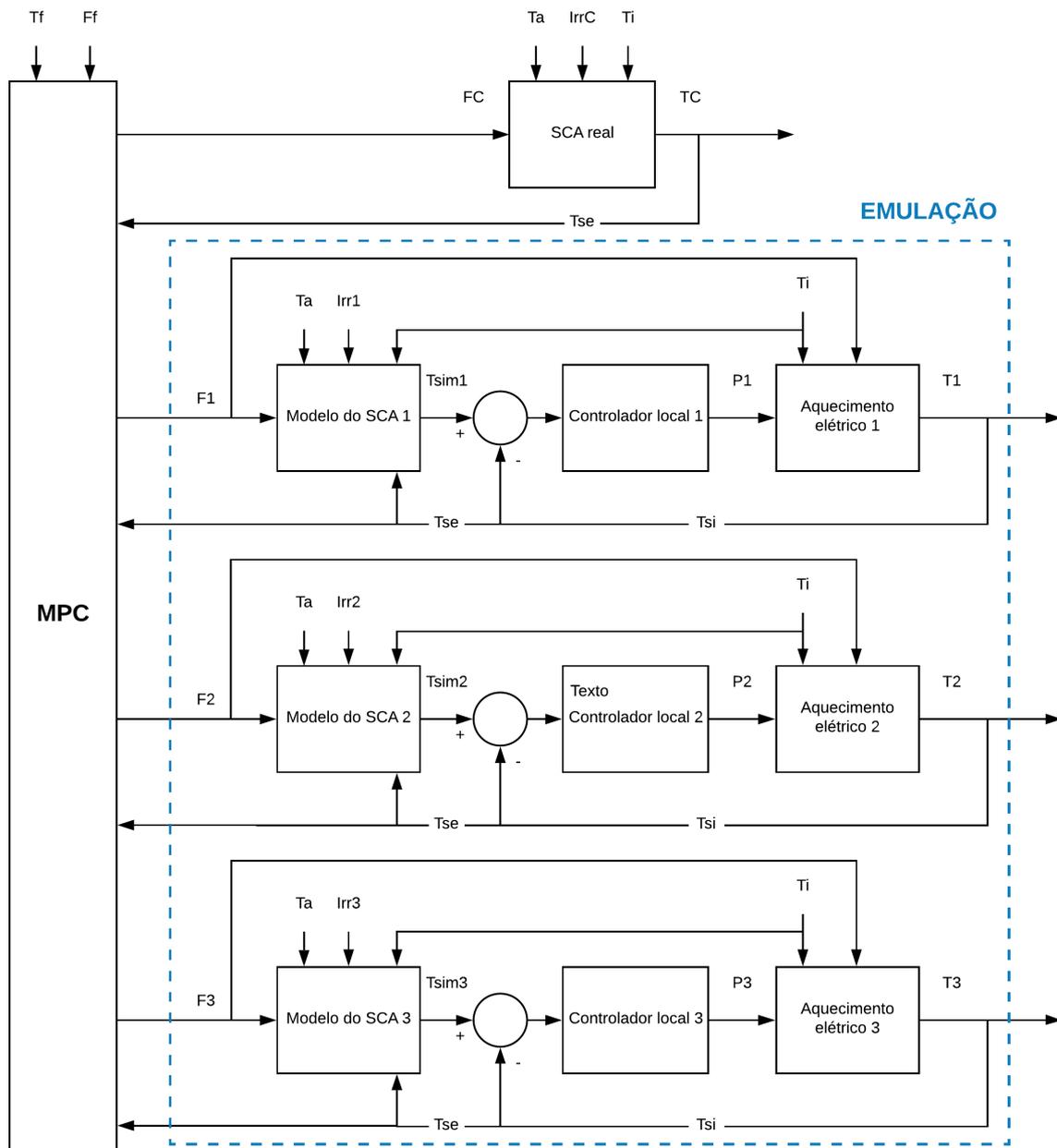


Figura 33 – Diagrama de blocos do sistema completo

Pela equação 5.3 observa-se que o termo $F(k-1)[T_i(k-1) - T(k-1)]$ é não-linear. Para resolver esse problema propõe-se aqui usar uma estratégia de realimentação linearizante. Assim, a parte não linear do sistema é anulada pela realimentação. Como $P(k-1)$ é a variável manipulada do sistema de controle, a realimentação linearizante se calcula como:

$$P(k-1) = \frac{u(k-1) - F(k-1)[T_i(k-1) - T(k-1)]}{\theta_2} \quad (5.4)$$

Sendo $u(k-1)$ a nova variável de controle. Substituindo $P(k-1)$ da equação 5.4 na equação 5.3 se obtém um modelo linear que representa a relação entre o novo sinal de

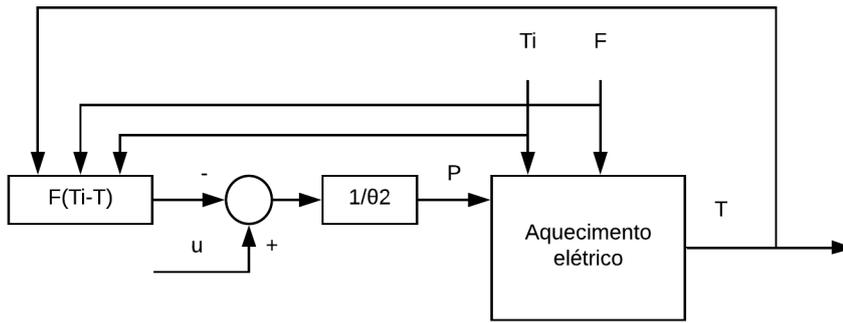


Figura 34 – Diagrama de blocos da realimentação linearizante

controle e a temperatura T .

$$\theta_1[T(k) - T(k-1)] = u(k-1) \quad (5.5)$$

Aplicando a transformada Z na equação 5.5:

$$T(z)\theta_1(1 - z^{-1}) = u(z)z^{-1} \quad (5.6)$$

$$\frac{T(z)}{u(z)} = \frac{z^{-1}}{\theta_1(1 - z^{-1})} = \frac{1}{\theta_1(z - 1)} \quad (5.7)$$

A função de transferência discreta da equação 5.7 mostra que a relação entre $T(z)$ e $u(z)$ é de uma planta integradora. Dessa forma para seguimento de referência do tipo degrau basta projetar um controlador proporcional, uma vez que todas as possíveis perturbações que possam ocorrer são medidas e realimentadas no cálculo de P . Porém, como o sistema usa a água que circula pelo tanque de água quente, essa água está constantemente sendo aquecida pelos aquecedores e pelo painel solar, então a temperatura da entrada da água T_i é do tipo rampa. De acordo com o teorema do valor final, para rejeitar perturbação do tipo rampa deve-se projetar um PI para esse caso onde a planta já é integradora. A figura 35 mostra o sistema em malha fechada depois de aplicar a realimentação linearizante.

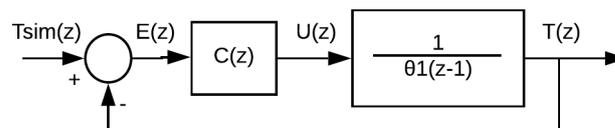


Figura 35 – Malha fechada do sistema com realimentação linearizante

Sendo C um controlador proporcional integral no domínio Z :

$$C(z) = \frac{C_1 z + C_0}{z - 1} \quad (5.8)$$

A função de transferência em malha fechada usando o controlador da equação 5.8 de $T(z)$ para $T_{sim}(z)$ é:

$$\frac{T(z)}{T_{sim}(z)} = \frac{\frac{C_1 z + C_0}{\theta_1}}{z^2 + z \left(\frac{C_1 - 2\theta_1}{\theta_1} \right) + \frac{\theta_1 + C_0}{\theta_1}} \quad (5.9)$$

O sistema de aquecimento elétrico deve atingir regime permanente com tempo menor que T_{SE} para garantir que a temperatura na saída do aquecedor elétrico seja igual à temperatura de simulação do painel solar. Dessa forma, como $T_{SE} = 30$ segundos, o $t_{5\%}$ do sistema de aquecimento em malha fechada é definido em 20 segundos.

O sistema em malha fechada apresentará dois polos: um da própria planta integradora e outro do controlador. Então para atingir as especificação de tempo desejado, o projeto do PI será pela alocação de dois polos reais e iguais.

Como $t_{5\%} = 20s = 0,33min$ e $T_{SI} = 1,5s = 0,025min$ e sabendo que $4,8\tau \approx t_{5\%}$, então:

$$\tau \leq \frac{t_{5\%}}{4,8} = 0,068min \quad (5.10)$$

O polo em Z pode ser calculado como:

$$Pol(z) = e^{-\frac{T_{SI}}{\tau}} \approx 0.69 \quad (5.11)$$

Com o polo calculado na equação 5.11, obtém-se o polinômio do sistema desejado em malha fechada ($z^2 - 1,38z + 0,476$). Igualando esse polinômio desejado com o polinômio característico do sistema da equação 5.9 obtém-se os valores $C_1 = 0,62\theta_1$ e $C_0 = -0,524\theta_1$. A figura 36 apresenta a estrutura de controle para a temperatura na saída dos aquecedores.

5.1.2 Compensação do atraso de transporte

Como o modelo da equação 4.22 tem atraso de transporte $d = 4$ amostras, então é necessário compensar o atraso na estrutura de controle. Portanto essa seção destina-se a explicar e resolver tal problema.

O modelo descrito na seção 4.1 com a equação 4.2 refere-se à temperatura no interior de um tanque aquecido com uma potência elétrica $P(k)$. Porém, na instalação

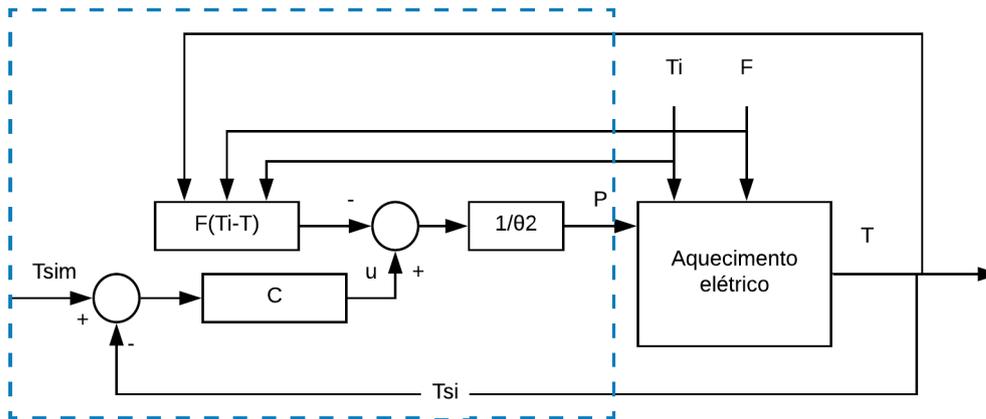


Figura 36 – Estrutura de controle interno

física do sistema o sensor de temperatura está a aproximadamente 20cm de distância do local de aquecimento. Então, a temperatura medida é a temperatura no interior do tanque atrasada d amostras.

$$T_m(k) = T(k - d)$$

Para evitar a perda de performance em um sistema com atraso, usa-se o Preditor de Smith, como na figura 37.

