

**Universidade Estadual de Campinas**

**(UNICAMP)**

**Faculdade de Engenharia Química**

**(FEQ)**

**Área de Concentração: Sistemas de Processos Químicos e**

**Informática**

**(ACSPQI)**

Desenvolvimento e implementação de um sistema de controle

*fuzzy* e sua aplicação a um reator de polimerização em

batelada de MMA

Aluno: Arnaldo José Bannwart Antunes

Orientadora: Profa. Dra. Ana Maria Frattini Fileti

Co-orientador: Prof. Dr. João Alexandre Ferreira da Rocha Pereira

Tese de doutorado apresentada para  
obtenção de título de doutor em  
Engenharia Química.

Campinas, 22 de fevereiro de 2005.

**UNICAMP**  
**BIBLIOTECA CENTRAL**  
**SEÇÃO CIRCULANTE**

UNIDADE	BC
Nº CHAMADA	Tilvicony An89d
V	EX
TOMBO BC/	64482
PROC.	16-P-00086-05
C	<input type="checkbox"/>
D	<input checked="" type="checkbox"/>
PREÇO	11,00
DATA	22/06/05
Nº CPD	

bid: 354273

FICHA CATALOGRÁFICA ELABORADA PELA  
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

An89d Antunes, Arnaldo José Bannwart  
Desenvolvimento e implementação de um sistema de controle *fuzzy* e sua aplicação a um reator de polimerização em batelada de MMA / Arnaldo José Bannwart Antunes.--Campinas, SP: [s.n.], 2005.

Orientadores: Ana Maria Frattini Fileti e João Alexandre Ferreira da Rocha Pereira.

Tese (Doutorado) - Universidade Estadual de Campinas, Faculdade de Engenharia Química.

1. Sistemas difusos. 2. Metil metacrilato. 3. Polímeros. 4. Controle automático. 5. Reatores químicos. I. Fileti, Ana Maria Frattini. II. Pereira, João Alexandre Ferreira da Rocha. III. Universidade Estadual de Campinas. Faculdade de Engenharia Química. IV. Título.

Título em Inglês: Development implementation of a fuzzy system to control a MMA polymerization batch reactor.

Palavras-chave em Inglês: Fuzzy systems, Methyl methacrylate, Automatic control, Polymers, Chemical reactors

Área de concentração: Sistemas de Processos Químicos e Informática

Titulação: Doutorado

Banca examinadora: Vivaldo Silveira Jr., Song Won Park, Flávio Vasconcelos da Silva e José Vicente Hallak d'Ángelo

Data da defesa: 22/02/2005

Este exemplar corresponde à versão final de Tese de Doutorado em Engenharia Química.



---

Profa. Dra. Ana Maria Frattini Fileti

47921500

Tese de doutorado defendida por Arnaldo José Bannwart Antunes e aprovada em 22 de fevereiro de 2005 pela banca constituída pelos doutores:



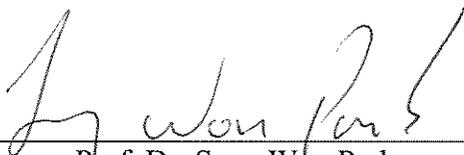
---

Profa. Dra. Ana Maria Frattini Fileti – Orientadora  
FEQ / UNICAMP



---

Prof. Dr. Vivaldo Silveira Jr.  
FEA / UNICAMP



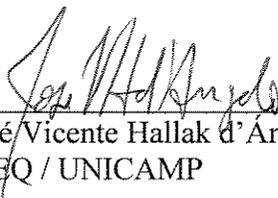
---

Prof. Dr. Song Won Park  
Escola Politécnica / USP



---

Prof. Dr. Flávio Vasconcelos da Silva  
FEQ / UNICAMP



---

Prof. Dr. José Vicente Hallak d' Angelo  
FEQ / UNICAMP

# Abstract

The present work is concerned with the design and experimental testing of control system with a fuzzy algorithm for temperature control of a methyl methacrylate (MMA) batch polymerization. Ethyl acetate is used as solvent and benzoyl peroxide is the reaction initiator. According to the literature, this reaction is complex and highly nonlinear, therefore there is some difficulty choosing the most appropriate controller. To verify the system complexity and the influence of the fuzzy control algorithm it is compared to a conventional PID (velocity form) controller. The controller tuning parameters were obtained via open-loop experiments. For the conventional PID, the Cohen-Coon method was applied followed by stability analysis. For the fuzzy controller, the open loop results helped to shape the knowledge base, with the determination of the membership functions. This work shows the several alternatives for tuning a fuzzy controller. It can be achieved by modifying the knowledge base or scaling factors. Besides, a digital temperature filter was successfully used in order to prevent the oscillatory behaviour of the reaction temperature. According to experimental results, the PID-fuzzy was considered more suitable and reliable for this polymerization process control since it outperformed PID velocity form algorithm.

KEY-WORDS: fuzzy control, methyl methacrylate, batch process.

2.3.1) Raciocínio <i>fuzzy</i> .....	18
Conjunto de regras.....	18
2.3.2) Estrutura do controle <i>fuzzy</i> .....	20
Fuzificação .....	21
Processo de inferência.....	21
Defuzificação.....	23
2.3.3) Estrutura do controlador <i>fuzzy</i> de Li .....	24
Vantagens e desvantagens .....	25
2.3.4) Revisão bibliográfica sobre controle <i>fuzzy</i> .....	26
2.4) Controle convencional por retroalimentação .....	30
2.5) Comunicação de dados.....	32
2.5.1) Protocolo aberto: Fieldbus Foundation .....	34
2.5.2) Blocos Funcionais do <i>Fieldbus</i> .....	36
<i>Resource block</i> .....	36
<i>Transducer Block</i> .....	37
Blocos funcionais .....	37
2.5.3) Controle distribuído .....	38
2.5.4) <i>Fieldbus</i> : vantagens e desvantagens .....	39
2.5.5) Conclusões .....	41
Capítulo 3 - Estudo experimental da polimerização do MMA .....	43
Introdução.....	44
3.1) Aparato experimental.....	44
3.2) Procedimento experimental.....	46
3.3) Configuração do sistema digital.....	47
3.4) Implementação do controle convencional: PID .....	50

3.4.1) Sintonia PID: parâmetros fixos .....	50
3.4.2) Sintonia PID: controle adaptativo .....	55
3.4.3) Análise de Estabilidade em Malha Fechada .....	56
Análise de estabilidade para parâmetros fixos .....	59
Análise de estabilidade para controle adaptativo .....	62
3.5) Controle <i>fuzzy</i> .....	66
3.5.1) Sintonia <i>fuzzy</i> .....	67
Parâmetros de sintonia.....	70
3.5.2) Implementação <i>fuzzy</i> .....	72
3.5.3) Resultados preliminares .....	74
Capítulo 4 - Resultados .....	77
Introdução.....	78
4.1) Controlador PID.....	78
4.2) Controlador PID adaptativo .....	81
4.3) Controlador <i>fuzzy</i> .....	84
4.4) Comparações.....	95
4.4.2) Qualidade do polímero produzido.....	96
4.4.2) Conclusões .....	98
Capítulo 5 - Conclusões e sugestões para trabalhos futuros .....	99
Referências .....	103
Apêndice A - Materiais usados .....	107
Produtos químicos:.....	107
Reator: .....	107
Protocolo de comunicação Fieldbus Foundation: .....	108
Apêndice B - Código fonte.....	109

Apêndice C - Teste de estabilidade (MatLab) e Calibração da válvula .....	139
Teste de estabilidade .....	139
Calibração da válvula .....	140

## Lista de figuras

Figura 2.1: Moléculas de monômero e do polímero. ....	4
Figura 2.2: Função de pertinência para altura. ....	11
Figura 2.3: Número zero, num contexto <i>fuzzy</i> .....	12
Figura 2.4: Temperatura, com variável lingüística.....	13
Figura 2.5: Conceito de número <i>fuzzy</i> . ....	14
Figura 2.10: Estrutura do controlador <i>fuzzy</i> . ....	20
Figura 2.11: Processo de inferência. ....	22
Figura 2.12: Critério de máximo: problema de estabilidade. ....	23
Figura 2.13: Estrutura do controlador PID- <i>fuzzy</i> híbrido. ....	24
Figura 2.14: Simplificação do conjunto de regras.....	25
Figura 2.15: Identificação do processo.....	26
Figura 2.16: Identificação do operador. ....	27
Figura 2.17: Diagrama de blocos da malha controle PID. ....	31
Figura 2.18: Esquema de barramento de campo (NUNES, 2003). ....	33
Figura 2.19: Comparação da estrutura entre DCS e FCS.....	39
Figura 3.2: Esquema do reator. ....	45
Figura 3.3: Fluxograma da reação. ....	47
Figura 3.4: Configuração do sistema <i>fieldbus</i> . ....	48
Figura 3.5: Interface do <i>software</i> desenvolvido (cliente OPC).....	49
Figura 3.6: Curva de reação do processo, após perturbação de vazão. ....	51
Figura 3.7: Perturbações na vazão.....	51
Figura 3.8: Curva de Reação do Processo após perturbação de 10,82 para 6,10 mL/s na vazão .....	52

Figura 3.9: Curva de Reação do Processo após perturbação de 10,82 para 23,74 mL/s na vazão.....	53
Figura 3.10: Perturbação no início da reação em malha aberta.....	55
Figura 3.11: Perturbação para o final da reação.....	56
Figura 3.14: Gráfico <i>Root Locus</i> para o fechamento da válvula no período inicial da reação (Tabela 3.4). ....	63
Figura 3.15: Gráfico <i>Root Locus</i> para a abertura da válvula.....	64
no período inicial da reação (Tabela 3.4). ....	64
Figura 3.16: Gráfico <i>Root Locus</i> para o fechamento da válvula no período final da reação (Tabela 3.5). ....	65
Figura 3.17: Gráfico <i>Root Locus</i> para a abertura da válvula no.....	66
período final da reação (Tabela 3.15).....	66
Figura 3.18: Ensaio em malha aberta. ....	67
Figura 3.19: Ensaio em malha aberta. ....	68
Figura 3.26: Teste de sintonia ( $K_1 = -1,0$ ).....	71
Figura 3.27: Teste de sintonia ( $K_1 = -0,5$ ).....	71
Figura 3.28: Teste de sintonia ( $K_1 = -0,2$ ).....	72
Figura 3.29: Teste de sintonia ( $K_1 = -0,02$ ).....	72
Figura 4.1: Reação com controle PID mal sintonizado.....	79
Figura 4.2: Acionamento do aquecedor. ....	79
Figura 4.3: Reação com controle PID. ....	80
Figura 4.4: Acionamento do aquecedor. ....	80
Figura 4.5: Reação com controle PID com filtro de temperatura. ....	81
Figura 4.6: Acionamento do aquecedor. ....	81
Figura 4.7: Reação com controle PID adaptativo com filtro de temperatura. ....	82
Figura 4.8: Acionamento do aquecedor. ....	82

Figura 4.9: Reação com controle PID adaptativo com filtro de temperatura. ....	83
Figura 4.10: Acionamento do aquecedor. ....	83
Figura 4.11: Reação com controle <i>fuzzy</i> com funções de pertinência igualmente espaçadas (n=7) .....	84
Figura 4.12: Acionamento do aquecedor. ....	85
Figura 4.13: Reação com controle <i>fuzzy</i> ( $K_1 = -0,02$ e $n=3$ ). ....	86
Figura 4.14: Acionamento do aquecedor. ....	86
Figura 4.15: Reação com controle <i>fuzzy</i> ( $K_1 = -0,01$ e $n=3$ ). ....	87
Figura 4.16: Acionamento do aquecedor. ....	87
Figura 4.17: Reação com controle <i>fuzzy</i> ( $K_1 = -0,03$ e $n=3$ ). ....	88
Figura 4.18: Acionamento do aquecedor. ....	88
Figura 4.19: Reação com controle <i>fuzzy</i> (com filtro). ....	89
Figura 4.20: Acionamento do aquecedor. ....	89
Figura 4.21: Reação com controle <i>fuzzy</i> (com + 2 mL/s). ....	90
Figura 4.22: Acionamento do aquecedor. ....	90
Figura 4.23: Reação com controle <i>fuzzy</i> (n=5). ....	91
Figura 4.24: Acionamento do aquecedor. ....	91
Figura 4.25: Reação com controle <i>fuzzy</i> (n=7). ....	92
Figura 4.26: Acionamento do aquecedor. ....	92
Figura 4.27: Massa específica. ....	93
Figura 4.28: Temperaturas de entrada e saída. ....	93
Figura 4.29: Reação com controle servo (n=7 e filtro) .....	94
Figura 4.30: Acionamento do aquecedor. ....	94
Figura 4.31: Polímero com aspecto normal (ensaio da Figura 4.15). ....	96
Figura 4.32: Polímero com aspecto fora dos padrões (ensaio da Figura 4.19). ....	97

Figura 4.33: Polímero com aspecto fora dos padrões (ensaio da Figura 4.21). .....97

## Lista de tabelas

Tabela 2.1: Conjunto de regras em tabela. ....	19
Tabela 3.1: Parâmetros de processo obtidos a partir da curva de reação do processo após perturbação de 10,82 para 6,10 mL/s na vazão do fluido de resfriamento. ....	53
Tabela 3.2: Parâmetros de processo obtidos a partir da curva de reação do processo após perturbação de 10,82 para 6,10 mL/s na vazão do fluido de resfriamento. ....	53
Tabela 3.3: Parâmetros de processo obtidos a partir da curva de reação do processo após perturbação de 10,82 para 6,10 mL/s na vazão. ....	54
Tabela 3.4: Comportamento no início da reação.....	55
Tabela 3.5: Comportamento no final da reação .....	56
Tabela 3.6: <i>Overshoot</i> para Diferentes Valores de $K_c$ .....	62
Tabela 3.7: <i>Overshoot</i> para Diferentes Valores de $K_c$ (início da reação).....	64
Tabela 3.8: <i>Overshoot</i> para Diferentes Valores de $K_c$ (final da reação).....	66
Tabela 3.9: Conjunto de regras para <i>fuzzy</i> -PI (card. 3) .....	73
Tabela 3.10: Conjunto de regras para <i>fuzzy</i> -PD (card. 3).....	73
Tabela 3.11: Conjunto de regras para <i>fuzzy</i> -PI (card. 5). ....	73
Tabela 3.12: Conjunto de regras para <i>fuzzy</i> -PD (card. 5).....	73
Tabela 3.13: Conjunto de regras para <i>fuzzy</i> -PI (card. 7). ....	73
Tabela 3.14: Conjunto de regras para <i>fuzzy</i> -PD (card. 7).....	74
Tabela 4.1: Comparação de desempenho (Controle PID).....	95
Tabela 4.2: Comparação de desempenho (Controle <i>fuzzy</i> ). ....	95
Tabela 4.3: Resultados das análises de GPC.....	98

## Nomenclatura

- **e**: erro de temperatura ( $^{\circ}\text{C}$ )
- **$\Delta e$** : variação do erro de temperatura ( $^{\circ}\text{C}$ )
- **$G_C$** : função de transferência do controlador
- **$G_P G_H$** : função de transferência de pulso
- **$K_{1-4}$** : fatores de escala da estrutura *fuzzy* de LI (1996)
- **$K_c$** : ganho do controlador ( $\text{mL}/^{\circ}\text{C}\cdot\text{s}$ )
- **$K_p$** : ganho do processo ( $^{\circ}\text{C}\cdot\text{s}/\text{mL}$ )
- **n**: número de cardinalidade
- **N**: valor lingüístico “negativo”
- **NM**: valor lingüístico “negativo médio”
- **NS**: valor lingüístico “negativo pequeno”
- **P**: valor lingüístico “positivo”
- **PM**: valor lingüístico “positivo médio”
- **PS**: valor lingüístico “positivo pequeno”
- **$t_d$** : atraso (s)
- **$\Delta t$** : intervalo de amostragem (s)
- **$T_f$** : temperatura filtrada ( $^{\circ}\text{C}$ )
- **$T(k)$** : temperatura do reator no instante k ( $^{\circ}\text{C}$ )
- **Z**: valor lingüístico “zero”
- **$\mu$** : grau de pertinência
- **$\tau_D$** : constante de tempo derivativo (s)
- **$\tau_I$** : constante de tempo integral (s)
- **$\tau_P$** : constante de tempo do processo (s)

# Capítulo 1

## Introdução

A função desempenhada pelo sistema de controle é de importância fundamental na indústria de processos, principalmente em certos casos onde existem dificuldades inerentes ao processo e penalidades adicionais numa eventual falha do sistema de controle. Isto é especialmente verdadeiro em processos em batelada envolvendo produtos de valor agregado mais elevado.

Para a polimerização, a qualidade do material é bastante sensível às condições operacionais, aumentando o custo no caso de um controle ineficiente. O controle de temperatura é um tema importante na polimerização por causa da forte dependência de fatores como massa molecular (MM) e distribuição da massa molecular (DMM), que possuem forte dependência com a temperatura e são reflexo da qualidade geral do polímero. Além da dificuldade típica na operação de reatores em batelada, ou seja, a alteração do meio reacional com o avanço da reação, a polimerização do metil metacrilato conta com alta não-linearidade. Como propriedades mais importantes do polímero, citam-se: semelhança ao vidro, resistência a agentes atmosféricos, resistência aos impactos e aos riscos. As aplicações mais destacadas são: vidraças, painéis, letreiros, decoração, partes de móveis e próteses dentárias.

Nesses casos, em que o processo apresenta severa não-linearidade o modo tradicional de controle PID não é recomendado, havendo necessidade do uso de técnicas mais avançadas como as baseadas na lógica *fuzzy*. Apesar da vantagem de lidar melhor com processos mais complexos, o controle *fuzzy* é de implementação difícil porque não há um consenso ou uma regra prática para as muitas decisões que devem ser tomadas durante o projeto do controlador.

No capítulo 2 são apresentados conceitos teóricos e revisão bibliográfica dos assuntos mais relevantes do trabalho, tais como o processo de polimerização, lógica *fuzzy*, controle *fuzzy*, controle convencional PID e comunicação de dados. No capítulo 3 são discutidos o aparato e o procedimento experimental, métodos de sintonia, análise de

estabilidade do controlador PID e implementação do controlador *fuzzy*. No capítulo 4 são mostrados os resultados das reações com os controladores desenvolvidos, além considerações sobre a qualidade do polímero formado, tanto no aspecto visual como em análises de Cromatografia por Permeação em Gel (GPC). As conclusões do trabalho estão apresentadas no capítulo 5.

No apêndice A são detalhados os materiais usados. No apêndice B é exibido o código fonte do programa de controle desenvolvido. Finalmente no apêndice C, são mostradas a listagem do *script* em Matlab para teste de estabilidade Root-Locus, assim como a calibração da válvula.

## **Capítulo 2**

### **Conceitos teóricos e revisão bibliográfica**

# Introdução

No capítulo presente serão apresentados alguns conceitos teóricos fundamentais ao entendimento do sistema desenvolvido, além da revisão bibliográfica do tema em estudo.

## 2.1) O processo de polimerização

### 2.1.1) Caracterização dos materiais

O monômero utilizado é o metacrilato de metila (MMA), tem ponto de ebulição a 100 °C e é descrito na Figura 2.1, juntamente com o polímero formado.

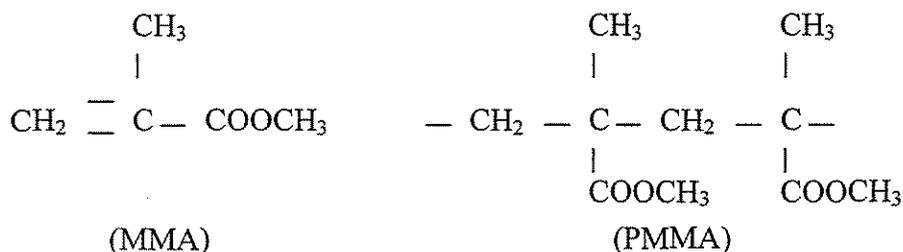


Figura 2.1: Moléculas de monômero e do polímero.

Para a polimerização, normalmente é feita uma adição em solução com temperatura em torno de 60 °C na presença de percompostos ou azonitrila.

O polímero PMMA tem cristalinidade predominantemente amorfa, com ponto de fusão de 160 °C e ponto de transição vítrea de 105 °C. Como propriedades mais importantes, citam-se: semelhança ao vidro, resistência a agentes atmosféricos, resistência a impactos e a riscos. As aplicações mais destacadas são: vidraças, painéis, letreiros, decoração, partes de móveis e próteses dentárias.

### 2.1.2) Polimerização

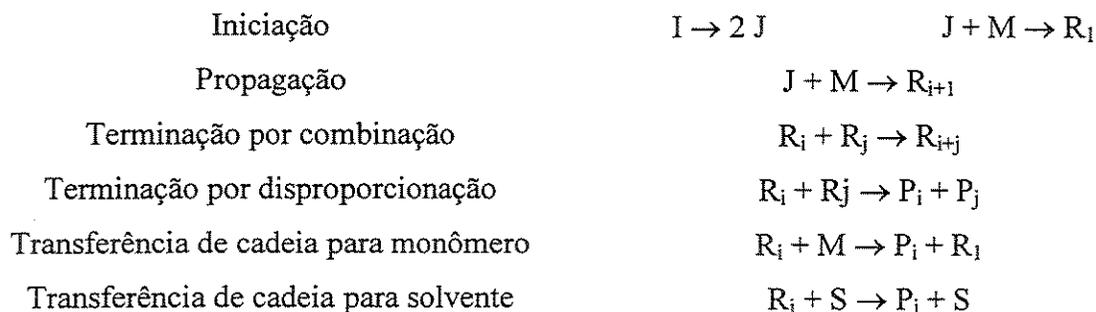
As reações de polimerização podem se dar por dois mecanismos básicos e distintos: adição e condensação. A poliadição envolve moléculas com funcionalidade maior ou igual a dois, aproveitando-se de pontos de insaturação para agregação de novas moléculas, e não havendo a formação de subprodutos. Esses centros ativos, que também podem ser íons além dos radicais, conduzem a um rápido crescimento das cadeias poliméricas. Mesmo para baixas conversões, surgem longas cadeias poliméricas juntamente com material não reagido. A iniciação pode resultar tanto em radicais livres ou em íons,

dependendo da cisão homolítica ou heterolítica, respectivamente. Os produtos comumente empregados na iniciação homolítica são: peróxidos, hidroperóxidos orgânicos, azocompostos e persais (MANO, 1985a).

