



RAPHAEL RIBEIRO CRUZ SANTOS

**Desenvolvimento de um Sistema Supervisório e  
Identificação Utilizando Redes Neurais Artificiais do  
Processo de Polimerização de Estireno**

CAMPINAS

2013





**UNIVERSIDADE ESTADUAL DE CAMPINAS  
FACULDADE DE ENGENHARIA QUÍMICA**

**RAPHAEL RIBEIRO CRUZ SANTOS**

**Desenvolvimento de um Sistema Supervisório e  
Identificação Utilizando Redes Neurais Artificiais do  
Processo de Polimerização de Estireno**

Orientador: Prof. Dr. Roger Josef Zemp

Dissertação de Mestrado apresentada ao Programa de  
Pós Graduação em Engenharia Química da Faculdade de  
Engenharia Química da Universidade Estadual de Campinas  
para obtenção do título de Mestre em Engenharia Química.

**ESTE EXEMPLAR CORRESPONDE À VERSÃO FINAL DA DISSERTAÇÃO  
DEFENDIDA PELO ALUNO RAPHAEL RIBEIRO CRUZ SANTOS E  
ORIENTADO PELO PROF. DR. ROGER JOSEF ZEMP**

A handwritten signature in black ink, appearing to be "R. Cruz Santos", written over a horizontal line.

**CAMPINAS**

**2013**

Ficha catalográfica  
Universidade Estadual de Campinas  
Biblioteca da Área de Engenharia e Arquitetura  
Rose Meire da Silva - CRB 8/5974

Santos, Raphael Ribeiro Cruz, 1987-  
Sa59d      Desenvolvimento de um sistema supervisorio e identificação utilizando redes neurais artificiais do processo de polimerização de estireno / Raphael Ribeiro Cruz Santos. – Campinas, SP : [s.n.], 2013.

Orientador: Roger Josef Zemp.  
Dissertação (mestrado) – Universidade Estadual de Campinas, Faculdade de Engenharia Química.

1. Controle de processos químicos - Automação. 2. Redes neurais artificiais. 3. Polimeização. I. Zemp, Roger Josef, 1962-. II. Universidade Estadual de Campinas. Faculdade de Engenharia Química. III. Título.

Informações para Biblioteca Digital

**Título em outro idioma:** Development of a supervisory system and identification using artificial neural networks for the styrene reaction process

**Palavras-chave em inglês:**

Chemical process control

Artificial neural networks

Polymerization

**Área de concentração:** Engenharia Química

**Titulação:** Mestre em Engenharia Química

**Banca examinadora:**

Roger Josef Zemp [Orientador]

Angel Pontin Garcia

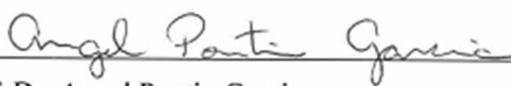
Flávio Vasconcelos da Silva

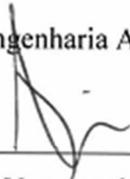
**Data de defesa:** 30-07-2013

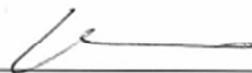
**Programa de Pós-Graduação:** Engenharia Química

## Folha de aprovação

Dissertação de Mestrado defendida por Raphael Ribeiro Cruz Santos e aprovada em 30 de julho de 2013 pela banca examinadora constituída pelos doutores:

  
\_\_\_\_\_  
Prof. Dr. Angel Pontin Garcia  
Faculdade de Engenharia Agrícola - Universidade Estadual de Campinas

  
\_\_\_\_\_  
Prof. Dr. Flávio Vasconcelos da Silva  
Faculdade de Engenharia Química - Universidade Estadual de Campinas

  
\_\_\_\_\_  
Prof. Dr. Roger Josef Zemp  
Faculdade de Engenharia Química - Universidade Estadual de Campinas



## ABSTRACT

This research work presents a methodology for the development of a supervisory system using Microsoft Excel<sup>®</sup>. The system was designed to operate a styrene polymerization plant. The development of computational codes in VBA (Visual Basic for Applications) allows for the creation of an OPC client for communication between computer and hardware. The spreadsheet has porting is able to receive, send, store and process information from the process. The polymerization of styrene is carried out using toluene as solvent and BPO (benzoyl peroxide) as an initiator. Acquisition of reactor temperature, inlet temperature and outlet shirt and specific mass of the reaction medium were performed. From the experimental data extracted from the polymerization reaction, an empirical model was generated using Artificial Neural Networks (RNA). RNAs are easily implemented in Excel simply, using operations arrays, making it appropriate for the development of predictive controllers. The use of Microsoft Excel<sup>®</sup> proved to be an interesting tool for application in automation experimental prototypes.

**Keywords:** chemical process control; artificial neural networks; polymerization



## RESUMO

Neste trabalho é apresentada uma metodologia de desenvolvimento de um sistema supervisorio utilizando Microsoft Excel<sup>®</sup>. O sistema foi desenvolvido para operar uma planta de polimerização em solução de estireno. A aplicação de códigos computacionais na em VBA (Visual Basic for Applications) na planilha torna possível a criação de um cliente OPC para comunicação entre o computador e *hardware*. A planilha possui portanto condições de receber, enviar, armazenar e tratar informações vindas do processo. A polimerização de estireno é realizada utilizando tolueno como solvente e BPO (Peroxido de Benzoíla) como iniciador. Aquisição da temperatura do reator, temperatura de entrada e saída da camisa e massa específica do meio reacional foram realizadas. A partir dos dados experimentais extraídos da reação de polimerização, foi gerado um modelo empírico utilizando Redes Neurais Artificiais (RNA). As RNAs são implementadas no Excel de forma simples, usando a própria planilha, por operar matrizes, tornando-se oportunas para o desenvolvimento de controladores preditivos. O Microsoft Excel<sup>®</sup> mostrou-se uma interessante ferramenta para aplicação em automação de protótipos experimentais.

**Palavras-chave:** controle de processos químicos – automação; redes neurais artificiais; polimerização



## SUMÁRIO

|   |    |
|---|----|
| 1. INTRODUÇÃO .....                               | 1  |
| 1.1. Motivação.....                               | 3  |
| 1.2. Objetivo.....                                | 4  |
| 1.3. Divisão da Dissertação .....                 | 5  |
| 2. REVISÃO DA LITERATURA.....                     | 7  |
| 2.1. Sistema Supervisório.....                    | 7  |
| 2.2. Identificação do Sistema .....               | 8  |
| 3. O SISTEMA SUPERVISÓRIO .....                   | 13 |
| 3.1. Protótipo experimental .....                 | 13 |
| 3.1.1. Modificação do Protótipo Experimental..... | 16 |
| 3.2. Variáveis de Processo.....                   | 17 |
| 3.3. Procedimento Experimental .....              | 19 |
| 3.4. Sistema de Comunicação .....                 | 19 |
| 3.4.1. Ferramenta Computacional.....              | 20 |
| 3.4.2. Programação CLP.....                       | 21 |
| 3.4.3. Comunicação OPC .....                      | 24 |
| 3.4.4. Planilha Eletrônica.....                   | 25 |
| 3.4.5. Sistema de Monitoramento e Controle .....  | 28 |
| 3.4.6. Algoritmo do sistema supervisório.....     | 30 |
| 3.5. Condições de Operação.....                   | 32 |
| 3.6. Reação de Polimerização .....                | 33 |
| 4. IDENTIFICAÇÃO DE SISTEMA.....                  | 35 |
| 4.1. Monitoramento do processo .....              | 35 |
| 4.2. Tratamento dos dados.....                    | 39 |

|   |    |
|---|----|
| 4.2.1. Escolha do Tempo de Amostragem.....        | 40 |
| 4.2.2. Variáveis Significativas.....              | 42 |
| 4.3. Rede Neural Artificial.....                  | 43 |
| 4.3.1. Treinamento da Rede Neural Artificial..... | 44 |
| 5. CONCLUSÕES.....                                | 47 |
| 5.1. Sugestão para Trabalhos Futuros.....         | 48 |
| REFERÊNCIAS.....                                  | 49 |
| APÊNDICE I.....                                   | 51 |
| APÊNDICE II.....                                  | 57 |

## LISTA DE FIGURAS

|   |    |
|---|----|
| <i>Figura 3.1 - Esquema do protótipo experimental.</i>  | 14 |
| <i>Figura 3.2 – Imagens do protótipo experimental</i>   | 15 |
| <i>Figura 3.3 - Interface do software SPDSW e código de conversão de unidades.</i>                    | 23 |
| <i>Figura 3.4 - Interface do driver HS1, da HI Tecnologia para comunicação OPC.</i>                   | 25 |
| <i>Figura 3.5 - Planilha Eletrônica para cliente OPC da empresa Takebishi.</i>                        | 27 |
| <i>Figura 3.6 - Programa sendo executado no Microsoft Excel<sup>®</sup>.</i>                          | 29 |
| <i>Figura 3.7 - Painel da planilha em Excel para monitoramento e manipulação das variáveis.</i>       | 30 |
| <i>Figura 3.8 - Fluxograma do algoritmo desenvolvido para aquisição e manipulação da planta.</i>      | 31 |
| <i>Figura 3.9 - Comportamento da temperatura do reator (TR) durante a reação de polimerização.</i>    | 34 |
| <i>Figura 3.10 - Perfil da massa específica do meio reacional durante a reação química.</i>           | 34 |
| <i>Figura 4.1 - Temperatura de Entrada (Te) durante a reação de polimerização.</i>                    | 36 |
| <i>Figura 4.2 - Perfil da potência (P) reação com valor de Te constante</i>                           | 36 |
| <i>Figura 4.3 - Perfil da temperatura do reator (Tr) reação com valor de Te constante.</i>            | 37 |
| <i>Figura 4.4 - Perfil da temperatura do reator (Tr) com controle manual.</i>                         | 38 |
| <i>Figura 4.5 - Temperatura de Entrada (Te) durante a reação de polimerização.</i>                    | 38 |
| <i>Figura 4.6 - Perfil da potência dissipada da resistência.</i>                                      | 39 |
| <i>Figura 4.7 - Auto-Covariância Cruzada de TR para os 4 conjunto de dados.</i>                       | 41 |
| <i>Figura 4.8 - Correlação Cruzada das variáveis de processo com TR.</i>                              | 42 |
| <i>Figura 4.9 - Topologia da RNA utilizada, com 3 entradas, 8 neurônios intermediários e 1 saída.</i> | 45 |
| <i>Figura 4.10 - Regressão Linear para validação do modelo neural de TR.</i>                          | 45 |



## LISTA DE TABELAS

|  |           |
|--|-----------|
| <i>Tabela 3.1 - Relação dos transmissores e da nomenclatura utilizada das variáveis. ....</i>                            | <i>18</i> |
| <i>Tabela 3.2 - Conversão de unidades no CLP de Bits para unidades de engenharia.....</i>                                | <i>22</i> |
| <i>Tabela 3.3 - Valores de Memória no SPDSW utilizadas e coeficientes da função linear das variáveis de entrada.....</i> | <i>23</i> |
| <i>Tabela 3.4 - Valores de Memória no SPDSW utilizadas e coeficientes da função linear das variáveis de saída. ....</i>  | <i>24</i> |
| <i>Tabela 3.5 - Principais parâmetros a serem configurados no driver HS1.....</i>  | <i>25</i> |
| <i>Tabela 3.6 - Condições de processo dos experimentos realizados. ....</i>  | <i>33</i> |

# 1. INTRODUÇÃO

No desenvolvimento de novos processos industriais é comum a construção de plantas pilotos visando o estudo de produção em escalas maiores que a de bancada em laboratório. A partir da planta piloto é possível realizar novos testes em uma escala intermediária e assim obter informações de possíveis limitações do processo em grande escala.

Com a realização de testes em escala piloto surge a necessidade de manter o controle de correntes, adicionar e remover energia do sistema e manter parâmetros constantes. Condições realizadas em processo, através de aquisição e tratamento de dados de forma automática. Além do mais, é preciso realizar análises e medidas em linha de processo, e para isso é importante que a planta seja instrumentada e automatizada.

Dessa forma, se faz necessário para a realização de testes em escala piloto que o protótipo experimental seja instrumentado e possua um sistema supervisório adequado para o processo. Com instrumentação e automação é possível obter melhores resultados de desempenho do processo, como também fazer análises e estudos do processo com maior confiança.

Porém a automação da planta piloto pode ser vista como um problema devido aos custos envolvidos no desenvolvimento, montagem e operação. No que se refere a programas para supervisão e operação, softwares comerciais específicos possuem considerável custo, envolvendo a necessidade de serviço de empresas específicas para instrumentação e automação da planta, e treinamento do recurso humano. Além disso, por ser específico para o processo, o próprio usuário não possui conhecimento de como o sistema é construído e que rotinas são executadas, confiando apenas nos fabricantes do software adquirido.

Outra solução, nesses casos, é a compra de softwares de desenvolvimento, utilizados para criar programas de comunicação e interfaces gráficas. Tornando-se possível elaborar e construir um supervisório para operar a planta. Todavia a utilização desse tipo de software cria a necessidade de usuários com conhecimento em desenvolvimento de programas, acarretando a necessidade de treinamento dos usuários.

Um ponto em comum nesses dois tipos de softwares é a utilização dos mesmos para aplicação de técnicas de controle avançado em processos químicos. A utilização dessas técnicas na academia, em projetos de pesquisas, é feita em maior número em relação a quantidade de aplicações em processos industriais. Assim, muitos softwares comerciais para operação de equipamento não possuem ferramentas disponíveis para desenvolvimento de controle avançado, nesse caso, sendo uma limitação.

Como alternativa a utilização de softwares comerciais específicos para determinado processo, devido aos problemas com custos que envolvem treinamento e aplicação de controle avançado, é iniciado estudo da aplicação do *Microsoft Excel*<sup>®</sup> como sistema de supervisão de operação. Embora que o *Microsoft Excel*<sup>®</sup> não seja um software desenvolvido com o objetivo de realizar operações em modo dinâmico, como por exemplo realizar aquisição de dados no tempo, e realizar comunicação com equipamentos externos, possui alta capacidade no processamento e armazenamento de dados.

A aplicação de identificação de sistemas é bastante comum nas indústrias e várias pesquisas são desenvolvidas a fim de tornar os modelos empíricos robustos. Sendo o modelo desenvolvido com base nos dados do processo, a qualidade do mesmo depende principalmente da qualidade dos dados utilizados para seu desenvolvimento. Isso gera a necessidade de sensores de qualidade, tempo de amostragem que represente bem o processo e dados com pouco ruído.

Existem diversos tipos de modelos empíricos, normalmente são lineares, baseado em funções estatísticas. Por outro lado, existem as ferramentas baseadas em inteligência artificial capazes de modelar um sistema de modo empírico. Das principais técnicas, se destaca o uso de redes neurais artificiais (RNA, do inglês *Artificial Neural Networks*).

A técnica de RNA foi desenvolvida com a tentativa de simular o funcionamento do cérebro e sistema sensorial biológico, o qual transmite um estímulo (sinal), vindo dos órgãos dos sentidos (sensor), através dos vários neurônios fazendo com que a informação chegue ao cérebro (saída) de forma a ser interpretado. RNA são bastante utilizadas no desenvolvimento de modelos devido sua capacidade de generalização e por superar problemas com falhas nos dados. Além disso, apresentam bom desempenho na modelagem de processos que possuem alto grau de não linearidade.

A vantagem da modelagem empírica de processos é que não se faz necessário desenvolvimento de equações matemáticas, partindo das leis de conservação de energia, cinética química e termodinâmica. Dessa forma, simplifica-se a modelagem, quando processos envolvem um elevado número de variáveis e propriedades de compostos.

Entre os exemplos de operações que apresentam dinâmica não linear podem ser citados os reatores em batelada. Reatores em batelada são bastante utilizados na indústria de polímeros, principalmente de maneira multipropósito, onde são utilizados para produção de diferentes tipos de polímeros, a depender da demanda.

A síntese de polímeros pode ser realizada realizando a adição ou não de solvente ou compostos emulsificante. A polimerização em solução é bastante utilizada, pois é adicionado solvente ao monômero, evitando problemas com alta viscosidade e má distribuição das cadeias poliméricas, já que o meio torna-se menos viscoso, ou seja, com um maior coeficiente de difusão.

No Laboratório de Engenharia de Sistemas Químicos (LESQ) da Faculdade de Engenharia Química, existe uma planta piloto de polimerização de estireno. A planta foi construída com o objetivo de estudos de controle avançado utilizando a conhecimento da área de inteligência artificial (IA) lógica *fuzzy*, utilizando MATLAB. O poliestireno é um polímero bastante produzido de diferentes formas, onde a forma mais utilizada é do poliestireno expandido, mais conhecido como Isopor nome do produto produzido pela empresa BASF.

### 1.1. MOTIVAÇÃO

Importantes propriedades dos produtos poliméricos, como sua fácil aplicação, flexibilidade, leveza, baixo custo e seu fácil processamento resultaram no crescimento da demanda desses produtos. Contudo, processos de polimerização envolvem complexos mecanismos de reação e a natureza da dinâmica de processo altamente não linear (HOSEN, HUSSAIN e MJALLI, 2011).

Nesses casos, modelos lineares não são adequados para descrever a dinâmica do processo e modelos não lineares devem ser utilizado. Além disso, os processos atuais necessitam de condições de operação mais restritas, já que mais restrições são impostas,

resultantes, por exemplo, de considerações de ordem ambiental e de segurança. Isso motiva a utilização de controles preditivos robustos que se adequam a qualquer condição operacional, conseqüentemente, um modelo robusto (FINDEISEN, 2002).

Uma possível solução para melhores manter as melhores condições de operação é a utilização de controle preditivo, que segundo Biyanto, Widjiantoro e Jabal (2011), tem sido um dos métodos de controle avançado mais utilizado na indústria, principalmente em controle de processos que são multivariáveis e instáveis.

Visto que a polimerização é um processo complexo e apresenta alto grau de não linearidade, segundo Chen e Billings (1992), as RNAs possuem como importante característica a capacidade de serem ajustadas como modelo não linear. Torna-se interessante a utilização de Redes Neurais Artificiais como alternativa e tentativa de utilizá-la como modelo na identificação do sistema.

Com o foco em ambiente industrial, uma limitação do uso de métodos avançados se mantém no custo e praticidade do software utilizado. Geralmente, a indústria se vê presa a códigos computacionais complexos e softwares comerciais caros, e torna-se de interesse a busca por softwares mais difundidos no meio computacional, como o Microsoft® Excel® (FUJIKI, 2009).