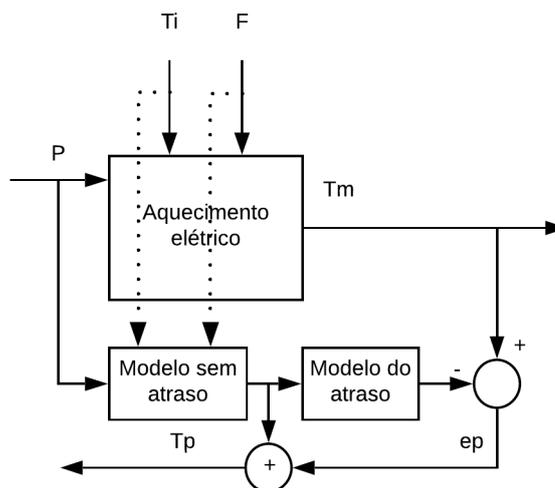


Figura 37 – Estrutura de predição do Preditor de Smith

Essa estrutura compara a temperatura medida $T_m(k)$ com a temperatura simulada atrasada, formando o erro de predição $e_p(k)$. Esse erro é então somado com a temperatura simulada sem atraso, formando a temperatura predita $T_p(k)$. Para um bom modelo e sem a

presença de perturbação $e_p(k) \approx 0$, portanto $T_p(k)$ fica em função apenas da temperatura simulada sem atraso.

Sendo assim, basta usar o controlador já projetado na subseção 5.1.1, porém usando a saída predita $T_p(k)$ ao invés de usar $T_m(k)$ como na figura 38. A seção seguinte apresenta a lei de controle projetada nessa seção.

O bloco "modelo sem atraso" da figura 37 é o modelo⁴ não-linear discreto do aquecedor elétrico dado por:

$$T(k) = \frac{F(k-1)[T_i(k-1) - T(k-1)] - \theta_2 P(k-1)}{\theta_1} + T(k-1) \quad (5.12)$$

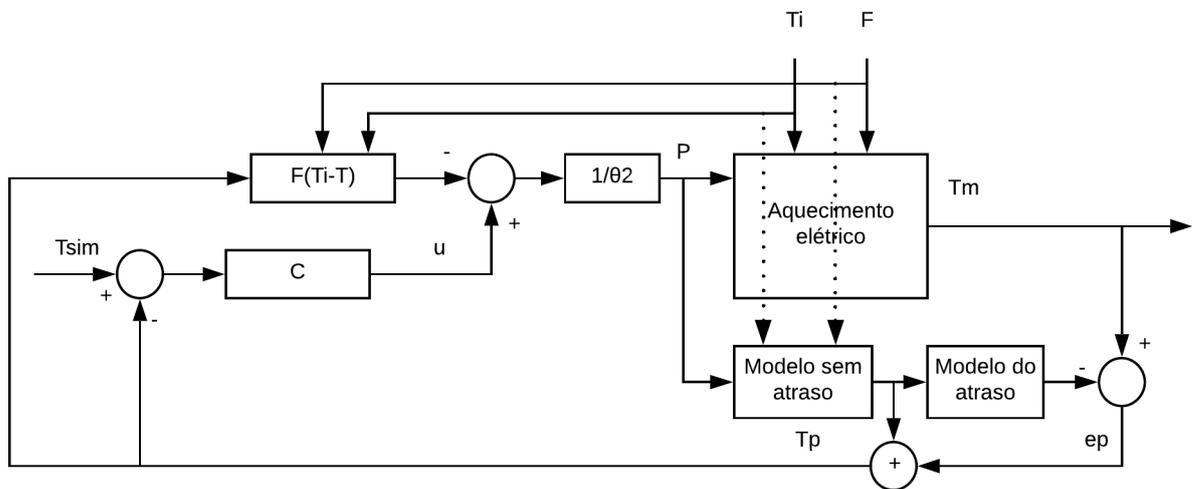


Figura 38 – Preditor de Smith

5.2 Lei de controle

A lei de controle foi desenvolvida para programação em MATLAB. O mesmo código pode ser embarcado no LabView através de uma função que executa comandos do MATLAB. Portanto o código a seguir está pronto para ser usado na planta caso não haja nenhuma outra alteração na estrutura de controle. Todas as simulações feitas nesse capítulo foram feitas usando o código abaixo ou variações dele.

```

1 function y = fcn(u)
2 F = u(1); %VAZAO MEDIDA
3 Ti = u(2); %TEMPERATURA DE ENTRADA
4 Ref = u(3); %REFERENCIA

```

⁴ Usando o sinal P como entrada do bloco, deve-se usar o modelo não-linear para fazer a previsão da resposta do sistema, sendo assim não se usa o modelo integrador (que não pode ser usado no Preditor de Smith)

```

5 Tm = u(4);    %TEMPERATURA MEDIDA
6
7 %Parametros
8 V = 3.7;
9 Tsi = 0.025;
10 theta1 = V/Tsi;
11 theta2 = 0.1752;
12 C1 = 0.62*theta1;
13 C0 = -0.524*theta1;
14
15 %VARIABLES PARA GUARDAR O PASSADO
16 persistent Pp1 TPp1 Tmodelp1 Tmodelp2 Tmodelp3 Tmodelp4 Tmodelp5 ...
    Tmodelp6 erropl up1 Fp1 Tip1;
17 if isempty(Pp1)
18     Pp1 = 0;           %Potencia em k-1
19     TPp1 = Tm;        %T predita em k-1
20     Tmodelp1 = Tm;    %T do modelo em k-1
21     Tmodelp2 = Tm;    %T do modelo em k-2
22     Tmodelp3 = Tm;    %T do modelo em k-3
23     Tmodelp4 = Tm;    %T do modelo em k-4
24     Tmodelp5 = Tm;    %T do modelo em k-5
25     Tmodelp6 = Tm;
26     erropl = 0;       %erro(k-1) para controle interno
27     up1 = 0;          %u(k-1) para controle interno
28     Fp1 = F;
29     Tip1 = Ti;
30 end
31
32 %Modelo
33 Tmodel = (Fp1*(Tip1-Tmodelp1)+theta2*Pp1)/theta1+Tmodelp1; %simula o ...
    sistema
34
35 %Predicao
36 erroP = Tm - Tmodelp4;    %erro de predicao
37 TP = Tmodel + erroP;
38 erro = Ref-TP;
39
40 %PI
41 u = C1*erro + C0*erropl + up1;
42
43 %Linearizacao
44 P = (u - F*(Ti-Tm))/theta2;
45
46 %atualizacao
47 Tmodelp6 = Tmodelp5;
48 Tmodelp5 = Tmodelp4;
49 Tmodelp4 = Tmodelp3;

```

```

50 Tmodelp3 = Tmodelp2;
51 Tmodelp2 = Tmodelp1;
52 Tmodelp1 = Tmodel;
53
54 erropl = erro;
55 up1 = u;
56 Fp1 = F;
57 Tip1 = Ti;
58 TPp1 = TP;
59 Pp1 = P;
60
61 y = P;

```

5.3 Simulações

Para se testar o algoritmo de controle antes de implementar na planta real, foi criado um simulador no software MATLAB (figura 39). A figura 40 apresenta a simulação do sistema identificado na seção 4.5 e com os mesmos sinais aplicados na figura 22.

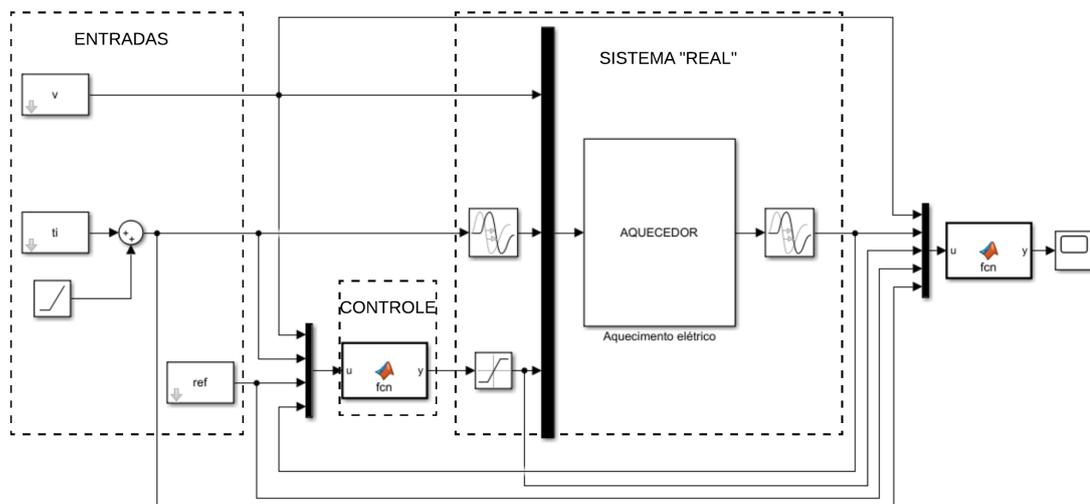


Figura 39 – Diagrama de blocos no Simulink

A figura 41 simula o sistema com entrada do tipo degrau usando a estrutura de realimentação linearizante. Como visto na equação 5.7, a planta é integradora.

A figura 42 simula o sistema com a aplicação do PI projetado na seção 5.1.1, sem a inclusão de atraso de transporte. De acordo o gráfico, percebe-se que segue referência dentro do tempo desejado. A realimentação linearizante em simulação rejeita a variação de vazão instantaneamente.

Com a inclusão do atraso de 7,5 segundos e a estrutura do Preditor de Smith no controlador, a resposta do sistema apresenta-se de acordo com a figura 43. Nesse caso,