A fase de crescimento molecular se dá na propagação, onde uma molécula de monômero combina-se com um centro ativo, aumentando o tamanho da cadeia. Por fim, nas reações de terminação, os centros ativos da macromolécula são neutralizados através da ligação com outra cadeia em crescimento resultando em uma única cadeia (combinação) ou em duas cadeias inativas (desproporcionamento). Pode haver ainda transferência de cadeia para uma molécula de solvente ou um monômero, iniciando uma nova cadeia.

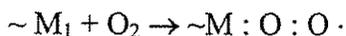
Por outro lado, a policondensação não é classificada em etapas. O incremento da cadeia polimérica é probabilístico, não havendo, como no caso de adição, uma configuração para o polímero ativo - passível de sofrer adição - e outro para o polímero morto - produzido na fase de terminação. Por isso, a alta massa molecular das cadeias só é conseguido em altas conversões. Outra diferença entre os dois mecanismos de polimerização é que a condensação sempre produz uma molécula pequena (como água ou ácido clorídrico) a cada monômero agregado à cadeia.

A polimerização por radicais livres, como no caso do MMA, é um processo muito complexo. Uma fonte importante de não-linearidade é devida à natureza autocatalítica da reação de polimerização, conhecido como efeito gel. Isso pode ocasionar reações incontroláveis, resultando em aumento de temperatura e danos ao equipamento (SHROUSH e KRAVARIS, 1993). Por essa razão, o calor gerado pode apresentar um pico durante a reação, ao contrário de reações normais que produzem o calor máximo no início da batelada, pela máxima concentração de reagentes. A densidade do polímero é maior que a do monômero (ELLIS *et al.*, 1994). Por isso, um densímetro *on-line* pode ser usado para estimar a conversão. A cinética para uma polimerização por radicais livres é descrita por

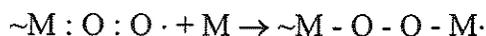


KIPARISSIDES (1996) descreve o aumento da viscosidade - reação controlada por difusão - em cada uma das reações elementares da polimerização por radicais livres (iniciação, propagação e terminação) levando a, respectivamente, efeito gel, efeito *glass* e efeito *cake*. O efeito gel dificulta a mobilidade no meio reacional, comprometendo as propriedades do material produzido. Também pode levar a um aumento descontrolado na temperatura, com reações em cadeia. O efeito *glass* manifesta-se na diminuição na taxa de propagação, ocorrendo a temperaturas menores que a temperatura de transição vítrea do polímero. A restrição na propagação “congela” a polimerização em conversões abaixo da máxima. Por fim, o efeito *cake* atua na conversão do monômero. A proximidade dos radicais formados facilita sua recombinação se a difusão até o monômero é dificultada pela viscosidade.

Para que o produto seja obtido conforme as especificações desejadas, é necessário que a reação de polimerização seja feita em condições controladas. Por isso, durante o armazenamento, estabilizadores devem ser adicionados ao monômero para evitar sua polimerização. Obviamente, esses agentes inibidores são retirados antes da reação, o que normalmente é feito com lavagem ou destilação. Geralmente os inibidores atuam combinando-se com centros ativos ou eliminando iniciadores. Os principais inibidores são: hidroquinona, p-t-butil-catecol, enxofre, cobre, oxigênio, etc.... O oxigênio forma um radical peróxido de baixa reatividade:



mas também pode prosseguir a reação, adicionando monômero:



Certos materiais são adicionados à mistura reacional ao fim da polimerização, para interrompê-la; são chamados terminadores (*shortstops*), e os mais importantes são a hidroquinona e os ditiocarbamatos alcalinos (MANO, 1985b).

### 2.1.3) Aspectos de controle

O controle de temperatura é um tema importante na polimerização por causa da forte dependência de fatores como massa molecular (MM) e distribuição da massa molecular (DMM), que possuem forte dependência com a temperatura e são reflexo da qualidade geral do polímero.

O período inicial de polimerização pode ser crítico. Sob um ponto de vista, quanto antes a temperatura chegar ao *set-point*, melhor. Em termos práticos, uma elevação rápida de temperatura reduz o tempo de batelada, o que é obviamente uma vantagem. Por outro lado, o sistema de controle deve prevenir uma sobrelevação de temperatura que poderia prejudicar a qualidade do polímero e, em casos extremos, conduzir a uma reação em cadeia deixando a temperatura sem controle.

Além do aspecto de *scaleup*, isto é, da redução da razão entre a área de troca térmica e o volume do reator, à medida que o volume do reator aumenta, existem outros aspectos que dificultam o controle de temperatura. De acordo com CROWLEY e CHOI (1996), o efeito gel causa um rápido aumento na taxa de reação, especialmente quando a viscosidade do meio aumenta com o avanço da reação. Este aumento de viscosidade também afeta negativamente a remoção de calor porque há diminuição no coeficiente efetivo de calor entre a parede do reator e a água na jaqueta. Incrustação de polímero pode prejudicar ainda mais a troca térmica.

## **2.1.4) Controle de reação de polimerização: Revisão**

### **bibliográfica**

Uma aplicação de controle *fuzzy* é relatada com resultados positivos em ROFFEL e CHIN (1991). O controle *fuzzy* é especialmente adequado à reatores de polimerização. Muitos podem ser modelados com precisão, mas em outros há problemas mais sérios. Mesmo sendo modelados corretamente, há fatores que precisam ser simplificados no controle *on-line*. Em alguns casos, mesmo quantidades indetectáveis de impurezas podem afetar o mecanismo de reação, resultando em comportamento transiente e não-linear.

Controle ótimo foi aplicado por vários autores para minimizar o tempo de batelada de polimerização e produzir um polímero com massa molecular (MM) e distribuição da massa molecular (DMM) desejadas (CHANG e LAI, 1992; THOMAS e KIPARISSIDES, 1984; O'DRISCOLL e PUNNUSWAMY, 1990). Uma grande quantidade de cálculo iterativo é necessário e a maioria dos pesquisadores basearam suas previsões em modelos simplificados. Suposições como densidade constante do meio reacional, ausência de transferência de cadeia para o monômero, estado pseudo-estacionário para polímero vivo e

uso de correlações simples para o efeito gel também são relatados. De acordo com AHN, CHANG e RHEE (1998), o cálculo de uma trajetória ótima a partir de tantas simplificações resultou praticamente numa isoterma, onde o efeito gel não foi apropriadamente considerado. Além disso, não houve relato da implementação prática.

Na literatura recente, não há muitos artigos relatando testes experimentais em reações de polimerização de PMMA. A determinação de trajetória ótima de concentração de iniciador no modo batelada da polimerização do MMA é feita por JEONG e RHEE (2000), visando níveis especificados de conversão e massa molecular. No controle de temperatura (cascata PID), uma micro bomba manipula a vazão de iniciador. Apesar das características não-lineares do processo, os resultados foram satisfatórios. CHANG e LIAO (1999) aplicaram um procedimento de estimativa de parâmetros para ajustar um modelo cinético com dados experimentais da polimerização do MMA. O modelo foi usado no cálculo da trajetória ótima de temperatura para controle de massa molecular, através de um controlador PI. Os parâmetros de sintonia são produzidos por tentativa e erro com base no simulador ou em testes com água. Duas variáveis manipuladas, vazão de água de resfriamento e de água de aquecimento são colocadas em função de uma única variável para simplificação. Os resultados são satisfatórios. PAN e LEE (2003) superaram a falta de um medidor *on-line* para predizer a qualidade do polímero, que pode ser manipulada ajustando-se o tempo e as condições de batelada. Para o MMA, as variáveis preditas são massa molecular médio e a polidispersão. Além da vantagem do modelo requerer apenas dados históricos, há também baixa demanda computacional. Houve sucesso na redução do efeito da variação na alimentação sobre a qualidade do produto.

## 2.2) Lógica *fuzzy*

No cotidiano, sempre se tenta buscar um grau máximo de clareza para se tomar uma decisão segura sobre uma situação. Assim, são evitados “tons cinzentos” que exigem a análise de várias nuances. Apesar de ter a capacidade de lidar com essas situações, caracterizadas como indefinidas ou contendo meias-verdades, elas são vistas com desconfiança e ceticismo.

A incerteza sempre foi tida como um elemento indesejado e prejudicial a uma análise rigorosa de um sistema. Assim, ela deveria ser reduzida através de um estudo mais detalhado do sistema investigado, resultando num modelo mais complexo e sofisticado.

Havendo ainda incerteza em algum componente do sistema, a análise poderia ser novamente aprofundada. Assim, a complexidade do modelo poderia ser incrementada indefinidamente, sob pena de impossibilitar o estudo do sistema. Por isso, a única possibilidade razoável parece ser, de alguma forma, *incluir* a incerteza na análise. Esse é o papel da lógica *fuzzy*: viabilizar uma análise rigorosa, levando-se em conta aspectos de incerteza e imprecisão.

O próprio uso das palavras incerteza e imprecisão pode levar o leitor a entender erroneamente aquilo que se pretende expressar realmente. A princípio, o sentido poderia significar a presença de um componente probabilístico, dependente de uma variável aleatória. Contudo, no contexto da lógica *fuzzy*, este não é o caso. O que se pretende exprimir é uma dificuldade em avaliar uma variável em termos pontuais, com a simples atribuição de um valor numérico. A intensidade da variável medida deve ser vista dentro de um contexto que têm fronteiras graduais.

### 2.2.1) Conjuntos escalares e *fuzzy*

A lógica *fuzzy* retrata a expansão da noção clássica de conjuntos, onde, para um determinado elemento, havia somente duas possibilidades: pertencer ou não pertencer a um conjunto. Assim, surge a flexibilidade dessa condição. Neste trabalho, esses conjuntos são chamados de escalares. Há várias referências sobre os conceitos básicos de lógica e controle *fuzzy*. Entre elas, destacam-se SHAW (1998), DRIANKOV (1996), PEDRYCZ (1989) e, principalmente, SHAW e SIMÕES (1999).

Dentro da lógica *fuzzy*, a pertinência de um elemento a um conjunto é uma questão de grau. Não é necessário incluir ou excluir integralmente o elemento, basta que ele tenha alguma semelhança com a idéia representada pelo conjunto para ser aceito com algum grau.

A teoria dos conjuntos *fuzzy* é melhor entendida quando comparada com a teoria clássica dos conjuntos, uma vez que aquela é uma generalização desta. Primeiramente deve ser assinalada a noção de pertinência de um elemento em relação a um conjunto. Nos conjuntos escalares, havia duas possibilidades: um elemento é ou não é parte do conjunto.

Essa relação entre elemento e conjunto pode ser expressa matematicamente através de uma função de pertinência, escrita na forma  $\mu_A(u)$ , que significa o valor da função para o

elemento  $u$  em relação ao conjunto  $A$ . Evidentemente, no caso de um conjunto escalar, a função de pertinência é binária, ou seja

$$\mu_A(u) = \begin{cases} 1 & \text{se } u \in A \\ 0 & \text{se } u \notin A \end{cases}$$

Para introduzir a idéia de pertinência em relação a um conjunto *fuzzy*, considere o exemplo de um conjunto de homens altos. Em certas situações é necessária uma divisão rígida entre as idéias de alto e não-alto. Na seleção para serviço militar, por exemplo, os uniformes são feitos de modo padronizado. Assim, são excluídos os candidatos que são excessivamente altos. Por simplicidade, a abordagem nos conjuntos escalares é tomada, com o estabelecimento de um limite rígido entre rapazes altos e não-altos. Para um candidato  $u$ , a função de pertinência poderia ser

$$\mu_A(u) = \begin{cases} 1 & \text{se altura} \geq 2,00 \text{ m} \\ 0 & \text{se altura} < 2,00 \text{ m} \end{cases}$$

Contudo, o entendimento que as pessoas têm sobre a realidade não é tão rígido assim. A diferença de altura entre uma pessoa de 1,97 m e outra de 2,02 m não é tão grande assim a ponto de justificar uma abordagem radical em que um deles é considerado alto e o outro não. Normalmente, as pessoas dirão que ambos são altos, mas que um deles é “mais alto que o outro”. A expressão “mais alto” é que marca a diferença entre o escalar e o *fuzzy*.

De acordo com a teoria de conjuntos *fuzzy*, a pertinência é uma questão de grau, não havendo uma separação rígida entre pertencer ou não pertencer. Evidentemente, a ausência de uma separação rígida é refletida na função de pertinência. Agora, esta função assume valores dentro de um intervalo, cada valor corresponde a um grau de pertinência. Por exemplo, para o conjunto dos homens altos, um elemento com 2,02 m tem o valor da função de pertinência maior ou igual do que um sujeito com uma altura de 1,97 m.

A forma usada para decorar a função de pertinência é de livre escolha, ainda no caso do conjunto de homens altos, a função poderia ser descrita pela expressão seguinte, graficada na Figura 2.2.

$$\mu_A(u) = \begin{cases} 0, & \text{se } u \leq 1,80 \text{ m} \\ 5(u - 1,80), & \text{se } 1,80 < u < 2,00 \text{ m} \\ 1, & \text{se } u \geq 2,00 \text{ m} \end{cases}$$

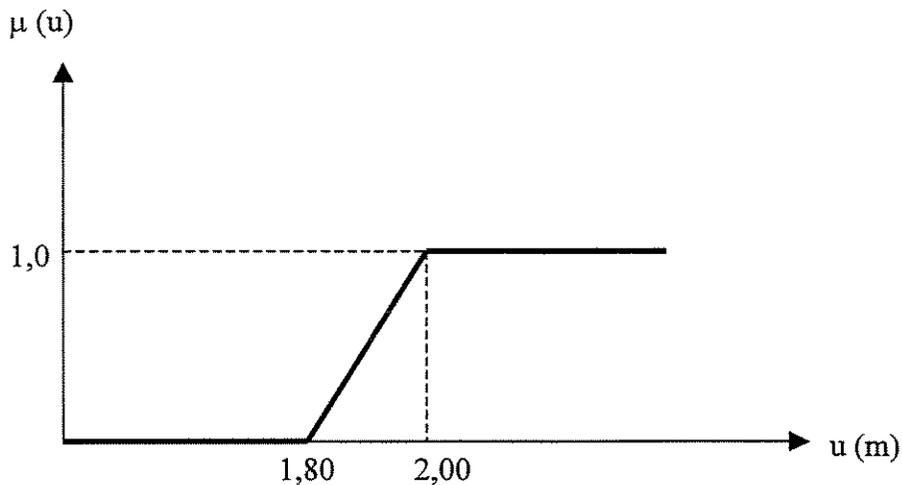


Figura 2.2: Função de pertinência para altura.

Com a idéia de conjuntos *fuzzy* também podemos definir, de maneira formal, proximidade de um variável em relação a um número determinado. Ou seja, no âmbito da lógica *fuzzy* faz sentido a expressão “próximo de zero”. A função de pertinência teria seu valor máximo, unitário, centrado em zero e decrescente para elemento mais afastados de zero, como na Figura 2.3.

Há uma forma consagrada de notação para representar os pares elemento-grau de pertinência num conjunto *fuzzy* discreto. O conjunto é posto como possuindo elementos do tipo  $\mu(u)/u$ , onde o operador de divisão serve apenas para separar um elemento de seu grau de pertinência. Por exemplo, um conjunto *fuzzy* A dos elementos próximos a 10 poderia ser representado como

$$\mu_A = \{0,0 / 7; 0,3 / 8; 0,6 / 9; 1,0 / 10; 0,6 / 11; 0,3 / 12; 0,0 / 13\}$$

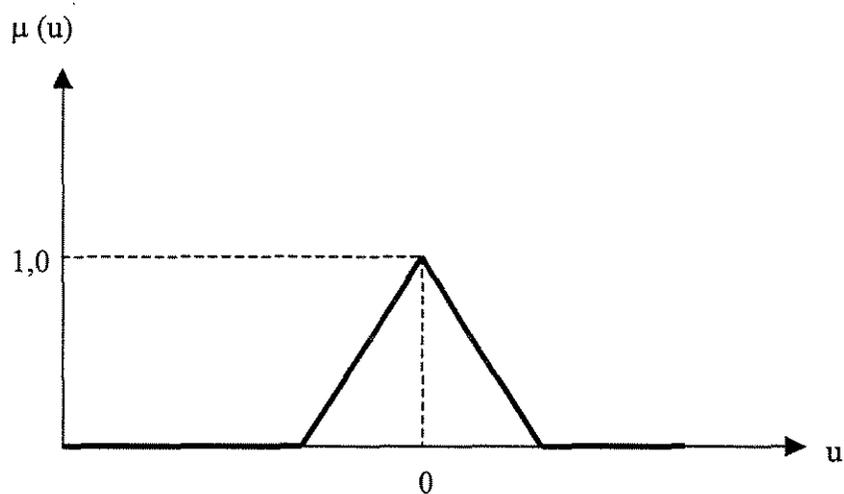


Figura 2.3: Número zero, num contexto *fuzzy*.

### 2.2.2) Caracterização de um conjunto *fuzzy*

Há uma série de definições e nomenclaturas comumente usadas na discussão de conjuntos *fuzzy*. Serão apresentadas a seguir algumas dessas definições mais importantes.

**Definição:** um conjunto *fuzzy*  $A$  é chamado de normal se existe um elemento  $u$  em que  $\mu_A(u) = 1$ . Caso contrário, ele é chamado de *subnormal*. Em qualquer caso, o valor máximo do grau de pertinência obtido por um elemento do conjunto é chamado de *altura*.

**Definição:** o suporte do conjunto *fuzzy*  $A$  é definido como o subconjunto de  $A$  em que os elementos têm um grau de pertinência não-nulo. Ou seja,  $\text{suporte}(A) = \{u \in A \mid \mu_A(u) > 0\}$ . Nos conjuntos *fuzzy* normais, o subconjunto do suporte, em que os elementos têm grau de pertinência unitário é chamado de *cume*. Logo,  $\text{cume}(A) = \{u \in A \mid \mu_A(u) = 1\}$ .

**Definição:** a questão subconjuntos e igualdade entre conjuntos é feita com base no grau de pertinência. Tendo os conjuntos  $A$  e  $B$ ,  $A$  seria subconjunto de  $B$  se  $\mu_A(u) \leq \mu_B(u)$ . Na relação de igualdade,  $A$  seria igual a  $B$  se  $\mu_A(u) = \mu_B(u)$  para todo  $u$ .

### 2.2.3) Variáveis linguísticas e cardinalidade

O aspecto de graduação contínua, de variação suave, representada por um conjunto *fuzzy* pode ser usada para associar a intensidade de uma variável com um atributo que

também não pode ser definido de modo abrupto, exibindo uma graduação tolerada. Num certo contexto, um conjunto de valores da variável velocidade não pode ser classificada de forma natural como membros de um conjunto escalar “velocidade alta”. Há um aumento gradual de similaridade entre este conjunto e valores considerados altos.

Assim, variáveis linguísticas são definidas em seus universos de variação em termos de adjetivos associados ao seu valor. Na verdade, esses adjetivos são conjuntos *fuzzy* que precisam ter uma função de pertinência definida no universo de variação.

Por exemplo, tomando a temperatura como variável linguística, os conjuntos *fuzzy* poderiam ser: frio, morno e quente. Num universo de  $0^{\circ}\text{C} \leq T \leq 80^{\circ}\text{C}$ , o perfil das funções poderia ser mostrada na Figura 2.4, para um contexto particular. O termo cardinalidade é usado para definir a quantidade de valores linguísticos usados no universo da variável linguística. Por exemplo, na Figura 2.4, a cardinalidade empregada é 3. Caso fossem adicionados os valores “muito frio” e “muito quente”, a cardinalidade seria 5.

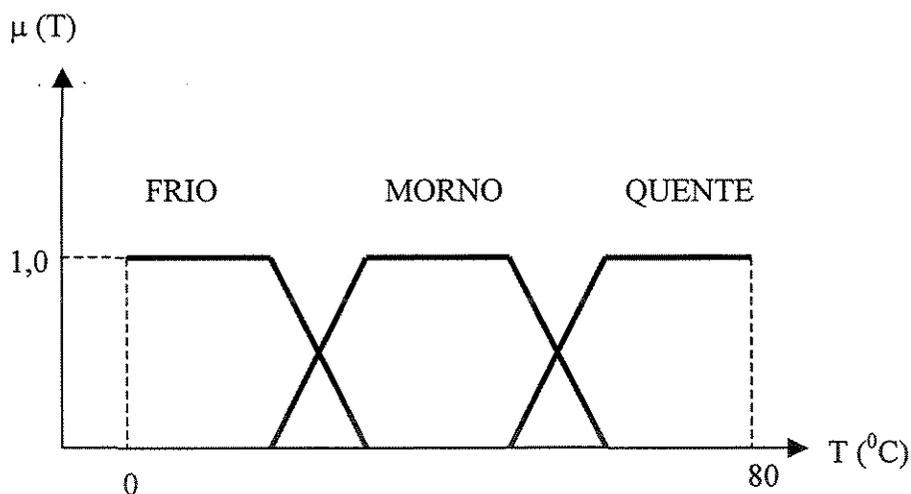


Figura 2.4: Temperatura, com variável linguística.