## 1.2. OBJETIVO

O estudo visa desenvolver um sistema de supervisão no Microsoft Excel® para o processo de polimerização de estireno, com o objetivo de operar o protótipo e retirar dados para desenvolvimento de um modelo empírico. O modelo do processo será obtido por meio da técnica de inteligência artificial Redes Neurais Artificiais, utilizadas para desenvolvimento de modelos de processos complexos.

O principal foco do trabalho é criar um sistema supervisorio utilizando o Excel®, assim como Fujiki (2009) e Santos (2011) que desenvolveram técnicas de controle utilizando a planilha eletrônica, demonstrando uma alternativa de fácil implementação devido a sua praticidade e disponibilidade do software.

Com intuito de desenvolver um sistema de supervisão, operar e extrair dados do processo para modelagem do mesmo com RNA, foi utilizado uma protótipo para

polimerização de estireno à poliestireno cristal em batelada do. Laboratório de Engenharia de Sistemas Químicos (LESQ), na Faculdade de Engenharia Química, UNICAMP.

### 1.3. DIVISÃO DA DISSERTAÇÃO

O trabalho está dividido em 2 principais partes, que é o desenvolvimento do supervisor no Excel<sup>®</sup> e a modelagem empírica do processo.

Dessa forma, o capítulo 2 trata da revisão de literatura, que visa dá um contexto mais abrangente de onde o trabalho está posicionado, analisando publicações relevantes relacionadas ao tema e explicar o porque de escolhas durante a realização do trabalho.

No capítulo 3 é apresentado o procedimento de desenvolvimento do sistema supervisor no Excel, utilizando Visual Basic, comunicação OPC e um CLP com programação em *ladder*. Também, o protótipo experimental para polimerização de estireno e seu funcionamento, e o procedimento experimental utilizado.

No capítulo 4 é apresentada a aplicação de Redes Neurais Artificiais como modelo para identificação do sistema, uma vez que são discutidos os dados obtidos do processo, utilizando o supervisor desenvolvido. Também, tratamento de dados utilizado para melhor qualidade do modelo.

A dissertação é encerrada com o capítulo 5, onde são apresentadas as conclusões obtidas a partir dos resultados apresentados nos capítulos 3 e 4 e sugestão para trabalhos futuros.



## 2. REVISÃO DA LITERATURA

A revisão de literatura foi realizada com dois enfoques, em busca de meios para o desenvolvimento de um sistema de comunicação e supervisorio utilizando Excel e a utilização de Redes Neurais Artificiais (RNA) como modelo empírico para identificação do sistema de polimerização.

### 2.1. SISTEMA SUPERVISÓRIO

Nesta seção é realizada uma revisão dos trabalhos utilizados como referência para motivação do desenvolvimento de um sistema supervisorio utilizando o Microsoft Excel<sup>®</sup>, visando a praticidade e popularidade da ferramenta. Na literatura é comum encontrar trabalhos realizados utilizando planilhas eletrônicas na área de automação, para pequenos protótipos de finalidade didática para ensino de técnicas de controle.

Fujiki (2009) estudou a aplicação de controle neural em um processo de precipitação de bromelina, visando o controle da temperatura do meio precipitante. O supervisorio foi desenvolvido utilizando Indusoft WebStudio<sup>®</sup>, porém os cálculos realizados para o modelo neural e otimização para minimização do erro do controlador, feitos em Excel, numa planilha eletrônica, utilizando códigos desenvolvidos em *Visual Basic*, para que o programa fosse executado a cada tempo de amostragem.

A comunicação entre o Indusoft WebStudio<sup>®</sup> e a planilha eletrônica foi realizada por ODBC (*Open Database Connectivity*). A grande vantagem apontada pela autora, para o uso da planilha eletrônica é tornar o desenvolvimento mais fácil e dinâmico, devido a familiaridade com o do usuário com o *software*. Demonstrando também, a capacidade do Excel, para simulação de Redes Neurais Artificiais (RNA) e da aplicação do Solver para otimização em tempo real, além da interação com softwares externos de automação.

Santos (2011) utilizou uma planilha eletrônica para controlar o processo de polimerização de estireno, utilizando o protótipo experimental construído por Leite (2011). Nesse caso, o autor implementou a comunicação do hardware com o Excel através do Indusoft, utilizando para comunicação do Excel com Indusoft WebStudio<sup>®</sup>, o protocolo de comunicação DDE (*Dynamic Data Exchanger*).

Nesse trabalho, a planilha eletrônica serviu como um sistema de supervisão e manipulação das variáveis, executando o *Solver*. O objetivo de desenvolver um controlador MPC foi obtido, porém existiram duas principais limitações, dado a utilização da planilha não foi possível implementar o controle tivesse um número maior que 1 de horizonte de controle; e utilizar um *software* comercial de maior custo para comunicação do Excel com o hardware.

No trabalho de Aliane (2010) é mostrada uma metodologia para o desenvolvimento de um sistema supervisorio utilizando o Excel mais aplicações em *Visual Basic*. Oferecendo *add-in* para supervisão e aquisição de dados e controle de processos. O primeiro, visando identificação de sistema e o segundo, para ajudar no ensino da disciplina de controle de processos. Dois equipamentos foram utilizados para demonstrar as ferramentas, de controle de nível de água e controle de temperatura em um tanque.

O trabalho do autor motivou a utilização do Excel, como ferramenta confiável e prática na utilização em automação de processos, visando ensino de disciplinas relacionadas. As ferramentas de trabalho obtiveram um interessante desempenho, sendo possível realizar a aquisição de dados em um tempo de amostragem 200 amostragens por segundo.

A partir dessas informações, optou-se por estender a utilização do Excel para processos químicos, não só como ferramenta de cálculo mas como um sistema supervisorio. Ultrapassando as limitações encontradas quanto à comunicação com o hardware.

A possibilidade de transformar uma planilha eletrônica em cliente OPC, permite que a mesma se comunique com um servidor OPC no computador. A disponibilidade de um ambiente gráfico, torna possível o desenvolvimento de uma interface para o operador monitorar o processo. Em vista da utilização de modelos para simulação, torna-se promissor a aplicação do Excel para simular um processo químico.

## 2.2. IDENTIFICAÇÃO DO SISTEMA

Nesta seção é abordado trabalhos referentes a identificação de sistemas e sistemas de polimerização utilizando a técnica de inteligência artificial Redes Neurais Artificiais. Influenciado pela necessidade de obter um modelo do processo, que seja possível executar

no Excel e aplicar técnicas de controle avançadas, como o controle preditivo baseado em modelo.

A aplicação de identificação de sistemas é bastante comum nas indústrias e várias pesquisas são desenvolvidas a fim de tornar os modelos empíricos robustos. Sendo o modelo desenvolvido com base nos dados do processo, a qualidade do mesmo depende principalmente da qualidade dos dados utilizados para seu desenvolvimento. Isso gera a necessidade de sensores de qualidade, tempo de amostragem que represente bem o processo e dados com pouco ruído.

A técnica de RNA foi desenvolvida com a tentativa de simular o funcionamento do cérebro e sistema sensorial biológico, o qual transmite um estímulo (sinal), vindo dos órgãos dos sentidos (sensor), através dos vários neurônios fazendo com que a informação chegue ao cérebro (saída) de forma a ser interpretado. RNA são bastante utilizadas no desenvolvimento de modelos devido sua capacidade de generalização e por superar problemas com falhas nos dados. Além disso, apresentam bom desempenho na modelagem de processos que possuem alto grau de não linearidade.

A vantagem da modelagem empírica de processos é que não se faz necessário desenvolvimento de equações matemáticas, partindo das leis de conservação de energia, cinética química e termodinâmica. Dessa forma, simplifica-se a modelagem, quando processos envolvem um elevado número de variáveis e propriedades de compostos.

Entre os exemplos de operações que apresentam dinâmica não linear podem ser citados os reatores em batelada. Em reatores em batelada, ao decorrer da reação, as propriedades do meio reacional são alteradas com o tempo, dessa forma, as condições de operação devem ser alteradas também até o término da reação, com a finalidade de manter a especificação do produto e segurança na operação.

Segundo, Hosen, Hussain e Mjalli (2011) reatores em batelada são bastante utilizados na indústria de polímeros, principalmente de maneira multipropósito, onde são utilizados para produção de diferentes tipos de polímeros, a depender da demanda. No caso de reatores de polimerização, devido a complexidade da cinética envolvida na reação, e pela mesma ser conduzida geralmente em batelada, um mesmo reator é utilizado para diferentes tipos de polímeros, o que torna bastante árdua a tarefa de desenvolver modelos fenomenológicos para os diferentes produtos que se deseja produzir

Vários trabalhos foram conduzidos para identificação de sistema, ou seja, desenvolvimento de modelos empíricos, baseado nos dados do processo. Da necessidade de modelos robustos, foram realizadas pesquisas buscando identificar o sistema utilizando redes neurais artificiais (RNA). Para reatores em batelada, torna-se uma importante ferramenta, devido a capacidade de generalização, mesmo com falhas nos dados do processo.

No trabalho de Chen e Billings (1992), os autores abordam a utilização das RNAs para identificação de sistema como uma poderosa ferramenta para sistemas não lineares. No trabalho, é discutida três principais tipos de arquitetura de RNAs, as *perceptron* multicamada, de função de base radial e rede de ligação funcional. Também, algoritmos de aprendizado e algumas vantagens e desvantagens das mesmas, para o caso de identificação de sistemas discretos.

Os autores mostram como os modelos neurais devem servir bem para o caso de modelagem de processos. Apresentando que muitas arquiteturas de rede *feedforward* têm sido consideradas como modelos não lineares e demonstram boas capacidades para representar sistemas complexos. Além de toda abordagem matemática sobre os métodos utilizados para essas ferramentas.

Em linhas gerais, as clássicas RNAs, chamadas *perceptron* multicamadas mostra-se como um modelo universal para sistemas não lineares. Que podem ser implementadas como modelos diretos ou recursivos. Porém, a desvantagem desse tipo de arquitetura é a dificuldade para minimização da função erro, e atingir o melhor resultado durante o treinamento.

O trabalho de Kuroda e Kim (2002), teve como objetivo a aplicação de Redes Neurais Artificiais para modelagem do balanço de energia do processo de polimerização de acrílico em emulsão. Aponta a importância e dificuldade de controlar a temperatura do reator, pois, influencia totalmente na distribuição uniforme das cadeias poliméricas e a que a reação é exotérmica, elevando a temperatura. No caso, as RNAs foram utilizadas devido a simplicidade para formulação do modelo diante da busca por parâmetros cinéticos e termodinâmicos da reação.

Nesse estudo foi demonstrado que as alterações de temperatura provocadas por reações exotérmicas poderiam ser facilmente calculada e preditos pelas RNAs. Obtiveram que essa ferramenta pode ser aplicada como ferramentas úteis para controle de processos de polimerização. Além disso, verificou-se que os números entradas afeta a eficiência em de treinamento, como também, o número de dados.

A utilização de RNA em diversas áreas também cresce com o avanço da capacidade computacional, em diversas áreas como reconhecimento de padrão e ajuste de função, tornando-se mais robustas, com melhorias em métodos de treinamento, e diferentes maneiras de desenvolvimento gerando a criação de vários tipos de RNA. Procura-se cada vez mais, aplicar RNAs para desenvolvimento de modelos dinâmicos, o que se cria uma nova linha de redes, chamadas de recursivas, as quais utilizam informação de saída, e das variáveis de entrada em instantes passados como entrada da rede.

Prasad e Bequette (2003) demonstram aplicação de RNAs para modelagem de diferentes tipos de processos, utilizando a simulação de um biorreator, duas colunas de destilação e um reator de polimerização de metacrilato. Nesse estudo, os autores exploram a ideia de diminuir o número de parâmetros (nós) utilizando a técnica SVD, com o objetivo de retirar os parâmetros redundantes da RNA. No trabalho, os autores mostram como o erro médio quadrado (MSE) diminui após essa abordagem.

O trabalho dos autores mostra a importância do tratamento dos dados (informação) para que o modelo seja robusto para um processo. Deste modo, diferentemente do que vem sendo aceito de outros trabalhos, as RNAs possuem capacidade de generalização, mas assim como um pequeno número de nós pode não ser o suficiente para modelar o sistema, um número grande de nós pode não ser capaz de se ajustar ao processo.

Kashani e Shahhosseini (2010) desenvolve uma metodologia para treinamento de RNAs em reatores bateladas. O problema apontado em reatores bateladas é o número de dados necessário para treinar a RNA, que tem como consequência a necessidade de um elevado número de bateladas. Porém, a metodologia é exaustiva, sendo necessário utilizar diferentes bateladas para vários treinamentos, a fim de encontrar o melhor modelo para o sistema.

Mais recentemente, Hosen, Hussain e Mjalli (2011) aplicou a técnica de RNA para modelagem de um processo de polimerização de estireno. Devido a dificuldade em

determinar os parâmetros cinéticos de reação. Então, a RNA foi utilizada para estimar parâmetros térmicos para o balanço da energia. Dessa forma, utilizou o conhecimento fenomenológico do sistema, estimando parâmetros, o que vem a ser um modelo caixa-cinza.

Nesse trabalho, os autores utilizaram o modelo como estimador para um controlador baseado em modelo. O controlador foi utilizado de modo a seguir um perfil ótimo, e comparado ao controle do processo utilizando PID. Os resultados foram satisfatórios, mostrando que o controlador MPC possui melhor desempenho em relação a conversão de monômero

Os trabalhos referenciados neste trabalho foram os mais relevantes para este trabalho, por justificar a utilização das RNAs e possuir aplicações em reatores bateladas, principalmente, de polimerização de estireno. O Microsoft Excel<sup>®</sup> é uma ferramenta recomendada e possui vantagens, na aplicação de criação de um sistema supervisor e a técnica de RNA gera modelos não lineares robustos, a partir dos dados do processo, dispensando a utilização de informações fenomenológicas. Por tanto, a partir dessas informações, optou-se por utilizar essas ferramentas.

### 3. O SISTEMA SUPERVISÓRIO

Neste capítulo são apresentados os meios utilizados para atingir o objetivo proposto. É feita a descrição do protótipo experimental e da modificação realizada no mesmo, utilizado na reação de polimerização. A descrição aborda a instrumentação e dispositivos utilizados para manipulação de variáveis durante a reação. O desenvolvimento do sistema para monitoramento e manipulação das variáveis de processo é mostrado com destaque em cada etapa utilizada para a criação da ferramenta desenvolvida em uma planilha do Microsoft Excel<sup>®</sup>, utilizando códigos em *Visual Basic for Applications* (VBA). Por último, é descrito o procedimento experimental adotado para realização da reação e obtenção dos resultados de reação.

#### 3.1. PROTÓTIPO EXPERIMENTAL

Nesse tópico são apresentadas informações detalhadas sobre o protótipo experimental, necessárias para a compreensão do sistema de monitoramento e manipulação de variáveis do sistema desenvolvido e do procedimento experimental.

O protótipo experimental construído por Leite (2011) pode ser dividido em duas partes:

- Mecânica: consiste no reator encamisado, tanque de armazenamento do fluido térmico, bombas de engrenagem, agitadores, cilindro de nitrogênio (N<sub>2</sub>).
- Elétrica: consistem nos sensores de temperatura, massa específica e vazão mássica; nos dispositivos no painel de controle, inversores de frequência, variador de potência e um controlador lógico programável (CLP) e resistor.

A estrutura do equipamento consiste em um reator de aço inox com capacidade de 2,3 L. O reator é encamisado, o que permite a circulação de fluido térmico na camisa do reator, regulando a temperatura interna do mesmo durante a reação. O meio reacional é bombeado pela bomba (BA-01) por uma linha de amostragem, a qual passa por sensores de densidade, vazão mássica e diferença de pressão, e retorna para o reator por um orifício na tampa, como mostra a Figura 3.1. A tampa do reator também possui orifícios para o eixo do



Além dos sensores de temperatura, existem sensor de massa específica (DIT-01), vazão mássica (FIT-01) e pressão diferencial (DPIT-01) na linha de amostragem do meio reacional. A Figura 3.2 ilustra a montagem experimental.



Figura 3.2 – Imagens do protótipo experimental

Os sinais dos sensores são enviados pelos transmissores para o CLP posicionado no painel de comando. O CLP é do tipo MCI02-QC, da HI Tecnologia. O sinal para comunicação entre as bombas e o CLP, se dá por meio de dispositivos intermediários (inversores de frequência) que servem para controlar a velocidade do motor. A potência do resistor recebe sinal do variador de potência, que depende do sinal enviado pelo CLP para o mesmo.

A comunicação dos transmissores com o CLP, ou seja, do sinal analógico para o digital se dá por meio das placas de conversão do CLP de entrada e saída. Através da porta serial de padrão 232, o PLC se comunica com o computador. A programação do PLC é feita com software SPDSW da Hi Tecnologia, utilizando linguagem *ladder*.

### **3.1.1. Modificação do Protótipo Experimental**

Duas principais modificações foram realizadas no protótipo experimental:

- Troca do resistor de potência máxima 3 kW por um de 2,5 kW,
- Adição de um condensador ao reator.

O resistor foi trocado por apresentar problemas. Um resistor de 2,5 kW foi utilizado, sendo considerada a potência máxima melhor dimensionada para o volume do tanque. Dessa forma, a utilização do variador de potência não se restringe de 0% a 80% da potência máxima, mas sim a 0% e 100%. Dessa forma, tornou-se possível operar usando uma faixa maior para o procedimento experimental.

A adição do condensador possui uma importância maior, visto que já fora detectado um problema causado pela diminuição excessiva da temperatura ao decorrer do tempo de reação, sendo assim necessário fornecer mais energia ao óleo térmico. Considerando que o problema provém da evaporação do tolueno (solvente), causando a redução do nível do produto no reator, diminuindo a área de contato para transferência de calor com a camisa. Assim, o condensador evita a perda de tolueno e estireno por arraste, devido a adição de nitrogênio e da alta temperatura de operação.

### 3.2. VARIÁVEIS DE PROCESSO

Nesse tópico serão apresentadas as variáveis importantes para o desenvolvimento do software e monitoramento do processo de polimerização, variáveis manipuláveis e variáveis não manipuláveis.

Para obtenção de polímeros com alta qualidade, com a distribuição das cadeias poliméricas uniforme, é necessário o controle da temperatura de reator ( $T_R$ ). Isso porque há variação da temperatura durante a reação química, devido a característica exotérmica da mesma.

O início da reação deve ser realizado a uma temperatura alta o suficiente para que o monômero seja consumido a uma taxa aceitável. Essa temperatura não pode ser alta a ponto de a reação acontecer de forma tão rápida que apenas algumas moléculas de monômero se unam em cadeias poliméricas, gerando um produto com distribuição de cadeias não uniformes. Também, com uma grande taxa de consumo de monômero a reação pode ocorrer de modo que as cadeias cresçam mais que o necessário.