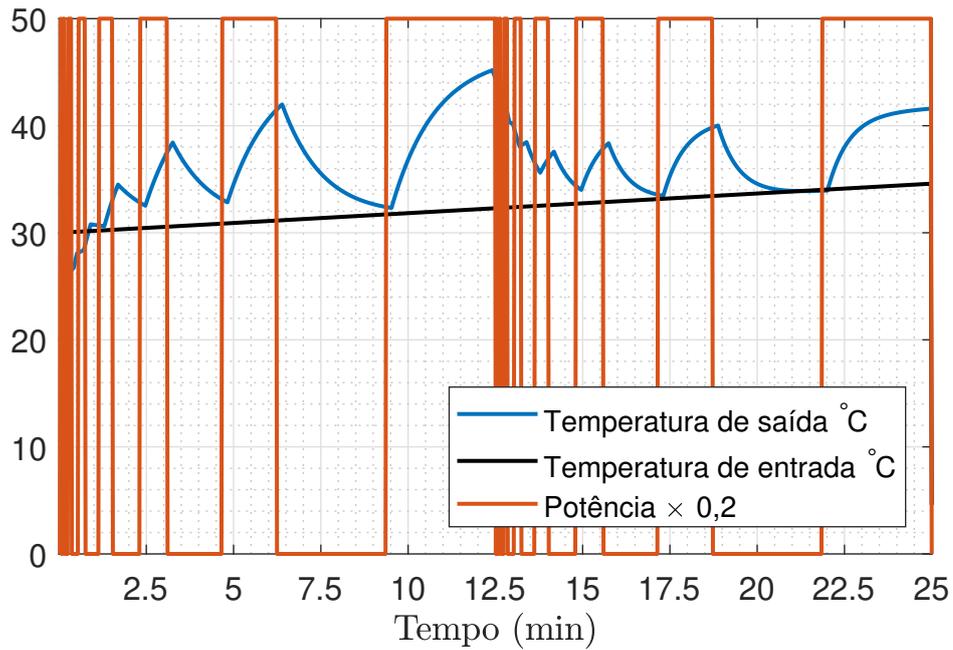


Figura 40 – Simulação do sistema identificado

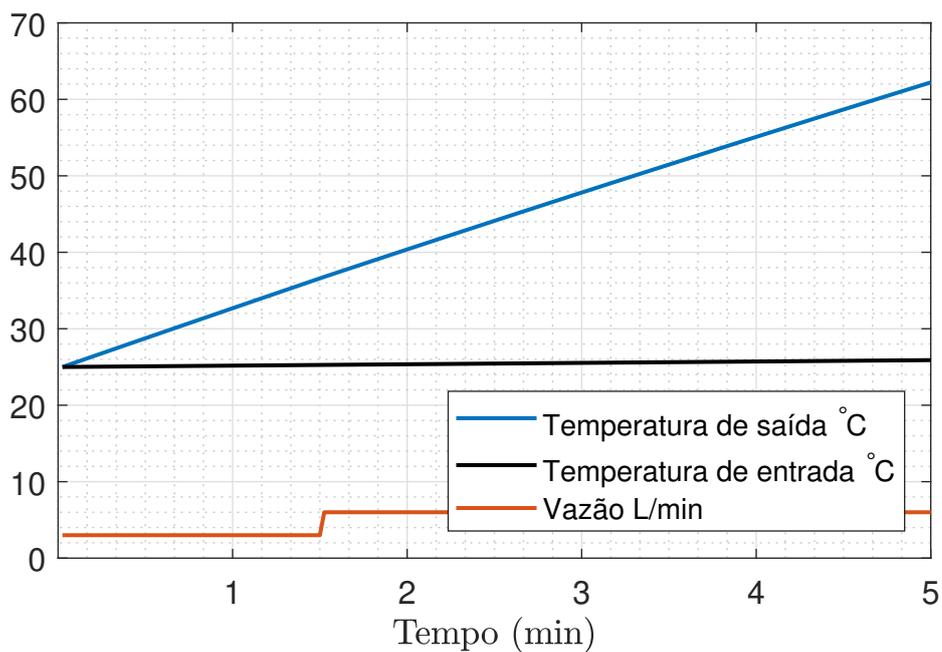


Figura 41 – Simulação do sistema com realimentação linearizante

o modelo do atraso é igual ao atraso real. Mesmo com o atraso o sistema atinge regime permanente dentro dos 30 segundos.

Na figura 44 foi acrescentado erro de modelagem do atraso, modelo tem atraso de 6 segundos enquanto o sistema mantém-se com atraso de 7,5 segundos. Ainda que a resposta piore muito, o sistema atinge um valor próximo da referência dentro de 30 segundos. Depois dessa simulação foi incluído um erro de 20% nos parâmetros estimados. Pela simulação da figura 45 percebe-se que o erro nesses parâmetros não altera tanto a

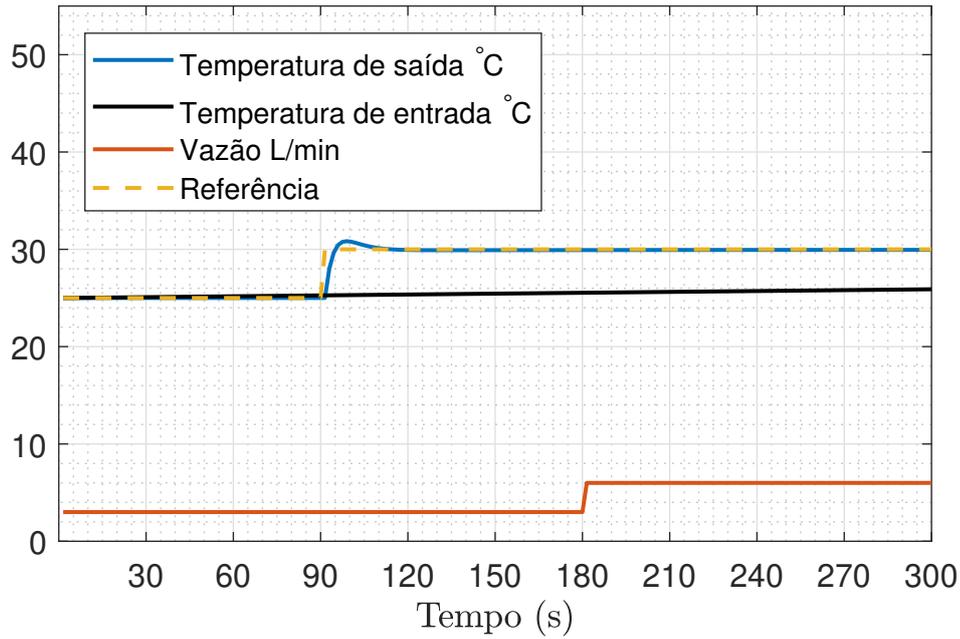


Figura 42 – Simulação do sistema com controle PI sem atraso

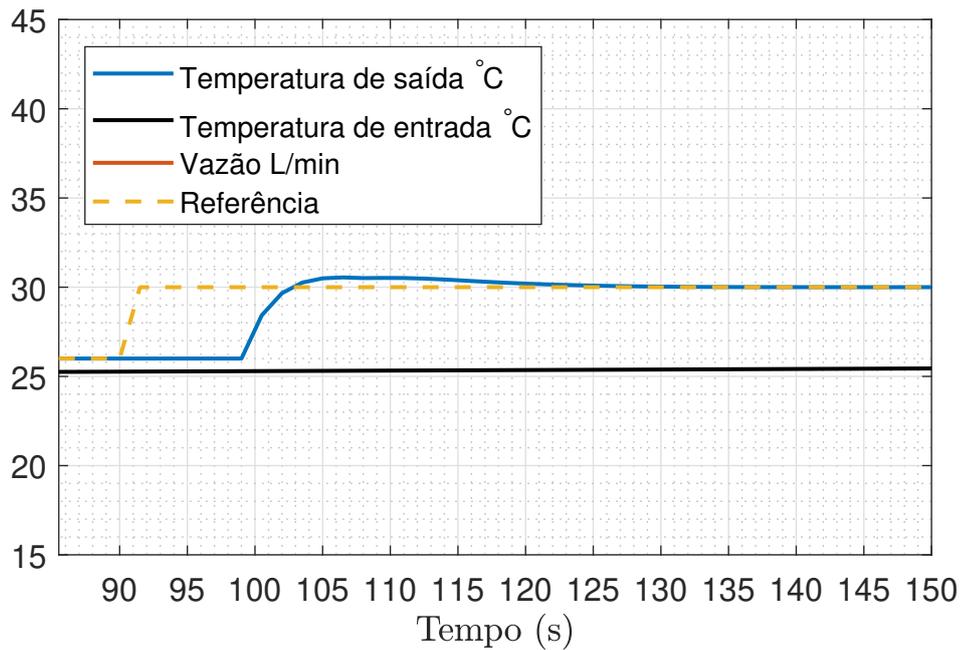


Figura 43 – Simulação do sistema com atraso e a estrutura do Preditor de Smith

resposta do sistema como o erro no modelo do atraso.

As simulações das figuras 42, 43, 44 e 45 têm comportamento razoavelmente bom, porém não são de um sistema com saturação na variável manipulada. A figura 46 apresenta o sinal da variável manipulada ($P(k)$) sem saturação para a mesma simulação da figura 42. Como o sistema real tem saturação na variável manipulada em 255, e na simulação foi necessário uma amplitude de aproximadamente 2500 para atingir regime permanente dentro do tempo de 30 segundos, dificilmente o sistema conseguirá atingir a especificação de

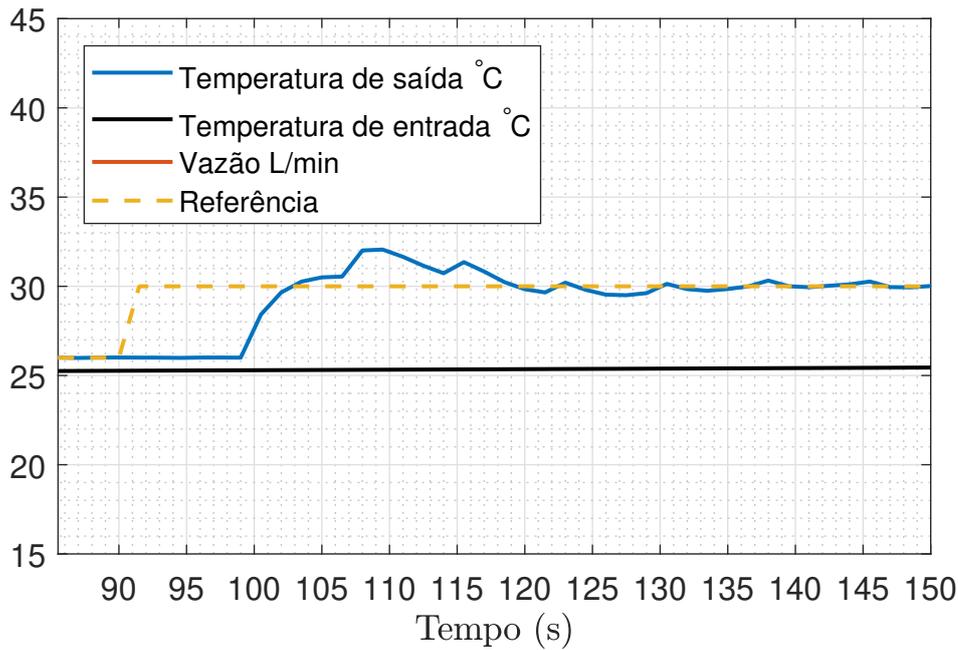


Figura 44 – Simulação do sistema com a estrutura do Preditor de Smith e atraso do modelo diferente do real

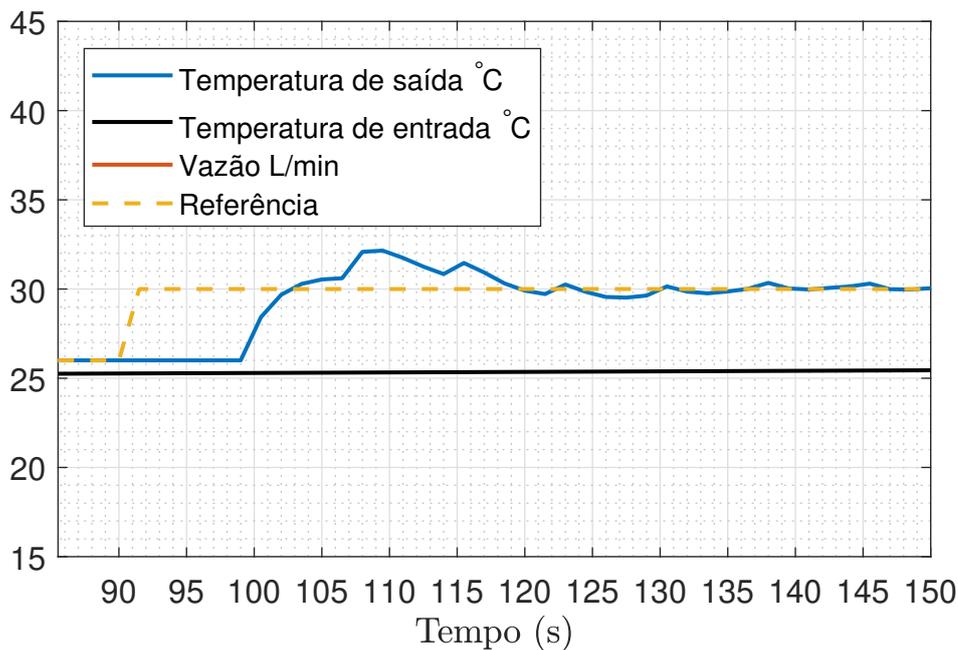


Figura 45 – Simulação do sistema com 20% de erro nos parâmetros

desempenho quando houver grande mudança na variável controlada, nesse caso a referência mudou de 26°C para 30°C . Além desse problema, para não acumular erro no integrador do PI, é necessário projetar um *anti-windup*. Como o controlador projetado aplica o sinal $P(k)$, que é uma combinação da realimentação linearizante e o controle PI, não houve tempo suficiente para estudar como seria a aplicação dessa estrutura *anti-windup*. Portanto, manteve-se as simulações sem saturação na variável manipulada.