Como já foi mencionado, a forma gráfica de uma função de pertinência de uma variável linguística é uma escolha subjetiva, feita livremente pelo projetista. Da mesma forma, o suporte desse conjunto também é fruto de um contexto subjetivo. Apesar de haver funções de variação suave como a sigmoideal, os padrões retilíneos, como triângulos e trapézios, são normalmente usados por facilidade de cálculo.

Para uma variável linguística, o caráter *fuzzy* pode ser atenuado ou acentuado conforme a aplicação de certas palavras que funcionam como transformadores do significado de um conjunto *fuzzy*. Dado um conjunto “temperatura alta”, poderia ser criado um novo conjunto “temperatura muito alta”, com valores de função de pertinência menores para uma temperatura determinada. Com a adição da palavra “muito”, aumentou-se o rigor para a similaridade entre um elemento e o novo conjunto. Em alguns casos, pode haver mudança de difusividade do conjunto. A Figura 2.5 mostra dois conjuntos: “em torno de 10” (linha sólida) e “aproximadamente em torno de 10” (linha tracejada).

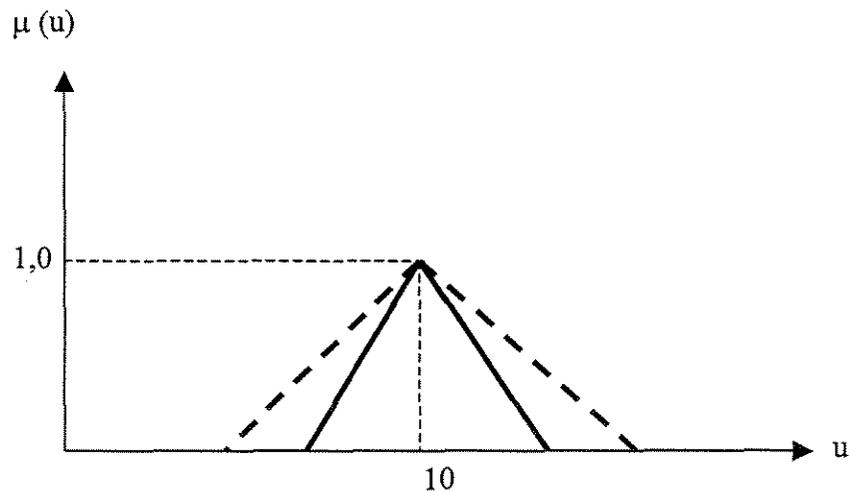


Figura 2.5: Conceito de número *fuzzy*.

#### 2.2.4) Operações com conjuntos *fuzzy*

Uma vez que as informações são colocadas em termos de conjuntos *fuzzy*, o desenvolvimento do algoritmo de controle deve contar com operações sobre esses conjuntos, descritas nesta seção. Da mesma forma que o conceito de um conjunto *fuzzy* estende o conceito de um conjunto escalar, as operações básicas num conjunto *fuzzy* também são generalizações das operações básicas dos conjuntos escalares. Ou seja, os operadores *fuzzy* poderiam também ser aplicados aos conjuntos escalares, com suas funções de pertinência binárias.

As operações de união e interseção podem ser realizadas por mais de um operador que satisfaça critérios de duas classes, chamadas de normas-s e normas-t, respectivamente.

Inicialmente, para fazer parte de um desses grupos, o operador deve ter coerência de resultados com os conjuntos escalares, além de manter características fundamentais como:

- Comutatividade:  $A \cap B = B \cap A$ ;
- Associatividade:  $(A \cap B) \cap C = A \cap (B \cap C)$ ;
- Monotonicidade:  $(A \subseteq B) \text{ e } (C \subseteq D) \rightarrow A \cap C \subseteq B \cap D$ ;
- Elemento neutro:  $A \cap X = A$ , ( $X$  é universo).

Segundo OLIVEIRA (1999), os múltiplos operadores servem para aumentar os graus de liberdade nos processos de avaliação, sendo que alguns operadores podem perder informação em alguns casos.

### União

Seja  $U$  o universo de discurso,  $x$  um elemento qualquer ( $x \in U$ ) e os subconjuntos  $A \subset U$  e  $B \subset U$ . O conjunto que representa a união  $A \cup B$  é o menor subconjunto de  $U$  que abrigue todos os elementos que pertençam ao  $A$  ou ao  $B$ . Assim, ele será sempre maior (ou igual no caso de um subconjunto conter o outro) que os subconjuntos individualmente.

Uma função de pertinência  $\mu_{A \cup B}(x)$  pode ser definida ponto a ponto por

$$\mu_{A \cup B}(x) = \mu_A s \mu_B(x) \leq \max[\mu_A(x), \mu_B(x)]$$

onde  $s$  é a co-norma triangular de união generalizada.

A norma- $s$  definida em  $[0, 1] \times [0, 1] \rightarrow [0, 1]$  é um mapeamento entre duas funções de pertinência, executado de acordo com um formato específico e deve obedecer às seguintes restrições (SHAW e SIMÕES, 1999):

- $x s w \leq y s z$ , se  $x \leq y$  e  $w \leq z$  (condição monotônica);
- $x s y = y s x$  (propriedade comutativa);
- $(x s y) s z = x s (y s z)$  (propriedade associativa);
- $x s 0 = x$  (condição de contorno);
- $x s 1 = 1$  (condição de contorno).

onde a função  $s$  é definida em  $[0, 1] \times [0, 1] \rightarrow [0, 1]$  tal que  $\forall x, y, z, w \in [0, 1]$ .

Normalmente o operador mais utilizado é o que retorna o valor máximo entre os argumentos recebidos, ou seja

$$x \text{ s } y = \max(x, y)$$

também é usual a aplicação da expressão algébrica

$$x \text{ s } y = x + y - x.y$$

### Interseção

Seja  $U$  o universo de discurso,  $x$  um elemento qualquer ( $x \in U$ ) e os subconjuntos  $A \subset U$  e  $B \subset U$ . O conjunto que representa a interseção  $A \cap B$  é o maior subconjunto de  $U$  que abrigue todos os elementos comuns à  $A$  e à  $B$ . Ele será sempre menor (ou igual no caso de um subconjunto conter o outro) que os subconjuntos individualmente.

A função de pertinência  $\mu_{A \cap B}$  é definida ponto a ponto por

$$\mu_{A \cap B}(x) = \mu_A \text{ t } \mu_B(x) \leq \min[\mu_A(x), \mu_B(x)]$$

onde  $\text{t}$  é a norma triangular, ou norma- $\text{t}$ . O mapeamento feito pela norma- $\text{t}$  deve cumprir os requisitos abaixo (SHAW e SIMÕES, 1999):

- $x \text{ t } w \leq y \text{ t } z$ , se  $x \leq y$  e  $w \leq z$  (condição monotônica);
- $x \text{ t } y = y \text{ t } x$  (propriedade comutativa);
- $(x \text{ t } y) \text{ t } z = x \text{ t } (y \text{ t } z)$  (propriedade associativa);
- $x \text{ t } 0 = 0$  (condição de contorno);
- $x \text{ t } 1 = x$  (condição de contorno).

Para isso, é usado principalmente a função de mínimo

$$x \text{ t } y = \min(x, y)$$

e o produto algébrico

$$x \text{ t } y = x.y$$

### Negação

Nos conjuntos escalares, a negação inverte a relação de pertinência entre um elemento e um conjunto. No caso *fuzzy*, é utilizado o complemento da unicidade, ou seja

$$\text{Neg}(x) = 1 - x$$

Desse modo, o complemento de um conjunto *fuzzy* é sua negação lógica. Se o operador de complemento é aplicado sobre a função de pertinência de um conjunto “rapazes altos” o resultado é a função de pertinência de um conjunto “rapazes não-altos”, quando uma função tem valor próximo da unidade, a outra é próxima de zero.

### Exemplos gráficos

Para ilustrar os conceitos anteriores, mostraremos o resultado dessas operações. Sendo os conjuntos A e B representados na Figura 2.6. Os resultados de união (max), interseção (min) e negação são mostrados nas Figuras 2.7, 2.8 e 2.9.

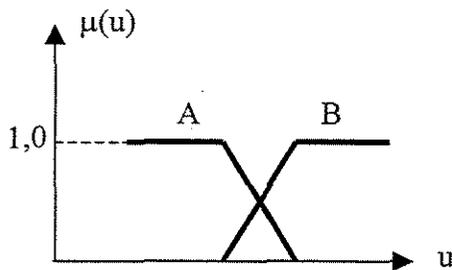


Figura 2.6: Conjuntos *fuzzy* A e B.

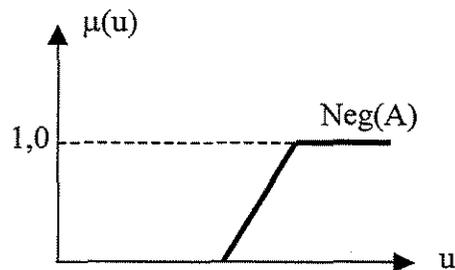


Figura 2.7: Negação do conjunto A.

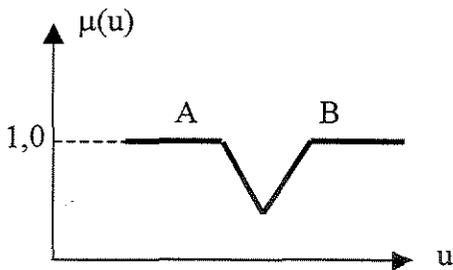


Figura 2.8: Operação de união.

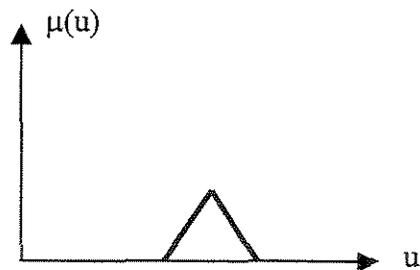


Figura 2.9: Operação de interseção.

## 2.3) Controle *fuzzy*

### 2.3.1) Raciocínio *fuzzy*

Através do raciocínio *fuzzy*, é possível chegar a uma conclusão razoável tendo-se um conjunto de regras, que determinam o mapeamento desejado entre a saída e os valores de entrada. As entradas representam as condições em relação as quais se quer tirar conclusões.

O raciocínio é chamado assim porque as regras estão escritas em termos de variáveis linguísticas, cujos valores associados são conjuntos *fuzzy*. Contudo, tanto os valores de entrada quanto os de saída necessitam ser expressos em números “palpáveis” e não em termos de variáveis linguísticas. Portanto, são necessárias duas etapas de tradução. Uma delas para converter os valores de entrada para o domínio *fuzzy*. Em seguida, os valores convertidos são processados pelo conjunto de regras. O resultado desse processamento deve ser novamente traduzido no sentido inverso, do domínio *fuzzy* para o domínio real.

#### Conjunto de regras

A resposta de um controlador *fuzzy* é dada de acordo com um conjunto de regras heurísticas, que são expressas com valores *fuzzy*. O processamento através de regras é capaz de produzir respostas não-lineares devido ao mapeamento entrada-saída ser feito ponto a ponto, utilizando uma expressão linguística ao invés de uma expressão numérica.

Assim como a lógica *fuzzy* flexibiliza a noção de pertinência de um elemento em relação ao conjunto, o processamento de informações de um controlador *fuzzy* também pode ser entendido como um modo flexibilizado de um sistema especialista. As regras de controle *fuzzy* refletem uma política de controle que se deseja adotar.

No caso de um sistema especialista, há um conjunto de regras do tipo “se <causa> então <consequência>” e, através dessas regras as respostas são obtidas. Ou seja, dada uma condição operacional, a ação apropriada de controle é avaliada através de regras que tenham como causa, a condição operacional dada. No entanto, é encontrada a mesma radicalidade vista na noção clássica de conjunto: se não houver identidade entre a condição medida e o item <causa> de uma regra, ela não é considerada.

No processamento dos controladores *fuzzy* as regras “se... então” são avaliadas desde que haja alguma similaridade, valor não nulo da função de pertinência, entre as condições obtidas e a causa das regras para que a consequência correspondente influencie na resposta do controlador. Um exemplo de regra seria

SE temperatura é alta ENTÃO reduzir vazão de vapor

A base de regras é um segmento crítico do projeto, responsável pela qualidade e adequação das respostas fornecidas pelo controlador. Um meio de se obter o conjunto de regras é fortemente relacionado com o conhecimento, em termos de causa e efeito, que se tem sobre a dinâmica do processo. Havendo, por exemplo, algum operador com grande experiência na interação com o sistema, pode-se tentar traduzir essa experiência no conjunto de regras usadas pelo controlador. Ou seja, há ocasiões em que a modelagem do processo é muito complexa e custosa, impedindo que o algoritmo de controle seja baseado em modelagem. Para resolver esse problema, faz-se presente uma mudança de foco, do processo para o operador do processo.

Um outro caminho, mais sistemático, é definir as variáveis de entrada a serem fuzificadas em termos de erro e suas variações. Num caso onde se considere o erro e a variação do erro, cada par desses valores corresponderia a uma intensidade desejada da variável manipulada. Desse modo, as regras podem ser colocadas na forma de uma tabela (Tabela 2.1).

Tabela 2.1: Conjunto de regras em tabela.

		erro		
		N	Z	P
$\Delta$ erro	N	N	N	Z
	Z	N	Z	P
	P	Z	P	P

O controle *fuzzy* também se presta a satisfazer variados objetivos de controle, como redução de poluentes ou economia de energia. Nesses casos, regras correspondentes a esses objetivos devem ser colocadas no conjunto.

### 2.3.2) Estrutura do controle *fuzzy*

A Figura 2.10 mostra a estrutura geral de um controlador *fuzzy*. Pode ser observado que há uma adaptação em relação a um esquema tradicional de controle (medição, decisão e atuação). Há duas etapas intermediárias responsáveis pela transformação dos dados antes e depois do controlador propriamente dito, representado pela inferência. São as etapas de fuzificação e defuzificação, que fazem a tradução entre os dois domínios envolvidos: escalar e *fuzzy*.

#### CONTROLE FUZZY

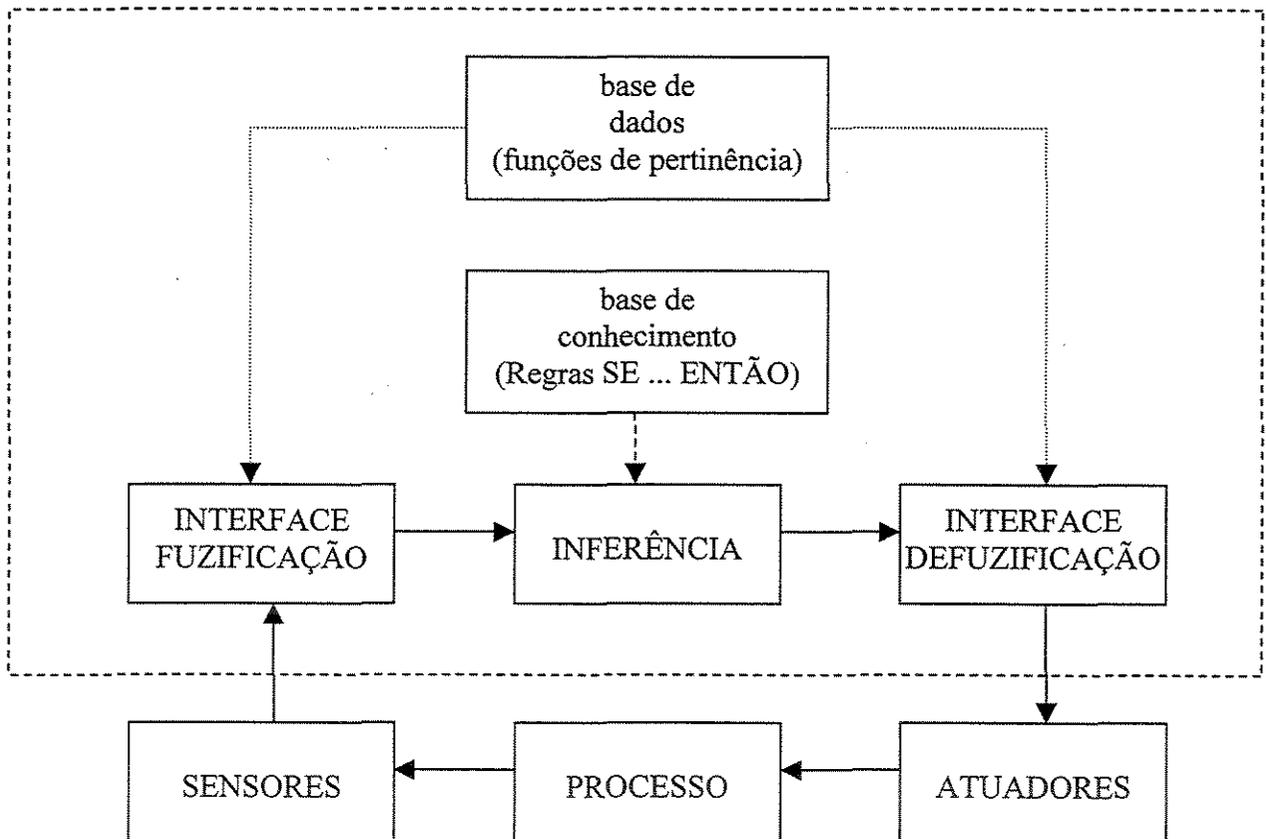


Figura 2.10: Estrutura do controlador *fuzzy*.

Posto isso, é válido estabelecer uma analogia entre o controle *fuzzy* e outro método que exige a tradução de informações entre dois domínios distintos. Trata-se da transformada de Laplace. Da mesma forma que na lógica *fuzzy*, certas informações (equações diferenciais, para a transformada) são colocadas no formato do domínio onde a solução se dá.

## **Fuzificação**

Como já foi mencionado, o conhecimento do controlador *fuzzy* para atuação no processo é codificado num conjunto de regras, cujo antecedente está colocado em termos de variáveis linguísticas. Ou seja, informar ao controlador *fuzzy* que há um erro no processo de cinco graus Celsius não tem nenhum significado. Assim, é necessário colocar essa informação no formato de graus de pertinência dos conjuntos *fuzzy* definidos para a variável erro. Essa é a função da fuzificação. Depois da fuzificação, o gráfico formado pelas regras ativadas é o retrato fuzificado do estado atual do processo.

Apesar da liberdade inerente à escolha da função de pertinência, alguns limites devem ser observados para não comprometer a atuação do controlador. Se esses cuidados não forem tomados, a comunicação entre o sistema e o controlador fica distorcida.

Havendo baixa cardinalidade, a resolução de controle se empobrece. No extremo oposto, haveria um excesso de regras e qualquer condição operacional seria, artificialmente traduzida em muitos graus de pertinência. O número de cardinalidade comumente usado é entre 3 e 7. Quanto à sobreposição das funções de pertinência, uma sobreposição exagerada traria, para uma certa condição, graus de pertinência próximos para conjuntos *fuzzy* distintos, prejudicando o significado dos valores linguísticos. Por outro lado, baixa sobreposição levaria a condições com pequeno grau de pertinência em relação a todos os conjuntos. Assim, a literatura (SHAW e SIMÕES, 1999) recomenda um sobreposição em torno de 50%, com as funções de pertinência interceptando-se no ponto médio entre os graus de pertinência mínimo e máximo (geralmente, zero e um). Os valores das funções de pertinência também podem ser obtida através de uma tabela, quantizando a faixa considerada (LEE, 1990a).

## **Processo de inferência**

De acordo com o conjunto de regras na base de conhecimento, as respostas do controlador *fuzzy* são calculadas em função das entradas. Entre os mecanismos de inferência possíveis, o presente trabalho utiliza o método *máx-mín*.

Para cada regra, são avaliados os graus de pertinência entre as variáveis de entrada e os conjuntos *fuzzy* correspondentes. Depois de obtido este grupo de valores, o grau de ativação da regra é definido como o mínimo. O grau de ativação da regra é atribuído ao grau de pertinência da parte conseqüente, uma vez que quanto maior for a similaridade na

causa da regra, maior será a ativação da consequência. No exemplo abaixo, é interessante notar que o conectivo “e” está associado à função norma-t adotada (*min*) e que o conectivo “ou” representa a função norma-s *máx*. O exemplo é ilustrado pela Figura 2.11.

Regra R1: SE  $x_1 = A$  e  $x_2 = B$  ENTÃO  $y = S1$   
 ou  
 Regra R2: SE  $x_1 = C$  e  $x_2 = D$  ENTÃO  $y = S2$

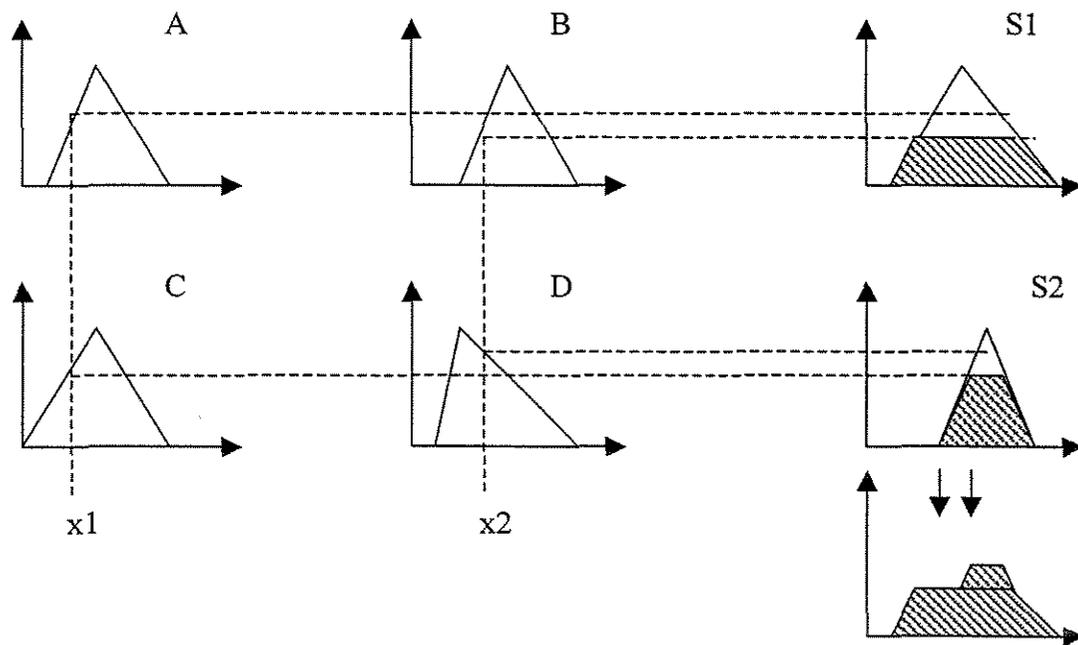


Figura 2.11: Processo de inferência.