Por a polimerização de estireno ser uma reação exotérmica, a importância da aplicação de controle torna-se maior ainda, pois ao início da reação, se não houver meios para refrigerar o reator, a qualidade do produto é comprometida.

Nesse caso, a variação de  $T_R$  depende de três fatores:

- Perda de energia na linha de amostragem
- Energia despreendida durante a reação de polimerização
- Temperatura do óleo na camisa do reator

A remoção de energia no reator se dá pela linha de amostragem, mesmo com isolamento. Essa perda depende da temperatura ambiente, sendo considerada constante e apenas um ponto negativo em relação a produção do polímero. Porém, não é um problema visando a aplicação de técnicas de controle no processo.

A energia despreendida durante a reação pode aumentar a temperatura do reator, a depender da concentração de estireno utilizado. Esse fator é considerado a perturbação do processo, pois se torna grande no início da reação e a medida que o reagente é consumido diminui, gerando a necessidade de meios para manter a temperatura constante.

O terceiro ponto é a temperatura do óleo, essa por outro lado, está diretamente relacionada à potência do resistor e é manipulável pelo operador do processo. É por meio desse ponto, que se torna possível o controle de  $T_R$ .

Com intuito de avaliar a transferência de calor entre o óleo e o meio reacional, também é medido a  $T_S$ , temperatura de saída da camisa do óleo térmico. Como são conhecidas as propriedades do fluido térmico e é possível estimar a vazão que o mesmo possui ao passar pela camisa, devido da diferença entre  $T_E$  e  $T_S$ , torna-se possível estimar o coeficiente de troca térmica e a quantidade de calor trocada no sistema.

Para finalizar, uma importante variável medida *online* é a massa específica (Dens). Por meio dessa é possível estimar a quantidade de estireno consumida durante a reação, ou seja, associar com a conversão de monômero. Sendo assim, uma importante variável para estimar qualidade do produto, porém não indica quão uniforme está a distribuição das cadeias poliméricas.

Por último, para que um sistema de monitoramento e controle seja completo para operação da planta, as variáveis  $F_{B1}$  e  $F_{B2}$ , frequência das bombas 1 e 2, bomba da linha de óleo térmico e da linha do meio reacional, devem ser adicionadas ao sistema, para que seja possível operar o sistema nas condições de interesse. Na Tabela 3.1 é mostrada a relação de nomes das variáveis aqui citadas e o número dos transmissores mostrados no *layout* do protótipo na Figura 3.1.

**Tabela 3.1 - Relação dos transmissores e da nomenclatura utilizada das variáveis.**

| <b>Dispositivo</b> | TT 01 | TT 02 | TT 03 | DT 01 | BA 01    | BA 02    |
|--------------------|-------|-------|-------|-------|----------|----------|
| <b>Variável</b>    | $T_R$ | $T_E$ | $T_S$ | Dens  | $F_{B1}$ | $F_{B2}$ |

A partir desse conjunto de variáveis que foram listadas como as principais para melhor conduzir a reação de polimerização, deu-se início ao desenvolvimento de um sistema para comunicação dos sensores com o computador, com a finalidade de atuar da maneira desejada na reação.

### 3.3. PROCEDIMENTO EXPERIMENTAL

A forma de obtenção do poliestireno utilizada é a polimerização em solução, ou seja, ao estireno é misturado um solvente, nesse caso, tolueno, para que durante a reação não ocorra aumento elevado da viscosidade, prejudicando a agitação e distribuição de energia no sistema, gerando cadeias poliméricas de tamanho irregular. O estireno utilizado foi da Sigma Aldrich, 99% de pureza. O tolueno é da ECIMBRA.

Além da adição de solvente é adicionado também Peróxido de Dibenzoila, composto químico que age como iniciador da reação. Ao ser adicionado ao meio aquoso, possui radicais livres, gerando a formação de radicais livres na molécula de estireno, favorecendo a união entre as moléculas e crescimento da cadeia polimérica.

O procedimento experimental é apresentado em passos, abaixo:

- 1) Adição de 330 mL de solvente, tolueno.
- 2) Adição de 770 mL de reagente, estireno.
- 3) Ligar agitadores, do tanque de óleo e do reator.
- 4) Ligar bomba de óleo com frequência de 30 Hz,
- 5) Ligar bomba do reator com frequência de 18 Hz,
- 6) Ligar o resitor para que a temperatura do meio reacional chegue a 90°C.
- 7) Abrir válvula de água para condensador.
- 8) Abrir válvula de N<sub>2</sub> para borbulhamento no meio reacional.
- 9) Ao estabilizar a temperatura, adiciona-se o BPO para iniciar a reação.
- 10) Ao finalizar, desligar a agitação e bomba de óleo.
- 11) Retirar o meio reacional do reator e limpar o sistema adicionando mais tolueno puro.

### 3.4. SISTEMA DE COMUNICAÇÃO

Com uma dinâmica complexa e não linear, o processo exige uma ferramenta computacional para manipulação e monitoramento das condições de operação e armazenamento do histórico de dados. Atualmente, diferentes *softwares* são utilizados na automação de plantas industriais, o que demanda custo financeiro e de tempo, ao treinar o recurso humano disponível para a tarefa de operar um sistema.

Para pequenos protótipos experimentais, no caso de desenvolvimento de pesquisa dentro da indústria ou academia, isso pode ser visto de forma negativa, onde um simples experimento demandaria bastante tempo para treinamento de recursos humanos em novos *softwares*.

Com o intuito de diminuir esse custo inicial em projetos, tratou-se nesse trabalho de desenvolver um sistema para comunicação com os *hardwares* da planta de polimerização, capaz de receber, enviar e processar a informação em Microsoft Excel<sup>®</sup>.

Existem várias vantagens associadas ao Excel citadas por MOURA (2009), onde as principais são:

- Disponibilidade, praticamente em ambiente industrial e acadêmico,
- Possui facilidade no uso de cálculos sofisticados,
- Possui uma vasta biblioteca de funções, para engenharia, estatística, matrizes, etc.
- Integração com outras linguagens, como FORTRAN.
- Linguagem de programação interna (*Visual Basic for Applications*).
- Portabilidade, já que existem vários problemas e métodos facilmente encontrados na internet em Microsoft Excel .

O Microsoft<sup>®</sup> Excel também possui a ferramenta *Solver*, desenvolvida para problemas de otimização. Também é versátil para o desenvolvimento de interface gráfica para comodidade do operador, não sendo necessário trabalhar diretamente com os códigos do programa ou a planilha.

Além das vantagens apresentadas do Microsoft Excel<sup>®</sup>, também é possível a utilização de interface gráfica no desenvolvimento do sistema supervisorio. Assim é possível evitar que a manipulação de variáveis seja feitas no código do programa. Assim, dá-se início o desenvolvimento da ferramenta computacional desenvolvida em Excel.

### **3.4.1. Ferramenta Computacional**

Uma planilha no Excel<sup>®</sup> foi desenvolvida utilizando programação em Visual Basic (macros do Excel) com a finalidade de obter uma maneira prática e simples de realizar

monitoramento e controle de um equipamento. No desenvolvimento de um sistema para tal finalidade algumas etapas a serem seguidas como mostradas a seguir:

1. Programação do CLP para conversão das variáveis e posição na memória para entrada e saída das variáveis manipuláveis e monitoradas.
2. Configuração e execução do *driver* de comunicação CLP - servidor OPC, que cria o servidor OPC no Windows.
3. Execução do código em VB com escolha dos valores de memória previamente configuradas no CLP.
4. Programação dos programas para processar os dados e interface para o usuário.
5. Criação da interface gráfica

Um dos problemas desse sistema é a realização de uma comunicação segura e instantânea com o equipamento e para o funcionamento de algoritmos de trabalho, como a aquisição de dados em tempo real, controladores como PI, PID, e controle avançado. Nos próximos tópicos será discutido cada passo desse desenvolvimento.

#### **3.4.2. Programação CLP**

O CLP utilizado no protótipo experimental pertence a HI Tecnologia, da série MCI02-QC, tendo sido utilizado duas placas de entrada analógica (AIM) e saída analógica (AOM) de sinal entre 4 a 20 mA, com 8 entradas e 4 saídas em relação ao equipamento.

A programação do CLP é realizada em *ladder*, uma linguagem de programação de CLP no qual as funções lógicas são representadas por meio de contatos e bobinas, de modo análogo a um esquema elétrico com os contatos dos transdutores e atuadores.

Dessa forma, o CLP é programado em linguagem *ladder* pelo software da HI Tecnologia, chamado SPDSW. O software consiste numa interface gráfica onde o usuário pode fazer programações de portas de entrada e saída dos sinais, como também operações matemáticas, lógicas e até mesmo controladores PID.

O CLP é acoplado a placas de entrada e saída, AIM e AOM, módulos de conversão de sinal analógico para digital e vice versa. A programação necessária feita ao CLP é de indicar quem são as variáveis de entrada e saída. O módulo de entrada possui capacidade para 8 entradas, enquanto o módulo de saída apenas 4. Dentro do CLP, essa indicação é

feita utilizando-se das variáveis que iniciam com “E” (entrada) e “S” (saída), variando de E0 a E7 e S0 a S3.

O sinal que a placa de aquisição (AIM) recebe, sinal analógico é convertido para digital, em *bytes*, e a placa de saída envia os valores digitais que são na forma de *bytes* para serem convertidos para corrente contínua (4 – 20 mA).

O CLP utilizado possui memória 12 *bits*, sendo possível atingir em decimal o valor 4095. Logo os valores de corrente que são enviados ao CLP, de 4 a 20 mA, são entendidos como valores decimais de 0 a 4095. Assim, é necessário fazer a conversão de *bits* para a unidade da variável medida no CLP, para que esse envie ao computador apenas os valores de interesse para o operador.

Na Tabela 3.2 é mostrada a faixa das variáveis e para quais unidades foram convertidas de forma linear, provindas dos valores decimais de 0 a 4095. A última coluna mostra as unidades reais em que as variáveis se encontram e foram calibradas, ou seja, a potência do resistor para 0 a 4095 no CLP corresponde a 0 – 100% na planilha, que em unidades de engenharia, seria 0 a 2,5 kW.

**Tabela 3.2 - Conversão de unidades no CLP de Bits para unidades de engenharia.**

| Variáveis           | Supervisório                | Real       |
|---------------------|-----------------------------|------------|
| Potência            | 0 – 100 %                   | 0 – 2,5 kW |
| Frequência da Bomba | 0 – 100 %                   | 0 – 60 Hz  |
| Temperatura         | 0 -200 °C                   | -          |
| Massa Específica    | 800 - 1060kg/m <sup>3</sup> | -          |

A Figura 3.4 exibe a interface e parte do programa, no SPDSW, desenvolvido em *ladder* para as conversões de unidades de variáveis. O valor de entrada E0000 é alocado em uma memória real (D0000), a qual pode ser utilizada em operações matemática, nesse caso, convertendo os valores decimais que o CLP recebe em unidades de trabalho, por meio da função linear.

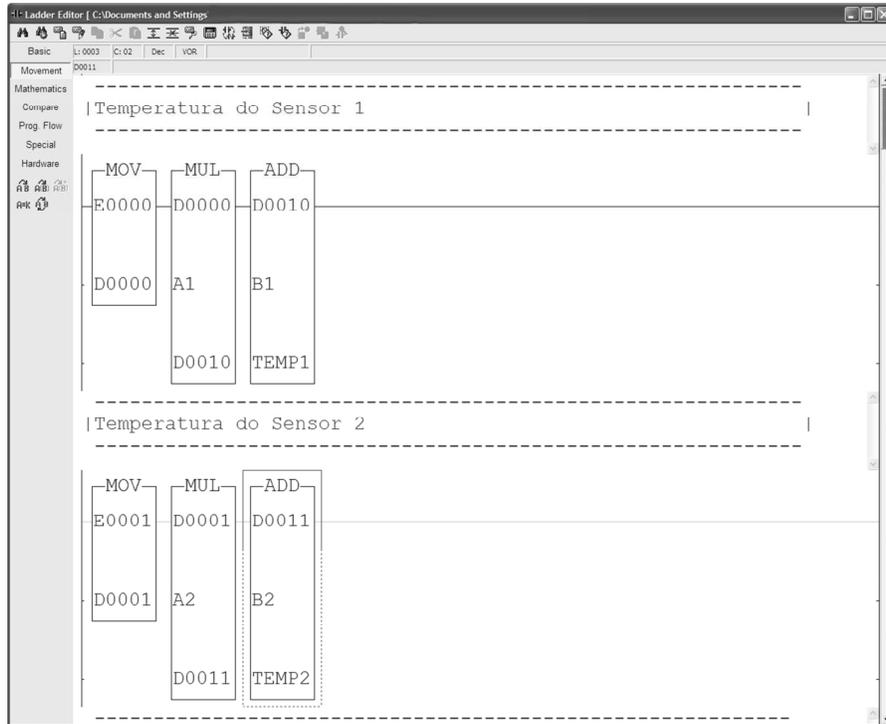


Figura 3.3 - Interface do software SPDSW e código de conversão de unidades.

A Tabela 3.3 e Tabela 3.4 mostram os valores utilizados para o coeficiente angular e linear de uma equação linear, correspondente a Equação 4.1. Também é mostrado as memórias utilizadas na programação do código em *ladder*.

$$\text{Valor de Trabalho} = A \times \text{Valor Decimal} + B \quad (3.1)$$

Tabela 3.3 - Valores de Memória no SPDSW utilizadas e coeficientes da função linear das variáveis de entrada.

| Variáveis de Entrada |         |               |               |              |
|----------------------|---------|---------------|---------------|--------------|
| Variáveis            | Memória | Memória Final | Coef. Angular | Coef. Linear |
| $T_R$                | E0      | D20           | 0,0522        | 3,6100       |
| $T_E$                | E1      | D21           | 0,0487        | 0,2289       |
| $T_S$                | E2      | D22           | 0,0477        | 3,7730       |
| <b>Dens</b>          | E4      | D24           | 0,1600        | 620,00       |

Os parâmetros, mostrados na tabela, da temperatura foram obtidos a partir da calibração dos sensores PT100. Já os parâmetros do sensor de Densidade é apenas uma conversão dos valores de bits para unidades de engenharia.

Tabela 3.4 - Valores de Memória no SPDSW utilizadas e coeficientes da função linear das variáveis de saída.

| Variáveis de Saída |         |               |               |              |
|--------------------|---------|---------------|---------------|--------------|
| Variáveis          | Memória | Memória Final | Coef. Angular | Coef. Linear |
| F <sub>B1</sub>    | S0      | D30           | 38,700        | 50,000       |
| P                  | S1      | D31           | 36,740        | 210,00       |
| F <sub>B2</sub>    | S2      | D32           | 39,015        | 55,000       |

### 3.4.3. Comunicação OPC

OPC é um conjunto comum de interfaces, métodos e propriedades de comunicação, agregados dentro de uma especificação padronizada e aberta para acesso público. O OPC permitiu aos fornecedores de automação reduzir seus custos de conectividade e assim manter o foco nas funcionalidades de sua solução, para os clientes o benefício foi à flexibilidade (PUDA, 2009).

A primeira versão funcional do OPC foi desenvolvida por volta de 1996, resultado do trabalho conjunto entre fornecedores de sistemas para automação industrial. A sigla OPC significa OLE para Controle de Processos (*OLE for Process Control*), no qual OLE é a sigla para Vinculação e Incorporação de Objetos. (*Object Linking and Embedding*).

A utilização de comunicação OPC se dá por meio da criação de dois componentes, o servidor OPC e o cliente OPC. Clientes OPC são os hardwares ou computadores que se ligarão ao servidor OPC, esse por sua vez, responsável por armazenar as informações adicionadas pelos clientes. Para uso criação do servidor OPC, normalmente é necessário a instalação no computador do *driver* do dispositivo utilizado.

O *driver* HS1, desenvolvido pela HI Tecnologia, é um servidor que possibilita a comunicação serial e cria um servidor OPC no computador, utiliza-se o SCP-HI para a comunicação com o CLP. A Figura 3.4 mostra a interface do programa que executa o *driver* HS1.

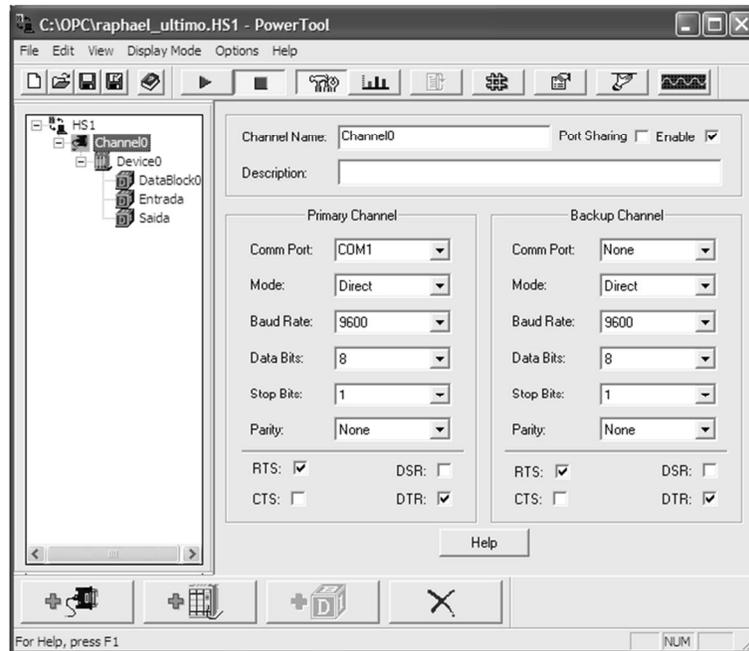


Figura 3.4 - Interface do driver HS1, da HI Tecnologia para comunicação OPC.

O *driver* deve ser configurado especificando-se o canal de comunicação, o dispositivo e os blocos de dados, nos quais são selecionados os valores de endereçamento da memória correspondente a cada variável, do CLP, que o *driver* tem que acessar para receber ou enviar informação. As principais configurações para o *driver* são mostradas na Tabela 3.5, como também os valores das memórias utilizadas no trabalho.

Tabela 3.5 - Principais parâmetros a serem configurados no driver HS1.

| Canal                | Dispositivo      | Bloco de Dados |           |
|----------------------|------------------|----------------|-----------|
| Porta de Comunicação | Estação Primária | Entrada        | Saída     |
| COM1                 | 255              | D20 – D24      | D30 – D33 |

Concluída a “ponte” que cria o servidor OPC e realiza a comunicação entre programas que atuam como um cliente OPC e o dispositivo CLP é desenvolvido o programa em VB para leitura, escrita e processamento dos dados.