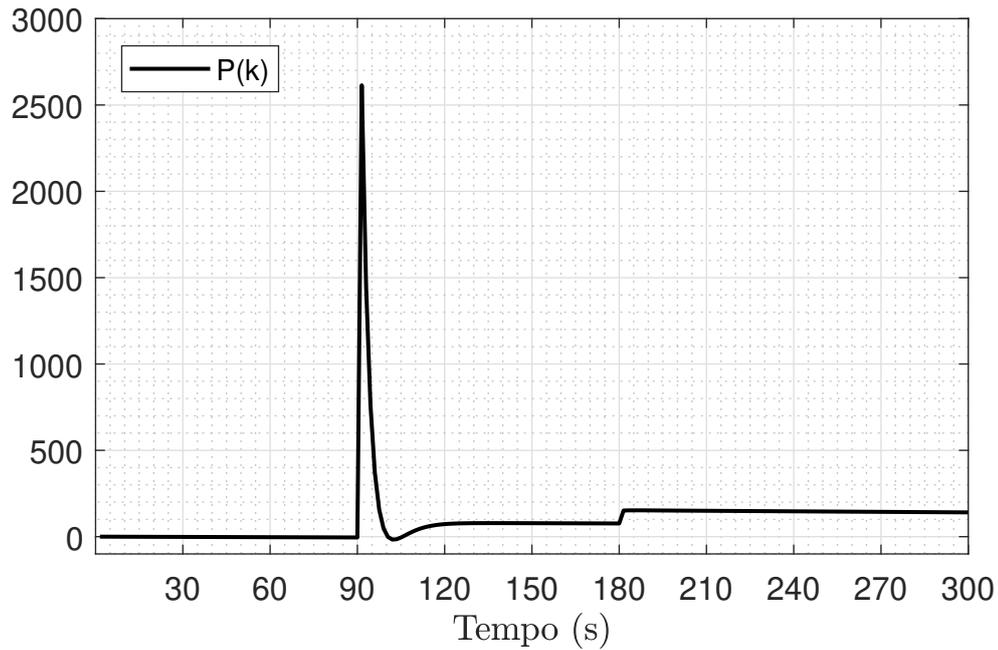


Figura 46 – Variável manipulada sem saturação com o sistema sem atraso

5.3.1 Simulação de temperatura experimental

Nessa seção foi feita uma simulação usando como referência o perfil de temperatura de 10 minutos do primeiro ensaio para a identificação do SCA. Portanto a temperatura de referência tem comportamento real e varia com período de 30 segundos (o mesmo tempo de amostragem do SCA). Além disso, manteve-se o erro de modelagem do atraso e o erro de 20% na modelagem dos parâmetros do aquecedor. A vazão foi mantida constante. A figura 47 mostra a saída do sistema e a figura 48 mostra a variável manipulada.

Ao analisar a figura 47, nota-se que para grandes variações de temperatura, o comportamento da saída não é bom. Porém, para pequenas variações, a temperatura na saída do aquecedor consegue seguir a referência dentro do tempo esperado. Já a figura 48 mostra que o sinal de potência é realmente um problema quando há grande variação na temperatura. A figura 49 mostra a saída do sistema com saturação entre 0 e 255 na variável $P(k)$ sem *anti-windup*.

5.4 Implementação

Como descrito na seção acima, não houve tempo hábil para estudar a estrutura *anti-windup* para esse caso: onde a saturação ocorre na variável $P(t)$ mas o acúmulo do erro acontece na integral do erro do controlador PI. Além disso, não foram concluídas as adaptações necessárias no LabView para embarcar a lei de controle. Portanto, devido a todos os obstáculos que ocorreram ao longo do período de execução do PFC, não foi possível finalizar a emulação dos campos solares até a data de entrega da versão final do

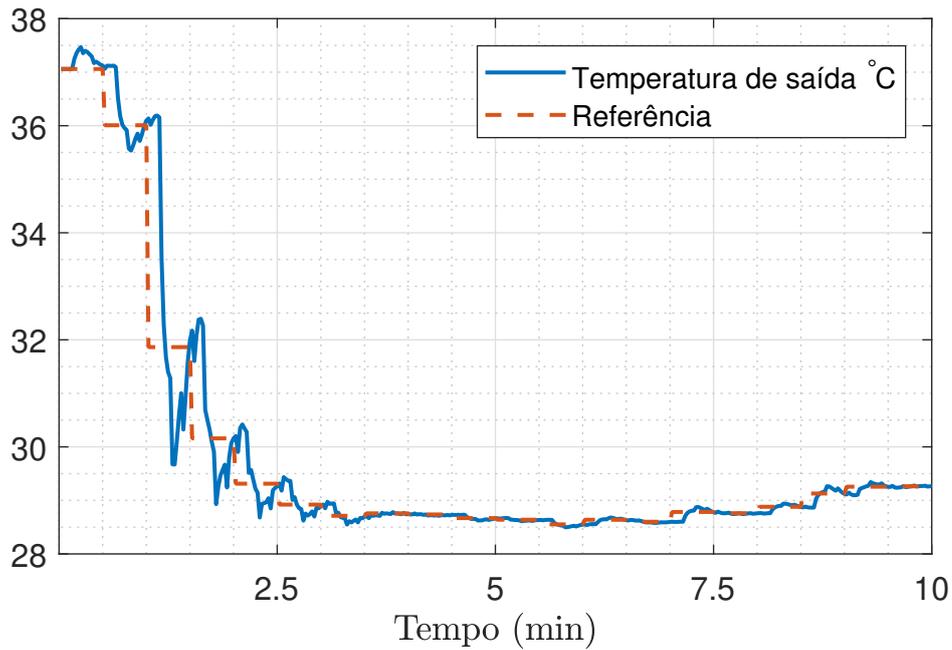


Figura 47 – Rastreamento de referência

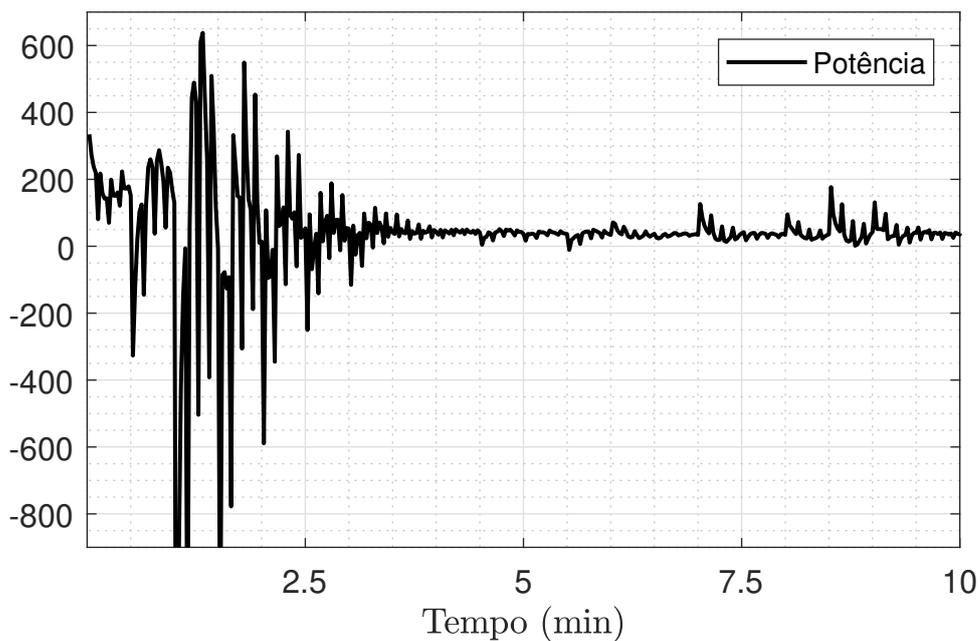


Figura 48 – Variável manipulada

documento para avaliação. Porém pretende-se, se tudo der certo, terminar a implementação da emulação dos campos e apresentar os resultados práticos no dia da defesa do PFC.

5.5 Conclusões

Como visto nas simulações deste capítulo, o maior problema encontrado foi a saturação da variável manipulada para grandes variações na temperatura de referência.

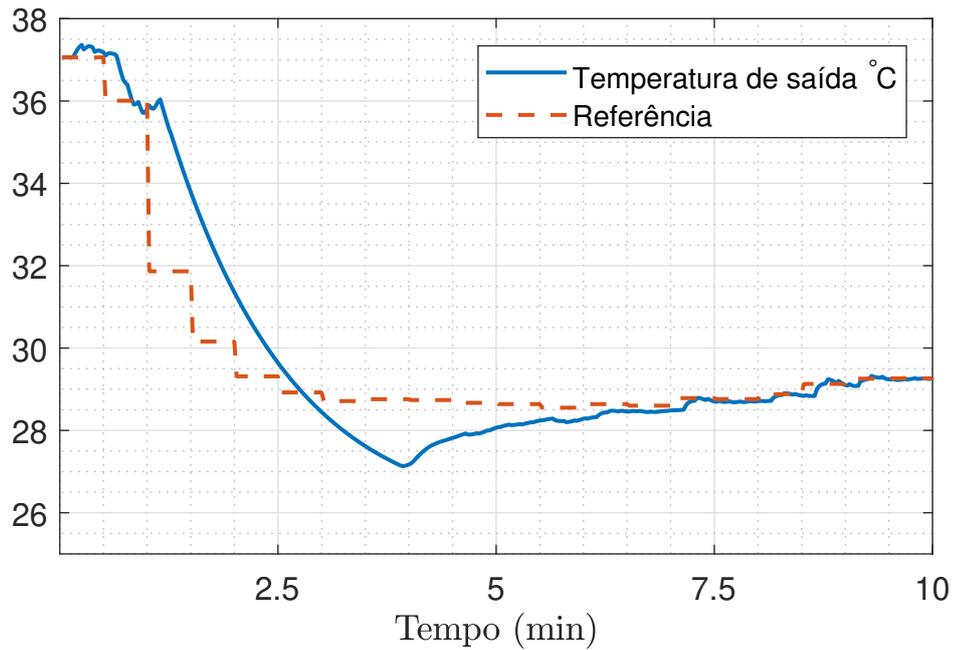


Figura 49 – Resposta do sistema com saturação na variável $P(k)$

Mesmo com a futura aplicação da estrutura *anti-windup*, o sistema não atingirá o regime permanente dentro do tempo hábil quando houver saturação na variável $P(k)$. Portanto deve-se, futuramente, analisar se é possível aumentar o tempo para atingir regime permanente ou aumentar a potência elétrica máxima ou diminuir o atraso de transporte. Todos os outros problemas previamente encontrados foram solucionados, como por exemplo a realimentação linearizante e a compensação do atraso de transporte.

6 Considerações finais e perspectivas futuras

As maiores dificuldades encontradas durante a realização desse Projeto de Final de Curso foram relacionadas a problemas práticos e específicos da natureza do sistema, como por exemplo os problemas na medição de temperatura. Porém, o trabalho em equipe coordenado pelo doutor Paulo Renato da Costa Mendes auxiliou na velocidade de resolução dos problemas. Já o desenvolvimento e a aplicação da teoria se mostraram bastante efetivos, uma vez que entendido o problema, foi rapidamente solucionado, como por exemplo a identificação dos sistemas e a emulação por simulação. Portanto fica claro que o alto nível de aprendizado teórico que o curso proporcionou se mostrou bastante eficaz na resolução de problemas de natureza conhecida, como por exemplo a compensação de atraso no sistema de controle.