Segundo OLIVEIRA (1999), um sistema *fuzzy* de  $n$  entradas e  $l$  saída pode ser visto como uma função

$$F: \mathfrak{R}^n \rightarrow \mathfrak{R}^l$$

que geralmente é fortemente não-linear e uma hipersuperfície de inferência *fuzzy* é descrita por  $y = f(\bar{x})$ .

LEE (1990a e 1990b) discute, de modo matematicamente formal, vários aspectos dos controladores *fuzzy*, desde conceitos básicos de lógica e conjuntos *fuzzy* até estratégias de defuzificação e problemas futuros. O procedimento da Figura 2.11 é mostrado como uma das formas possíveis de raciocínio *fuzzy*. Além disso, o autor aponta uma vantagem no mecanismo de um controlador *fuzzy* em relação a um sistema especialista. No controle

*fuzzy*, que utiliza processamento em paralelo, o conseqüente de uma regra não pode ser aplicado como antecedente de outra (inferência em cadeia).

### Defuzificação

Completada a operação do controlador, a saída da etapa de inferência deve ser convertida num sinal útil ao elemento de controle. Essa é função da interface de defuzificação. A operação também é chamada de condensação. Os principais métodos para se obter o valor defuzificado são: critério de máximo, média dos máximos e centro de área (também chamado de centro de gravidade). Este último é mais utilizado na prática e o valor resultante seria o centro de gravidade das figuras geradas pela inferência. No caso discreto e no contínuo, o resultado é dado respectivamente por

$$escalar = \frac{\sum_i x_i A(x_i)}{\sum_i A(x_i)} \quad (2.1)$$

e

$$escalar = \frac{\int xA(x)dx}{\int A(x)dx} \quad (2.2)$$

Na média dos máximos, computa-se a média entre os pontos de máximo das regras ativadas. Caso o máximo seja um patamar, seu ponto médio é considerado.

A defuzificação pelo critério de máximo, apenas o maior valor entre todas as ativações é considerado como o resultado mais plausível. Contudo este critério apresenta um grande problema de estabilidade. A Figura 2.12 mostra dois casos onde uma pequena mudança nas ativações produz uma grande alteração na defuzificação.

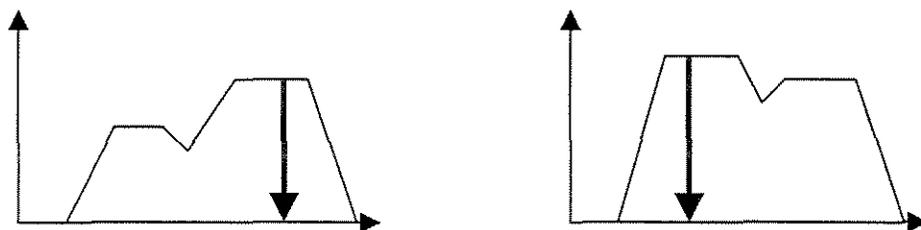


Figura 2.12: Critério de máximo: problema de estabilidade.

A sintonia de todo controlador é feita de forma específica para cada processo, uma vez que cada processo responde de maneira diferente à cada estímulo. Para as técnicas tradicionais, normalmente, é obtida uma primeira estimativa verificando-se a resposta do processo numa malha aberta. Fechando-se a malha de controle, os parâmetros podem ser melhorados analisando a consequência de perturbações em variáveis de carga. São conhecidas metodologias que relacionam a dinâmica do processo com os parâmetros do controlador. Para os sistemas de controle baseados em regras (sistemas especialista e controle *fuzzy*) a sintonia é tão empírica quanto a própria construção do conjunto de regras. Se é preciso observar durante muito tempo a dinâmica do processo para estabelecer as regras, um tempo ainda maior necessário para lograr sucesso na sintonia.

### 2.3.3) Estrutura do controlador *fuzzy* de Li

O artigo de LI e GATLAND (1996) mostra uma abordagem alternativa à estrutura tradicional de um controlador *fuzzy*. É proposta um controle híbrido PID-*fuzzy*, sendo que a saída do controlador tem duas componentes, calculadas independentemente. Uma delas é a saída de um controle PI-*fuzzy* em forma velocidade e a outra representa um controle PD-*fuzzy* em forma posição. Essa abordagem híbrida simplifica a sintonia do controlador, havendo quatro fatores de escala (Figura 2.13).

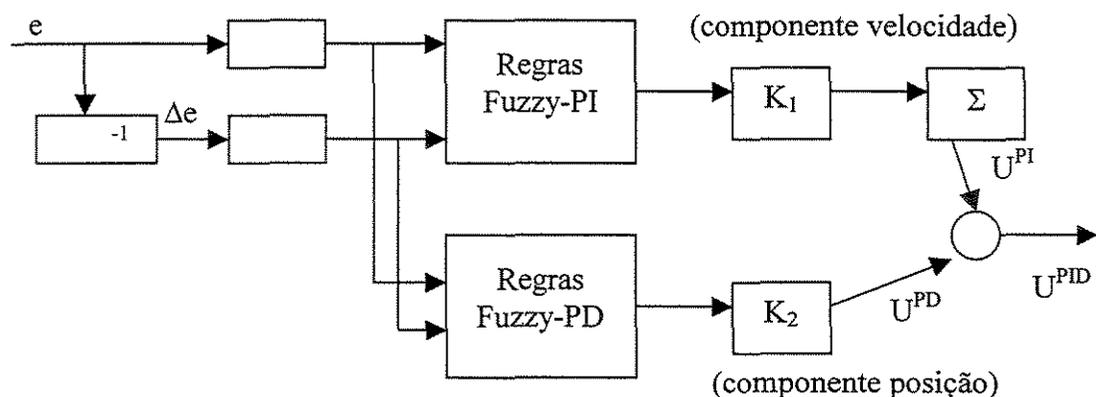


Figura 2.13: Estrutura do controlador PID-*fuzzy* híbrido.

Observando a uma tabela de regras (Tabela 2.1, por exemplo) seria possível concluir que o PID-*fuzzy* necessitaria uma estrutura cúbica do conjunto de regras porque cada uma das variáveis (*erro*), ( $\Delta$ *erro*) e ( $\Delta^2$ *erro*) representaria um eixo. Obviamente, seria

mais complexo trabalhar com uma estrutura do que com conjuntos de regras bidimensionais. No entanto, como a Figura 2.13, a estrutura híbrida tem a vantagem de usar dois conjuntos de regras bi-dimensionais, ao invés de um único conjunto tri-dimensional (Figura 2.14).

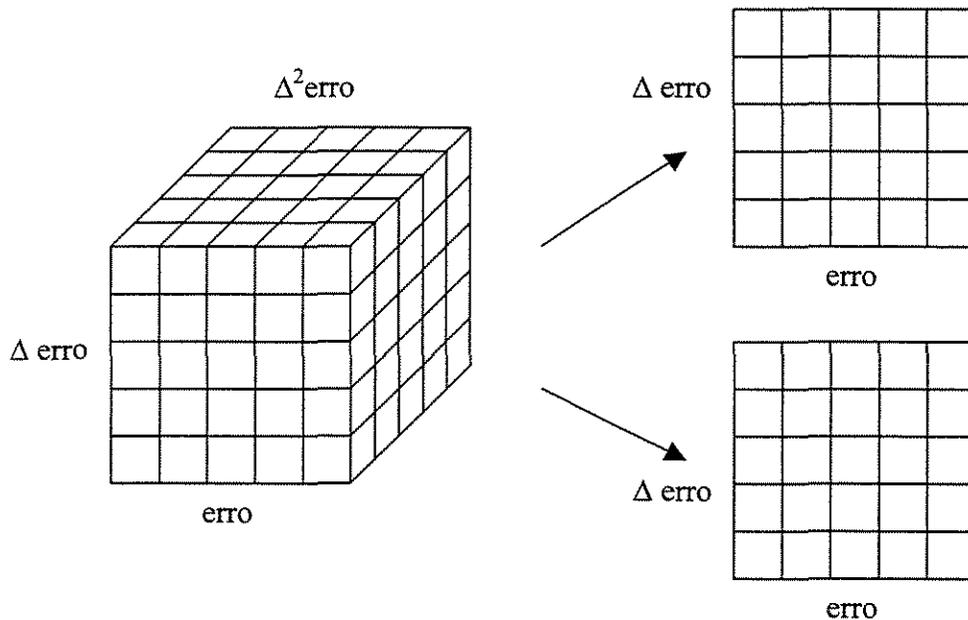


Figura 2.14: Simplificação do conjunto de regras.

### Vantagens e desvantagens

Além da vantagem em expressar a estratégia de controle em termos de regras “SE... ENTÃO”, a natureza qualitativa das regras fuzzy conseguem aproximar-se do pensamento humano. Ou seja, dada uma regra, quanto mais uma condição operacional aproximar-se da causa, maior será a influência da consequência no controle (resultado da defuzificação). Além disso, o uso de termos lingüísticos facilita a discussão de idéias entre as pessoas.

A introdução do controle *fuzzy* em vários processos cotidianos pode ser um passo simples porque é mais fácil encontrar um operador experiente do que compilar o conhecimento sobre um processo complexo.

Os controladores *fuzzy* são inerentemente confiáveis e robustos, resistentes a perturbações externas e ao desgaste e envelhecimento de componentes internos. Sistemas convencionais processam equações complexas em seqüência, e se houver um erro em uma

delas sequer o resultado final é totalmente questionável. Num controlador *fuzzy*, cada regra é processada independentemente e com isso seu efeito no resultado final é mínimo. Isso significa que uma falha parcial do sistema pode não deteriorar significativamente a performance do controlador (SHAW e SIMÕES, 1999).

Considerando um processo não linear, os conjuntos *fuzzy* são mais adequados na descrição de uma função de controle apropriada do que as equações de controle PI e PID, onde é complicado encontrar bons parâmetros que se ajustem a uma faixa larga de operação. Se o projetista de controle deseja satisfazer vários critérios como condições seguras e economia de energia, mais regras podem ser adicionadas ao conjunto usado para atingir esses objetivos.

A maior desvantagem no controle *fuzzy* é a falta de métodos consagrados de projeto e ajuste, de um controlador de caráter geral. Talvez esse seja o preço a ser pago para incluir conceitos vagos no controlador.

#### 2.3.4) Revisão bibliográfica sobre controle *fuzzy*

Tradicionalmente, o controlador é sintonizado com informações que descrevem o comportamento do processo que se deseja controlar, ou seja, faz-se a identificação do processo, observando as respostas dadas para variáveis de entradas conhecidas. Ou seja, a análise é focada sobre o processo, como na Figura 2.15.

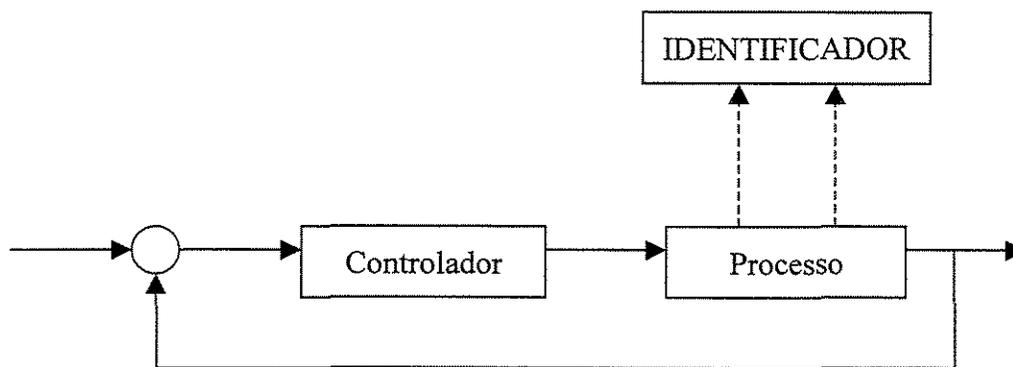


Figura 2.15: Identificação do processo.

Por outro lado, na abordagem para o controle *fuzzy* o foco é dirigido do processo para um operador capaz de controlar o processo, tentando identificar regras heurísticas conhecidas por ele (Figura 2.16). As informações advindas da identificação do processo servem apenas de suporte para a determinação das funções de pertinência.

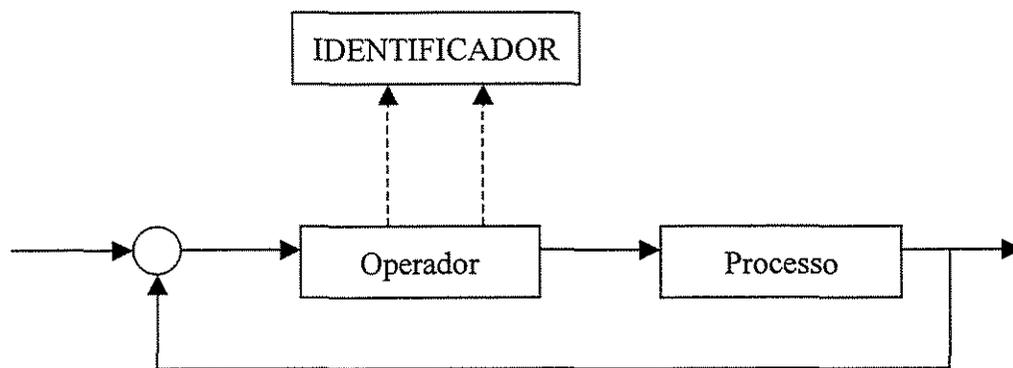


Figura 2.16: Identificação do operador.

O objetivo do controle *fuzzy* é dar um tratamento mais rigoroso ao raciocínio do operador, tipicamente feito em termos de noções vagas e imprecisas.

Os controladores baseados na lógica *fuzzy* selecionam a ação de controle através da similaridade entre a variável medida e algumas condições preestabelecidas. Ou seja, a ação de controle é tomada de acordo com a pertinência da variável medida com alguns conjuntos *fuzzy*.

Modelos não lineares podem ser usados em técnicas de controle, mas se o comportamento do processo muda, o algoritmo deve ser adaptativo para reduzir o resíduo entre o processo e o modelo para garantir a estabilidade do controlador às custas da sua performance (ROFFEL e CHIN, 1991).

LUIZ *et al.* (1997) apontam a deficiência de controladores de realimentação com ganhos fixos em lidar com variações nos parâmetros do processo e em mudanças no meio externo. São comparadas técnicas de controle preditivo, adaptativo e *fuzzy*, que apresentam essa limitação num processo de controle de nível. Para cada modalidade foram feitos dois ensaios: um com variação na carga e outro com variação no *set-point*. Os autores ainda comentaram que não há um método para projeto e sintonia do controlador.

Apesar de estarem conectados, há uma separação significativa entre o controlador *fuzzy* e o processo: um é invisível para o outro. Do processo, são tomadas variáveis de operação e retornado um sinal para o elemento final de controle. Para o controlador, são recebidos graus de pertinência significativos e, de acordo com um conjunto de regras, são calculados os graus de pertinência da variável *fuzzy* de saída. Assim, são necessárias duas etapas no algoritmo de controle: a conversão das condições medidas no processo em

funções de pertinência dos conjuntos apropriados (etapa de fuzificação) e a transformação das informações de saída do controlador num valor de atuação no processo (etapa de defuzificação).

O algoritmo de controle pode ser entendido como uma função relacionando as variáveis de entrada e saída. Para um sistema não-linear, a obtenção e solução das equações dinâmicas pode ser difícil, dependendo da complexidade do sistema. Existem teoremas (OLIVEIRA, 1999) que garantem que podemos aproximar qualquer função no espaço euclidiano através de sistemas *fuzzy* pelo treinamento com pontos conhecidos de entrada e saída, num caso semelhante ao de redes neurais.

Apesar de não haver relação direta entre redes neurais e controle *fuzzy*, essas técnicas compartilham alguns aspectos positivos para o controle de processo. Ambas têm natureza distribuída, não centralizada. Assim como as redes neurais são formadas por nódulos que dividem o trabalho de processar os sinais recebidos, o controle *fuzzy* tem conjuntos que contribuem, cada um com sua parcela, para o resultado final, tanto na etapa de fuzificação quanto a etapa de defuzificação. Com essa idéia de estabelecer um valor final a partir de várias contribuições, o sistema como um todo ganha em robustez, havendo tolerância maior na presença de ruídos ou mau funcionamento de sensores do processo. Essa vantagem não é observada em sistemas que não possuem natureza compartilhada, como o controle PID.

BERNARD (1988) mostra que a implementação da lógica *fuzzy* no controle de processos não foi aceita com facilidade. Primeiramente porque faltava uma metodologia para implementação do controle *fuzzy*. Somado a isso havia poucos estudos comparativos entre o controle *fuzzy* e as técnicas tradicionais. Por fim, a dificuldade de se compor as regras do controlador. Além disso, o autor também aponta a dificuldade de, a partir de critérios de performance, modificar ou mesmo construir o conjunto de regras. Ainda sobre os critérios de performance, é de difícil compreensão o impacto de vários aspectos do controlador sobre o desempenho, de modo que algumas perguntas ficam sem uma resposta objetiva:

- Há cardinalidade suficiente?
- Qual o conjunto *fuzzy* que poderia ser eliminado ou modificado para melhoria de desempenho?

- A base de regras é completa? Ou seja, existe alguma situação que não é coberta pelas regras?

O interesse da lógica *fuzzy* como instrumento de controle pode ser sentido pelo número de aplicações em que é usada variando desde análise financeira até prevenção a terremotos. Entre os mais importantes podemos citar: trocadores de calor, lama ativada, controle de tráfego, fornalha de cimento, controle de aeronave, purificação de água, sistemas de potência e reatores de usinas nucleares, componentes de memória e computadores *fuzzy* (LEE, 1990b). Abaixo, são descritos, de forma mais detalhada, dois exemplos de aplicação de controle *fuzzy*.

Numa coluna de flotação, o pouco conhecimento sobre os fenômenos químicos e físicos impede o desenvolvimento de um modelo, mesmo em regime permanente. Em vista disso, CARVALHO e DURÃO (2002), usaram um modelo linear dinâmico capaz de descrever relações entre variáveis manipuladas e controladas. Na verdade, apenas algumas dessas relações se mostraram satisfatoriamente modelados. Primeiramente, malhas SISO independentes falharam devido a interferências mútuas, causando frequente desativação de algumas malhas para reabilitar a estabilidade. Mas a existência de operadores habilidosos que poderiam lidar com o processo, foi uma pista para aplicação de controle *fuzzy*. Como geralmente acontece, a sintonia do controlador é feita experimentalmente, de acordo com a performance.

Um caso de emprego do controle *fuzzy* onde todas as tentativas anteriores de automação falharam é relatado (FODIL *et al.*, 2000). A aplicação ocorreu na França, onde 80% da energia elétrica é gerada por vapor pressurizado de reatores nucleares, e a variação na demanda dificulta a obtenção de um sistema de controle que responda eficazmente. A melhor forma encontrada era através de um operador. O processo é altamente complexo, não-linear e descrito somente por um modelo numérico (não analítico), além de ter um grande atraso na resposta. A potência gerada é resultado de balanço entre emissão absorção de neutrons. Assim, a atuação é feita pela concentração de boro, que também é um elemento absorvente na água do circuito.

Analisando a literatura, observa-se ainda que o uso de controle *fuzzy* não se trata nem de um modismo que precisa demonstrar sua eficiência, nem de uma técnica antiga que já se encontra superada. Na década de 80, SUGENO (1985) já descrevia vários casos de

aplicações industriais. Mais recentemente, na década de 90, também foi utilizada em diversos segmentos da indústria (HIROTA e SUGENO, 1995). Atualmente, também encontram-se vários autores com artigos dedicados ao controle *fuzzy*. CHEN e PENG (2004) mostram uma abordagem neuro-*fuzzy* no controle de processos químicos não-lineares com dinâmica pouco conhecida, como um reator CSTR com reações secundárias não modeladas. ZHOU *et al.* (2004) desenvolveram um controle *fuzzy* adaptativo robusto para um sistema multivariável, não-linear e com comportamento dinâmico desconhecido. TONG *et al.* (2004) também utilizam técnicas baseadas na lógica *fuzzy* para controle de sistemas SISO caracterizados por incertezas e não-linearidades. FOULY e GALICHET (2003) usam o mapeamento de um controlador *fuzzy* para introduzir meios de manipular entradas *fuzzy*. Os conceitos propostos são aplicados no controle PI de um processo de primeira ordem e num processo não-linear para ilustrar um caso mais complicado. ODETUNJI e KEHINDE (2004) desenvolveram um sistema de controle *fuzzy* para simulação de fermentação com bons resultados como rápida resposta de controle, redução de energia e melhoria do produto.