#### 3.4.4. Planilha Eletrônica

O sistema de monitoramento e controle para o processo de polimerização deve ser eficiente quanto à transmissão e processamento de informação no computador e simples de

modo a não exigir extenso conhecimento técnico sobre a ferramenta computacional utilizada. Além disso, ser uma ferramenta viável economicamente.

Com esse direcionamento, buscou-se desenvolver a ferramenta utilizando o *software* Microsoft Excel<sup>®</sup>, ou seja, uma planilha eletrônica onde é possível realizar aquisição de dados, salvar históricos e enviar comando ao protótipo. Por outro lado, a necessidade de utilizar um sistema para comunicação com a planta é considerado um problema, pois o Excel não é uma ferramenta voltada para essa finalidade.

O próximo tópico aborda esse passo da adaptação do sistema de comunicação, provindo de uma planilha disponível pela empresa Takabeshi, desenvolvida para servir de cliente OPC, realizando comunicação com o servidor OPC presente no computador.

### **Cliente OPC**

A viabilidade do desenvolvimento da planilha eletrônica como cliente OPC se dá através da linguagem computacional presente no Excel, o Visual Basic for Applications (VBA), assim o Excel conecta-se e transfere informação com do servidor OPC. A facilidade dessa ferramenta no Excel se dá pela ativação do pacote OLE *Automation*, nas referências do VBA. Um código em Visual Basic, disponibilizado pela empresa japonesa Takabeshi em seu site, foi utilizado com esse propósito.

A planilha da Takabeshi possui uma rotina “*Auto\_Open*”, a qual declara variáveis iniciais da biblioteca do OLE *Automation* e busca pelos servidores OPC que estão funcionando no PC no momento, reproduzindo a lista dos mesmos na planilha, para a escolha do servidor que se pretende acessar. A mostra a interface da planilha Excel da Takebishi.

Na planilha da Takabeshi é preciso especificar o nome do dispositivo, o código e os valores dos endereços de memória para as variáveis desejadas, ou seja, o endereçamento de memória especificado no CLP, em linguagem *ladder*. O nome do dispositivo é o especificado no *driver* HS1, da HI Tecnologia.

Após detectar os servidores OPC existentes no Windows é possível conectar-se a eles através do botão “*Connect*”, criando-se o vínculo com o servidor e torna possível a utilização dos botões “*Advise*” e “*Write*”, para receber e enviar o sinal dos sensores e atuadores do protótipo, respectivamente.

A planilha da Takabeshi possui também um módulo VB chamado OPCDataAccess que é dividido em 7 rotinas, sendo elas:

- *Connect*
- *AddGroup*
- *Advise*
- *UnAdvise*
- *SyncWrite*
- *OPCMygroup\_DataChange*
- *Class\_Terminate*

A rotina *connect* é acionada pelo botão “Connect” na planilha. Essa rotina é responsável pela identificação do servidor OPC selecionado e pela busca do dispositivo e memórias especificadas na planilha. *AddGroup* é uma rotina chamada no bloco do botão “Connect”. Possui aplicação de adicionar os parâmetros especificados na planilha ao grupo de variáveis OPC utilizados pelo serviço OLE Automation 2.0, ativado anteriormente.

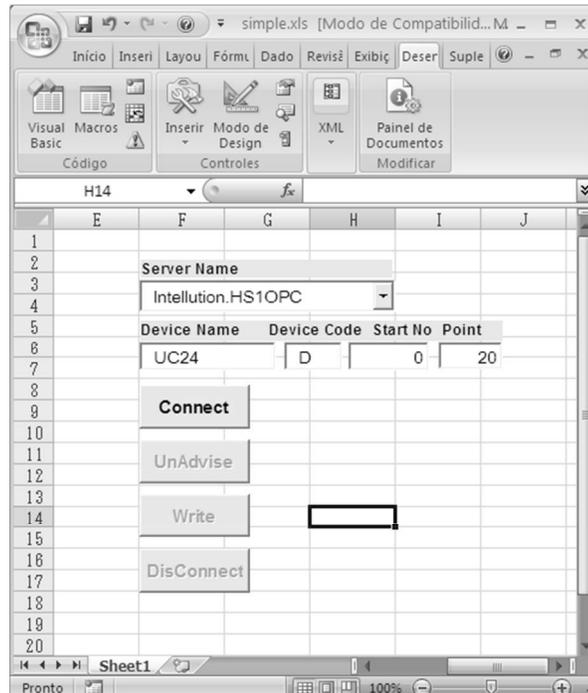


Figura 3.5 - Planilha Eletrônica para cliente OPC da empresa Takebishi.

A rotina *connect* é acionada pelo botão “Connect” na planilha. Essa rotina é responsável pela identificação do servidor OPC selecionado e pela busca do dispositivo e memórias especificadas na planilha. *AddGroup* é uma rotina chamada no bloco do botão “Connect”. Possui aplicação de adicionar os parâmetros especificados na planilha ao grupo de variáveis OPC utilizados pelo serviço OLE *Automation*, ativado anteriormente.

As rotinas *Advise* e *UnAdvise* são adicionadas para iniciar e finalizar, respectivamente, a aquisição de dados do *hardware*. A informação é mostrada na planilha eletrônica, na primeira coluna. *SyncWrite* é chamada pelo botão “Write” para enviar dados para o *hardware*, variáveis *output*.

A principal dessas rotinas e mais complexa é a *OPCMygroup\_DataChange*. Essa rotina está vinculada com o sinal vindo do CLP, de forma monitorar os valores vindos do servidor OPC e a qualquer mudança desses, os mesmos são escritos nas células da planilha.

A partir da manipulação das três ferramentas explicadas anteriormente, programação em *ladder*, *driver* de comunicação OPC, e código em VB utilizando OPC *Automation 2.0*, foi possível construir um sistema computacional que permitisse a comunicação segura entre o equipamento e computador, possibilitando a leitura de variáveis de processo e o envio de variáveis manipuladas. Para a utilização deste sistema no acompanhamento e controle do processo de polimerização, ainda resta a implementação do sistema que contém a interface com o usuário, com possibilidade de:

- Visualização das variáveis de processo,
- Alteração de variáveis manipuladas,
- Acompanhamento das variáveis ao longo do tempo,
- Armazenamento do comportamento dinâmico do sistema,
- Aplicação de procedimentos de controle.

### **3.4.5. Sistema de Monitoramento e Controle**

O código em VBA da Takabeshi foi utilizado no desenvolvimento de uma nova planilha eletrônica. Essa nova planilha contém uma interface para interação com o usuário própria para o sistema de polimerização.

A rotina *OPCMygroup\_Datachange* realiza a aquisição dos dados quando estes são alterados no processo, porém torna-se um problema a criação de um histórico de dados sem um tempo de amostragem constante. Análises estatísticas e avaliações do processo são realizadas com tempo uniforme de aquisição.

Para que a rotina seja executada com frequência e essa frequência especificada na janela inicial, foi feita uma adaptação do código, sendo essa mudança de extrema importância na aplicação de controle de processos.

A planilha atual é mostrada na Figura 3.6. Na tela principal do programa de supervisão foram inseridos seis campos distintos para:

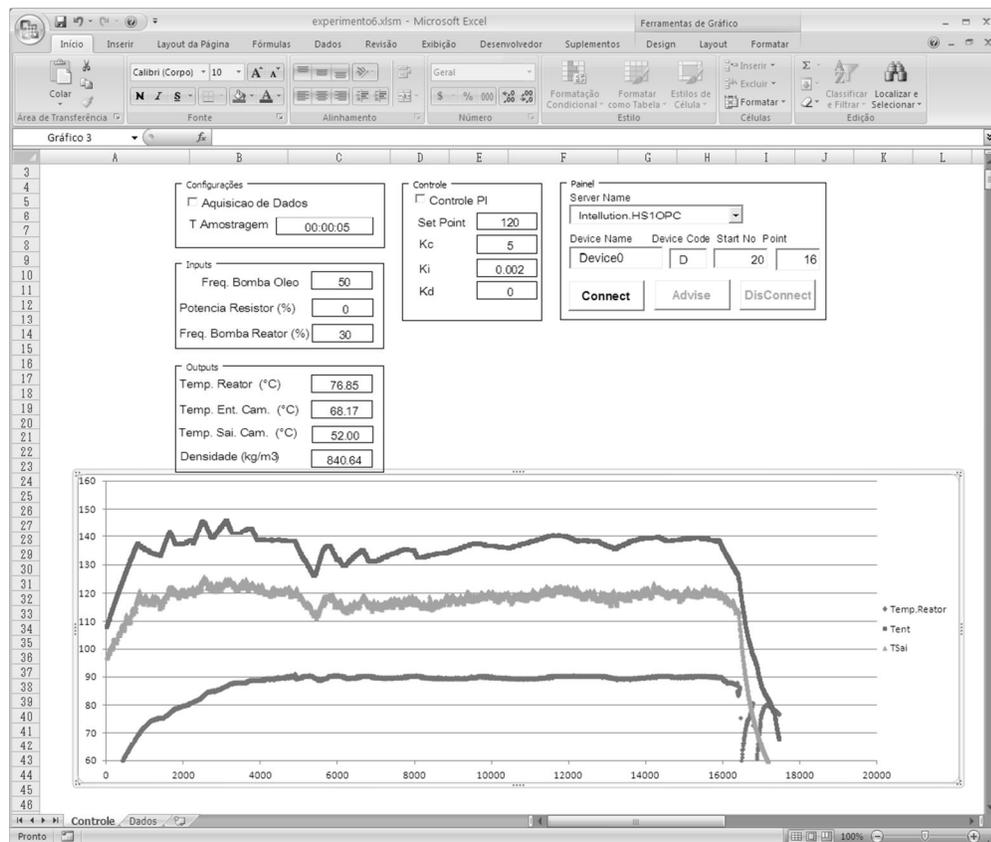


Figura 3.6 - Programa sendo executado no Microsoft Excel®.

- Definição do intervalo de amostragem,
- Inserção de valores para variáveis manipuladas  $F_{B1}$ ,  $P$  e  $F_{B2}$ ,
- Visualização das variáveis de processo  $T_R$ ,  $T_E$ ,  $T_S$ ,  $Dens$ ,

- Inserção dos parâmetros do dispositivo para comunicação (códigos VBA da Takabeshi).
- Inserção de parâmetro de um controlador proporcional integrativo (PI), para controle de  $T_E$  em função de P.
- Visualização gráfica das variáveis de entrada em função do tempo.

Na Figura 3.7 é mostrado o painel do sistema supervisor, feito utilizando as ferramentas de desenvolvedor do Microsoft<sup>®</sup> Excel.

The screenshot shows an Excel-based supervisor interface with three main panels:

- Configurações:** Includes a checkbox for 'Aquisicao de Dados' and a 'T Amostragem' field set to '00:00:05'.
- Inputs:** Contains three input fields: 'Freq. Bomba Oleo' (50), 'Potencia Resistor (%)' (00), and 'Freq. Bomba Reator (%)' (30).
- Outputs:** Displays four output values: 'Temp. Reator (°C)' (47.48), 'Temp. Ent. Cam. (°C)' (67.00), 'Temp. Sai. Cam. (°C)' (68.44), and 'Densidade (kg/m³)' (891.84).
- Controle:** Features a checkbox for 'Controle PI' and three control parameters: 'Set Point' (120), 'Kc' (5), 'Ki' (0.002), and 'Kd' (0).
- Panel:** Includes a 'Server Name' dropdown menu (Intellution.HS1 OPC), a table for device configuration, and three buttons: 'Connect', 'Advise', and 'DisConnect'.

| Device Name | Device Code | Start No | Point |
|-------------|-------------|----------|-------|
| Device0     | D           | 20       | 16    |

Figura 3.7 - Painel da planilha em Excel para monitoramento e manipulação das variáveis.

### 3.4.6. Algoritmo do sistema supervisor

Nesta seção será discutido o algoritmo desenvolvido do sistema de supervisão, programado em VB, por meio do VBA. A planilha, ao ser aberta, executa o código de busca dos servidores OPC presentes no computador. Essa rotina funciona de forma independente de ações do operador, de forma automática. O fluxograma do algoritmo é apresentado na Figura 3.8, e nesse tópico será apresentando o funcionamento do mesmo. O código tem como objetivo realizar a conexão com o servidor OPC e permitir a manipulação e leitura das variáveis de processo.

- 1) O botão *Conectar* ativa a rotina que realiza a conexão entre o Microsoft<sup>®</sup> Excel<sup>®</sup>, quanto a cliente OPC, com o servidor OPC criado pelo *driver*.

- 2) O botão *Aquisição* inicia a fase de aquisição dos dados, ou seja, leitura das variáveis de processo. Inicia realizando a leitura do tempo, para servir de referência para cálculo do tempo de amostragem e inicia a rotina *Principal*.

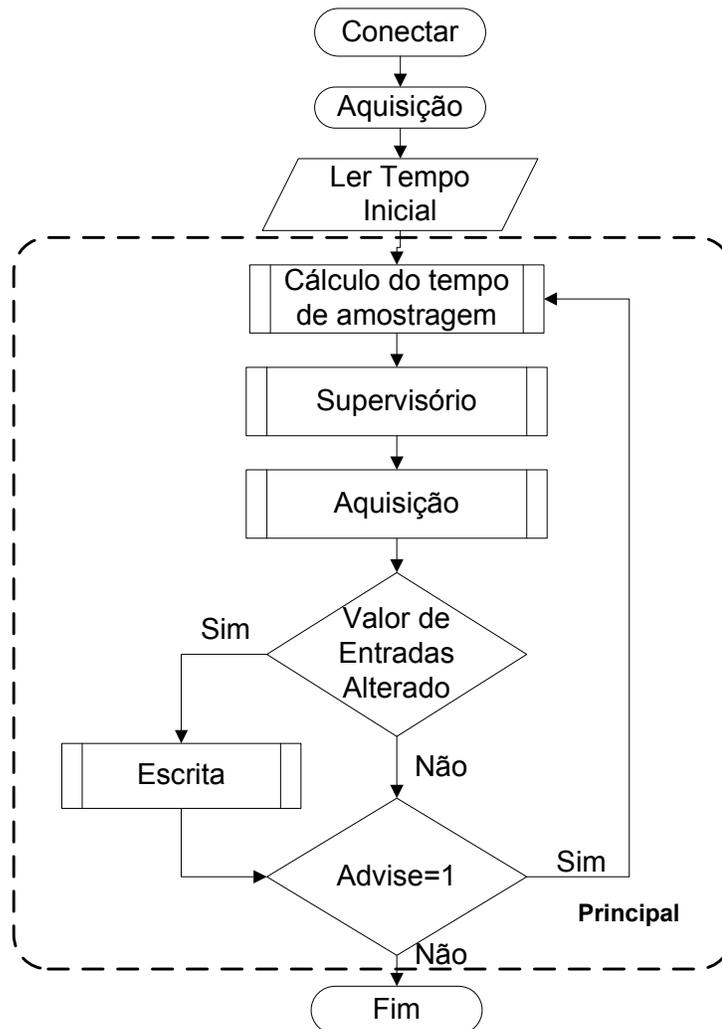


Figura 3.8 - Fluxograma do algoritmo desenvolvido para aquisição e manipulação da planta.

- 3) A rotina *Principal* é a responsável por sua auto execução com uma frequência igual ao tempo de amostragem. Dessa forma, ela é reiniciada a cada tempo de amostragem. Dentro dela, existem as rotinas mais simples:
- Aquisição dos dados do processo, por meio da leitura das memórias do PLC e armazenamento em variáveis VBA.
  - Tratamento dos dados: visualização na tela, estimativa de parâmetros para algoritmos de controle, procedimentos de controle, ajuste de dados, etc.

- c) Escrita dos valores calculados na memória do PLC para atuação no processo. Esta rotina é executada continuamente, até que seja interrompida pelo usuário. Também está inserido nesta rotina um procedimento para “congelar” a execução do processamento até que o próximo instante previsto para a aquisição de dados.
- 4) O sistema mantém funcionamento até que o botão *Advise*, seja novamente pressionado.

Utilizando esse algoritmo, torna-se simples criar rotinas para controle de processos, em VB, e adicioná-las ao sistema. Basta que a nova rotina seja colocada antes da verificação de manipulação das variáveis.

A limitação do sistema fica a critério de códigos utilizados para processamento dos dados em tempo real. Pois, se a os cálculos de uma nova rotina gerarem um grande esforço computacional, pode ser que o tempo de execução da rotina Principal torne-se maior que o tempo de amostragem escolhido. Se isso realmente acontecer a aplicação que se deseja fazer torna-se inviável se o tempo de amostragem não puder ser maior.

Com a comunicação estável e bem configurada entre VB e *hardware*, e a programação em VB funcionando da maneira apropriada para ser trabalhado como um supervisor de processo, o programa foi testado na planta piloto de polimerização apresentada acima, com o intuito de retirar informação e possivelmente aplicar estratégias de controle. A seguir são apresentadas explicações sobre o processo de polimerização e a reação química ocorrida, assim como a apresentação dos resultados obtidos.

### 3.5. CONDIÇÕES DE OPERAÇÃO

Nesse capítulo são apresentados os resultados obtidos na utilização do sistema supervisor em Excel<sup>®</sup>, tratamento dos dados e da utilização de Redes Neurais Artificiais como modelo para identificação do processo dinâmico para estimativas futuras.

Foram realizados quatro testes com reação química. Os experimentos foram realizados seguindo as condições de processo e da frequência das bombas. A Tabela 3.6 mostra a quantidade inicial de iniciador da reação ( $I_0$ ), monômero ( $M_0$ ) e solvente ( $S_0$ ); velocidade de agitação do reator e frequências das bombas.

Tabela 3.6 - Condições de processo dos experimentos realizados.

| $I_0$          | $M_0$        | $S_0$        | Agitador   | $Tr_0$   | $F_{B1}$ | $F_{B2}$ |
|----------------|--------------|--------------|------------|----------|----------|----------|
| 0,018<br>mol/L | 70<br>% vol. | 30<br>% vol. | 130<br>RPM | 90<br>°C | 30<br>Hz | 18<br>Hz |

### 3.6. REAÇÃO DE POLIMERIZAÇÃO

O primeiro experimento teve com objetivo avaliar o processo da reação química, mantendo todas as variáveis manipuladas constantes durante todo o tempo de reação. Torna-se possível analisar como o protótipo está funcionando para realizar alterações na planta, caso necessário. Os experimentos foram monitorados pela planilha do Excel<sup>®</sup>. Utilizando tempo de amostragem de 5 segundos. Nos gráficos a seguir, foi utilizado como filtro média móvel para 13 pontos.