Como visto no capítulo 5, para trabalhos futuros, deve-se encontrar uma solução para evitar a saturação da variável manipulada quando houver grande variação na temperatura de referência. Resolver, também, os problemas encontrados nos ramos 1 e 3 (seção 4.5). Resolver de forma definitiva os problemas com os sensores de temperatura (sugestão: trocar a posição dos cabos) e por final implementar o algoritmo de controle preditivo *MPC*.

Referências

- 1 ENERGIA Heliotérmica. Acessado: 09-05-2018. Disponível em: <<http://energiaheliotermica.gov.br/pt-br/energia-heliotermica/como-funciona>>. Citado 3 vezes nas páginas 9, 19 e 20.
- 2 SOLAR Flat Plate Collectors for Solar Hot Water. Acessado: 28-06-2018. Disponível em: <www.alternative-energy-tutorials.com/solar-hot-water/flat-plate-collector.html>. Citado 2 vezes nas páginas 9 e 21.
- 3 THE Multidrop 1-wire bus. Acessado: 27-05-2018. Disponível em: <<https://www.iot-programmer.com/index.php/books/22-raspberry-pi-and-the-iot-in-c/chapters-raspberry-pi-and-the-iot-in-c/39-the-multidrop-1-wire-bus>>. Citado 2 vezes nas páginas 10 e 84.
- 4 TIBA, C. *Atlas Solarimérico do Brasil*. Recife, Brazil: Ed. Universitária da UFPE, 2000. Citado na página 19.
- 5 DIETRICH, J. D. V. et al. Controle de temperatura em campos solares de grande porte utilizando a abordagem do pnmpc - practical nonlinear model predictive control. 2016. Citado na página 21.
- 6 LORENZETTI. *Manual de Instruções de Instalação e Garantia*. [S.l.]. Citado 2 vezes nas páginas 26 e 79.
- 7 NATIONAL INSTRUMENTS. *LabVIEW 2017 Help*. [S.l.], 2017. Citado na página 31.
- 8 ROCA, L. et al. Solar field control for desalination plants. 2008. Citado na página 39.
- 9 AGUIRRE, L. A. *Introdução à Identificação de Sistemas*. 3. ed. Av. Antônio Carlos, 6627, CAD II Bloco III, Campus Pampulha, 31270-901, Belo Horizonte-MG: [s.n.], 2015. Citado 3 vezes nas páginas 41, 43 e 95.
- 10 1-WIRE Protocol. Acessado: 27-05-2018. Disponível em: <<https://www.maximintegrated.com/en/products/digital/one-wire.html>>. Citado na página 84.
- 11 MAXIM INTEGRATED PRODUCTS. *Programmable Resolution 1-Wire Digital Thermometer*. [S.l.], 2015. Citado na página 84.
- 12 DALLAS Temperature Control Library. Acessado: 26-05-2018. Disponível em: <["https://www.milesburton.com"](https://www.milesburton.com)>. Citado na página 84.
- 13 GUIDELINES for Reliable Long Line 1-Wire Networks. Acessado: 28-05-2018. Disponível em: <<https://www.maximintegrated.com/en/app-notes/index.mvp/id/148>>. Citado na página 85.

Apêndices

APÊNDICE A – Cálculo da vazão de segurança

O cálculo da vazão mínima de segurança foi feito antes de qualquer utilização dos aquecedores elétricos para garantir que o acionamento não danificaria o sistema. Ele é baseado no manual do aquecedor elétrico [6] e nas seguintes suposições:

- O sistema real comporta-se de forma parecida com o modelo da equação 4.1;
- O volume interno do aquecedor V é aproximadamente $0,5L$;
- ρ da água é $1kg/L$ e não varia;
- c_p da água é $1cal/g^\circ C$

De acordo com o manual [6], a potência máxima para 220V na posição 1 é de 2900W. Então a equação A.1 converte a unidade de Watts para Quilocalorias/min igualando a unidade com o calor específico da água c_p e a unidade de tempo da vazão.

$$2900W = 2493,55kcal/h = 41,56kcal/min \quad (A.1)$$

Então, a análise dimensional do termo que relaciona a potência com a temperatura da equação 4.1 é:

$$\frac{P}{V\rho c_p} \left[\frac{\frac{kcal}{min}}{L \frac{kg}{L} \frac{cal}{g^\circ C}} \right] \quad (A.2)$$

$$\frac{P}{\rho c_p} \left[\frac{^\circ C}{min} \right] \quad (A.3)$$

Para encontrar a vazão de segurança, foi usado o critério $\Delta T_{max} \leq 25^\circ C$, ou seja, a máxima variação de temperatura em relação à temperatura de entrada é $25^\circ C$. A partir da equação 4.1, com $T_i = 0$, $\frac{dT(t)}{dt} = 0$ e $P_{max} = 41,56$ chega-se à equação A.4 e à solução A.5.

$$0 \geq -F\Delta T_{max} + P_{max} \quad (A.4)$$

$$F \geq 1,66L/min \quad (A.5)$$

APÊNDICE B – Definição de T_{SI}

Para definir o período de amostragem do aquecimento elétrico será considerado o modelo da equação 4.1 e supondo uma vazão constante F .

$$\frac{dT(t)}{dt} = \frac{FT_i(t) - FT(t)}{V} + \frac{P(t)}{V\rho c_p} \quad (\text{B.1})$$

A partir do termo $-FT(t)$ é possível perceber que a vazão afeta a dinâmica da temperatura. Para melhor entender é aplicada a transformada de Laplace na equação B.1.

$$sT(s) = \frac{FT_i(s) - FT(s)}{V} + \frac{P(s)}{V\rho c_p} \quad (\text{B.2})$$

$$T(s) = \frac{FT_i(s)}{V \left(s + \frac{F}{V} \right)} + \frac{P(s)}{V\rho c_p \left(s + \frac{F}{V} \right)} \quad (\text{B.3})$$

A equação B.3 mostra que, quanto maior F , menor é o polo do sistema (mais negativo), portanto mais rápido é a dinâmica. Então considerando a faixa de operação para a vazão entre $3L/min$ e $6L/min$, será usada a maior vazão para o cálculo da dinâmica mais rápida e conseqüentemente a definição do período de amostragem (T_{SI}).

$$\tau = \frac{V}{F} = \frac{0.5}{6} = 0,083min$$

O tempo para atingir o regime permanente (ou 95%) de um sistema de primeira ordem é $t_{5\%} \approx 3\tau$ então:

$$t_{5\%} \approx 0,25min = 15s$$

Então o $T_{SI} = 1,5s = 0,025min$. Isso proporciona 10 amostras para o regime transitório para o caso com a dinâmica mais rápida.

APÊNDICE C – Problemas na medição das temperaturas

Esse apêndice destina-se a apresentar de forma resumida a análise e procedimento para resolver o problema, apresentando o raciocínio utilizado para direcionar a busca pela solução.

Após a verificação de que os sensores e atuadores necessários para coletar os dados apenas do painel solar estavam funcionando todos corretamente, optou-se por capturar esses dados mesmo antes de toda a planta estar funcionando corretamente.

Porém a partir da primeira coleta das variáveis para identificação do modelo matemático do painel solar, foi notado a presença aleatória e persistente de medições de 127°C nos sensores de temperatura (Figura 50).

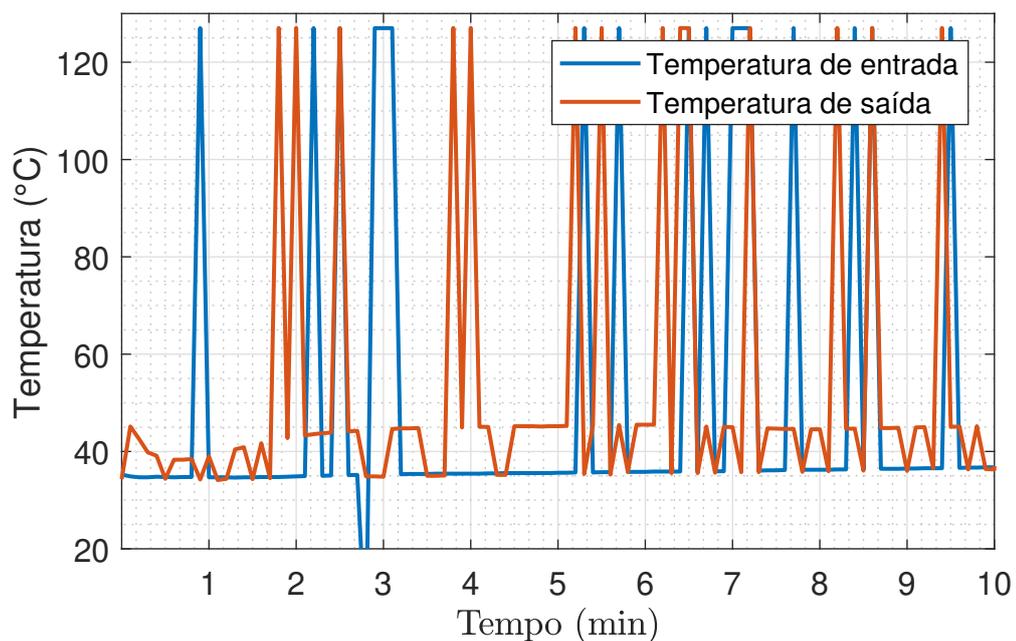


Figura 50 – Exemplo de 2 sinais de temperatura

Essas medições de 127°C ocorriam em todos os sinais de temperatura, contudo esse gráfico foi usado para mostrar um outro problema: a mudança repentina do sinal da temperatura da saída do painel solar para valores parecidos com a temperatura de entrada.

Inicialmente achava-se que seria um problema de contra-fluxo de água, já que os ramos emulados estavam abertos e sem aquecimento nenhum. Ou seja, a água fria de três ramos se misturava à água quente na saída e eventualmente poderia entrar contra o fluxo de água da saída do painel, gerando a leitura de água fria. Mas essa hipótese foi logo

descartada ao se observar a estrutura hidráulica e a posição do sensor de temperatura e a válvula de saída.

C.1 Sobre os sensores

Os sensores de temperatura utilizados na planta são digitais e produzidos pela empresa Maxim Integrated, também proprietária do protocolo de comunicação OneWire [10]. Esse protocolo permite que diversos sensores transmitam a informação pelo mesmo fio de dados conforme a Figura 51.

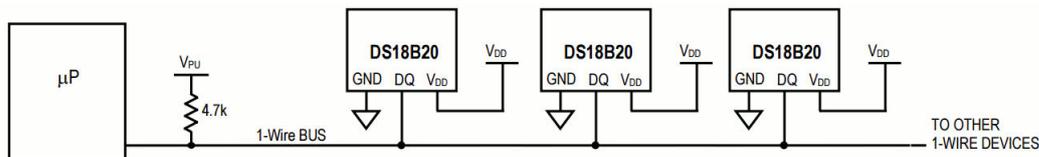


Figura 51 – Exemplo de esquemático OneWire. Retirado de: [3]

Cada sensor possui um endereço físico exclusivo de 64 bits [11]. Isso permite que a biblioteca OneWire saiba de qual sensor está vindo cada informação. Enquanto a biblioteca OneWire identifica de onde vem a informação, a biblioteca Dallas Temperature interpreta o dado recebido como um valor de temperatura em Celsius ou Fahrenheit. Essas duas bibliotecas foram implementadas no Arduino para facilitar a comunicação com os sensores.