## 2.4) Controle convencional por retroalimentação

O desenvolvimento de controlador digital proporcional-integral-derivativo (PID) é um método desenvolvido para corrigir desvios indesejados na variável controlada, através de uma manipulação de uma outra variável, denominada variável manipulada. O controlador aqui empregado é do tipo *feedback* digital, o qual baseia-se na medição da variável controlada, seguido por uma comparação com o valor do *set-point* (valor de referência pré-estabelecido). Na seqüência, o erro é processado pelo algoritmo de controle, que retorna um valor para o elemento final de controle, fechando a malha de controle. Desta forma, a variável controlada deve permanecer em torno do valor de *set-point* e esta oscilação dependerá da sintonia do controlador principalmente.

O controlador PID trabalha com uma soma de três termos que serão responsáveis pela manipulação do erro, cada um correspondente às parcelas que constituem o PID: proporcional, integral e derivativo. O primeiro é uma multiplicação do erro por uma constante de proporcionalidade ( $K_c$ ), que será o ganho do controlador, obtido em sua sintonia. O segundo termo é referente à integração deste erro que introduz ao controlador uma melhoria, pois com esta nova parcela o controlador passa a ser PI (proporcional-

integral) e sua função é eliminar o *off-set*, o que torna o controlador mais eficaz, porém obtendo-se um período de grande oscilação antes de se chegar ao valor desejado. O último termo que é adicionado ao controlador, agora chamado de PID (proporcional-integral-derivativo), para suavizar esta oscilação inicial e trata-se da derivação do erro no tempo. A equação original do controlador PID analógico encontra-se a seguir:

$$\mu = \mu_{ss} + K_C \left\{ e(t) + \frac{1}{\tau_I} \int e(t) dt + \tau_D \frac{d e(t)}{dt} \right\} \quad (2.3)$$

onde  $\mu_{ss}$  é o valor da variável manipulada no estado estacionário, o  $\tau_I$  é denominado de constante de tempo integral, o  $\tau_D$  é a constante de tempo derivativo e  $K_C$  é a constante de proporcionalidade ou ganho do controlador.

Para ilustrar claramente o esquema de controle descrito acima, apresenta-se o diagrama de blocos (Figura 2.17) para representar a malha de controle *feedback*.

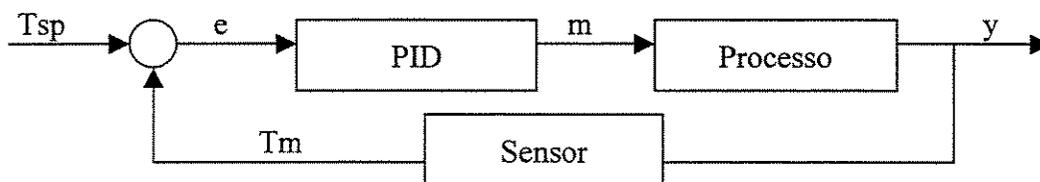


Figura 2.17: Diagrama de blocos da malha controle PID.

Evidentemente, esses parâmetros têm grande influência na atuação do controlador e no seu desempenho. Valores inadequados podem, para uma dada condição de erro, produzir uma resposta insuficiente para o retorno da variável controlada para próximo do *set-point*, mantendo uma margem persistente. Outro resultado ainda mais indesejado é um conjunto de parâmetros que resulte numa resposta exagerada, levando a um controle do tipo liga/desliga e, tendo como consequência, a instabilidade na operação do processo. A sintonia (*tuning*) desses parâmetros é feita através da identificação do processo, determinando sua resposta frente a uma perturbação. Essa resposta é avaliada tanto em termos de ganho de estado estacionário como no tempo gasto para se atingir o novo estado (constante de tempo), além da possível existência do tempo morto.

O método empregado na sintonia do controlador é o da curva de reação do processo, seguida da aplicação das fórmulas semi-empíricas de Cohen-Coon, para manter

possibilitar comparações com o trabalho de NUNES (2003). Este método se baseia em causar uma perturbação, em degrau (*step*), na variável manipulada (*m*) em malha aberta e analisar a dinâmica da variável controlada (*y*), graficada em função do tempo. Vale ressaltar que se considera como tempo zero o instante em que a perturbação é implementada em malha aberta.

A equação do controlador, usada no programa desenvolvido, não é a equação clássica do PID, mas sim uma modificação para um controlador digital, a qual continua sendo a mesma equação, mas aproximada de forma a não haver a necessidade de se calcular a integral nem a derivada (PID forma velocidade). Tal fato é fruto da aplicação computacional, pois visa reduzir o número de cálculos, o requisito de memória do computador e, sobretudo, visa evitar a saturação da válvula de controle, típica do controlador PID clássico (em sua forma posição).

A equação a seguir é do PID, na sua forma velocidade, encontrada através da subtração das equações do PID, forma posição, aplicada a dois intervalos de amostragem consecutivos:

$$\begin{aligned} \Delta C_n = C_n - C_{n-1} = K_C \left( 1 + \frac{\Delta t}{\tau_I} + \frac{\tau_D}{\Delta t} \right) e_n \\ - K_C \left( 1 + 2 \frac{\tau_D}{\Delta t} \right) e_{n-1} + K_C \left( \frac{\tau_D}{\Delta t} \right) e_{n-2} \end{aligned} \quad (2.4)$$

Como o controlador é digital (vide programa no apêndice 1), trabalha-se com intervalos de amostragem ( $\Delta t$ ), fornecendo como resposta um incremento que será adicionado ou subtraído ao valor atual, a fim de trazer o valor da variável controlada em direção ao *set-point*.

## 2.5) Comunicação de dados

A primeira forma usada na comunicação de informações entre sensores e atuadores com o controlador era analógica, sendo que a intensidade do sinal poderia ser transmitida na forma pneumática (pressão) ou elétrica (corrente ou voltagem). Apesar do potencial de representar todos os valores numa faixa de variação, o sinal analógico é pobre no sentido da quantidade de informações sobre o processo, trazendo somente o valor atual da variável. Cada informação necessita de um canal exclusivo para sua transmissão. O sinal

do sensor para o controlador não poderia ocupar o mesmo par de fios do sinal de controle, do controlador para o atuador. O algoritmo de controle é executado remotamente, numa sala de controle, e de forma centralizada. Ou seja, a parada de um controlador comprometeria as malhas sob sua observação.

Com o surgimento de sistemas de comunicação digital, foi possível o estabelecimento de protocolos de comunicação, atribuindo-se significado às seqüências de bytes usados na comunicação, assim como é feito numa rede de computadores. No formato digital, a quantidade de informações que podem ser transmitidas aumenta sensivelmente. Além do valor da variável, é possível transmitir dados sobre diagnóstico de falhas, alarmes, *set-points* e parâmetros de sintonia de forma bidirecional. Além disso, pacotes de dados endereçados à máquinas diferentes podem compartilhar o mesmo canal, havendo economia e simplificação no cabeamento.

O *Fieldbus* é definido como o elo de comunicação serial entre os dispositivos de automação, primários localizados junto ao processo, e o nível de controle imediatamente superior na sala de controle com mais capacidade de processamento (Figura 2.18). Ele tem algumas funções interessantes de comunicação, não presentes na maioria dos protocolos, como detecção automática de dispositivos, espaço reservado para *Plug and Play* e sincronização de tempo.

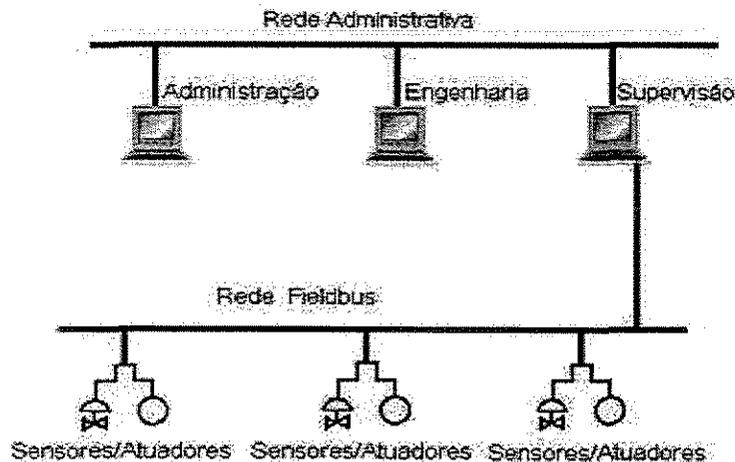


Figura 2.18: Esquema de barramento de campo (NUNES, 2003).

Ao contrário das iniciativas anteriores de controle distribuído onde protocolos foram projetados para ter ampla atuação e depois modificados a fim de obter maior

desempenho nas aplicações críticas, os *fieldbuses* já foram concebidos para o nível de controladores e transmissores.

Aproveitando-se a flexibilidade permitida pelo sistema *fieldbus*, este trabalho propõe a utilização de um sistema de monitoramento e controle híbrido (analógico e digital) através de uma interface *fieldbus* (barramento de campo). Assim, excetuando os elementos de saída digital (bomba e aquecedor), os outros dispositivos do sistema de controle proposto funcionam de forma analógica.

### **2.5.1) Protocolo aberto: Fieldbus Foundation**

No desenvolvimento dos primeiros protocolos de comunicação digital, cada empresa decidiu por um formato proprietário que fosse incompatível com os demais, na tentativa de conquistar mercados. Assim, uma indústria deveria comprar todo o sistema de controle de uma mesma empresa, elevando o custo do sistema de controle. Outra alternativa era sujeitar-se a ter ilhas de automação, sem troca de informação entre si, reduzindo o escopo de automação. Essa situação era claramente desfavorável para as indústrias de processos porque, os componentes do sistema de controle não podiam ser escolhidos caso a caso, elegendo os melhores fornecedores para cada componente. Além disso, os planos de expansão ou melhoria do sistema de controle ficariam seriamente limitados.

Para resolver o problema de padronização de protocolos na automação, o protocolo *Fieldbus Foundation* foi proposto inicialmente em 1994 por uma fundação internacional de mesmo nome, sediada nos Estados Unidos, composta pela união de duas organizações de *fieldbus* existentes, a ISP (Interoperable System Protocol) e uma divisão da WorldFIP francesa, mais um grupo de 85 empresas. Aliados aos organismos internacionais de normalização IEC e ISA, a *Fieldbus Foundation* é uma organização independente sem fins lucrativos que tem o objetivo de desenvolver e manter um barramento de campo para trabalhos de automação internacionalmente uniforme e de qualidade. Entre os membros estão fabricantes de dispositivos de campo e sistemas de automação. Um dos conceitos chave do *fieldbus* é a interoperabilidade, que é a possibilidade do emprego de dispositivos de fornecedores diferentes num mesmo sistema, perfeitamente compatíveis. Além disso, um componente Fieldbus Foundation pode ser substituído por um de outro fabricante, também membro do Fieldbus Foundation.

Organismos de padronização pioneiros nos trabalhos com *fieldbus*, como o IEEE, o IEC e a ISA, iniciaram os trabalhos por elaborar um conjunto de requisitos iniciais que o novo protocolo deveria suportar, no que se diz respeito a tempos de resposta, topologia física, redundância, confiabilidade, número de dispositivos por ramo de rede, método de acesso ao meio físico, alimentação dos transmissores, comprimento do barramento e métodos de configuração do sistema.

As especificações Foundation não estão finalizadas, portanto, atualmente fabricantes e usuários trabalham para definir e aprovar os capítulos inacabados da norma. Por outro lado, é natural que mesmo as especificações que já foram aprovadas passem por um contínuo processo de evolução. Observa-se que periodicamente novas revisões e novos capítulos são publicados pela *Foundation*, na medida em que os requisitos dos sistemas de automação vão se tornando cada vez mais complexos, e que os fabricantes e usuários vão acumulando experiência na operação e instalação de sistemas.

PINCETI (2004) explica o conceito de “instrumento virtual”, que consiste em atribuir um grau maior de inteligência a um instrumento, usando tecnologia digital. O autor ainda aponta tendências para o futuro das redes *fieldbus*. Num estágio ideal, todo dispositivo é visto como um recurso computacional, podendo realizar qualquer função como executar um software PI no processador de um transmissor de temperatura ou calcular o custo energético num transmissor de pressão. Isso mostra que o *fieldbus* é mais que um *link* de comunicação. Também é mencionado que qualquer instrumento de campo no mercado atual comunicam-se em dois protocolos: *Profibus* e *Foundation Fieldbus*.

Um sinal de que as redes estão evoluindo é seu uso combinado com outras tecnologias que facilitam seu desenvolvimento e potencializam seus recursos. KUMES e SAUTER (2001) propõem uma abordagem para conectar uma rede *fieldbus* à internet, visando monitoramento e manutenção por acesso remoto, com um protocolo que seja independente do tipo de rede *fieldbus*. A integração com a linguagem de programação Java também deve ser ressaltada como um ponto positivo. BÜHLER *et al.* (2001) utilizam o *Java Fieldbus Control Framework* para criar funções de controle que sejam executadas em paralelo num sistema de automação, baseado em *fieldbus*, tirando proveito da reusabilidade de um programa orientado a objetos e da portabilidade do Java. BÜHLER e KÜCHLIN (2000) representam os dados de um sistema remoto *fieldbus* como documentos XML,

formando um repositório central de informações sobre o processo para toda empresa e que pode ser acessado por ferramentas XML. LUMPP *et al.* (1998) introduzem o conceito de Dispositivo Virtual Java para integrar sistema *fieldbus* na internet/intranet para fins de monitoramento, manutenção e controles remotos, usando a tecnologia Web.

### **2.5.2) Blocos Funcionais do *Fieldbus***

Os blocos *fieldbus* são estruturas que podem realizar funções básicas de uma malha de controle, como algoritmos de compensação, aquisição e atuação e são realizados respectivamente fechando assim as malhas de controle da aplicação.

Também possuem parâmetros de configuração, de entrada e de saída além do algoritmo em si e são identificados e acessados nos equipamentos por um nome ou tag exclusivo, e estão integrados aos equipamentos.

Dependendo das características do bloco, este pode ser definido com *transducer*, *resource* ou *function block* (bloco funcional). Um *resource block* é utilizado para especificar as características gerais do equipamento onde este se encontra, assim só é permitido um bloco *resource* por equipamento e como é um bloco de especificações, não possui parâmetros de entrada ou de saída. O bloco *transducer* serve para desacoplar os outros blocos de operações específicas de I/O nos diferentes sensores e atuadores, é ele quem opera o hardware do dispositivo físico.

#### ***Resource block***

As características do hardware associado ou recurso estão descritas nos parâmetros de configuração do *resource block*. Também o *resource* possui os parâmetros universais, e através destes parâmetros de configuração, o *resource* isola os blocos funcionais das características de *hardware* dos dispositivos.

O algoritmo do *resource* é utilizado para monitorar o estado de operação do *hardware*, e indicar possíveis alarmes neste aspecto.

A execução do *resource* não entra no escalonamento dos blocos, portanto este bloco não possui parâmetros de entrada ou de saída. A execução do *resource* segue regras que são definidas pelo fabricante. Um dos parâmetros de configuração presentes no *resource* é o *resource state*. Este parâmetro indica o estado do hardware:

- *Start / Restart;*

- Inicialização;
- *On-line*;
- *On-line linking*;
- *Failure*;
- *Stand by*.

Em alguns casos outros blocos podem utilizar os parâmetros do *resource* por acesso local.

### ***Transducer Block***

O bloco *transducer*, ou transdutor tem a função de isolar os blocos funcionais das funções de I/O específicas assim como acionamento de sensores, atuadores ou chaves.

O algoritmo do *transducer* realiza a aquisição, linearização e filtragem do dado, disponibilizando para os blocos de entrada um valor em formato padrão independente do sensor. A comunicação do *transducer* com um bloco funcional de entrada ou saída é feita através de canais, sendo que um *transducer* pode ter vários canais. Uma das vantagens deste isolamento do bloco funcional com o hardware é a possibilidade da execução do *transducer* quantas vezes forem necessárias para gerar um dado de boa qualidade sem sobrecarregar os blocos funcionais.

Os blocos *transducer* são classificados em três tipos:

- *Input* - realizam a interface com sensores;
- *Output* - realizam a interface com atuadores ou dispositivos de saída;
- *Display* - operam dispositivos de interface locais.

Assim como o *resource block*, o *transducer* não possui parâmetros de entrada e de saída e sua execução não é comandada pelo *schedule*, mas definida pelo fabricante.

### **Blocos funcionais**

Os blocos funcionais realizam as ações e funções básicas presentes em um sistema de automação e controle, independentemente da plataforma de *hardware*. Eles possuem parâmetros de entrada, saída e de configuração e a partir destes parâmetros realizam um algoritmo específico.

Os blocos funcionais são classificados em:

- Blocos de entrada: acessam medidas físicas através da comunicação com *transducer blocks* de entrada por meio de canais. Estas medidas então são disponibilizadas para outros blocos através dos parâmetros de saída;
- Blocos de saída: acionam *transducers* de saída pelos canais a partir de um valor recebido pelos *links* com outros blocos nos seus parâmetros de entrada. Realizam também a *back-calculation*;
- Blocos de controle: Realizam cálculos com parâmetros de blocos de entrada e enviam parâmetros para outros blocos de controle ou para blocos de saída. Realizam também a *back-calculation* e possuem processamentos que evitam efeito de *bump* e de *wind-up*;
- Blocos de cálculo: Realizam cálculos matemáticos sobre parâmetros de entrada, gerando parâmetros de saída.

### **2.5.3) Controle distribuído**

A estrutura de um sistema de controle, considerando não apenas malhas individuais, mas toda uma fábrica é claramente triangular, com um lado junto ao processo, e níveis intermediados de barramentos de dados. Este lado está um grande número de dispositivos que sensoreiam e atuam no processo, com um ciclo de operação rápido processando muitos dados de significado local, apenas variáveis do processo. Acima deles fica a nível dos controladores, que recebem os sinais dos sensores e envia para os atuadores, trabalhando com processamento mais sofisticado e podendo ser um pouco mais lento que o nível abaixo. Os controladores, além de interagirem com os componentes junto ao processo, também trocam informações com máquinas de um nível hierárquico acima, ocupadas em gerenciar a produção de um modo global, através do ajuste de *set-points* e da detecção setores da linha de produção com problemas, trabalhando com informações de significado mais amplo. Ou seja, a medida que se sobe na hierarquia, as informações processadas têm maior importância e o número de componentes é menor. Esta estrutura tem relação com a distribuição do sistema de controle.

Num sistema absolutamente centralizado, todos os algoritmos de controle seriam processados na mesma máquina. Obviamente, esta configuração não é confiável porque todo o sistema depende de uma única máquina. Assim, uma das medidas para tornar o sistema mais confiável é distribuir a função de controle entre várias máquinas com canais de comunicação entre elas. Desse modo, quando uma delas pode suprir o processamento de uma que apresentar problemas. O sistema Fieldbus possibilita uma distribuição, e confiabilidade, ainda maior. Os dispositivos que atuam como sensores e atuadores podem acumular outras funções. A adição dos blocos funcionais apropriados (por exemplo, blocos de controle PID ou PD) leva a tarefa de controle para os dispositivos em contato com o processo. A vantagem dessa maior distribuição é a diminuição de carga nos controladores propriamente ditos e até a substituição deles em pequenas instalações. Ou seja, uma malha de controle pode ser constituída apenas de um sensor e um atuador, havendo comunicação entre eles, evidentemente.

#### 2.5.4) *Fieldbus*: vantagens e desvantagens

A redução do custo de fiação do *fieldbus* sobre a tecnologia SDCD, foi avaliada por MANGANO e DUNDIE (1998) em 40% a 60%, resultando, juntamente com outros fatores, em uma redução do custo de projeto (hardware, software, engenharia e instalação) de 55% sobre o SDCD. Tal economia em cabeamento se deu principalmente pela substituição da topologia ponto a ponto pela topologia de barramento multiponto, Figura 2.19.

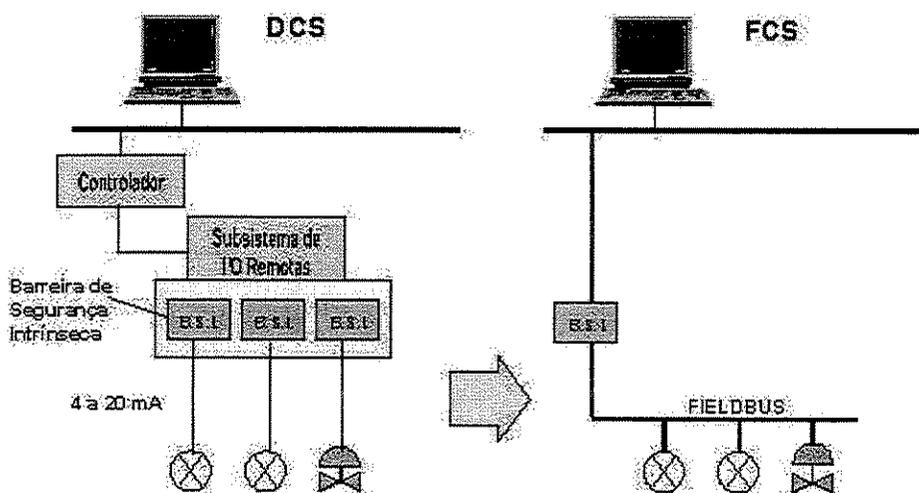


Figura 2.19: Comparação da estrutura entre DCS e FCS.

Segundo BRANDÃO (2000), as vantagens que devem ser observadas neste tipo de barramento são:

- Possibilidade da migração do processamento do controlador para os transmissores;
- Redução do custo de fiação;
- Maior facilidade e economia na manutenção e na instalação, devido a possibilidade de configuração remota dos transmissores;
- Maior desempenho na comunicação, totalmente digital;
- Disponibilização rápida de informações em qualquer ponto da rede, além de uma maior gama de informações disponíveis;
- Detecção automática de falhas nos transmissores e no barramento;
- Expansão e evolução flexíveis;
- Diminuição do tempo dos loops de controle;
- Diminuição do tamanho e complexidade das salas de controle;
- Dispositivos de campo energizados pelo barramento;
- Topologia em árvore ou em linha;
- Modelos de blocos padronizados;
- Segurança intrínseca para ambientes perigosos;
- Capacidade de comunicação multi-master;
- Padronização no modelo de blocos.