O tempo de amostragem de 5 segundos foi utilizado, pois foi o melhor tempo encontrado para se trabalhar com segurança. Testes com a planilha eletrônica levaram a encontrar que o tempo de amostragem mínimo utilizado pode ser de 2 segundos. A média móvel com 13 pontos é utilizada porque foi a maior quantidade de pontos encontrada para filtrar os dados, de modo que não criou uma variação significativa nos cálculos de correlação cruzada dos dados, apresentados no capítulo 4.

A Figura 3.9 mostra o comportamento da temperatura do meio reacional dentro do reator durante a reação química. Nesse caso, dados foram coletados por 3 h. Como esperado, há elevação da temperatura no reator ( $T_R$ ), devido a característica exotérmica da reação química.

O desvio da elevação da temperatura em relação a temperatura inicial atinge 5 °C, depois de aproximadamente 15min da adição do iniciador. Após 1 h de reação a temperatura retorna a condição inicial. Ao decorrer da reação há uma diminuição da temperatura de 3,5 °C. Essa diminuição é possivelmente causada pelo aumento da massa específica dentro do reator, elevando a viscosidade do meio reacional. Com o aumento da viscosidade, a troca térmica com a camisa torna-se menos eficiente.

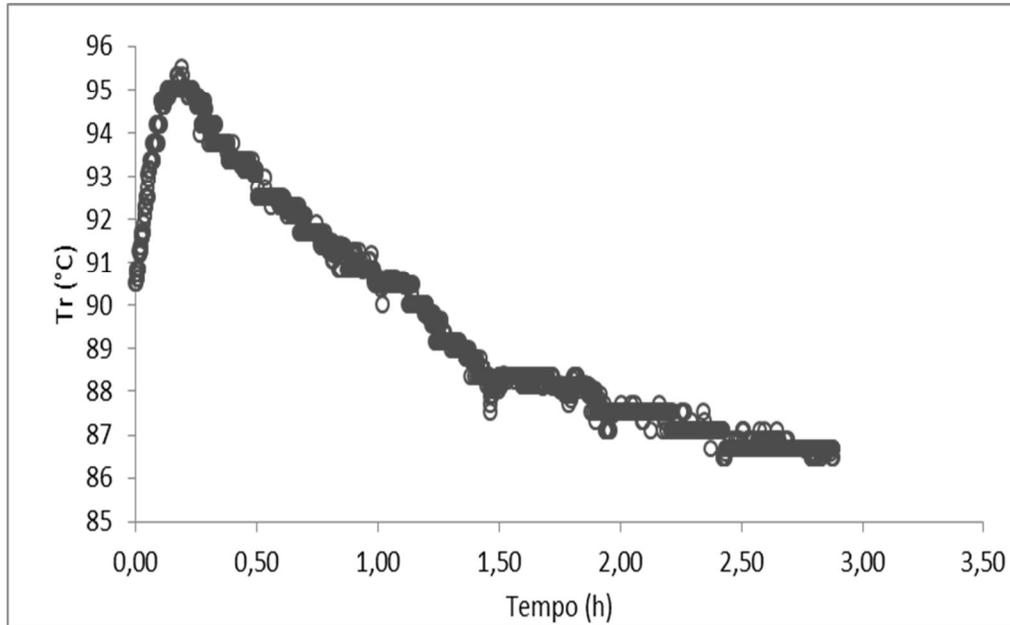


Figura 3.9 - Comportamento da temperatura do reator (TR) durante a reação de polimerização.

A elevação da massa específica é percebida pelos dados coletados pelo densímetro. A Figura 3.10 ilustra o comportamento dessa propriedade em função do tempo, durante a reação. Após duas horas de reação, percebe-se que a taxa de aumento é reduzida sendo consistente a variação da temperatura no início devido a formação das cadeias poliméricas.

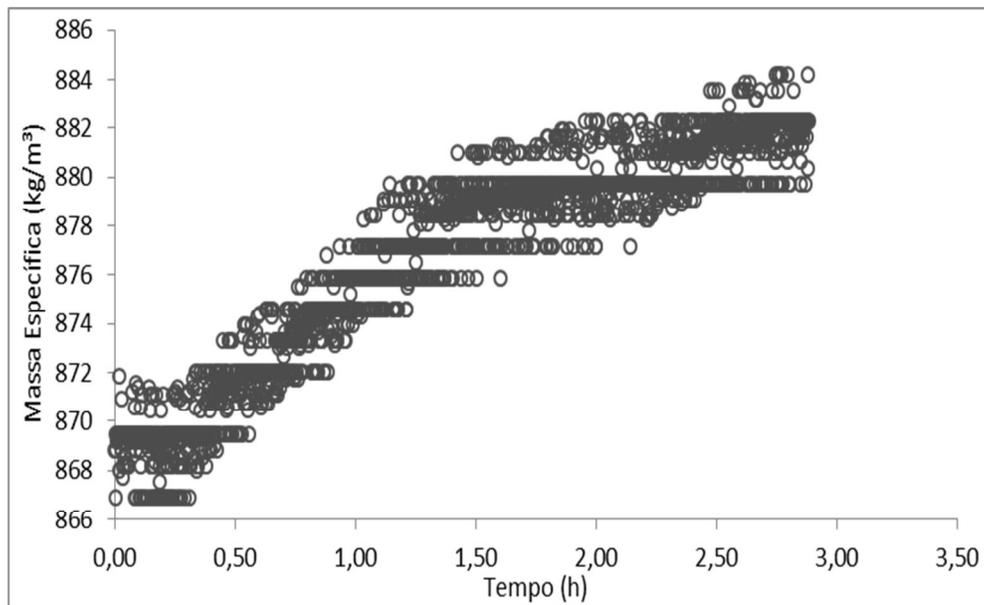


Figura 3.10 - Perfil da massa específica do meio reacional durante a reação química.

## 4. IDENTIFICAÇÃO DE SISTEMA

Uma vez estabelecido o funcionamento adequado do sistema supervisor a próxima etapa é a identificação do processo, visando determinar a relação entre as variáveis do mesmo. Para tanto foram realizados 3 experimentos com duas metodologias distintas:

- Controle da temperatura do óleo por meio da manipulação da potência da resistência;
- Controle da temperatura do reator por meio da manipulação da potência da resistência.

Os dois casos são extremos, onde o primeiro trata de avaliar o perfil da temperatura do reator para temperatura do óleo constante, enquanto o segundo mostra as alterações necessárias da temperatura do óleo para que  $T_R$  mantenha-se constante. O segundo caso foi feito duas vezes para obter melhor controle de  $T_R$  e conseqüentemente, gerar dados para identificação do sistema.

### 4.1. MONITORAMENTO DO PROCESSO

No primeiro experimento a temperatura do óleo térmico  $T_E$  foi controlada manualmente para se manter fixa em  $138^\circ\text{C}$ , temperatura necessária para que o reator permaneça inicialmente a  $90^\circ\text{C}$ . O objetivo é acompanhar o perfil de  $T_R$  ao manter a temperatura do óleo constante.

A Figura 4.1 mostra o perfil da temperatura do óleo na entrada da camisa do reator ( $T_E$ ). A temperatura se manteve em uma faixa entre  $137,6^\circ\text{C}$  e  $139,7^\circ\text{C}$ , aproximadamente. Em média, manteve-se a  $138,7^\circ\text{C}$ , durante o decorrer da reação. A Figura 4.2 mostra o perfil de potência utilizada na resistência. A Figura 4.3 mostra a evolução da temperatura do reator.

Nota-se um aumento da temperatura do óleo no início da reação, mesmo mantendo a potência em um valor constante ( $P = 15\%$ ). Isso pode ser atribuído à elevação da temperatura do reator devido à reação ser exotérmica (Figura 4.3), que assim reduz a transferência de calor (redução da diferença de temperatura) entre o óleo e o reator, portanto a um menor resfriamento do óleo. A sua posterior passagem pelo tanque de

armazenamento (potência constante) eleva assim a sua temperatura em relação à condição inicial (Figura 4.1). Uma redução da potência leva a uma redução na temperatura do óleo, como pode ser observado nas Figura 4.1 e Figura 4.2. De forma geral, nota-se uma boa concordância entre a variação de potência e a temperatura do óleo.

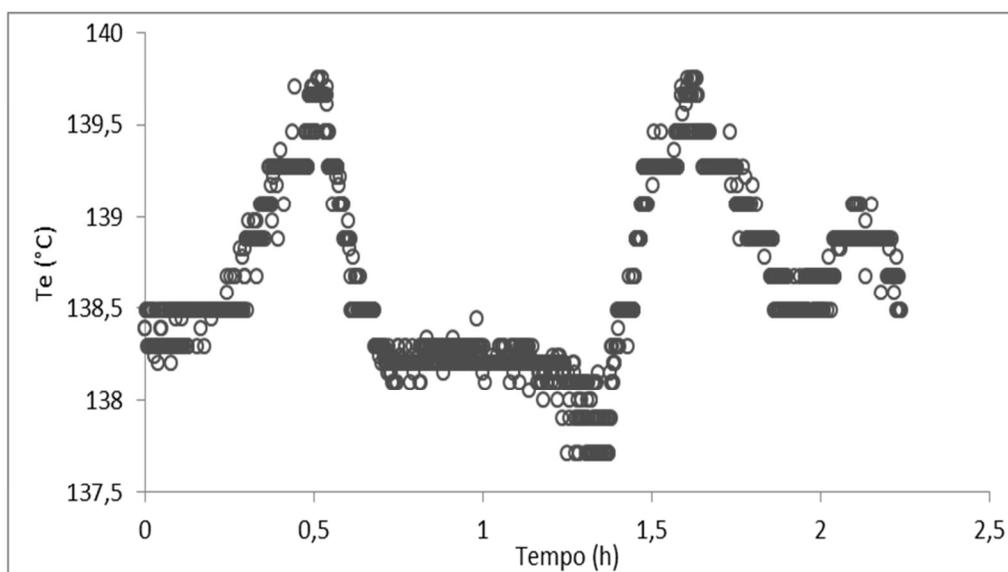


Figura 4.1 - Temperatura de Entrada ( $T_e$ ) durante a reação de polimerização.

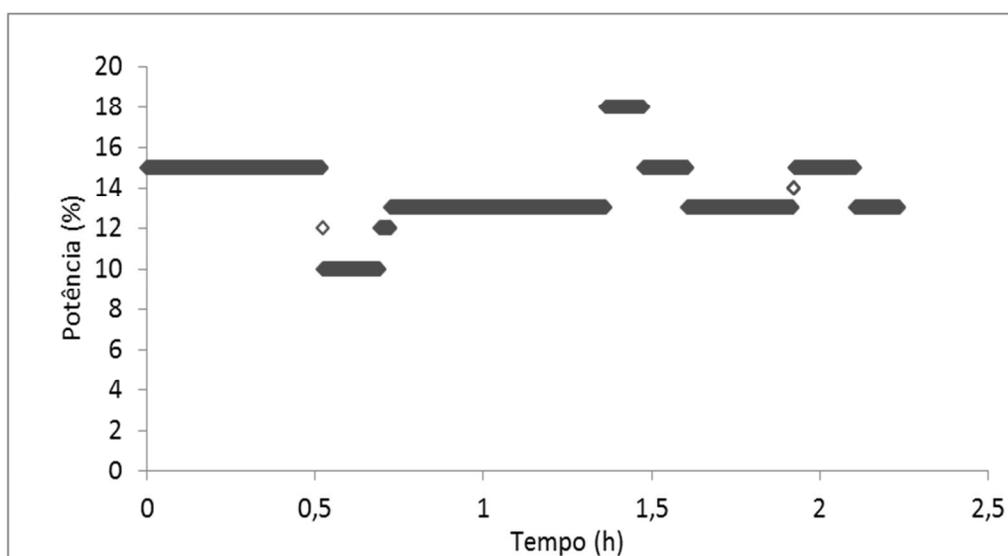


Figura 4.2 - Perfil da potência ( $P$ ) reação com valor de  $T_e$  constante

Assim, foi verificado que é possível manter a temperatura do óleo dentro de uma faixa de 2 a 3°C, por meio da manipulação da potência dissipada na resistência, sem causar perturbações significativas a  $T_R$ . Também, a influência que a reação pode gerar na temperatura do óleo, devido as condições de operação.

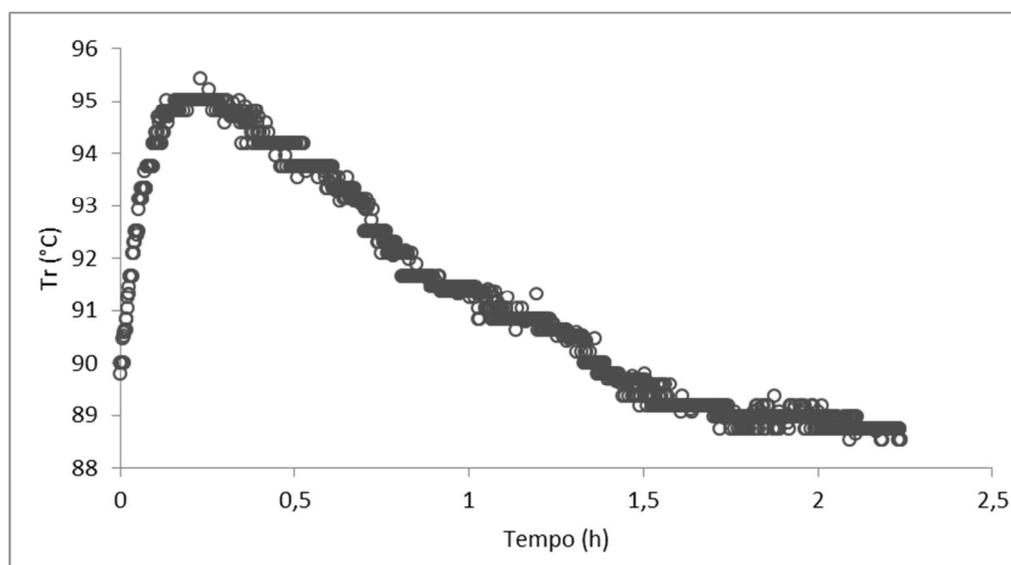


Figura 4.3 - Perfil da temperatura do reator ( $T_R$ ) reação com valor de  $T_E$  constante.

No segundo experimento tentou-se o controle manual da temperatura da reação, por meio da manipulação da potência dissipada na resistência. Este caso é mais complexo que o anterior uma vez que o comportamento da variável de interesse depende da temperatura do óleo, que por sua vez é função da potência dissipada. Assim, a relação entre a variável controlada ( $T_R$ ) e a variável manipulada ( $P$ ) passa por uma variável intermediária ( $T_E$ ).

O experimento foi realizado com o objetivo de manter a temperatura do reator em 90°C, por meio da manipulação da potência. A Figura 4.4 mostra o perfil de  $T_R$  para este experimento, mostrando que foi possível manter a temperatura numa faixa estreita em volta de 90°C, com desvios de  $\pm 0,8^\circ\text{C}$ .

A figura 4.5 mostra a variação da temperatura do óleo ( $T_E$ ) para o controle de  $T_R$ . No decorrer dos primeiros 10 minutos de reação,  $T_E$  foi reduzida em 12 °C aproximadamente, por meio do desligamento da resistência (Figura 4.6), para conter a elevação de  $T_R$  devido a reação.

Apesar de ter conseguido manter a temperatura do reator em uma faixa razoavelmente estreita, ocorreram oscilações ao longo do experimento devido ao tempo morto entre P e TR, uma vez que alterações na variável P não acarretam mudanças instantâneas em  $T_E$ , e muito menos em  $T_R$ .

A maior dificuldade encontrada foi na primeira meio-hora de reação, devido a rápida elevação da temperatura do meio reacional.

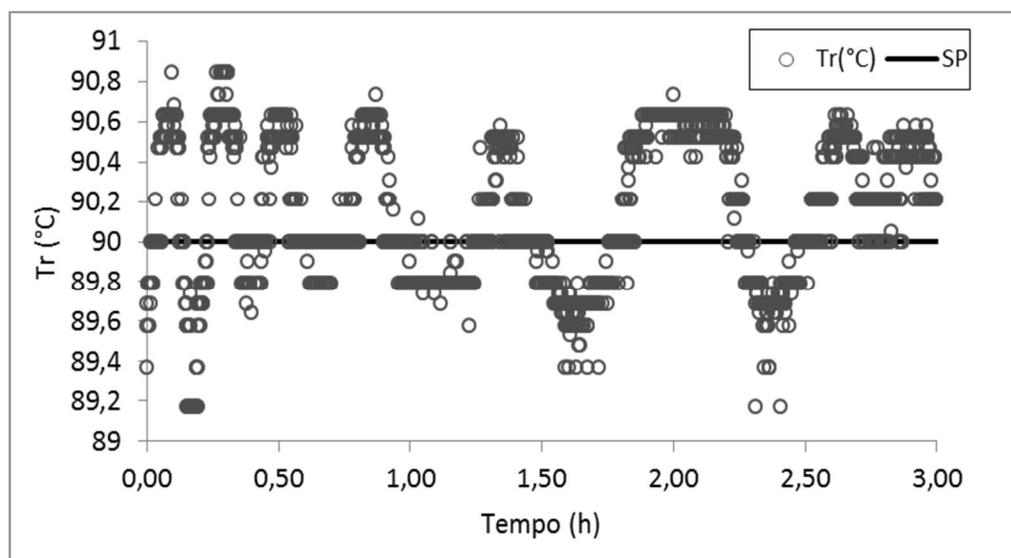


Figura 4.4 - Perfil da temperatura do reator ( $T_r$ ) com controle manual.

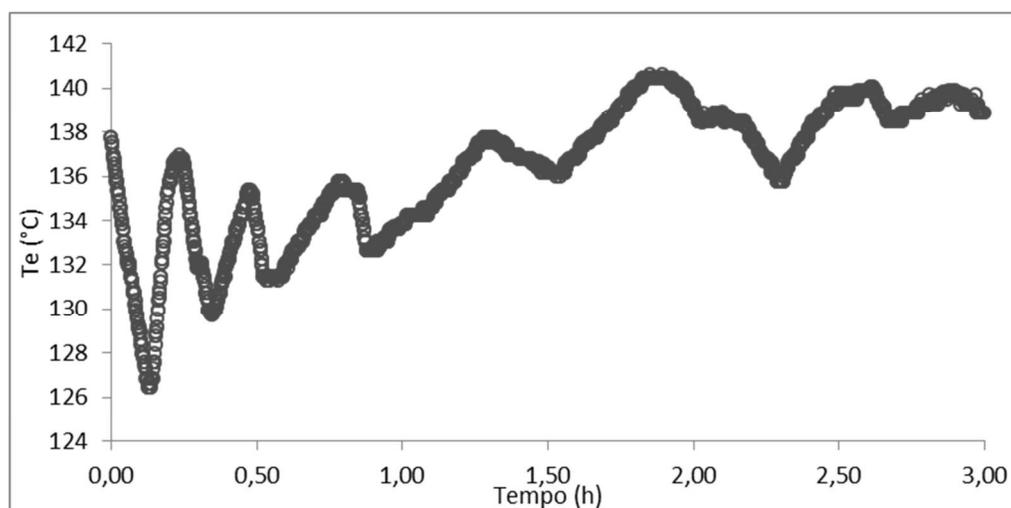


Figura 4.5 - Temperatura de Entrada ( $T_e$ ) durante a reação de polimerização.