C.2 Problema de 127C

Como os sensores usados são digitais, buscou-se entender como funcionava a biblioteca usada no Arduino para interpretar a leitura dos dados. A biblioteca em questão chama-se "Dallas Temperature Control". Na página do autor da biblioteca foi possível encontrar uma referência a esse erro: "*Changed DEVICE_DISCONNECTED value to -127 so it can be passed as a result in functions returning a char value*" [12]. No caso, o real valor da saída serial do Arduino nano é -127, a leitura positiva se dá pelo fato de existir uma rotina no Arduino Mega que converte números negativos em positivos.

Com essa informação, é intuitivo acreditar que os valores de 127 não acontecem somente quando o dispositivo está desconectado, mas também quando há falha na transmissão dos dados e a biblioteca não consegue interpretar os bits recebidos. Então a suspeita é de que esse problema está na transmissão entre sensor e Arduino nano.

A partir dessa suspeita, acreditou-se que a causa do problema de transmissão fosse os cabos de dados estarem juntos com os cabos de força (que são usados para alimentação da bomba e dos aquecedores). O correto nesse caso, seria trocar os cabos de lugar, porém isso levaria muito tempo graças à falta de prática dos envolvidos na instalação elétrica.

C.2.1 Resistor pullup

Então a ideia inicial para resolver o problema foi relacionada ao resistor pullup de $4k7$. No caso, foi diminuir esse resistor de pullup para possivelmente diminuir a quantidade de erros 127 até que a frequência do erro se tornasse insignificante¹.

Para isso, uma estratégia rápida para verificar se havia influência do resistor pullup, foi soldar em paralelo ao resistor presente, outros resistores $4k7$. Dessa forma a resistência equivalente é R/nR onde $R = 4k7$ e nR é o número de resistores em paralelo.

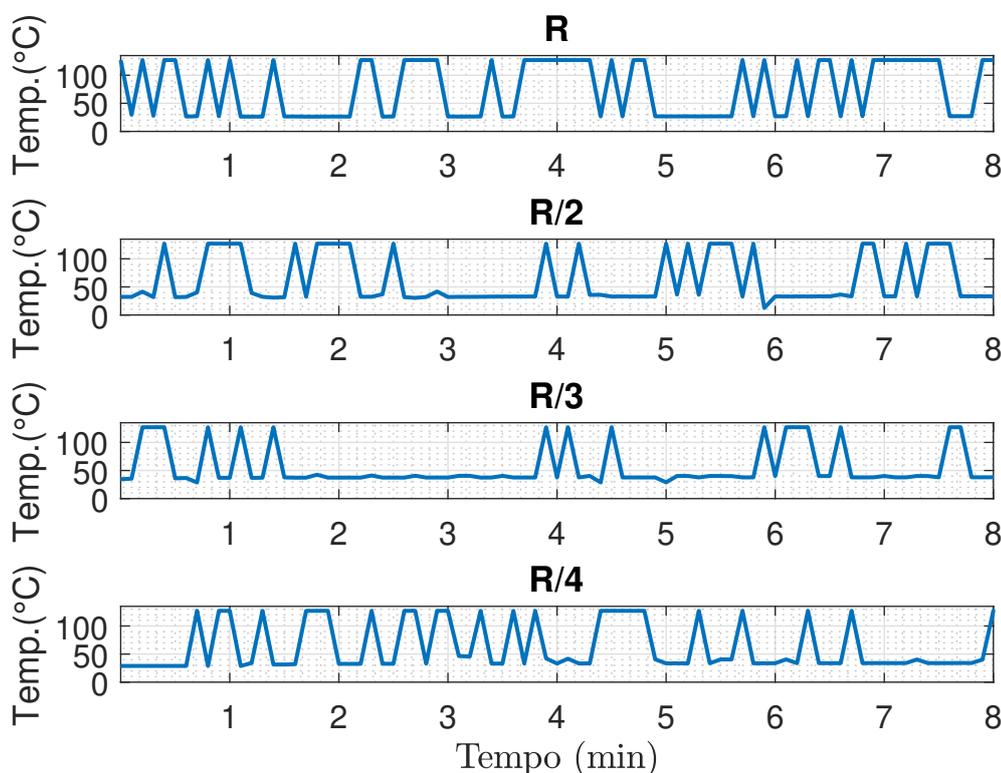


Figura 52 – Temperatura de saída do painel para diferentes resistores pullup

A figura 52 indica que diminuir a resistência equivalente para $R/2$ e para $R/3$ resultou em menor frequência de erro. Porém com $R/4$ a frequência voltou a ser alta. Talvez exista um valor do resistor pullup ótimo que consiga resolver o problema entre os valores $R/2$ e $R/4$. Porém, não se sabe se existe esse valor ótimo, e existindo, poderia demorar muito para se encontra-lo. Então foi decidido comprar novos resistores e usar um Arduino externamente para fazer a aquisição dos dados do painel solar. O bolsista de iniciação científica Henrique Roos ficou encarregado de refazer a instalação dos cabos.

¹ Um dos aspectos que podem influenciar a confiabilidade de grandes redes de sensores é o dimensionamento desse resistor [13].

C.2.2 Circuito de alimentação desligado

Além do ensaio trocando o resistor pullup foi feito um teste com o circuito que alimenta a bomba desligado, cujo resultado é mostrado na figura 53². É possível perceber que o cabo de força é o causador do problema pois não há um único valor de 127 (dispositivo desconectado). Além disso, o outro problema de mudança repentina ocorre com frequência muito menor.

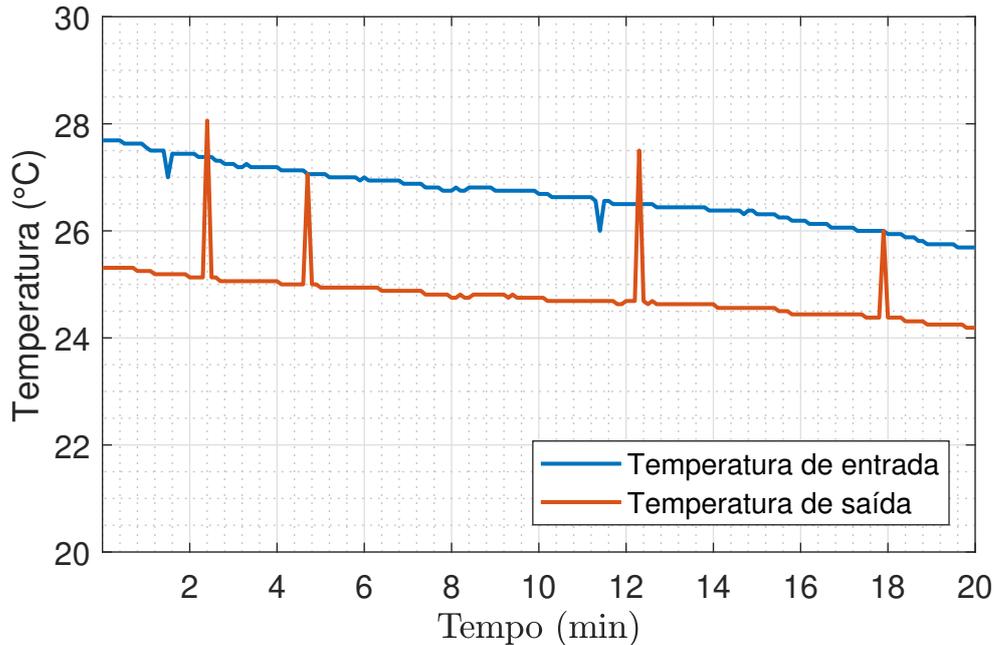


Figura 53 – Sinais sem o circuito de potência elétrica ligado

C.2.3 Resolução do problema

Enquanto os novos sensores estavam a caminho, foi feito um teste para tentar resolver o outro problema. O teste consistiu em trocar a forma como se acessava os valores de temperatura usando as bibliotecas OneWire e Dallas Temperature. Mais detalhes na subseção C.3.1.

C.3 Problema de mudança repentina de valor

Para melhor visualizar o problema de mudança repentina de valor, foi criada uma função no MATLAB para eliminar os valores 127 dos gráficos, substituindo-os por valores que interpolam os valores corretos. Mais detalhes sobre a implementação dessa rotina no apêndice D.

² Observação: A temperatura de entrada está maior que a de saída pois não há fluxo de água. O dia estava frio, e o sensor de temperatura da saída do painel estava na sombra.

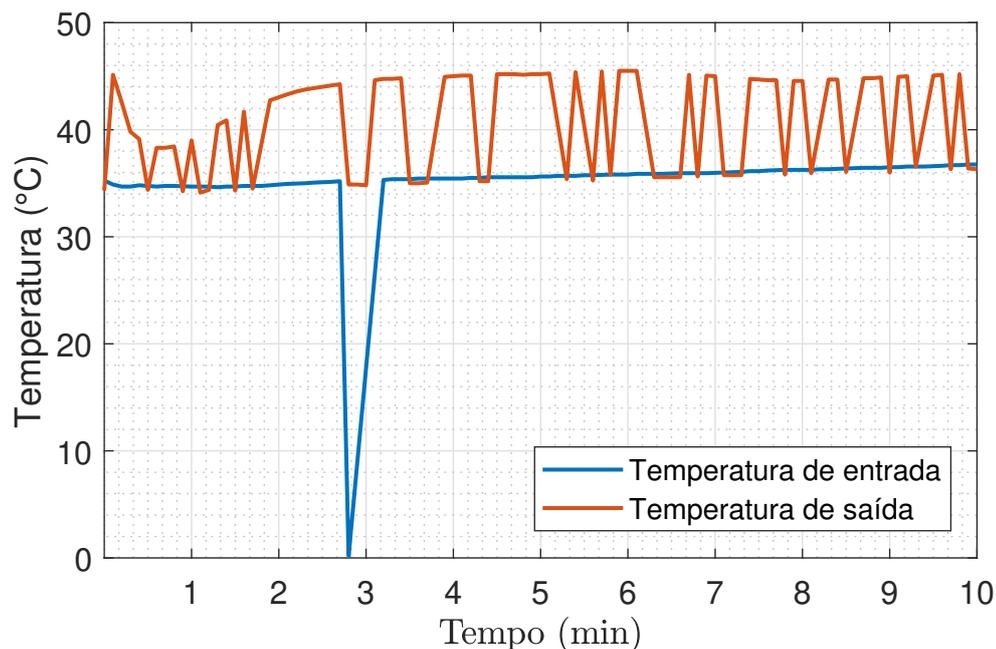


Figura 54 – Mesmos sinais da figura 50 com remoção de valores 127

Com ajuda do doutor Paulo Renato da Costa Mendes, do doutorando José Dolores Vergara Dietrich, do graduando Henrique Roos, do professor Julio Elias Normey Rico e do professor Rodolfo César Costa Flesch, foram levantadas algumas hipóteses sobre o que poderia estar causando essas medições frias na saída do painel solar.

Uma das hipóteses era de que o painel solar não estava aquecendo a água e portanto a temperatura fria seria o valor de medição correto. Então as medições erradas seriam os valores superiores e poderiam ser causados pela troca de algum bit durante o envio entre o sensor e o Arduíno nano.

Para verificar se realmente existia a possibilidade de o painel solar não estar sendo aquecido, foi feito o ensaio da Figura 55. Segue abaixo uma breve descrição do que foi feito nesse ensaio.