Como desvantagens do *fieldbus* tem-se a concentração da comunicação em poucos canais de rede, o que compromete o desempenho do tráfego de informação e torna todo o sistema mais suscetível a um rompimento do barramento, portanto a concentração de malhas por canal deve ser estudada caso a caso, conforme comenta PRZYBYLSKI (1996).

Várias iniciativas de normalização de *fieldbus* foram observadas e existe atualmente um número considerável de tipos de *fieldbus* que atendem aos requisitos de diferentes tipos de chão de fábrica.

A padronização internacional do barramento de campo tornará fácil a elaboração de interfaces para os diversos tipos de aplicações industriais, baseados em um mesmo protocolo, além de tornar o mercado de sistemas aberto, ou seja, garantir a interoperabilidade de equipamentos de diferentes fabricantes.

### 2.5.5) Conclusões

Neste capítulo, foram apresentados os aspectos mais importantes dos assuntos envolvidos no trabalho. Com a lógica *fuzzy*, foram introduzidos conceitos de números e conjuntos *fuzzy*, criando a possibilidade de definir as variáveis envolvidas no sistema de controle de modo linguístico. Feito isso, a estrutura do controlador é explicada, fechando a malha de controle com as etapas de fuzificação, inferência e defuzificação. Foi mostrado que o conhecimento utilizado pelo controlador é colocado no formato de regras, que são processadas na inferência. As etapas de fuzificação e defuzificação são responsáveis pela tradução entre os domínios escalar e nebuloso.

Este capítulo descreve também o processo de polimerização, cuja temperatura se deseja controlar. São apontados aspectos potencialmente problemáticos para o controle de temperatura que justificam o uso do controle *fuzzy*.

Na literatura recente, não há muitos artigos relatando testes experimentais de controle em reações de polimerização de PMMA.

Pelo potencial de utilização de controle *fuzzy* em vários processos, descrito no item 2.3.4, e pelas dificuldades de controle de polimerização do MMA relatadas no item 2.1.3, propõe-se a aplicação experimental deste controlador a um reator em escala de laboratório. Ou seja, o objetivo principal do trabalho é verificar qual o método de controle mais adequado para o processo de polimerização partindo-se de resultados de perturbações em malha aberta. Através destes métodos, são alterados parâmetros de sintonia buscando minimizar os índices de desempenho IAE e ISE.

No centro do trabalho está o problema de seleção de metodologias de controle para um processo específico. É sabido que processos simples e de comportamento linear são

controlados de modo satisfatório com os controladores convencionais PID, com parâmetros fixos. No outro extremo de espectro, estão processos complexos e altamente não-lineares que necessitam de um algoritmo de controle mais elaborado, como o *fuzzy*.

Várias fontes da literatura afirmam que a polimerização do MMA é altamente não-linear, mas há poucos trabalhos que medem essa complexidade com resultados experimentais comparativos entre metodologias de controle, de modo a saber até que ponto o controle *fuzzy* é necessário, ou se um esquema PID convencional ou PID adaptativo seria suficiente. A maioria dos trabalhos aplica um determinado controlador e analisa os resultados em relação a ele.

Nos casos mais difíceis, o controlador *fuzzy* é especialmente útil porque, além da capacidade intrínseca de mapeamento não-linear, a estrutura do controlador (Figura 2.10) disponibiliza várias alternativas de sintonia. Entre as principais estão a mudança no formato e na posição das funções de pertinência, alteração na tabela de regras e aumento na cardinalidade. Como foi usada a estrutura de LI (Figura 2.13), também é possível ajustar os fatores de escala. Contudo, a sintonia do controlador *fuzzy* pela variação nos fatores de escala deve ser evitada porque, neste caso, o controlador poderia ser substituído por uma caixa preta, sem qualquer relação com a lógica *fuzzy*.

## **Capítulo 3**

### **Estudo experimental da polimerização do MMA**

## Introdução

O objetivo do presente trabalho é estabelecer comparações entre metodologias de controle, PID convencional e *fuzzy*, para manutenção de temperatura na polimerização do MMA, processo complexo e não-linear. Neste capítulo são apresentados o procedimento da reação de polimerização, a implementação dos controladores convencionais (PID) e *fuzzy*. Também são discutidos os procedimentos de sintonia dos controladores PID, PID adaptativo e *fuzzy*. Uma comparação entre os resultados da aplicação dos controladores é apresentada no capítulo de resultados. A relação dos materiais usados no desenvolvimento do presente trabalho encontra-se no Apêndice 1. No final do presente capítulo são apresentados alguns resultados preliminares.

### 3.1) Aparato experimental

A reação é feita de modo batelada num reator de vidro de um litro de capacidade. O elemento final de controle é uma válvula regulando a vazão de água pela jaqueta. O calor necessário é fornecido por dois aquecedores elétricos, um deles (75 W) permanecendo ligado durante todo o tempo da batelada e outro controlado de forma liga/desliga com atuação quando há um desvio maior do que 4 °C em relação ao *set-point* (45 W). Acetato de etila é usado como solvente e peróxido de benzofila (BPO) como iniciador. Como o oxigênio atua como inibidor da polimerização, nitrogênio deve ser borbulhado antes e durante a reação. O metacrilato de metila é inibido por hidroquinona, e é separado desta através de destilação à vácuo. O apêndice 1 especifica, com maior detalhamento o material usado. A Figura 3.1 mostra o reator usado.

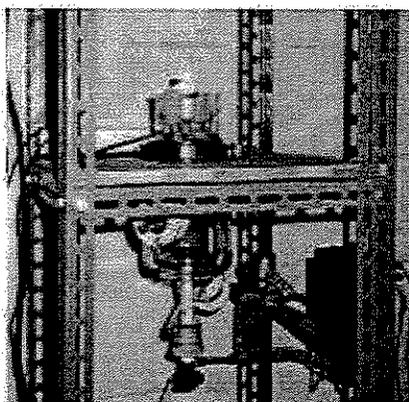


Figura 3.1: Reator piloto.

Um esquema com os componentes do reator é mostrado na Figura 3.2.

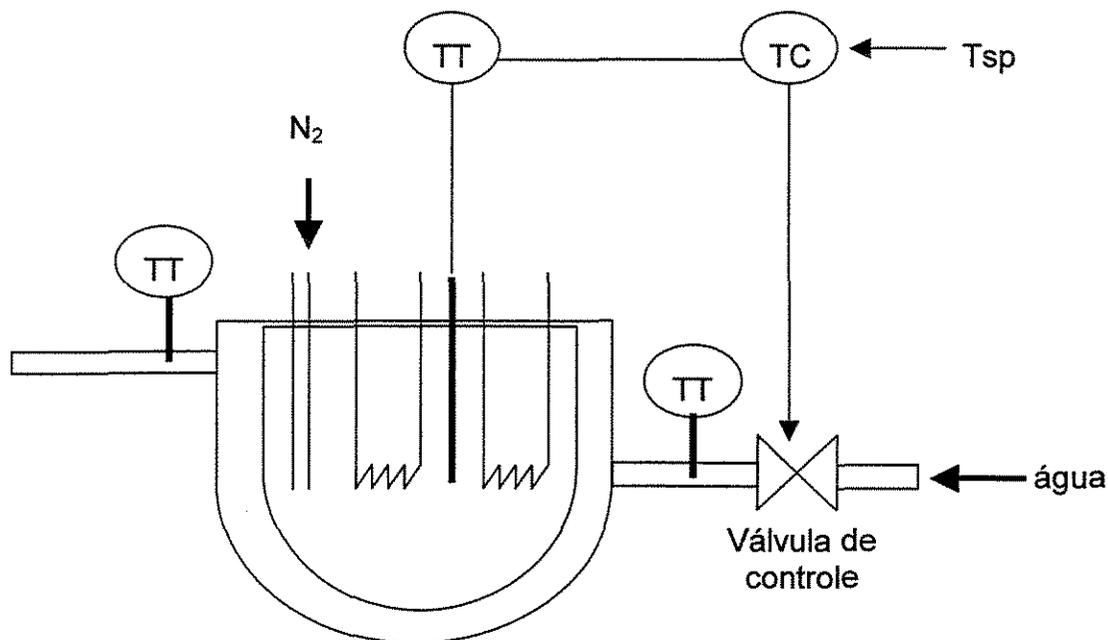


Figura 3.2: Esquema do reator.

O maior objetivo é o controle de temperatura, através da manipulação da vazão de água de resfriamento. A literatura indica que a reação ocorre numa faixa de temperatura em torno de  $60^{\circ}\text{C}$  (CROWLEY e CHOI, 1996; GHOSH *et al.* 1998). O *set-point* de temperatura foi obtido empiricamente. Uma quantidade razoável de polímero foi produzida a  $63^{\circ}\text{C}$ . Além da temperatura do reator, as temperaturas de entrada e saída da água de resfriamento também são medidas. Um densímetro on-line é usado para tentar inferir a conversão. Uma interface *Fieldbus* é usada para estabelecer comunicação entre o processo e a rede local de computadores, além de auxiliar na configuração dos dispositivos de entrada e saída. O sinal entre esta interface e os sensores de temperatura, massa específica e a válvula de controle é analógico (4-20 mA). Do lado da rede local, um servidor OPC é responsável para fornecer ao programa de controle (cliente OPC) as variáveis medidas e também comunicar à interface a ação de controle a ser implementada. OPC é um padrão de comunicação baseado na tecnologia OLE da Microsoft que traz interoperabilidade aos sistemas de automação e controle, com a definição padronizada de métodos de interface dos objetos.

Há uma razão para a utilização de um sistema de aquecimento e colocado juntamente com um sistema de resfriamento do reator. Primeiramente, deve-se lembrar que, por razões de segurança, a reação é conduzida em diluição considerável e, assim, o calor liberado pela reação não é suficiente para manter a temperatura num nível adequado, dispensando os aquecedores. Por outro lado, o sistema de resfriamento justifica-se pela falta de controle fino de aquecimento e para evitar um aumento excessivo de temperatura em decorrência do efeito gel.

### **3.2) Procedimento experimental**

O monômero MMA com 10 a 100 ppm de hidroquinona deve ser destilado, pois a hidroquinona é um forte inibidor dos radicais livres formados na reação de polimerização. O processo de separação escolhido foi a destilação à pressão reduzida (ou a vácuo), sendo à temperatura de 34-36 °C e pressão de 60 mmHg, conforme descrito em VOGUEL (1988). O monômero purificado é armazenado à baixa temperatura (-5 °C) para evitar a ocorrência da polimerização.

Para a reação, são colocados 75 mL (25%) de MMA juntamente com 225 mL (75%) de acetato de etila, usado como solvente. Como o oxigênio é prejudicial à reação, gás nitrogênio é borbulado durante 30 minutos na mistura do monômero com o acetato de etila. O aquecedor mais potente é ligado e quando a temperatura do meio atinge um valor entre 55-60 °C, o iniciador é adicionado (2,12 g, correspondente a 2% relativo a massa de MMA) e o sistema de controle entra em operação. O tempo de batelada adotado, de seis horas, é estimado com base na literatura (PAN e LEE, 2003; SOROUSH e KRAVARIS, 1993). O nitrogênio é borbulhado o tempo todo. A leitura dos dados e atuação no processo é feita a cada 60 segundos (aproximadamente 20% da constante de tempo). Para melhorar a mistura do meio reacional e forçar sua passagem pelo densímetro, a bomba é acionada a cada 180 segundos. Terminado isso, o aquecimento é ligado e, quando a temperatura se aproxima de 55 °C, o iniciador (peróxido de benzoíla, BPO, na quantidade de 2% da massa do monômero) é colocado e o programa do sistema de controle é executado. O borbulhamento do nitrogênio é mantido durante toda a reação. O procedimento experimental igual ao usado em NUNES (2003), para manter uma comparação válida. A Figura 3.3 mostra o fluxograma da reação.

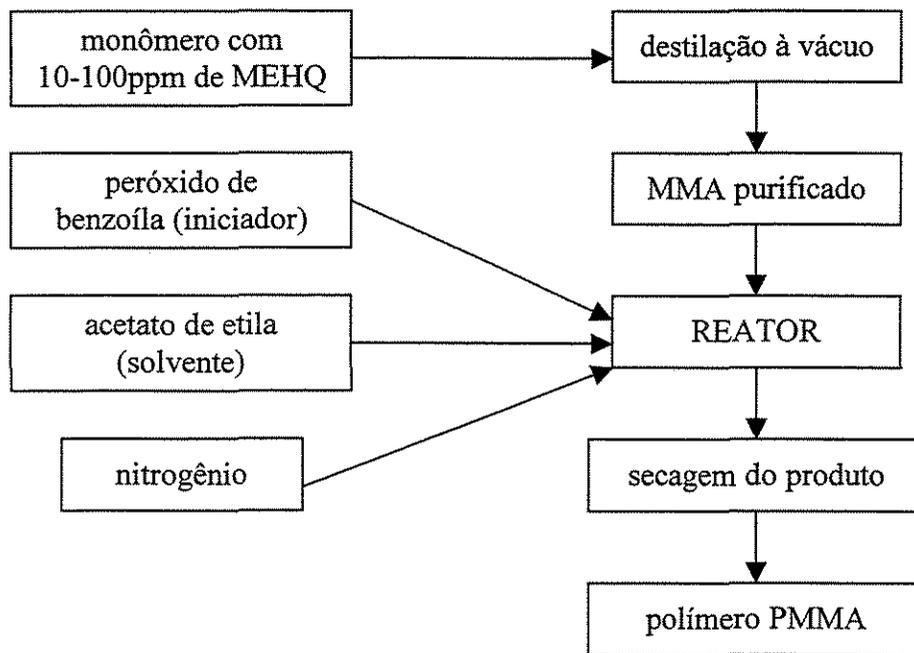


Figura 3.3: Fluxograma da reação.

A reação é mantida por seis horas, sob controle da temperatura de reação. No final a reação de polimerização é parada através de redução de temperatura do meio (desligamento dos aquecedores), injeção de oxigênio, e adição de hidroquinona.

Passa-se então à etapa de secagem, com a evaporação natural (ou com emprego de um exaustor) do solvente e posterior pesagem do polímero formado na reação. Esta massa final representa o rendimento da reação, ou seja, a quantidade de monômero que formou polímeros.

### 3.3) Configuração do sistema digital

Entre os programas usados para a configuração do sistema de comunicação de dados *Fieldbus*, o Syscon foi usado para utilização neste trabalho, mostrado na Figura 3.4.

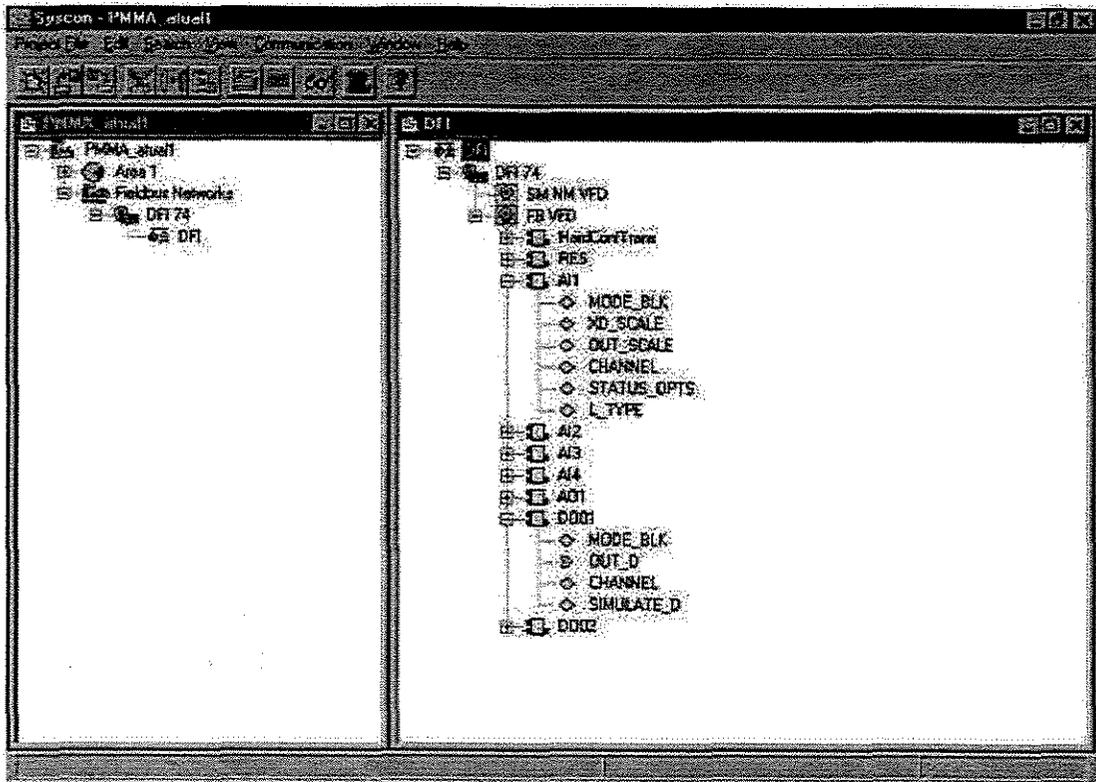


Figura 3.4: Configuração do sistema *fieldbus*.

Atualmente existem vários softwares configuradores e de supervisão para sistemas *Fieldbus Foundation*. Estes softwares via de regra utilizam o padrão de comunicação não-proprietária de interface Windows denominado OPC (*OLE for Process Control*).

A Figura 3.4 mostra os blocos funcionais de entrada e saída usados na configuração do sistema de controle. Os blocos de entrada são analógicos (*analogic input*, AI) para medição de temperatura e massa específica. Nos sinais de saída, há um bloco analógico (*analogic output*, AO) que regula a voltagem na válvula de controle e blocos de saída digitais (*digital output*, DO) referentes à bomba e ao aquecedor de 45 W. São eles:

- AI1: temperatura de entrada da água de resfriamento;
- AI2: temperatura no interior do reator;
- AI3: temperatura de saída da água de resfriamento;
- AI4: massa específica do meio reacional;
- AO1: voltagem para a válvula;
- DO01: bomba;

- DO02: aquecedor.

O padrão de comunicação OPC é uma tecnologia cliente/servidor que permite a configuração e monitoramento de equipamentos a partir de uma estação de trabalho (cliente) conectada remotamente a interfaces localizadas no servidores, via rede ethernet. As *tags* exportadas para OPC têm suas bases de dados disponibilizadas para qualquer HMI (Interface Homem-máquina), não restringindo o usuário a apenas uma HMI, permitindo-o escolher a de preferência, ou até mesmo desenvolver seu próprio software (como o software desenvolvido neste trabalho, em Visual Basic), havendo total independência do software de controle do fabricante do hardware. A Figura 3.5 mostra a interface do *software* desenvolvido no presente trabalho.

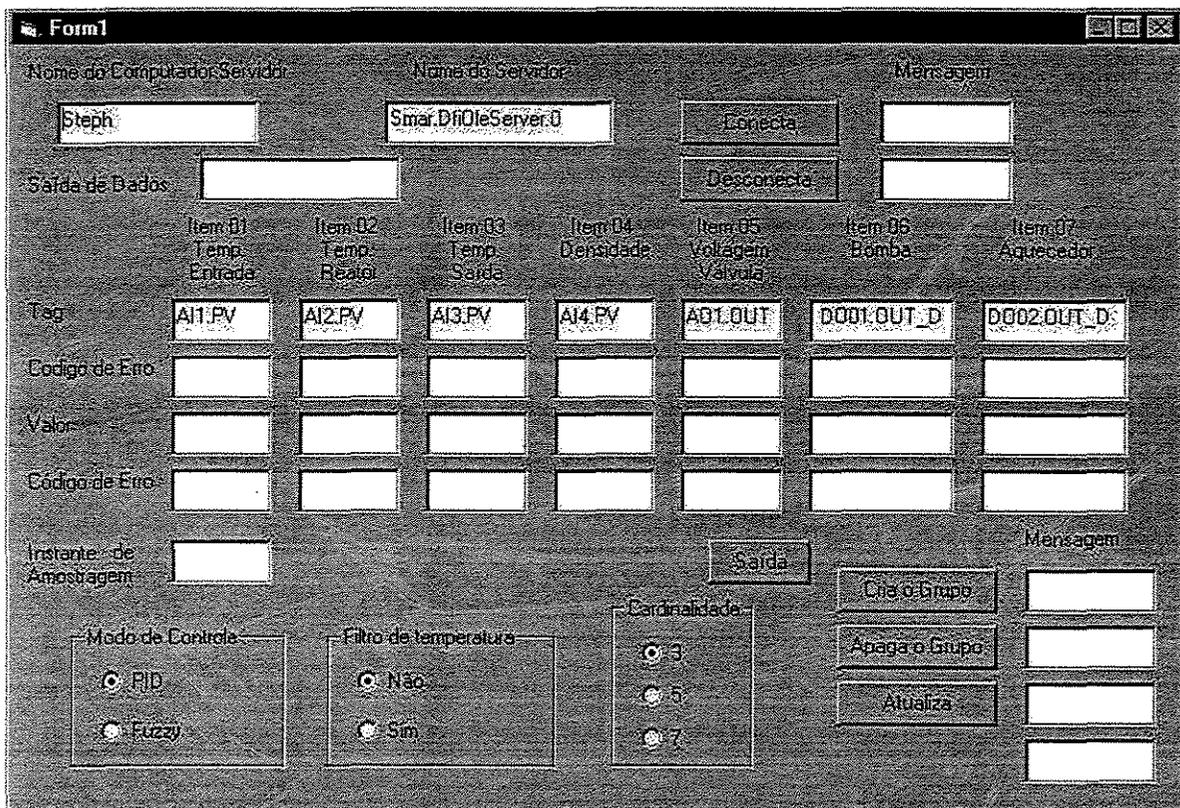


Figura 3.5: Interface do *software* desenvolvido (cliente OPC).