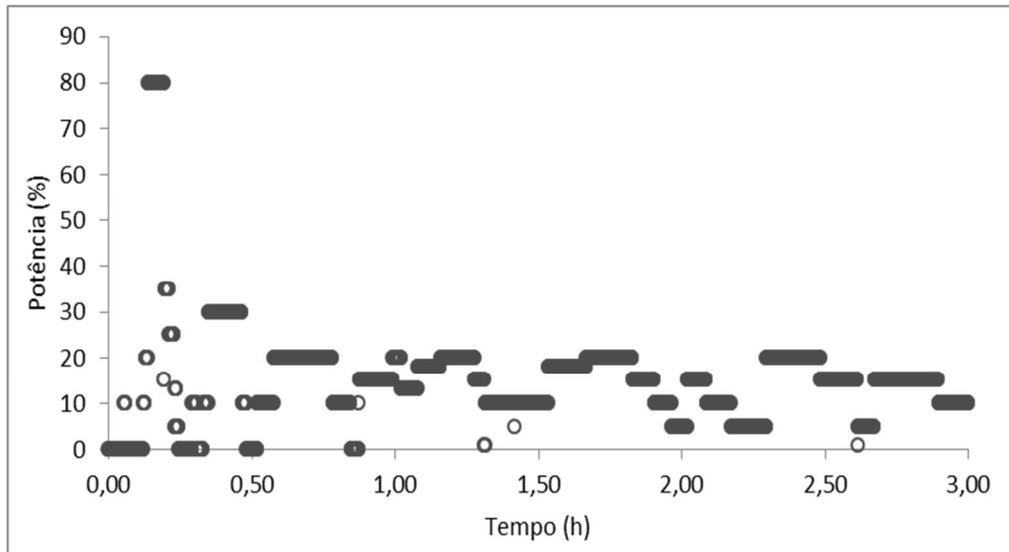


Figura 4.6 - Perfil da potência dissipada da resistência.

Com os experimentos foi percebido que é possível controlar  $T_E$ , e portanto, esta é uma variável indiretamente manipulável, visto que depende em maior grau da potência dissipada ( $P$ ). Os experimentos também serviram para retirar dados do processo para serem utilizados na criação do modelo empírico.

#### 4.2. TRATAMENTO DOS DADOS

Como o sistema implementado permite o armazenamento das variáveis do processo em função do tempo, é possível utilizar estes dados para analisar o processo e gerar um modelo empírico do mesmo. O fator mais importante para o desenvolvimento de um modelo empírico é a qualidade dos dados obtidos da planta. É por meio das informações presentes nos dados que será extraído os parâmetros para o ajuste do modelo.

Por isso faz-se importante tratar os dados para melhorar a performance e qualidade do modelo neural, principalmente, aplicado a um processo dinâmico. Os principais pontos que influenciam nesses casos, são:

- Ruído nos dados,
- Tempo de amostragem,
- Quantidade de variáveis de entrada para o modelo.
- Tempo de atraso entre as variáveis.

- Normalização dos dados,

Os dados obtidos a partir dos experimentos foram separados por bateladas. Os dados foram filtrados utilizando média móvel para 1 minuto, no caso, 13 amostras. As técnicas para determinar tempo de amostragem e variáveis significantes para o processo foram utilizadas em cada conjunto de dados. A utilização dos métodos para análise não são afetados por causa da utilização da média móvel como filtro.

#### 4.2.1. Escolha do Tempo de Amostragem

O tempo de amostragem é importante porque se este for muito pequeno, o conjunto de dados obtidos será muito grande e os mesmos terão informações redundantes, já que a dinâmica do processo é lenta. Por outro lado, se o tempo de amostragem adotado for muito grande, informação será perdida, e a rede não será capaz de simular a dinâmica do processo com precisão.

Para determinar o tempo de amostragem ideal, utilizou-se a técnica de auto-covariância cruzada. A técnica consiste na utilização de análise de covariância da variável de interesse, por ela mesma, por isso, auto covariância. Porém, desloca-se a variável no tempo para que seja obtido um conjunto de covariâncias com o dobro do número de dados, já que a variável é deslocada para tempos passados e futuros.

A técnica na é apresentada por Aguirre (2007), que busca pela melhor taxa de decimação para ser utilizado no conjunto de dados, partindo do princípio de um conjunto de dados superamostrado. A covariância cruzada entre dois conjuntos de dados chamados de  $x$  e  $y$  é realizada utilizando a Equação 4.1, onde  $N$  é o número de dados, e  $\tau$  é o termo que indica o valor de atraso ou adiantamento dos valores de  $x$  em relação a  $y$ . Então, para cada  $\tau$  é determinada uma covariância.

$$\text{Cov}_{x,y}(\tau) = \frac{1}{N} \cdot \sum_{k=1}^N [(x(k) - \bar{x}) \cdot (y(k - \tau) - \bar{y})] \quad (4.1)$$

A técnica de auto covariância cruzada foi utilizada na variável de interesse,  $T_R$ , variável de saída do modelo, nos 4 experimentos. A Figura 4.7 mostra o gráfico obtido a partir dessa técnica.

Assim, é possível identificar, no gráfico das covariâncias pelos tempos de atraso deslocado, onde a variável tem a maior e menor correlação, com ela mesma. O importante nessa técnica, é a obtenção do primeiro mínimo, sendo o valor que será utilizado para determinar a taxa de decimação. O primeiro ponto de mínimo a partir do zero é do experimento 6, que ocorre a  $\tau = -132$  pontos de amostra.

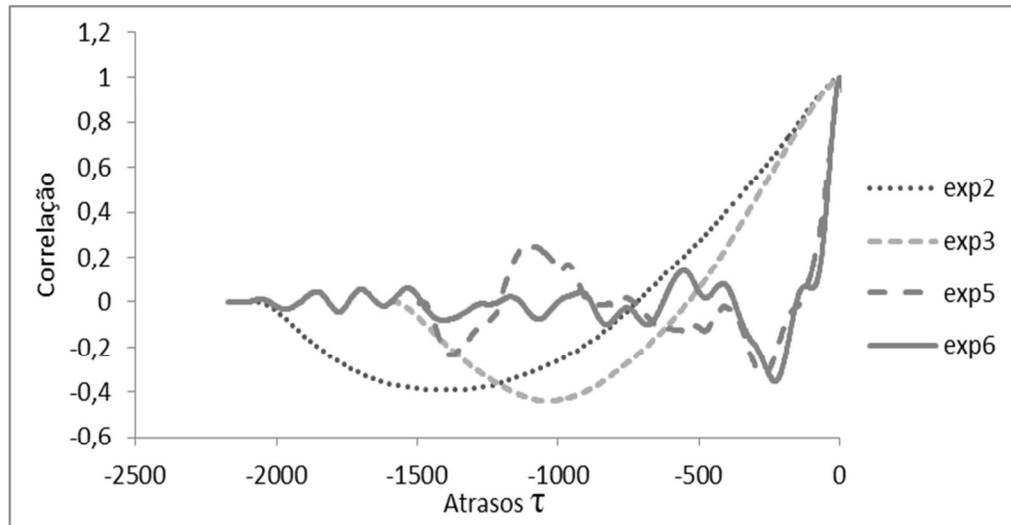


Figura 4.7 - Auto-Covariância Cruzada de TR para os 4 conjuntos de dados.

O menor valor de atraso é utilizado porque garante que todos os outros casos não tenham problema com informação escassa para o modelo. O valor encontrado é utilizado na equação 4.2, onde existe uma relação do valor do primeiro mínimo na relação de auto-covariância cruzada com a taxa de decimação  $\Delta$ . Usando a metodologia apresentada para encontrar o valor de decimação ( $\Delta$ ), nesse caso, temos:

$$10 \cdot \Delta \leq \tau \leq 20 \cdot \Delta \quad (4.2)$$

$$6 \leq \Delta \leq 14 \quad (4.3)$$

Pelo método empregado, considerando o valor mínimo  $\Delta = 6$ , tem-se que é possível utilizar um tempo de amostragem (T) igual a:

$$T = \Delta \cdot t = 6 \cdot 5 = 30s \quad (4.4)$$

Para garantir não perder informação do processo, optou-se utilizar os dados com  $T = 20s$ , um valor menor que o determinado pela técnica, pois as análises realizadas apenas servem como aproximações, já que as equações utilizadas são para variáveis com comportamento linear. O número de dados totais são de 7329, de 5 em 5 segundos. Considerando o  $T = 20s$ , o número de dados totais foi reduzido a 1831.

#### 4.2.2. Variáveis Significativas

Embora o número de variáveis seja pequeno nesse caso, 5 variáveis ( $T_R$ ,  $T_E$ ,  $T_S$ ,  $P$ ,  $Dens$ ), optou-se por analisar variáveis mais influentes na estimativa de  $T_R$ , pois dessa forma diminui-se o esforço computacional no momento de determinação dos parâmetros do modelo, como também na simulação do processo.

Adotou-se realizar análise de sensibilidade, analisando quais variáveis influenciam mais a variável de interesse ( $T_R$ ) como saída do modelo, por meio de técnicas de análise de covariância. A determinação de variáveis importantes para o modelo foi feita por análise de correlação cruzada das principais variáveis do processo com a variável de interesse ( $T_R$ ). A partir dessa análise é possível perceber quais variáveis e em qual tempo apresentam maior correlação com  $T_R$ . A Figura 4.8 ilustra o resultado obtido.

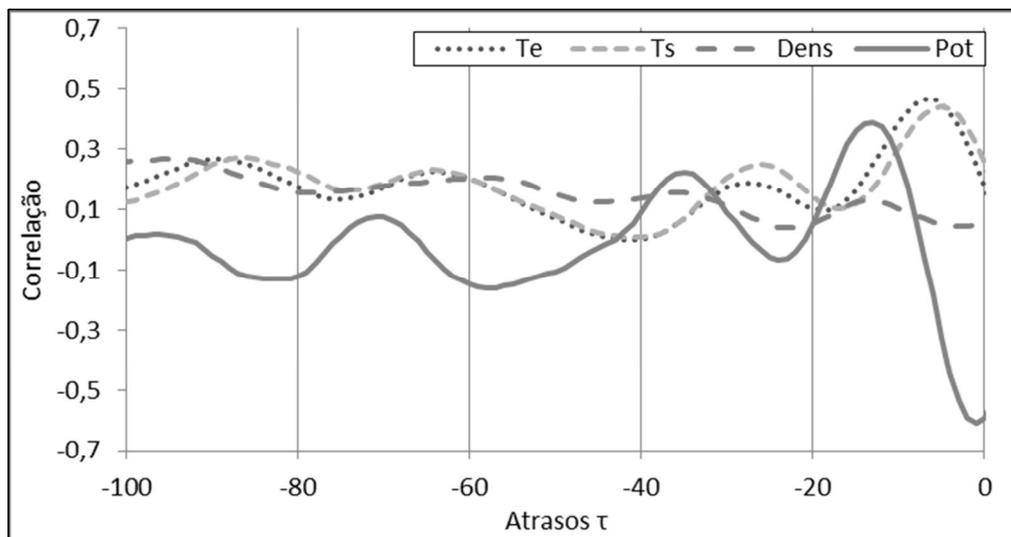


Figura 4.8 - Correlação Cruzada das variáveis de processo com  $T_R$ .

Pela análise, a variação da massa específica é a que tem menos a contribuir com as variações da variável  $T_R$ . Inicialmente, para o modelo neural, optou-se por escolher  $P$  e  $T_E$ , já que são as variáveis com maior facilidade de manipulação.

Em relação ao tempo morto entre as variáveis, existe um tempo de atraso para a variável de saída responder a uma perturbação de outras variáveis influentes. Esse tempo tem que ser levado em consideração para uma melhor qualidade do modelo. Pode-se determinar durante o processo, o tempo morto aproximado, ou fazer análise dos dados pela covariância cruzada, verificando em quanto tempo a variável de saída tem maior correlação com a variável de entrada.

Para estimar o valor de  $T_R[k]$ , no instante de tempo  $k$ , será utilizado  $P[k-2]$  e  $T_E[k-6]$ . Além dessas duas variáveis, também é considerada a realimentação da variável  $T_R$  como entrada do modelo, ou seja, para esse caso, utilizando  $T_R[k-1]$ .

#### 4.3. REDE NEURAL ARTIFICIAL

A técnica de RNA, a princípio, pode ser considerada simplesmente um ajuste de parâmetros para o desenvolvimento de um modelo. Tendo como maior potencial, sua capacidade de generalizar o problema, dessa forma, por causa de dados descontínuos, gerar uma boa simulação do processo, com a finalidade de prevê-lo. O modelo RNA é mostrado na Equação 4.5.

$$y = f_2 \left( \sum_{j=1}^{NN} \mathbf{w}''_{1,j} \cdot f_1 \left( \sum_{i=1}^{NE} \mathbf{w}'_{j,i} \cdot u_i \right) + \mathbf{b}'_j \right) + \mathbf{b}''_1 \quad (4.5)$$

Onde  $y$  é a variável de saída do modelo,  $u$  são as variáveis de entrada,  $NE$  o número de variáveis de entrada,  $NN$  o número de neurônios na camada intermediária,  $\mathbf{w}'$  e  $\mathbf{w}''$  são as matrizes ( $NN \times NE$  e  $1 \times NN$ , respectivamente) de pesos (parâmetros ajustáveis) referentes as camadas de entrada e de saída da rede, e  $\mathbf{b}'$  e  $\mathbf{b}''$  são as matrizes de “bias” referente as camada intermediária e camada de saída. Já  $f_1$  e  $f_2$  são as funções de ativação. As mais comuns são funções não lineares, contínuas e limitadas, funções do tipo sigmoidal. Duas

funções bastante utilizadas no caso das RNA são do tipo tangente hiperbólica e a função logística, mostradas nas equações (4.6) e (4.7), respectivamente.

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (4.6)$$

$$\text{logsig}(x) = \frac{1}{1 - e^{-x}} \quad (4.7)$$

A vantagem de unir com RNA no Excel<sup>®</sup> é que a simulação apenas depende de operações matemáticas simples, e a multiplicação de matrizes também é realizada com facilidade na planilha. O treinamento (ajuste dos parâmetros) necessita de rotinas de otimização.

Foi desenvolvida uma planilha para treinamento de RNA. Essa utiliza o *Solver* para ajuste dos parâmetros. Isso permitiria o treinamento “on-line” da rede, permitindo a sua adaptação à diferentes condições de operação do reator.

#### 4.3.1. Treinamento da Rede Neural Artificial

Foi proposta uma rede neural na qual os neurônios de entrada são as variáveis  $P[k-1]$ ,  $T_E[k-5]$ , e  $T_R[k]$ , e único neurônio de saída a variável  $T_R[k+1]$ , como mostrado na Figura 4.9. Desta forma, a rede proposta possibilita a estimativa da temperatura de reação em um instante futuro, a partir do conhecimento da potência em um instante anterior, a temperatura de óleo em cinco instantes anteriores, e a temperatura do reator no instante atual.

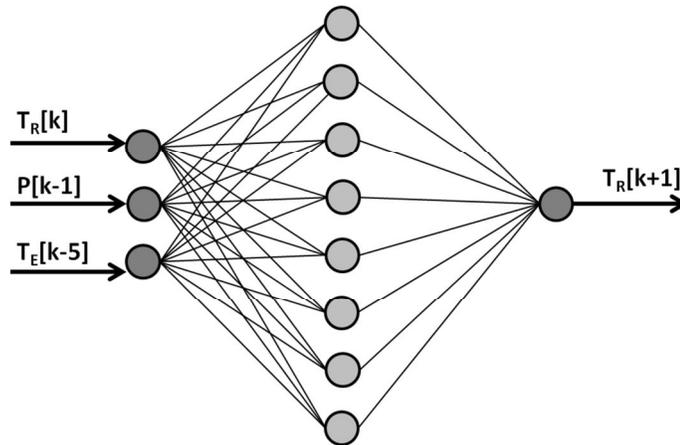


Figura 4.9 - Topologia da RNA utilizada, com 3 entradas, 8 neurônios intermediários e 1 saída.

A RNA foi treinada utilizando o algoritmo *trainbr*, com funções de ativação *tansig* e *purelin*. Os dados foram normalizados entre -1 e 1, para melhor desempenho da RNA, pois pode ocorrer de variáveis de maior magnitude que outras influencie nos pesos durante o treinamento, gerando um modelo não adequado para todas as condições.

O modelo possui 3 entradas, como citado anteriormente, e 8 neurônios na camada intermediária. Para treinamento foram utilizados dados de três experimentos, sendo um sem controle e dois com controle manual. Como conjunto de validação foi utilizado um experimento com controle. A Figura 4.10 mostra a regressão linear do melhor modelo neural obtido para esse caso.

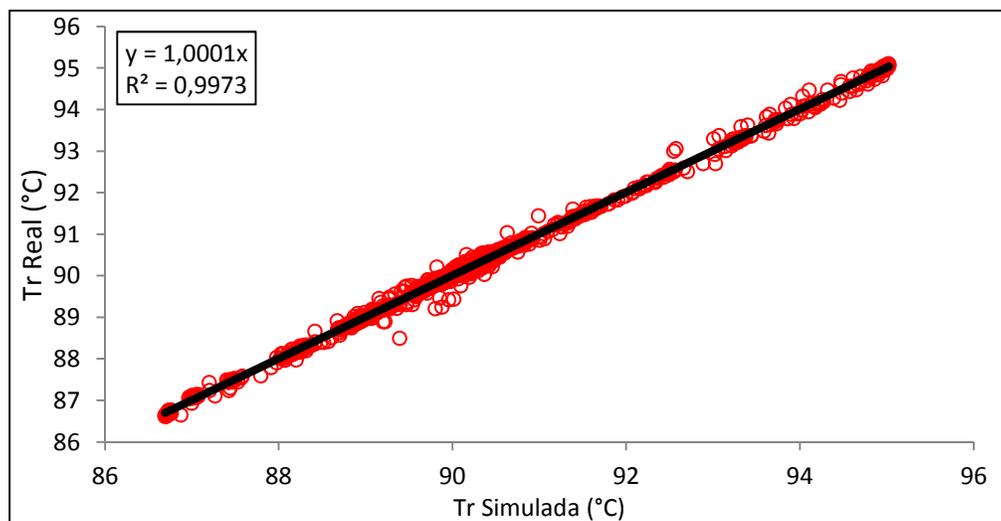


Figura 4.10 - Regressão Linear para validação do modelo neural de TR.