O sistema foi ligado normalmente. Por volta de 6 minutos é desligada a bomba, então, não há circulação de água. Pouco antes dos 8 minutos, liga-se a bomba e fecha-se a válvula de saída da do painel. Logo em seguida, troca-se a entrada de água (do tanque de água quente para a caixa d'água). Antes de completar 9 minutos, volta-se a usar a água do tanque de água quente.

No início, a água parada no painel está quente e quando começa a circular, a temperatura é medida pelo sensor. Depois, mantém-se constante por um tempo, e quando se fecha a válvula de saída, nota-se um rápido aumento de temperatura. Portanto conclui-se que o perfil superior da temperatura de saída do painel representa a medição correta, ou seja, o painel está aquecendo a água.

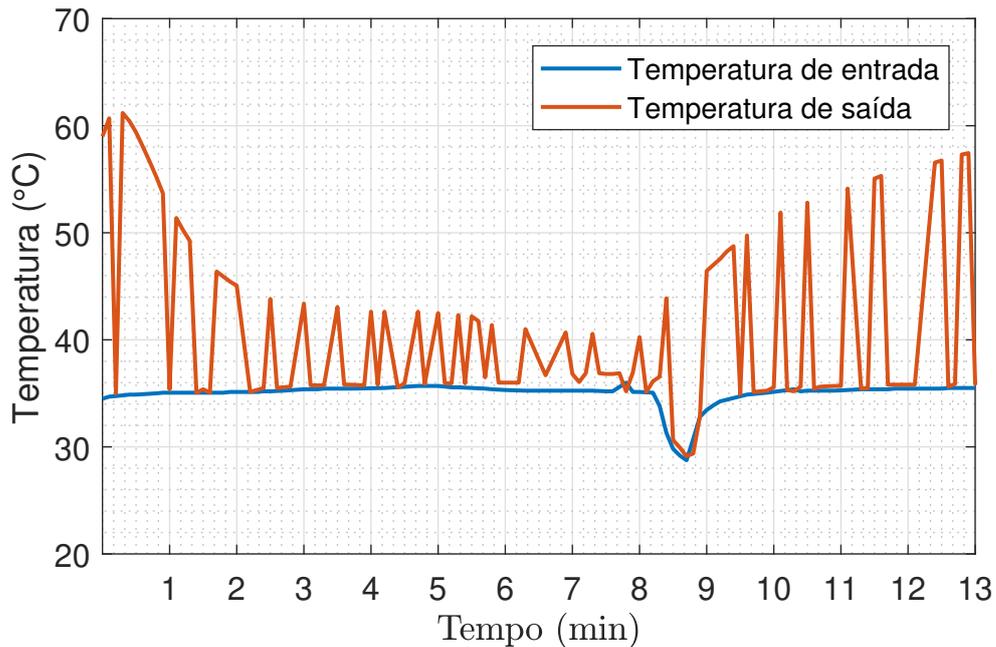


Figura 55 – Verificação da possibilidade de não aquecimento da água

Com esse mesmo gráfico percebe-se que a temperatura de saída do painel erra para valores muito parecidos com a temperatura de água fria. Logo a ideia de haver um único bit de erro está descartada. Porém essa hipótese abriu espaço para se pensar que os valores vindos dos sensores vêm em ordem aleatória. Dessa forma, não seria possível gravar, por exemplo, sempre o primeiro valor lido em uma variável específica.

C.3.1 Análise do código

Ao se analisar o código já implementado (usando as bibliotecas citadas na seção 2.1.4), constatou-se que a variável recebia o valor de temperatura baseado em um índice e não no endereço físico do sensor (ver a seção C.1), como no exemplo abaixo:

```
1 float temp_2 = sensors.getTempCByIndex(0); //Índice 0 ...
   corresponde a temperatura 2
```

Com isso abriu-se a hipótese de que quando um sensor não é reconhecido pela biblioteca (erro de -127) pode ser que o espaço ocupado por ele no índice seja substituído pelo sensor seguinte da ligação OneWire.

Dessa forma, o correto é obter o endereço de 64 bits de cada sensor e fixá-los como uma constante na inicialização do Arduino. Então a temperatura lida não seria mais pelo índice, mas pelo endereço de cada sensor.

Porém, fazer isso na planta seria muito trabalhoso, e poderia não resolver o problema,

já que haveria a necessidade de se abrir uma boa parte da instalação elétrica, separar cada sensor (estão todos ligados em OneWire), e obter o endereço de cada um separadamente usando uma rotina específica no Arduino.

Após a verificação de que quando o circuito de potência está desligado não há erro 127 nas medições (subseção C.2.2), foi decidido obter os endereços de cada sensor. Isso foi possível assumindo que a sequência dos índices estavam corretamente relacionadas à cada sensor da planta e que com o circuito causador dos problemas 127 desligado, não haveria nenhum sensor desconectado. Dessa forma não seria necessário abrir a instalação elétrica para se obter os endereços.

Índice	Sensor relacionado	Endereço
<i>Sensor0</i>	temp_2	{0x28, 0xFF, 0xAC, 0x53, 0x93, 0x16, 0x04, 0xE1}
<i>Sensor1</i>	temp_F	{0x28, 0xFF, 0x6C, 0x1A, 0x8B, 0x16, 0x03, 0x5C}
<i>Sensor2</i>	temp_Amb	{0x28, 0xFF, 0x52, 0x13, 0x8B, 0x16, 0x03, 0xE9}
<i>Sensor3</i>	temp_123	{0x28, 0xFF, 0x52, 0x13, 0x8B, 0x16, 0x03, 0xE9}
<i>Sensor4</i>	temp_3	{0x28, 0xFF, 0x31, 0x7B, 0x93, 0x16, 0x04, 0xD6}
<i>Sensor5</i>	temp_1	{0x28, 0xFF, 0x47, 0x53, 0x93, 0x16, 0x04, 0x4B}
<i>Sensor6</i>	temp_PS	{0x28, 0xFF, 0x17, 0xC3, 0x92, 0x16, 0x05, 0xCA}
<i>Sensor7</i>	temp_0	{0x28, 0xFF, 0x2F, 0xD5, 0x92, 0x16, 0x05, 0x41}

Tabela 6 – Endereços dos sensores

Com os endereços definidos na tabela C.3.1, foi feito um novo ensaio (figura 56) e implementado o seguinte exemplo de código:

```

1 DeviceAddress Sensor0 = { 0x28, 0xFF, 0xAC, 0x53, 0x93, ...
    0x16, 0x04, 0xE1 };
2 .
3 .
4 .
5 void loop() {
6     float temp_2 = sensors.getTempC(Sensor0); //2
7 }
```

Percebe-se que a mudança na forma como se acessa os valores de temperatura dos sensores acabou resolvendo o problema da temperatura igual a 127³. Ainda diminuiu a frequência da troca de valores entre as variáveis.

Portanto o problema de 127 foi resolvido ao se testar uma possível solução para a troca de valores entre as variáveis.

³ O 127 ocorre uma única vez nesse gráfico, apenas quando o disjuntor que aciona a bomba é ligado.

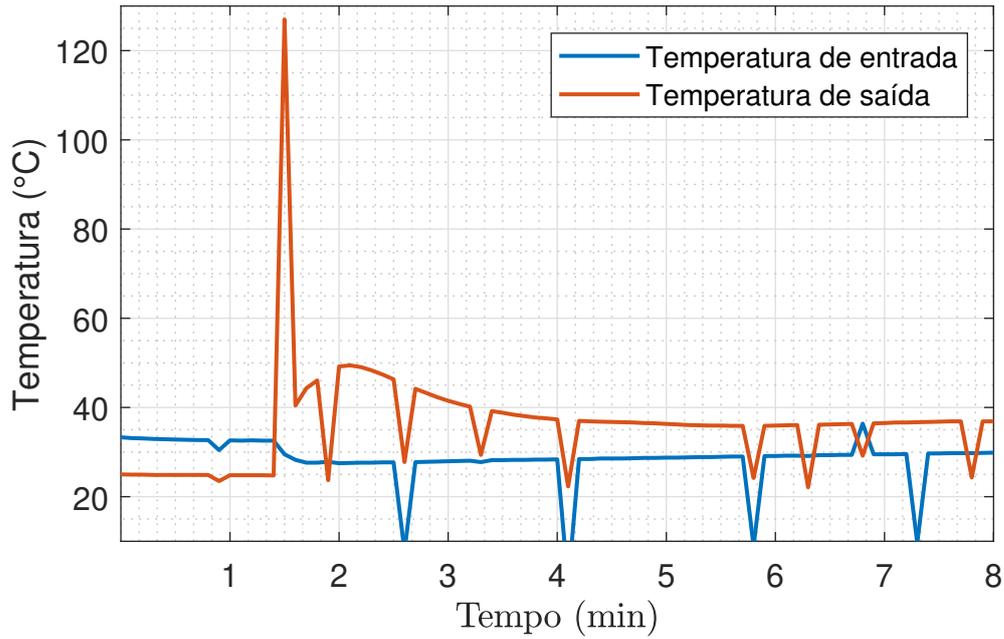


Figura 56 – Sinais depois de fixar os endereços dos sensores

C.3.2 Relação com a irradiação

Ao se analisar a variável de irradiância de ensaios anteriores, percebeu-se que também apresentava picos estranhos em 2540 (figura 57).

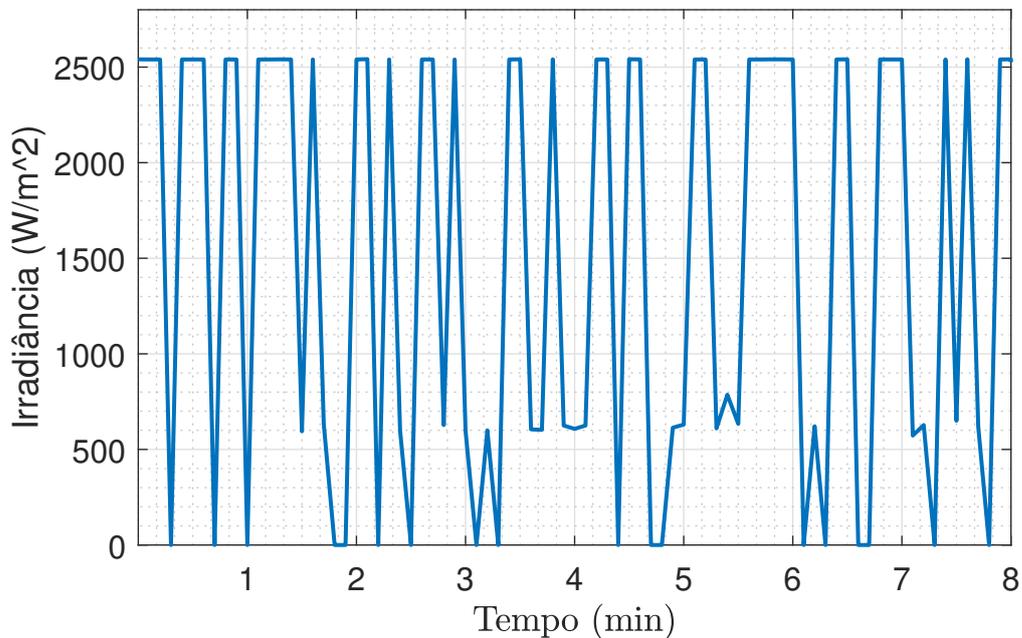


Figura 57 – Sinais de irradiação

Sabendo que o sinal de irradiação vindo do Arduino Nano ao Arduino Mega era multiplicado por 20 para se ter a correta unidade em W/m^2 , o graduando Henrique Roos intuitivamente dividiu o valor de 2540 por 20 e o resultado era exatamente 127. Com

essa informação pôde-se concluir que essa troca de valores entre as variáveis não estava associada à transmissão entre sensor e Arduino Nano, já que a informação de irradiância não chega pelo mesmo protocolo OneWire, e mesmo assim a irradiância eventualmente obtinha valores 127. Então a nova hipótese é de que o problema estaria entre o Arduino Nano e o Arduino Mega, seja por problemas na suas rotinas ou na comunicação.