A tecnologia OPC é análoga à comunicação entre instrumentos eletrônicos e pneumáticos, que fariam o papel de componentes de software. Os instrumentos eletrônicos e pneumáticos têm uma interface comum - um sinal de 4-20 mA ou 3-15 psi. Assim, é permitido que os consumidores tenham liberdade de escolher qualquer fornecedor que

atenda suas necessidades. Os produtos de controle digital, no final dos anos 80 não ofereciam essa mesma padronização.

Com a facilidade de transferência de dados entre aplicativos Windows, as informações do *fieldbus* tornam-se acessíveis às redes de comunicação corporativas dentro ou fora da fábrica ou laboratório.

### **3.4) Implementação do controle convencional: PID**

Todo o aparato experimental do Laboratório de Controle e Automação de Processos do DESQ/FEQ não foi implementado apenas para o desenvolvimento do controle *fuzzy*. NUNES (2003) objetivou a implementação do controle PID (forma velocidade) para este mesmo processo. Assim, uma comparação entre os dois modos de controle pode ser estabelecida. Parte do material apresentado neste trabalho sobre o controle PID com parâmetros fixos (implementação, sintonia, análise de estabilidade, resultado experimental) foi compilado a partir de NUNES (2003).

A variável controlada do processo é a temperatura do meio reacional (que tem implicação direta sobre a conversão do monômero) e a variável manipulada é a abertura da válvula localizada na entrada da água de resfriamento na jaqueta do reator. Através da curva de calibração da válvula, obtida a partir da medida de volume em um determinado tempo para aberturas diferentes da válvula de controle, o programa envia sinais de tensão para esta válvula de controle. Os sensores de campo são os termopares e o controlador propriamente dito é um programa desenvolvido em Visual Basic, cujo código apresentado no apêndice 2. O programa foi desenvolvido e tornou o sistema independente de *softwares* comerciais.

#### **3.4.1) Sintonia PID: parâmetros fixos**

O procedimento adotado para a determinação da curva de reação do processo, visando a obtenção dos parâmetros de sintonia, foi conduzir experimentos em malha aberta, observando o comportamento da temperatura em função de distúrbios na vazão de resfriamento. Para a reação usando o controlador PID com parâmetros de sintonia fixos, foi realizado um ensaio mantendo a vazão de resfriamento em 6,3 mL/s, temperatura do reator em 55-65 °C e com as taxas de amostragem a cada 15 segundos. Levando-se a reação por 100 minutos e aplicando-se distúrbios na variável controlada, a partir de distúrbios na

variável manipulada, vazão do fluido de resfriamento, que pode ser observado nas Figuras 3.6 e 3.7.

Nelas pode-se observar a não-linearidade do sistema, mais nitidamente nos intervalos A, B, C, quando a temperatura no patamar A é submetida a uma perturbação de  $x$  para  $y$  na vazão do fluido de resfriamento, atingindo o estado estacionário B, quando submetido a uma nova perturbação de mesma intensidade, mas sentido contrário, na abertura da válvula de controle,  $y$  para  $z$ , observa-se que o novo regime permanente obtido, C, não corresponde ao mesmo A, caracterizando a não linearidade do sistema.

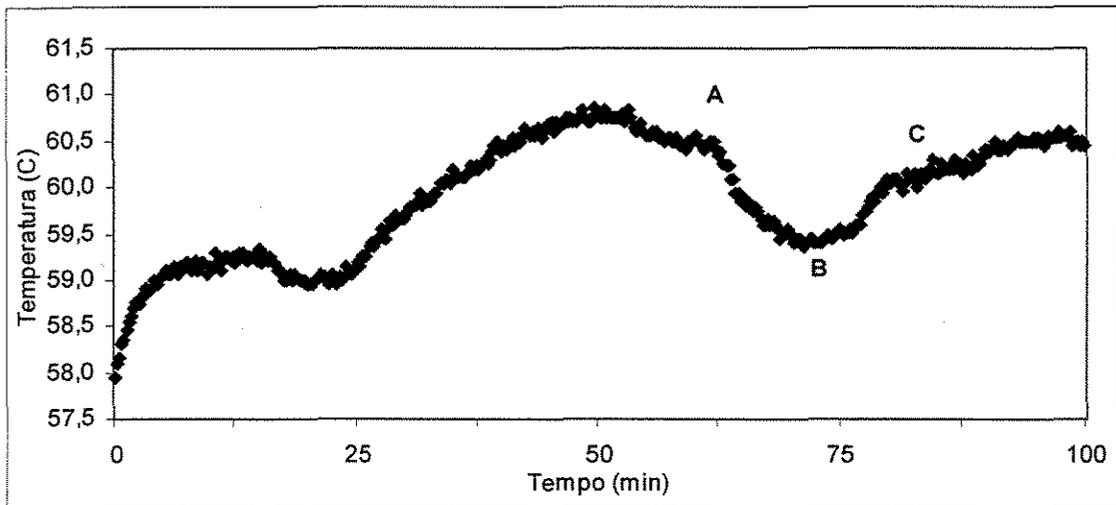


Figura 3.6: Curva de reação do processo, após perturbação de vazão.

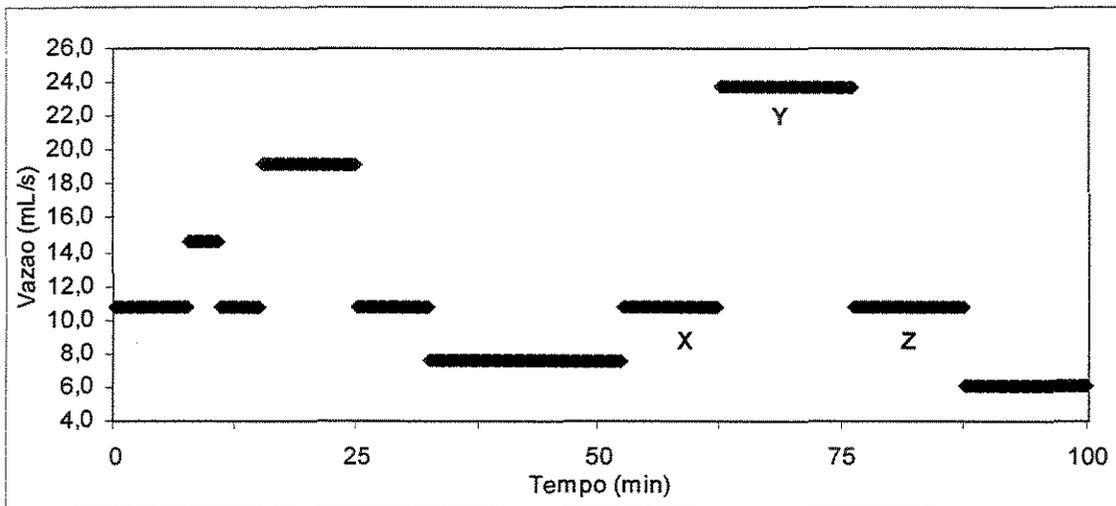


Figura 3.7: Perturbações na vazão.

Os valores para os parâmetros são encontrados obtendo-se primeiramente o valor de  $t_d$  que é o valor no eixo x no qual a reta tangente (reta C) cruza a reta horizontal inferior (reta B) mostrada na Figura 3.8. Posteriormente, encontra-se o valor no eixo x no qual a reta tangente cruza a reta horizontal superior (reta A, valor do suposto estado estacionário) e subtrai-se o valor de  $t_d$ , encontrando-se o valor de  $\tau_p$  (constante de tempo do processo). Finalmente, subtrai-se a temperatura de estabilização final da inicial e divide-se pelo valor do salto de perturbação, a fim de se obter o valor de  $K_p$  (ganho do processo), o mesmo processo deve ser adotado no caso da Figura 3.9. Invertendo-se apenas os pontos de intersecção com a reta C a tomada de  $t_d$  na reta superior (reta A) e o valor do suposto estado estacionário final na reta inferior (reta B). Os parâmetros obtidos nas Tabelas 3.1 e 3.2.

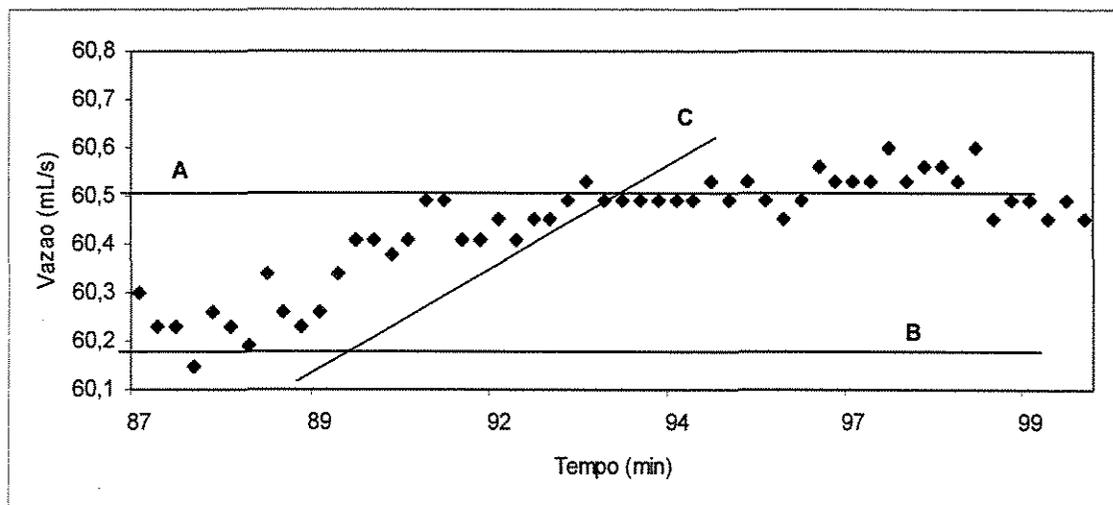


Figura 3.8: Curva de Reação do Processo após perturbação de 10,82 para 6,10 mL/s na vazão.

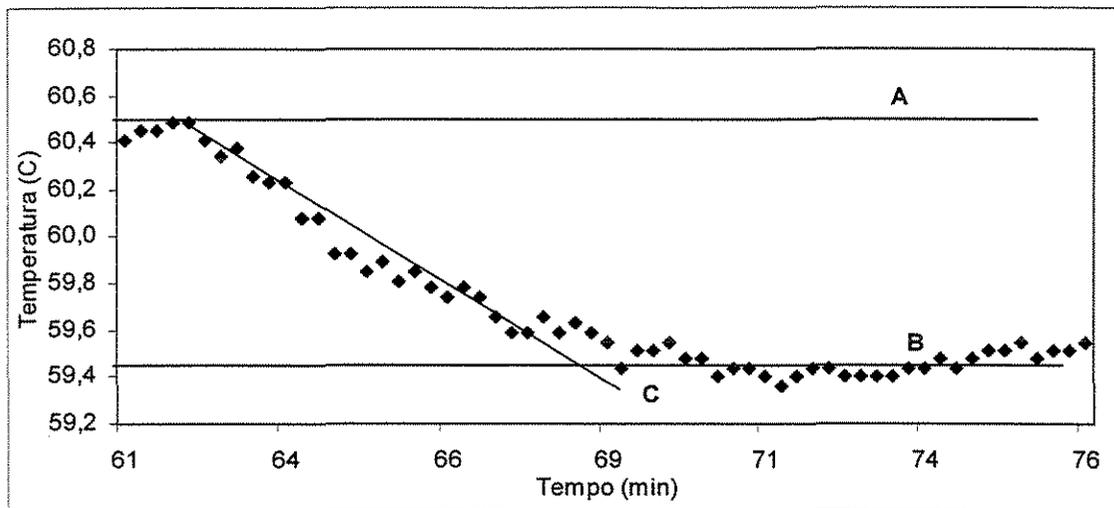


Figura 3.9: Curva de Reação do Processo após perturbação de 10,82 para 23,74 mL/s na vazão.

Tabela 3.1: Parâmetros de processo obtidos a partir da curva de reação do processo após perturbação de 10,82 para 6,10 mL/s na vazão do fluido de resfriamento.

---

<b>Parâmetros de Processo</b>
<b>Reação de Polimerização em Malha Aberta</b>
$T_{SSo} = 60,2 \text{ } ^\circ\text{C}$ , $T_{SSf} = 60,5 \text{ } ^\circ\text{C}$ , $\Delta T_{SS} = 0,3 \text{ } ^\circ\text{C}$
$t_d = 100 \text{ s}$ , $\tau_p = 138 \text{ s}$ , $K_p = -0,062 \text{ } ^\circ\text{C}\cdot\text{s/mL}$

---

Tabela 3.2: Parâmetros de processo obtidos a partir da curva de reação do processo após perturbação de 10,82 para 6,10 mL/s na vazão do fluido de resfriamento.

---

<b>Parâmetros de Processo</b>
<b>Reação de Polimerização em Malha Aberta</b>
$T_{SSo} = 60,6 \text{ } ^\circ\text{C}$ , $T_{SSf} = 59,5 \text{ } ^\circ\text{C}$ , $\Delta T_{SS} = -1,2 \text{ } ^\circ\text{C}$
$t_d = 50 \text{ s}$ , $\tau_p = 213 \text{ s}$ , $K_p = -0,086 \text{ } ^\circ\text{C}\cdot\text{s/mL}$

---

Em seguida, substituem-se os parâmetros do processo obtidos anteriormente nas equações semi-empíricas determinadas por Cohen-Coon, que se encontram a seguir:

$$K_C = \frac{1}{K_P} \left( \frac{\tau_P}{t_d} \right) \left( \frac{4}{3} + \frac{t_d}{4\tau_P} \right) \quad (3.1)$$

$$\tau_I = t_d \frac{32 + 6 \left( \frac{t_d}{\tau_P} \right)}{13 + 8 \left( \frac{t_d}{\tau_P} \right)} \quad (3.2)$$

$$\tau_D = t_d \frac{4}{13 + 8 \left( \frac{t_d}{\tau_P} \right)} \quad (3.3)$$

Os conjuntos de parâmetros do controlador calculados são apresentados na Tabela 3.3.

Tabela 3.3: Parâmetros de processo obtidos a partir da curva de reação do processo após perturbação de 10,82 para 6,10 mL/s na vazão.

<b>Conjunto de parâmetros 1</b> Para os parâmetros do processo	<b>Conjunto de parâmetros 2</b> Para os parâmetros do processo
$t_d = 100$ s	$t_d = 50$ s
$\tau_P = 138$ s	$\tau_P = 213$ s
$K_P = -0,062$ °C.s/mL	$K_P = -0,086$ °C.s/mL
Parâmetros obtidos do controlador	Parâmetros obtidos do controlador
$K_C = -53,49$ mL/°C.s	$K_C = -57,95$ mL/°C.s
$\tau_I = 126,02$ s	$\tau_I = 132,52$ s
$\tau_D = 20,22$ s	$\tau_D = 20,76$ s

Com os valores encontrados dos parâmetros  $K_C$ ,  $\tau_I$  e  $\tau_D$ , substitui-se na equação do controlador PID na forma velocidade (Equação 2.4) presente no código do programa feito em Visual Basic. Desta forma, conclui-se uma primeira estimativa para estes valores, restando agora verificar a análise de estabilidade para estes sistemas, o comportamento do controlador na prática e se necessário fazer o ajuste fino destes parâmetros.

### 3.4.2) Sintonia PID: controle adaptativo

Na tentativa de se melhorar o desempenho do controlador PID foi proposta uma forma adaptativa, onde o parâmetro  $K_c$  variasse ao longo da polimerização. Para isso, foram feitos ensaios em malha aberta observando-se o comportamento da temperatura frente à manipulação da vazão de água de resfriamento. A Figura 3.10 mostra os resultados de um experimento feito no início da reação. Na Tabela 3.4 constam os resultados produzidos.

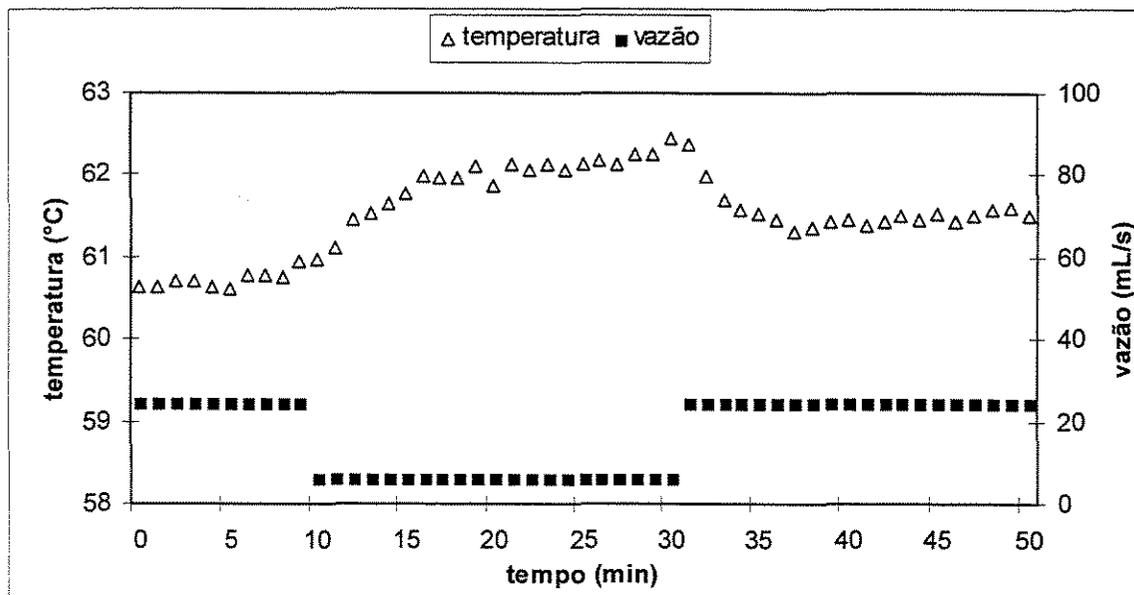


Figura 3.10: Perturbação no início da reação em malha aberta.

Tabela 3.4: Comportamento no início da reação

Fechamento da válvula	Abertura da válvula
$t_d = 58,4$ s	$t_d = 57,17$ s
$\tau_p = 271,2$ s	$\tau_p = 273,7$ s
$K_p = -0,078$ °C.s/mL	$K_p = -0,039$ °C.s/mL
Parâmetros obtidos do controlador	Parâmetros obtidos do controlador
$K_C = -83,05$ mL/°C.s	$K_C = -171,25$ mL/°C.s
$\tau_I = 132,06$ s	$\tau_I = 129,44$ s
$\tau_D = 20,44$ s	$\tau_D = 20,01$ s

Para estimar a variação do parâmetro de controle, o mesmo ensaio foi feito no final da polimerização, especificamente, depois de 4,5 horas. A Figura 3.11 e a Tabela 3.5 mostram os resultados.

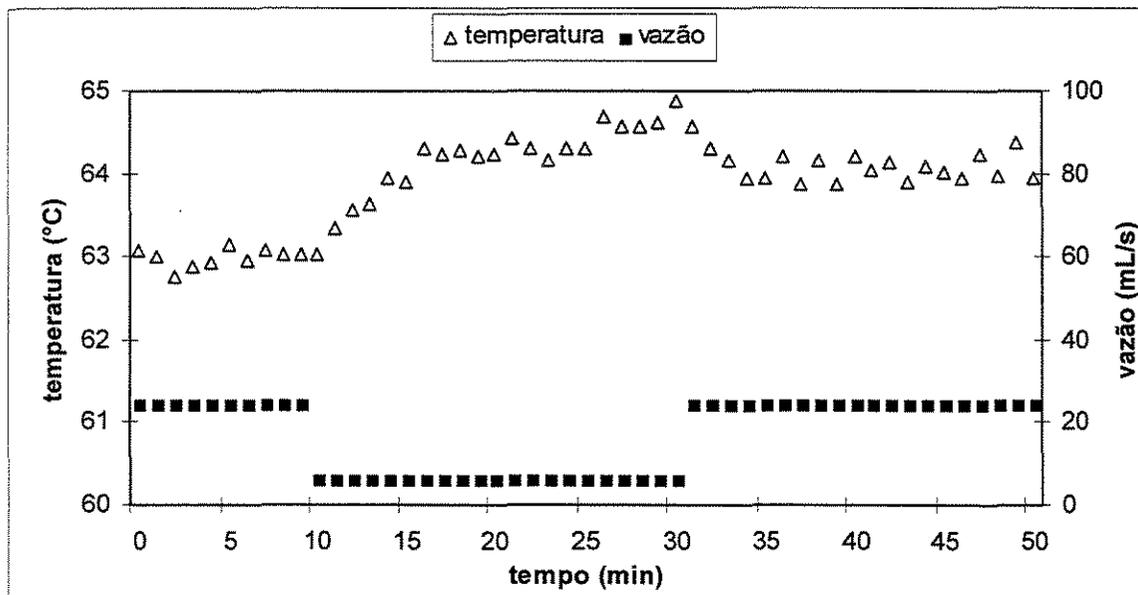


Figura 3.11: Perturbação para o final da reação.