O número de 8 neurônios para a camada intermediária foi determinado sendo o menor número de neurônios encontrado, durante alguns testes de treino, para conseguir resultados aceitáveis. Nesse caso, foram utilizados 1241 dados para treino e 538 para validação. E a estimativa de  $T_R$  é satisfatória, visto que o coeficiente de rendimento  $R^2 = 0,9973$ .

Os parâmetros (pesos) obtidos a partir deste treinamento no ambiente Matlab podem ser importados para o ambiente de supervisão em Excel e utilizados como modelo neural do processo. Uma vez que não exige mais o procedimento de treinamento, o cálculo da saída de uma rede neural em função de dados de entrada é extremamente rápido.

## 5. CONCLUSÕES

O objetivo original deste trabalho foi o desenvolvimento de um sistema computacional de supervisão e controle de processos químicos no ambiente Microsoft Excel<sup>®</sup>, com a finalidade de aplicar uma estratégia de controle MPC em um processo de polimerização de estireno. Para isso, numa primeira etapa, a planta piloto existente no laboratório e Engenharia de Sistemas Químicos foi modificada para permitir uma operação mais eficiente.

Numa segunda etapa foi desenvolvido um sistema supervisorio, em linguagem VBA, no ambiente Excel<sup>®</sup>. O aplicativo implementado utiliza o padrão OPC para a comunicação com o PLC da planta, e possibilita a aquisição e envio de dados do e para o processo, além do armazenamento dos dados em arquivos. O uso de recursos da planilha também permite a entrada dados por meio de janelas, e a visualização dos parâmetros do processo por meio de gráficos.

O sistema supervisorio desenvolvido apresentou resultados satisfatórios, sendo capaz de receber e enviar sinais para a planta e armazenar o histórico de dados das variáveis de processo. Como a rotina foi desenvolvida em VBA torna-se possível a implementação de novos algoritmos de tratamento de dados, identificação do processo e de estratégias de controle.

Numa etapa posterior o programa foi utilizado para acompanhar e controlar manualmente as variáveis de processo durante a polimerização de estireno. Foi verificado que a temperatura do reator podia ser controlada em uma faixa bastante estreita por meio da manipulação da potência dissipada da resistência, manipulando indiretamente a temperatura do óleo.

A identificação do sistema foi realizada para gerar um modelo empírico do processo, com a finalidade de uma possível aplicação numa estratégia de controle MPC. Utilizando a técnica de Redes Neurais Artificiais foi obtido um modelo para simular o comportamento da temperatura do meio reacional em função de valores passados de potência e temperatura do óleo, e o valor atual da temperatura de reação. O modelo foi validado utilizando dados do processo, mostrando um bom desempenho. Por apresentar

uma notação matricial, as Redes Neurais podem ser facilmente simuladas no Excel<sup>®</sup> na própria planilha, e utilizadas como modelo em algoritmos de controle.

#### 5.1. SUGESTÃO PARA TRABALHOS FUTUROS

Assim, fica como uma proposta para continuação deste trabalho a utilização do sistema implementado para desenvolver, implementar e avaliar diferentes estratégias de controle de processos, com o objetivo de direcionar a evolução da reação para obter um polímero com as qualidades desejadas.

## REFERÊNCIAS

AGUIRRE, L. A., 2007, **Introdução à Identificação de Sistemas** - Técnicas Lineares e Não-Lineares Aplicadas a Sistemas Reais, 3º ed, UFMG, Minas Gerais, Brasil.

ALIANE, N. Data acquisition and real-time control using spreadsheets: interfacing Excel with external hardware. **ISA transactions**, v. 49, n. 3, p. 264–9, doi:10.1016/j.isatra.2010.03.009, 2010.

BIYANTO, T.; WIDJIANTORO, B. e JABAL, A. Artificial neural network based modeling and controlling of distillation column system. **International Journal of Engineering, Science and Technology**, v. 2, n. 6, p. 177–188, 2011.

CHEN, S. e BILLINGS, S. a. Neural networks for nonlinear dynamic system modelling and identification. **International Journal of Control**, v. 56, n. 2, p. 319–346, doi:10.1080/00207179208934317, 1992.

FINDEISEN, R. An introduction to nonlinear model predictive control. **Benelux Meeting on Systems and Control**, 2002.

FUJIKI, T. L. **Desenvolvimento de Controlador Digital Baseado em um Modelo Neural e um Otimizador de Uma Planilha Eletrônica**. Universidade Estadual de Campinas - [S.l.]. 2009.

HOSEN, M. A.; HUSSAIN, M. A. e MJALLI, F. S. Control of polystyrene batch reactors using neural network based model predictive control (NNMPC): An experimental investigation. **Control Engineering Practice**, v. 19, n. 5, p. 454–467, doi:10.1016/j.conengprac.2011.01.007, 2011.

KASHANI, M. N. e SHAHHOSSEINI, S. A methodology for modeling batch reactors using generalized dynamic neural networks. **Chemical Engineering Journal**, v. 159, n. 1-3, p. 195–202, doi:10.1016/j.cej.2010.02.053, 2010.

KURODA, C. e KIM, J. Neural network modeling of temperature behavior in an exothermic polymerization process. **Neurocomputing**, v. 43, n. 1-4, p. 77–89, doi:10.1016/S0925-2312(01)00621-X, 2002.

LEITE, M. S. **Projeto, Montagem e Instrumentação de um Protótipo Experimental de Sistema de Polimerização para o Desenvolvimento e Implementação de Diferentes Técnicas de Controles Inteligentes**. Dissertação (Doutorado em Engenharia Química) - Campinas - SP. 2011.

MOURA, L. F. de. **Excel para engenheiros**. 1º. ed. São Carlos: [s.n.], 2009. p. 151

PRASAD, V. e BEQUETTE, B. W. Nonlinear system identification and model reduction using artificial neural networks. **Computers & Chemical Engineering**, v. 27, n. 12, p. 1741–1754, doi:10.1016/S0098-1354(03)00137-6, 2003.

PUDA, A. P. Padronização da Comunicação Através da Tecnologia Opc. In: ISATEC RIO DE JANEIRO. **Anais...** [S.l.: s.n.]. Disponível em: <<http://www.isarj.org.br/artigos.php>>, 2009.

SANTOS, B. F. Dos. **Desenvolvimento de um Controlador Preditivo Baseado em Redes Neurais para Aplicação em uma Planta de Polimerização de Estireno**. Universidade Estadual de Campinas - [S.l.]. 2011.

SANTOS, R. R. C. et al. Application of Artificial Neural Networks in an Experimental Batch Reactor of Styrene Polymerization for Predictive Model Development. **Chemical Engineering Transactions**, v. 32, 2013.

# APÊNDICE I

Artigo publicado na Chemical Engineering Transactions, volume 32,2013.

[DOI:10.3303/CET1332234](https://doi.org/10.3303/CET1332234)

## Application of Artificial Neural Networks in an Experimental Batch Reactor of Styrene Polymerization for Predictive Model Development

Raphael R. C. Santos<sup>\*</sup>, Brunno F. Santos, Ana M. F. Fileti, Flávio V. da Silva, Roger J. Zemp

State University of Campinas, Chemical Engineering Faculty. Cidade Universitária Zeferino Vaz, Av. Albert Einstein, 500 - CEP 13083-852 - Campinas - SP - Brasil  
rrcs.al@gmail.com\*

Batch reactors are widely used in the polymer industry, especially for multi-purpose processes where different types of polymers are produced on demand. Batch polymerization reactors impose rather great operational difficulties due to the complex reaction kinetics and inherent process nonlinearities. Thus it is a difficult task to develop mathematical models for polymerization processes. If required for process control purposes the model should be accurate but simple, so that it can be used in a control loop. The present work shows the application of the neural network approach in the development of a predictive model for a styrene polymerization pilot plant, located at the Laboratory of Chemical Systems Engineering, School of Chemical Engineering at UNICAMP. Artificial Neural Networks have become a usual application in many areas of engineering, and are well suited for chemical processes due to their ability to describe multi-variable non-linear models. However to control purposes, the consideration of the variation of the process variables in real time is required as input to the model, to ensure the representation at the dynamics of the process. The experimental prototype consists of a jacketed stirred reactor, using thermal fluid as a heat source. Reaction progress was measured by a density sensor situated in a external recycle loop. Temperature sensors were positioned both inside the reactor and in the inlet and outlet of the jacket. Traditional *feedforward* neural networks with back-step inputs and the *Elman* network were applied to obtain the best model to be employed in a control loop. A comparison between the networks was performed, showing that, for process dynamics modeling, both networks were able to create suitable polymerization models.

### 1. Introduction

One of the many advantages of batch processes is ability to allow for the change of operating conditions, both at start up and along the reaction, in order to conduct them to desired products and material properties. According to Hosen et al. (2011), it is common to manufacture different products in the same equipment, for instance the polymer industry, where batch reactors are used to produce materials aggregate to demands of all potential customers in a specific market.

Many batch processes present serious difficulties for control due, especially, to the inherent complexity of the involved reactions, like the nonlinear behaviour and with properties that vary whole batch period, namely, require the frequent change of operating parameters (Galván et al. 1997). Lastly, the results are products yield lower than expected.

Searching for performance improvements of process is usual to develop mathematical models for the dynamic simulation of the systems, so that studies to improve yield, cost minimization and test of new control strategies can be performed.

The development of phenomenological models requires much time and can lead to high costs due the range of operating conditions that have to be considered, and the large number of components and reactions involved.

Also, the additional cost might not be compensated for, as many batch reactors are used for smaller scale production.

In this context, a faster way of developing models for control purposes is the so-called system identification, which uses mathematical models to create relationship between process variables, using actual data from the process, but without resorting to phenomenological models. Through optimization techniques the model parameters are adjusted so that the prediction model performs close to the actual dynamic behaviour. In this field artificial neural networks (ANN) have become of widespread use. According Chen et al. (1992), the ANNs have the ability to be adjusted as nonlinear models, being well suited to be used as models in system identification.

## 2. Experimental Configuration

The styrene polymerization pilot plant, located at the Laboratory of Chemical Systems Engineering, School of Chemical Engineering at UNICAMP, was used for the process data acquisition. The experimental prototype consists of a stirred jacketed reactor (total volume of 1.2 L). Thermal oil was used as heat transfer medium in the jacket. An electrical heater, connected to a thyristor, provided heating to the thermal fluid inside a storage tank. The dissolved oxygen was purged by bubbling pure nitrogen gas through the reaction mixture. The monomer was obtained with 99% purity from *Sigma Aldrich*. Toluene was used as solvent and it was purchased from *Ecibra* with a purity of 99%. No further purification was used. The initiator agent for the reaction was benzoyl peroxide (BPO) from *Sigma Aldrich*, presenting 70% purity. There are temperature sensors inside the reactor and in the inlet and outlet of the jacket. A scheme layout of the prototype is shown in Figure 1.

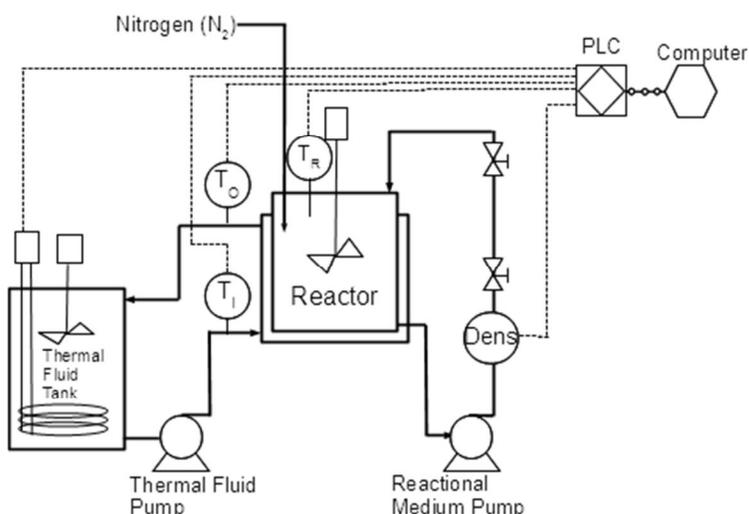


Figure 1: Scheme layout of the polymerization prototype.

Table 1: Experimental Conditions of the Styrene Solution Polymerization Reactor

| Parameter               | Value        |
|-------------------------|--------------|
| Reaction volume         | 800 mL       |
| Reactor stirring speed  | 200 rpm      |
| Jacket flow rate        | 300 L/h      |
| Heating power           | 3000 W       |
| Initiator concentration | 0.0185 mol/L |

After loading the monomer and the solvent into the reactor, the system was heated to reach the desired operating temperature of 90 °C (Ghasem et al., 2007), when the initiator BPO was added to start the polymerization reaction. Typical experimental operating conditions are shown in Table 1. Experimental runs were conducted using several monomer/solvent ratios: 30, 50 and 70 V/V %.

For this kind of reaction the prediction of reactor temperature is important for product quality. Therefore, the aim of the sought model was to predict reaction temperature  $TR$  at time  $k + 1$ , given the following process

variables at time  $k$ : actual reactor temperature  $TR [k]$ , jacket inlet temperature  $Ti [k]$  and the power  $P [k]$  being sent for the oil heater. In the *feedforward* topology the same variables were inputted at time  $[k-1]$  and  $[k-2]$ .

### 3. Neural Networks Models

Artificial Neural network is an artificial intelligence technique that has been widely used in the chemical industry for the development of nonlinear process models, due to its great ability to generalize and represent nonlinear systems satisfactorily. These models are of the black-box type as the model is generated by adjusting parameters (weights and bias) until the error between the dummy variable and real variable is small enough. In the case of ANN's the building blocks of the models are the neurons, which generate an output signal by processing an input signal through an activation function.

The ANN is built by the union of a several neurons. Figure 2 illustrates the topology of a typical *feedforward* network with three inputs, one output and an intermediate layer.

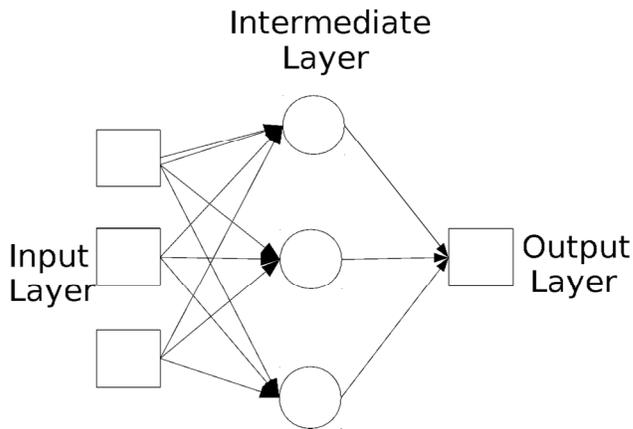


Figure 2: Topology layout of a feedforward ANN.

In the *feedforward*, the input signals are processed from left to right as shown in Figure 2. For a network with two neurons, one in each layer, see Figure 3, the output  $Y$  as a function of input  $X$  is given by:

$$Y = f_2(w_2 \cdot f_1(w_1 \cdot X + b_1) + b_2) \quad (1)$$

where  $f_1$  and  $f_2$  are the activation functions,  $w_1$  and  $w_2$  the weights, and  $b_1$  and  $b_2$  the bias of neurons 1 and 2, respectively.

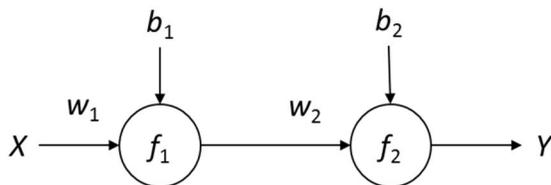


Figure 3: Feedforward ANN topology for 1 neuron in each layer

In an *Elman* network (recurrent network), there is a return signal from the output of the intermediate layer neuron to the input of the same neuron, see Figure 4. This signal is delayed by one time step and multiplied by a weight ( $w_2$ ) thus allowing for the consideration of the dynamic behaviour of the process.

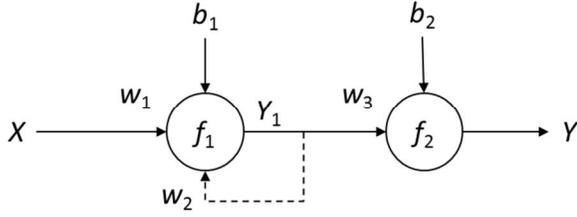


Figure 4: Elman ANN topology for 1 neuron in each layer

The equation representing the output of an *Elman* network is shown by Eq(2) and Eq(3).

$$Y[k] = f_2(w_3 \cdot f_1(w_1 \cdot X[k] + w_2 \cdot C + b_1) + b_2) \quad (2)$$

where,

$$C = Y_1[k - 1] \quad (3)$$

Compared to the *Elman* network the *feedforward* network is much simpler. For a two-neuron network the *feedforward* network has four adjustable parameters, two weights ( $w$ ) and two bias ( $b$ ). An *Elman* network of the same size requires five adjustable parameters, three weights ( $w$ ) and the two bias value ( $b$ ). With the growth in the number of inputs and neurons in the intermediate layer, it is expected for the *Elman* ANN to become harder to be trained, as the number of parameter to be fitted increases. For both cases the activation functions are of a non-linear type, usually of sigmoid shape.

#### 4. Results and Discussion

The dynamic behaviour of the reaction system was measured in four different batches, resulting in 9068 samples. Due to the short period of time used to sample variables five seconds from each batch run the number of data points used for the ANN training had to be assessed. The training data was evaluated by a cross-covariance technique, as suggested by Aguirre (2007). This technique is a simple heuristic method to identify the most adequate time span between data points, so avoiding either redundancy or lack of suitable data points for training, and is based on the covariance of the sampled data:

$$\text{Cov}_{x,y}(\epsilon) = \frac{1}{N} \cdot \sum_{k=1}^N [(x(k) - \underline{x}) \cdot (y(k - \epsilon) - \underline{y})] \quad (4)$$

where:

- $N$  is the number of elements in vectors  $x$  and  $y$ .
- $x$  and  $y$  are the input and output vectors, respectively
- $\underline{x}$  and  $\underline{y}$  are the mean values of the respective vectors
- $\epsilon$  is the number of delays

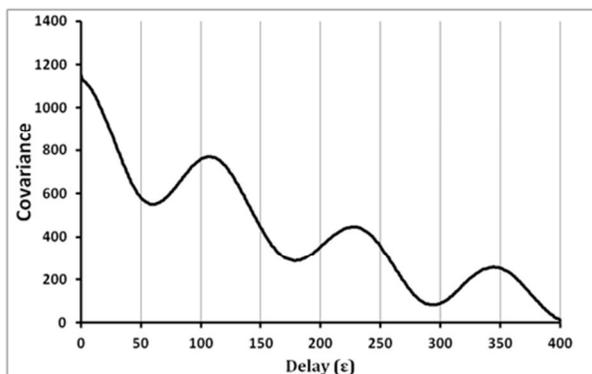


Figure 5: Covariance as a function of delay

Applying the technique to the sampled data leads to the identification of the first local minimum, in Figure 5, leading to an optimal sampling time of 20 s, and a reduction of the data set from 9068 to 453 points.