Com a ajuda dos novos sensores e um Arduino Mega extra, foi possível implementar testes mais rápidos na bancada, onde se podia trabalhar em conjunto com o LabView no computador e simular aspectos de transmissão de dados pela saída serial⁴.

O primeiro teste na bancada levou em consideração um rotina do Arduino Nano que convertia os valores lidos pelos sensores em String para enviar pela saída serial. O problema nesse caso era a dimensão da string criada. Quando ocorria valores 127 (3 dígitos), os números em seguida se perdiam, o que resultava em troca de valores. Mas o problema não era só esse, ou seja, havia mais alguma causa para a troca de valores. Isso pode ser visto na figura 56, em que a ausência de valores 127 resultou em poucas trocas de valores entre as variáveis. A solução para esse problema foi simplesmente eliminar a rotina que convertia números em string, pois a função `Serial.Print()` do Arduino já faz isso muito bem.

A segunda causa da troca de valores foi descoberta ao se analisar a saída serial do Arduino Mega sem converter os valores recebidos do Arduino Nano, ou seja, foi feito o `Serial.Print()` na informação inteira que vinha do Arduino Nano. Isso mostrou que eventualmente, a informação chegava em partes.

Ao analisar-se essa rotina, notou-se a ausência de um caractere para sinalizar o fim de mensagem. Então a solução para mensagens problemáticas foi: somente atualizar as variáveis quando a mensagem completa chegar. Ou seja, nesse caso foi incluído um caractere "S" para sinalizar o fim.

C.4 Conclusões

Os dois problemas foram resolvidos, sendo que a causa do problema de valor 127 foi a interferência eletromagnética devido ao acionamento do circuito de potência e sua solução foi alterar a forma como a biblioteca Dallas Temperature lê os valores obtidos pelo OneWire⁵. Já o segundo problema teve 2 motivos: a rotina de conversão de número para string que tinha tamanho limitado para números float de 6 caracteres (incluindo ponto, casas decimais e sinal negativo) e era desnecessária, e o segundo motivo foi a eventual falha na entrega dos dados entre os 2 Arduínos.

⁴ Os detalhes dessas implementações não são importantes para esse documento

⁵ Acredita-se que a forma antiga de se obter os valores (pelo índice) diminuía de alguma forma o tempo do sinal lógico de cada bit na rede, e a interferência eletromagnética associada a esse menor tempo resultava em não identificação correta dos bits.

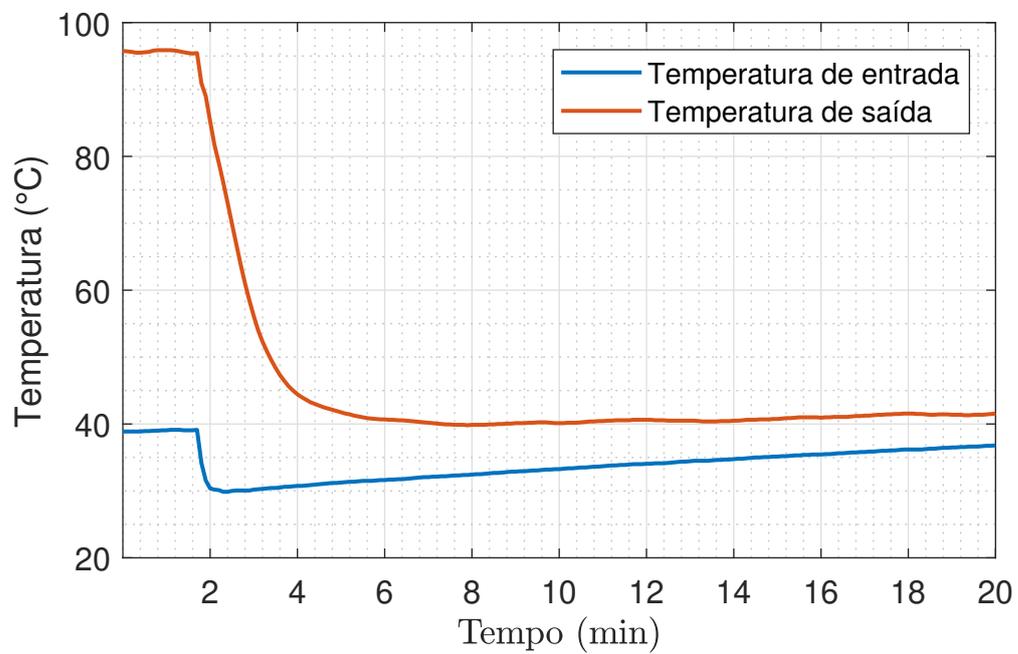


Figura 58 – Problemas resolvidos

APÊNDICE D – Rotina para eliminação de valores incorretos

Para remover os valores de 127 e melhorar a visualização do outro problema com as variáveis de temperatura, foi criada a seguinte rotina:

```

1 clear all; close all; clc;
2 NdeCorte = 6;
3 A = [2 2 3 3 3 7 7 9 9 9 5 5 5 8 8 8 4 4 4 4];
4 Ai = A;
5 [nl nc] = size(Ai);
6 for i=1:nc
7     if Ai(i)>NdeCorte
8         if i == 1
9             c=i;
10            while Ai(c)>NdeCorte
11                c=c+1;
12            end
13            for w=i:c-1
14                Ai(w) = Ai(c+1);
15            end
16        elseif i == nc
17            Ai(i) = Ai(i-1);
18        end
19        if i>1 && i<nc
20            c=i;
21            while Ai(c)>NdeCorte
22                if c>nc-1
23                    break
24                end
25                c = c+1;
26            end
27            incCorrecao = (Ai(c)-Ai(i-1))/(c-i+1);
28            for w=i:c-1
29                Ai(w) = Ai(w-1)+incCorrecao;
30            end
31        end
32    end
33 end
34 plot(Ai, 'r')
35 hold on
36 plot(A)

```

```

37 hold on
38 NdeCorte = NdeCorte*ones(1,nc);
39 plot(1:nc,NdeCorte,'g--')

```

Essa rotina, quando encontra um valor acima do valor máximo (valor indesejado), começa a contar quantos valores seguidos são indesejados, até encontrar um valor que esteja abaixo do valor máximo (valor desejado). Feito isso, calcula-se a diferença entre o novo valor desejado e o antigo valor desejado, isso representa um dos catetos de um triângulo retângulo. O outro cateto é formado pela quantidade de valores indesejados. Portanto, dividindo-se a diferença entre valores desejados pela quantidade de valores indesejados obtêm-se o quanto deve ser incrementado para cada posição de valor ruim.

$$Inc = \frac{V_{DF} - V_{DI}}{n_{IND}} \quad (D.1)$$

$$V(k) = V(k - 1) + Inc \quad (D.2)$$

Onde V é o valor corrigido, k começa no primeiro valor indesejado e termina no último valor indesejado. Inc é o incremento dado para cada valor corrigido, V_{DF} e V_{DI} são os valores desejado final e inicial, respectivamente e n_{IND} é o numero de valores indesejados.

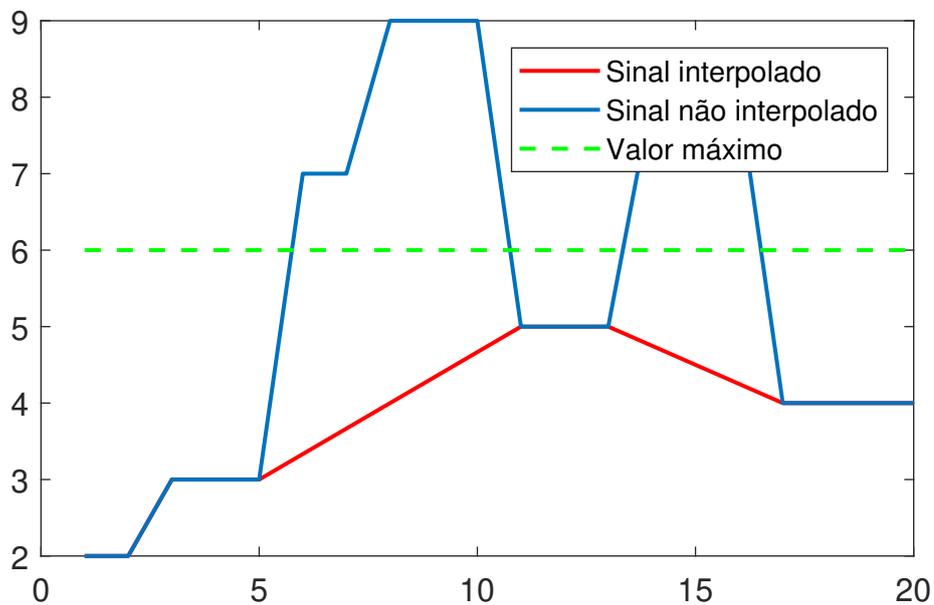


Figura 59 – Exemplo de sinal interpolado usando a rotina criada

APÊNDICE E – Implementação do sinal PRS

Abaixo encontra-se a rotina modificada a partir do código disponível no livro [9]. Essa modificação faz com que o sinal gerado também tenha amplitude aleatória.

```

1 function saida=prbsganhoavar(b,m,Ymax,Ymin);
2
3 % y=prbs(b,m)
4 % generates a PRBS signal with b bits each value is held during m ...
   sampling times.
5 % For b=8 the PRBS will not be an m-sequence.
6
7 % Luis A. Aguirre - BH 18/10/95
8 % - revision 01/02/1999
9
10 %Com amplitude variavel aleatoriamente - Criado por Lucas Peruzzo ...
   18/11/2017
11 reset(RandStream.getGlobalStream,sum(100*clock));
12 Num = 2^b - 1;
13 saida=zeros(1,Num);
14 bits=round(rand(1,b));
15
16 j=1; % for most cases the XOR of the last bit is with the
17 % one before the last. The exceptions are
18 if b==5
19     j=2;
20 elseif b==7
21     j=3;
22 elseif b==9
23     j=4;
24 elseif b==10
25     j=3;
26 elseif b==11
27     j=2;
28 end;
29 nptos = floor(Num/m);
30 sinal = (Ymax+Ymin)/2;
31 bit_final_ant = 0;
32
33 for i=1:nptos
34     hold_signal(m*(i-1)+1:m*i)=bits(b)*ones(1,m);    %yl = y logico

```

```
35     bits=[xor(hold_signal(m*(i-1)+1),bits(b-j)) bits(1:b-1)];
36
37     if bits(b)≠ bit_final_ant
38         Amp = normrnd(0.5,0.25) %distribuicao normal com media 0,5 ...
39             e desvio padrao 0,25
40
41         sinal = Amp*(Ymax-Ymin)+Ymin;
42
43         if sinal > Ymax
44             sinal = Ymax;
45         end
46         if sinal < Ymin
47             sinal = Ymin;
48         end
49
50         bit_final_ant = bits(b);
51         saida(m*(i-1)+1:m*i) = sinal;
52
53         %corrigir os ultimos m pontos de simulacao
54         if i==nptos
55             saida(nptos*m:Num)=saida(nptos*m);
56         end
57     end;
```