Once the training data set was obtained, the dynamic behaviour of the reaction systems was modelled with both the *Elman* and *feedforward* ANNs. The networks were developed and trained using *Matlab*<sup>®</sup> 7.0, Neural Networks Toolbox.

Both networks showed a good performance with regard to minimizing the error for training and validation, obtaining correlation coefficient of 0.96. Figure 6 a comparison between both networks is presented, showing a good agreement between actual and estimated data. In both cases a maximum variation of 0.5% for the reactor temperature was achieved.

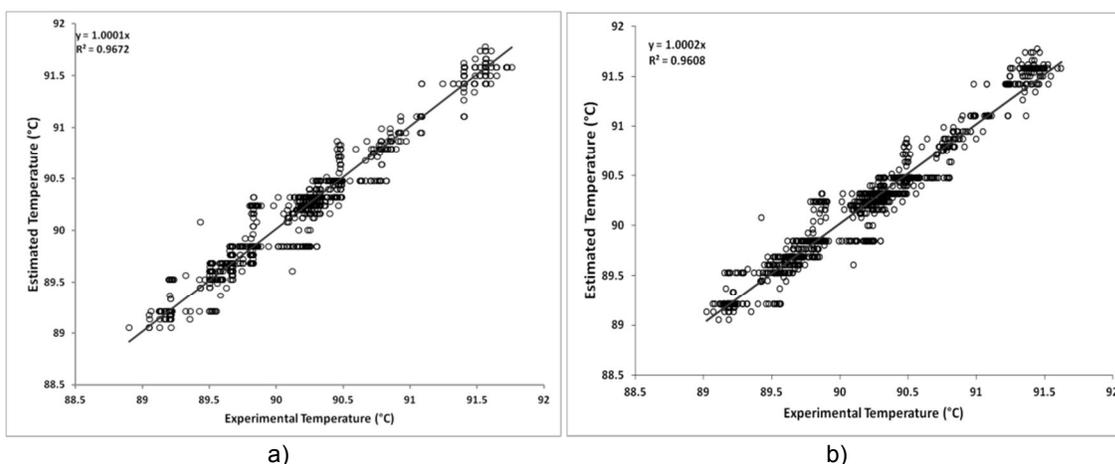


Figure 6: Comparison between performance on the validation of a) *Elman* ANN and b) *Feedforward* ANN.

Other parameters were also evaluated in order to point out differences between these two types of ANN. Table 2 shows additional information about time, training algorithms and number of adjustable parameters. Although the performance to predict reaction temperature is similar for both cases, the number of parameters and training time for *Elman* network is much greater than for the *feedforward* network of the same topology.

A problem found in the *Elman* network is the difficulty in training, as it was not possible to use a training procedure with bayesian regularization, which avoids overfitting. The use of the Levenberg–Maquardt method increased the training time for the *Elman* network 28 fold, compared to the *feedforward* network. The best performance of the *Elman* network was only obtained when the hyperbolic tangent function for the intermediate and output layers was chosen, which demands greater complexity to the model.

In point of view of the model implementation in a control loop, the *Elman* network is also more complex as it requires more memory to store signal until the next loop.

Table 2: Neural Networks Performance Comparison

| Parameters | Elman | Feedforward |
|------------|-------|-------------|
|------------|-------|-------------|

|                       |                       |                       |
|-----------------------|-----------------------|-----------------------|
| R <sup>2</sup>        | 0.9672                | 0.9608                |
| MSE (°C)              | 1.87x10 <sup>-3</sup> | 1.92x10 <sup>-3</sup> |
| Topology <sup>1</sup> | 3x3                   | 9x20                  |
| Train Time(s)         | 60.19                 | 2.18                  |
| Transfer Functions    | Tansig/Linear         | Tansig/Tansig         |
| Train Algorithm       | LM <sup>2</sup>       | BRB <sup>3</sup>      |
| Number of Weights     | 21                    | 200                   |
| Number of Bias        | 4                     | 21                    |

1. Number of input neurons and neurons in the intermediate layer.
2. Levenberg-Maquardt algorithm
3. Bayesian regularization backpropagation algorithm.

## 5. Conclusions

In this work an application of artificial neural networks to the identification of a polymerization system was presented. With data obtained from four experiments, the *Elman* neural networks and traditional *feedforward* were trained to model the dynamic behaviour of the process and used to estimate the reactor temperature. The comparison between the performances of both networks showed that although *Elman* networks intrinsically have the capacity to describe dynamic models, they suffer in aspects related to the computational effort and training time, and in implementation in a real system due to the additional complexity of requiring intermediate storage.

## References

- Aguirre, L. A., 2007, An Introduction to System Identification – Linear and Nonlinear Techniques Applied to Real Systems, 3<sup>o</sup> ed, UFMG, Minas Gerais, Brazil. (In Portuguese).
- Chen, S.; Billings, 1992, S. A. Neural networks for nonlinear dynamic system modelling and identification. *International Journal of Control*, 56, 319-346.
- Galván, I. M. E Zaldívar, J. M., 1998, Application of recurrent neural networks in batch reactors Part II: Nonlinear inverse and predictive control of the heat transfer fluid temperature. *Chemical Engineering and Processing*, 37, 149-161.
- Ghasem, N. M., Sata, S. A. and Hussain, M. A., 2007, Temperature Control of a Control of a Bench-Scale Batch Polymerization Reactor for Polystyrene Production, *Chemical Engineering Technology* 30, 9: 1193-1202.
- Hosen, M. A.; Hussain, M. A.; Mjalli, f. S. 2011, .Control of polystyrene batch reactors using neural network based model predictive control (NNMPC): An experimental investigation. *Control Engineering Practice*, 19, 454-467.
- Leite M.S., Santos B.F., Lona L.M., Silva F.V., Fileti A.M.F., 2011, Application of Artificial Intelligence Techniques for Temperature Prediction in a Polymerization Process, *Chemical Engineering Transactions*, 22, 385-390, DOI: 10.3303/CET1124065.

## APÊNDICE II

Listagem dos códigos em VBA desenvolvidos.

### **Workbook**

```
Private Sub Workbook_Open()  
    Plan1.Auto_Open  
End Sub
```

### **Módulo**

```
Option Base 1  
Option Explicit  
  
Dim tcontrol As Double  
Public a As Integer  
Public t As Double  
Public outputs(8, 1) As Double  
Public inputs(5, 1) As Double  
Public flag1 As Boolean  
Dim erro_int As Double  
  
Sub principal()  
    'On Error GoTo ErrorTrap  
    Dim tempo As Date  
    Dim samptime As Date  
    Dim nexttime As Date  
    'Application.Calculation = xlCalculationManual  
    Application.ScreenUpdating = False  
    'your code here  
  
    'Variáveis de Tempo  
    If t = Empty Then  
        t = Time  
        a = 2  
    End If  
    tempo = Time
```

```

sampletime = TimeValue(Plan1.Txt_sample.Text)
nexttime = sampletime + tempo

tcontrol = CDb1(sampletime) * (1000 / 0.011574)
If Plan1.controlePI.Value = "Verdadeiro" Then
'controladorPI
End If

```

Supervisorio

Aquisicao

```

'Chamada da função principal#####
If flag1 = True Then
Application.OnTime (nexttime), "principal"
End If
'#####

```

```

'Comandos para otimização do código#####
'Application.Calculation = xlCalculationAutomatic
DoEvents
Application.Cursor = xlNorthwestArrow
'Application.ScreenUpdating = True
'#####
End Sub

```

```

Sub Escrita()
Application.ScreenUpdating = False
On Error Resume Next:
With Plan1
inputs(1, 1) = .txt_bomb_oleo.Text: inputs(2, 1) =
.txt_Pot_Resistor.Text:
inputs(3, 1) = .txt_Bomb_Reator.Text: inputs(4, 1) = .txt_SP.Text
End With
End Sub

```

```

Sub Supervisorio()
With Plan1

```

```

.txt_Temp1.Text      =      Format(outputs(1,      1),      "##,##0.00"):
.txt_temp2.Text      =      Format(outputs(2,      1),      "##,##0.00"):
.txt_temp3.Text = Format(outputs(3, 1), "##,##0.00")
.txt_dens.Text = Format(outputs(5, 1), "##,##0.00")
If .controlePI.Value = "Verdadeiro" Then
.txt_Pot_Resistor.Text = outputs(8, 1)
End If
End With
'temp reator      'ent camisa      ' sai camisa
End Sub

```

```

Sub Aquisicao()
Dim anitem As OPCItem
Dim aq_sample_tempo As Double
'Application.ScreenUpdating = False
aq_sample_tempo = Cdbl(Time - t) * (1000 / (0.011574))

a = a + 1
With Plan2
.Cells(a, 1) = Format(aq_sample_tempo, "##,##0.00"):
.Cells(a, 2) = Format(outputs(1, 1), "##,##0.00"): .Cells(a, 3) =
Format(outputs(2,      1),      "##,##0.00"):      .Cells(a,      4)      =
Format(outputs(3, 1), "##,##0.00")
.Cells(a, 5) = Format(outputs(5, 1), "##,##0.00")
.Cells(a, 6) = inputs(1, 1): .Cells(a, 7) = inputs(2, 1):
.Cells(a, 8) = inputs(3, 1):
.Cells(a, 9) = Plan1.txt_SP.Text: .Cells(a, 10) = outputs(8, 1)
End With
End Sub

```

```

Sub controladorPI()
Dim erro, dh, Kc, Ki, y, SP As Double
'Application.ScreenUpdating = True
'dh = Timer - tcontrol
'Plan2.Cells(a, 9) = erro_int
With Plan1
Kc = .txt_Kc.Text
Ki = .txt_Ki.Text
SP = .txt_SP.Text
erro = SP - outputs(2, 1)

```

```

Plan1.Cells(6, 10) = erro
Plan1.Cells(5, 10) = tcontrol
If Abs(erro) < 1 Then
erro_int = erro_int + erro * tcontrol
y = Kc * erro + Kc * Ki * erro_int
Else
y = Kc * erro
erro_int = 0
End If
'Plan1.Cells(1, 11) = y
'Plan1.Cells(1, 12) = Kc
'Plan1.Cells(1, 13) = erro_int
If y < 0 Then
y = 0
End If

If y > 50 Then
y = 50
End If

'plan1.Cells(12, 1) = y
.txt_Pot_Resistor.Text = y

End With
End Sub

```

### **Módulo de Classe - OPCDataAccess**

```

Option Explicit
Option Base 1

```

```

Const MaxItemCount = 100

```

```

Dim WithEvents OPCMyserver As OPCServer 'OPCfT [fo,ÌfIfufWfFfNfg
Dim WithEvents OPCMygroups As OPCGroups ' fOf< [fv,ÌfRfEfNfVf†f“
Dim WithEvents OPCmygroup As OPCGroup ' fOf< [fvfIfufWfFfNfg
Dim OPCmyitems As OPCItems ' fAfCfef€f,ÌfRfEfNfVf†f“
Dim OPCMyitem As OPCItem ' fAfCfef€fIfufWfFfNfg
Dim ItemCount As Integer

```

```

Private Sub Class_Terminate()
    Dim Errors() As Long      'fAfCfef€ , ìfGf% [
    On Error Resume Next
    OPCmygroup.IsActive = False
    OPCMygroups.Remove OPCmygroup.ServerHandle

    Set OPCmyitems = Nothing ' fAfCfef€ , ìfRf€fNfVf†f“ í œ
    Set OPCMyitem = Nothing  ' fAfCfef€fIfufWfFfNfg í œ
    Set OPCMygroups = Nothing ' fOf< [fv, ìfRf€fNfVf†f“ í œ
    Set OPCmygroup = Nothing ' fOf< [fvfIfufWfFfNfg í œ
    OPCMyserver.Disconnect   'fT [fo, ì Ø'f
    Set OPCMyserver = Nothing 'OPCfT [fo, ìfIfufWfFfNfg í œ
End Sub

```

```

Function Connect(ServerName As String)
    On Error GoTo ConnectError
    Set OPCMyserver = New OPCServer
    OPCMyserver.Connect ServerName, ""
    Connect = True
    Exit Function
ConnectError:
    Connect = False
End Function

```

```

Function AddGroup(DeviceName As String, DeviceCode As String,
Deviceno As Integer, point As Integer)
    On Error GoTo AddError
    Dim i As Integer
    Dim ItemServerHandles() As Long
    Dim ClientHandles(1) As Long
    Dim OPCItemIDs(1) As String
    Dim Errors() As Long

    Set OPCMygroups = OPCMyserver.OPCGroups
    Set OPCmygroup = OPCMygroups.Add("Group1")
    OPCmygroup.UpdateRate = 1000
    Set OPCmyitems = OPCmygroup.OPCItems
    ItemCount = point
    For i = 1 To ItemCount

```

```

        ClientHandles(1) = i
        If Devicecode = "X" Or Devicecode = "Y" Or Devicecode =
"B" Or Devicecode = "W" Then
            OPCItemIDs(1) = Devicename + "." + _
                Devicecode + Hex$(Val("&h" + Deviceno) + i - 1)
        Else
            OPCItemIDs(1) = Devicename + "." + _
                Devicecode + Format$(Val(Deviceno) + i - 1)
        End If
        OPCmyitems.AddItem 1, OPCItemIDs, ClientHandles,
ItemServerHandles, _
            Errors '', RequestedDataTypes, AccessPaths
        AddGroup = True
        Exit Function
        AddError:
        AddGroup = False
    End Function

```

```

Sub Advise(badvise As Boolean)
    On Error Resume Next
    OPCmygroup.IsActive = badvise
    OPCmygroup.IsSubscribed = badvise
    If badvise = True Then
        Dim anitem As OPCItem
        For Each anitem In OPCmygroup.OPCItems
            If anitem.ClientHandle < 9 Then
                outputs(anitem.ClientHandle, 1) =
                    anitem.Value
            End If
            Set anitem = Nothing
        Next anitem
    End If
End Sub

```

```

Sub UnAdvise()
    OPCmygroup.IsSubscribed = False
    outputs = 0
End Sub

```

```

Sub SyncWrite()

```

```

On Error Resume Next
Dim anitem As OPCItem

    For Each anitem In OPCmygroup.OPCItems
        If anitem.ClientHandle > 10 Then
            anitem.Write inputs(anitem.ClientHandle - 10, 1)
            Set anitem = Nothing
        End If
    Next anitem
End Sub

Sub OPCMygroup_DataChange(ByVal TransactionID As Long, ByVal
NumItems As Long, ClientHandles() As Long, ItemValues() As
Variant, Qualities() As Long, TimeStamps() As Date)

    Dim result, valueX, qualityX, timestampX As Variant
    Dim angell As Variant
    Dim i As Integer
    For i = 1 To NumItems
        outputs(ClientHandles(i), 1) = ItemValues(i)
    Next i
End Sub

Private Sub OPCMyserver_ServerShutDown(ByVal Reason As String)
    MsgBox "Server shutdown"
End Sub

```

### **Planilha 1**

```

Option Base 1
Option Explicit
Dim OPCDA As OPCDataAccess
Dim badvise As Boolean

Sub Auto_Open()
    Dim Getserver As OPCServer
    badvise = "Falso"
    OPCConnect.Enabled = True

```

```

OPCAdvise.Enabled = False
OPCDisconnect.Enabled = False
Dim Servers As Variant
Dim i As Integer
ServerName.Clear
Set Getserver = New OPCServer
Servers = Getserver.GetOPCServers
For i = LBound(Servers) To UBound(Servers)
    ServerName.AddItem Servers(i)
Next i
Set Getserver = Nothing
ServerName.ListIndex = 0
End Sub

Private Sub controlePI_Click()
    Application.ScreenUpdating = False
    If controlePI.Value = "Verdadeiro" Then
        inputs(5, 1) = 1
    Else
        inputs(5, 1) = 0
    End If
    Escrita
    OPCDA.SyncWrite
    DoEvents
    Application.Cursor = xlNorthwestArrow
End Sub

Private Sub OPCConnect_Click()
    Set OPCDA = New OPCDataAccess
    If OPCDA.Connect(ServerName.List(ServerName.ListIndex)) =
False Then
        MsgBox "Connect Error"
        Exit Sub
    End If
    If OPCDA.AddGroup(DeviceName.Text, DeviceCode.Text,
DeviceNo.Value, point.Value) = False Then
        MsgBox "AddGroup/Item Error"
        Exit Sub
    End If
    OPCConnect.Enabled = False

```

```

    OPCAdvise.Enabled = True
    OPCAdvise.Caption = "Advise"
    OPCDisconnect.Enabled = True

    'apagar celulas
    Plan2.Cells.Clear: a = 0
    t = Empty
End Sub

Private Sub OPCAdvise_Click()
    badvise = Not badvise
    OPCDA.Advise (badvise)
    If badvise = True Then
        OPCAdvise.Caption = "UnAdvise"
        flag1 = True
        principal
    Else
        OPCAdvise.Caption = "Advise"
        flag1 = False
    End If
End Sub

Private Sub OPCWrite_Click()
    OPCDA.SyncWrite
End Sub

Private Sub OPCDisConnect_Click()
    Set OPCDA = Nothing
    OPCConnect.Enabled = True
    OPCAdvise.Enabled = False
    OPCDisconnect.Enabled = False
End Sub

Private Sub txt_bomb_oleo_Change()
    On Error Resume Next
    Application.ScreenUpdating = False
    Escrita
    OPCDA.SyncWrite
End Sub

```

```
Private Sub txt_Bomb_Reator_Change()  
    On Error Resume Next  
    Application.ScreenUpdating = False  
    Escrita  
    OPCDA.SyncWrite  
    DoEvents  
    Application.Cursor = xlNorthwestArrow  
End Sub
```

```
Private Sub txt_Pot_Resistor_Change()  
    On Error Resume Next  
    Escrita  
    OPCDA.SyncWrit  
End Sub
```

```
Private Sub txt_SP_Change()  
    On Error Resume Next  
    Application.ScreenUpdating = False  
    Escrita  
    OPCDA.SyncWrite  
    DoEvents  
    Application.Cursor = xlNorthwestArrow  
End Sub
```