Tabela 3.5: Comportamento no final da reação

Fechamento da válvula	Abertura da válvula
$t_d = 53,3 \text{ s}$	$t_d = 106,7 \text{ s}$
$\tau_p = 386,7 \text{ s}$	$\tau_p = 93,3 \text{ s}$
$K_p = -0,089 \text{ } ^\circ\text{C.s/mL}$	$K_p = -0,033 \text{ } ^\circ\text{C.s/mL}$
Parâmetros obtidos do controlador	Parâmetros obtidos do controlador
$K_C = -111,95 \text{ mL/}^\circ\text{C.s}$	$K_C = -42,60 \text{ mL/}^\circ\text{C.s}$
$\tau_I = 124,07 \text{ s}$	$\tau_I = 187,21 \text{ s}$
$\tau_D = 18,91 \text{ s}$	$\tau_D = 32,12 \text{ s}$

### 3.4.3) Análise de Estabilidade em Malha Fechada

A análise de estabilidade do processo com controlador PID é realizada a partir do estudo do lugar geométrico das raízes (*root locus*) da equação característica (3.4) do

processo em questão. Para isso apresenta-se o tratamento matemático para a obtenção da equação característica no domínio Z.

$$1 + G_C(z).G_P.G_H.G_M.G_f(z) = 0 \quad (3.4)$$

A função de transferência do processo no domínio de Laplace é dada por:

$$G_P(s) = \frac{K_P e^{-tds}}{\tau_P.s + 1} \quad (3.5)$$

A função de transferência *data hold* de ordem zero, relativa ao elemento de *hardware* que emantém o valor da variável manipulada durante o intervalo de amostragem, até sua atualização é descrita como segue:

$$G_H(s) = \frac{1 - e^{-s.\Delta t}}{s} \quad (3.6)$$

Considerando que o sensor de temperatura e a válvula não interferem na dinâmica do sistema ( $G_M(s) = G_f(s) = 1$ ) obtém-se a função de transferência de pulso:

$$G_P.G_H(s) = \frac{K_P e^{-td.s}}{s(\tau_P.s + 1)} - \frac{K_P e^{-(td+\Delta t).s}}{s(\tau_P.s + 1)} \quad (3.7)$$

Evidenciando a constante do processo e dividindo a numerador e o denominador da Equação 3.7 por  $\tau_P$ , tem-se que:

$$G_P.G_H(s) = K_P \left[ \frac{e^{-td.s} - e^{-(td+\Delta t).s}}{s.\tau_P \left( s + \frac{1}{\tau_P} \right)} \right] \quad (3.8)$$

Utilizando-se a definição da transformada Z, dada pela equação 3.9, tem-se as seguintes correspondências:  $e^{-td.s} = z^{-1}$  e  $e^{-(td+\Delta t).s} = z^{-2}$ .

$$Z\{e^{-td.s}\} = z^{-1} \quad (3.9)$$

Considerando  $t_d \approx \Delta t$  (60 s, média ajustada) e passando a função de transferência da Equação 3.8 do domínio de Laplace para o domínio de Z, obtém-se a Equação 3.10.

$$G_p.G_H(s) = K_p.z^{-1}(1-z^{-1}) \frac{\left(1 - e^{-\frac{\Delta t}{\tau_p}}\right).z^{-1}}{\left(1-z^{-1}\right)\left(1 - e^{-\frac{\Delta t}{\tau_p}}.z^{-1}\right)} \quad (3.10)$$

Pode ser simplificada para obter-se a Equação 3.11.

$$G_p.G_H(s) = K_p \frac{\left(1 - e^{-\frac{\Delta t}{\tau_p}}\right).z^{-2}}{\left(1 - e^{-\frac{\Delta t}{\tau_p}}.z^{-1}\right)} \quad (3.11)$$

Utilizando-se o controlador PID na forma velocidade (Equação 2.4), tem-se a seguinte função de transferência para o mesmo:

$$G_C(z) = K_C \left\{ \left(1 + \frac{\Delta t}{\tau_I} + \frac{t_d}{\Delta t}\right) - \left(1 + \frac{2t_d}{\Delta t}\right)z^{-1} + \frac{\tau_D}{\Delta t}z^{-2} \right\} \quad (3.12)$$

Desta forma, obtém-se a equação característica da malha fechada, cujas raízes (z) devem encontrar-se dentro de um círculo unitário no plano complexo (Root Locus no domínio Z) para o sistema ser estável em malha fechada:

$$1 + G_C(z).G_p.G_H(z) = 0 \quad (3.13)$$

Substituindo-se as funções de transferência de pulso e do controlador encontra-se a Equação 3.14 que de posse dos parâmetros do processo e do controlador (Tabela 3.3) pode

ser aplicada no software Matlab 6.3 para determinar-se as regiões de estabilidade para os parâmetros encontrados.

$$G_C(z).G_P.G_H(z) = \frac{K_C K_P \left(1 - e^{-\frac{\Delta t}{T}}\right)}{1 - e^{-\frac{\Delta t}{T}} z^{-1}} \left[ \left(1 + \frac{\Delta t}{\tau_I} + \frac{t_d}{\Delta t}\right) z^{-2} - \left(1 + \frac{2t_d}{\Delta t}\right) z^{-3} + \frac{\tau_D}{\Delta t} z^{-4} \right] \quad (3.14)$$

Como o software Matlab 6.3 trabalha com polinômios em  $z$  e não  $z^{-1}$  deve-se transformar esta função de transferência em um polinômio em  $z$ , assim, multiplica-se o numerador e o denominador por  $z$  elevado ao valor absoluto do mais baixo exponencial da equação 3.14, no caso,  $z^4$ :

$$G_C(z).G_P.G_H(z) = \frac{K_C K_P \left(1 - e^{-\frac{\Delta t}{T}}\right)}{z^4 - e^{-\frac{\Delta t}{T}} z^3} \left[ \left(1 + \frac{\Delta t}{\tau_I} + \frac{t_d}{\Delta t}\right) z^2 - \left(1 + \frac{2t_d}{\Delta t}\right) z + \frac{\tau_D}{\Delta t} \right] \quad (3.15)$$

### **Análise de estabilidade para parâmetros fixos**

Implementa-se então a Equação 3.15 através do programa em Matlab (Código no Apêndice 3), com o objetivo de se determinar as faixas de  $K_C$  para as quais o controlador é estável em malha fechada. Desta forma, foram gerados os gráficos das Figuras 3.12 e 3.13 com os conjuntos de parâmetros da Tabela 3.3.

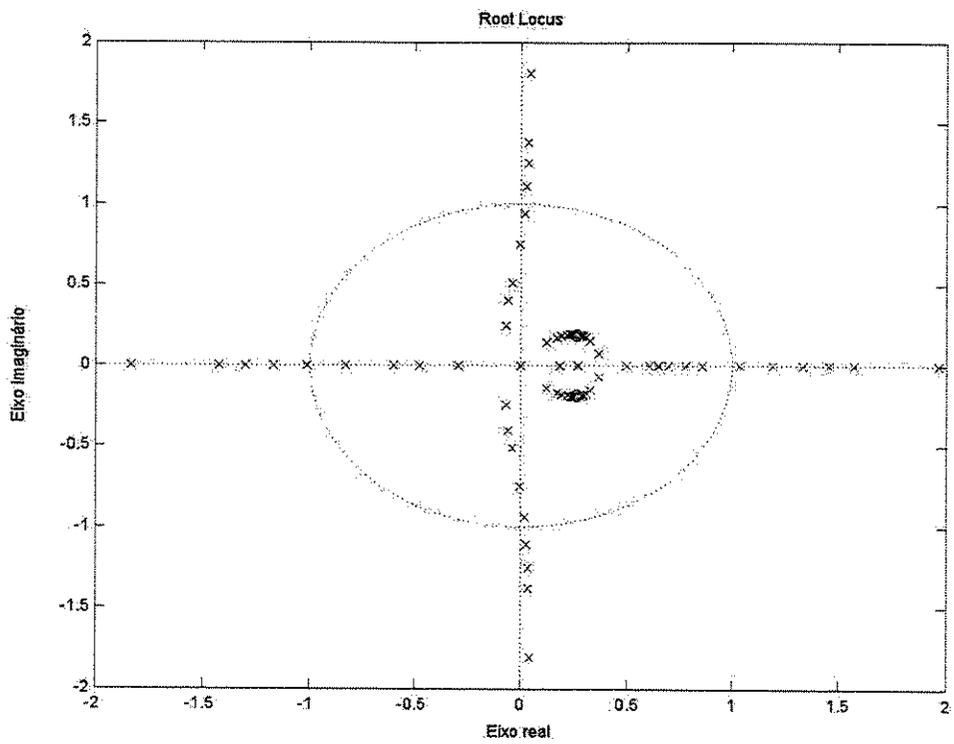


Figura 3.12: Gráfico *Root Locus* para grupo de parâmetros 01 (Tabela 3.3)

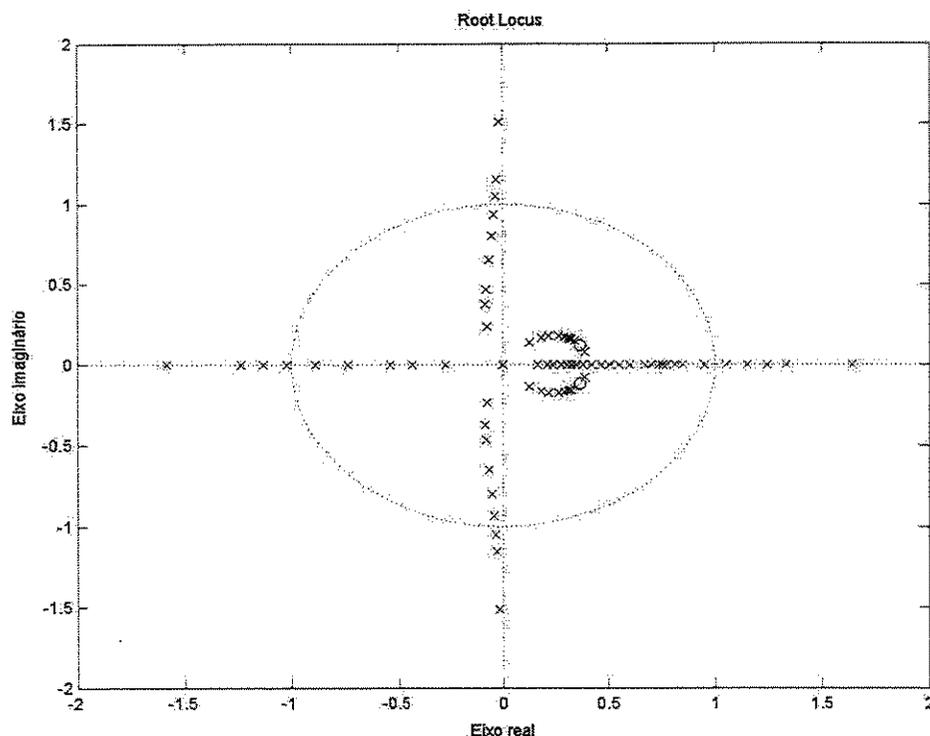


Figura 3.13: Gráfico *Root Locus* para grupo de parâmetros 02 (Tabela 3.3)

Para ambos os conjuntos de parâmetros da Tabela 3.3 o  $K_c$  limite para estabilidade do sistema esta em torno de  $-15 \text{ mL}^0\text{C.s}$ , valor absoluto máximo de  $K_c$  para que as raízes encontrem-se dentro do círculo unitário.

Uma vez que o ganho do controlador estimado por Cohen-Coon (Tabela 3.3) indica um valor fora do círculo unitário de estabilidade no gráfico do lugar das raízes (*root locus*), evidencia-se a necessidade deste tipo análise. O ganho do controlador tão alto caracterizaria uma atuação do tipo *on-off*, com baixas chances de sucesso no controle da temperatura do reator e ainda a possibilidade desta não atingir novo estado estacionário, produzindo instabilidade.

Na Tabela 3.6 é possível observar o “overshoot” da resposta do processo sob controle, obtido no programa feito no Matlab de  $K_c = 1$  até  $K_c$  do sistema marginalmente estável.

Tabela 3.6: *Overshoot* para Diferentes Valores de  $K_C$

<b>Determinação do <i>overshoot</i> do controlador PID</b>			
Conjunto de parâmetros 01		Conjunto de parâmetros 02	
$K_C = -1 \text{ mL}^0\text{C.s}$	Overshoot = 10%	$K_C = -1 \text{ mL}^0\text{C.s}$	Overshoot = 10%
$K_C = -5 \text{ mL}^0\text{C.s}$	Overshoot = 30%	$K_C = -5 \text{ mL}^0\text{C.s}$	Overshoot = 26%
$K_C = -14 \text{ mL}^0\text{C.s}$	Overshoot = 100%	$K_C = -15 \text{ mL}^0\text{C.s}$	Overshoot = 100%

Um overshoot de 10% foi encontrado para  $K_C$  em torno de  $K_C = -1 \text{ mL}^0\text{C.s}$ , considerado um valor intermediário aceitável (baixa intensidade de oscilação inicial) para início dos testes em malha fechada.

Embora isto pareça simples à primeira vista, na prática sabe-se que fazer uma sintonia de um controlador PID não é tarefa simples, pois se têm três parâmetros que podem variar independentemente e uma pequena mudança em qualquer um deles altera significativamente o comportamento do controlador. Mesmo que o ajuste semi-empírico de Cohen-Coon não seja o mais exato, ele é sem dúvida o mais utilizado para este fim devido à sua facilidade no procedimento e nos cálculos, em relação aos outros métodos que são encontrados na literatura clássica de controle, como o Ziegler-Nichols, e mais eficaz que um método de tentativa-e-erro.

As não-linearidades e a transitoriedade do processo em questão agravam o problema de sintonia do controlador PID, resultando em parâmetros do controlador que não funcionam adequadamente para o tempo todo da batelada.

#### **Análise de estabilidade para controle adaptativo**

Assim como no caso do controlador PID com parâmetros fixos, também é necessário saber os valores limites de  $K_C$  para aplicar o controle adaptativo. Por isso, também foram feitas análises de estabilidade, visando determinar os valores máximos de  $K_C$  para determinado *overshoot*.

Em relação ao início da reação, as Figuras 3.14 e 3.15 mostram a análise de *Root Locus* para o fechamento e a abertura da válvula, respectivamente.

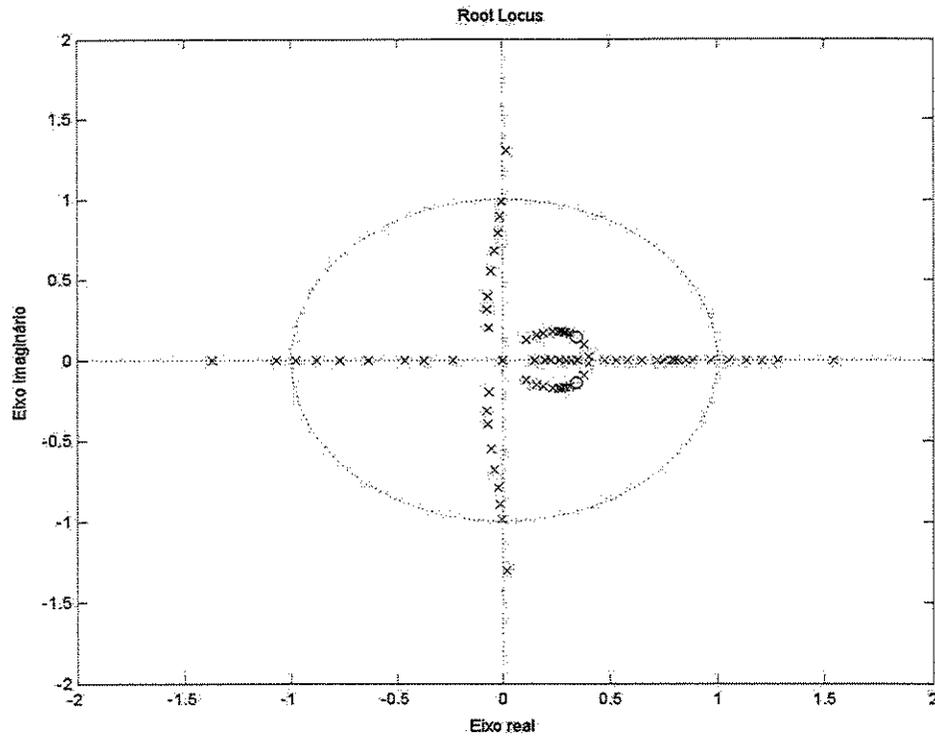


Figura 3.14: Gráfico *Root Locus* para o fechamento da válvula no período inicial da reação (Tabela 3.4).

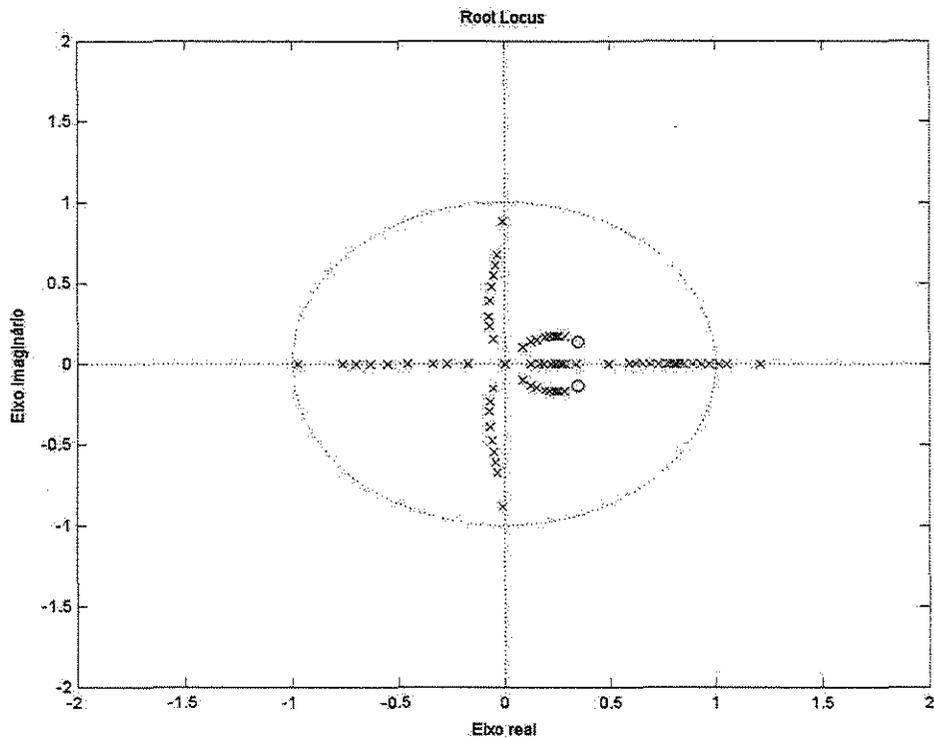


Figura 3.15: Gráfico *Root Locus* para a abertura da válvula no período inicial da reação (Tabela 3.4).

Como resultado, a Tabela 3.7 mostra os valores de  $K_C$  em função do *overshoot* resultante.

Tabela 3.7: *Overshoot* para Diferentes Valores de  $K_C$  (início da reação)

<b>Determinação do <i>overshoot</i> do controlador PID adaptativo</b>			
Fechamento da válvula		Abertura da válvula	
Overshoot (%)	$K_C$ (mL <sup>0</sup> C.s)	Overshoot (%)	$K_C$ (mL <sup>0</sup> C.s)
10	-1,6	10	-3,7
30	-9,0	30	-18,7
100	-30,9	100	-62,7

Considerando o período final da polimerização, as Figuras 3.16 e 3.17 mostram a análise de *Root Locus* para o fechamento e a abertura da válvula, respectivamente.

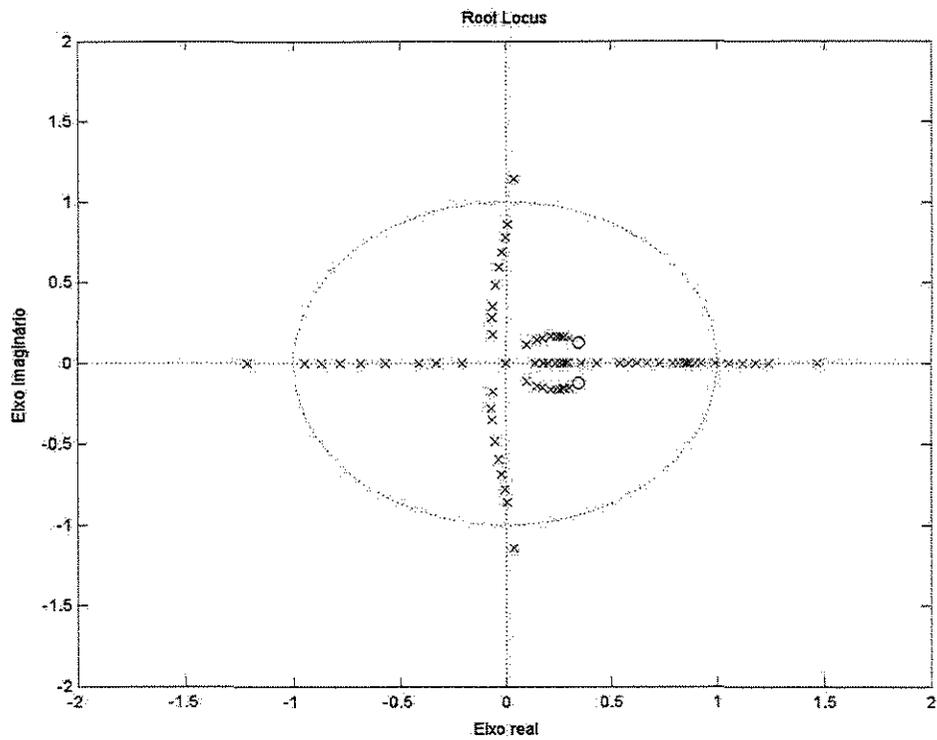


Figura 3.16: Gráfico *Root Locus* para o fechamento da válvula no período final da reação (Tabela 3.5).

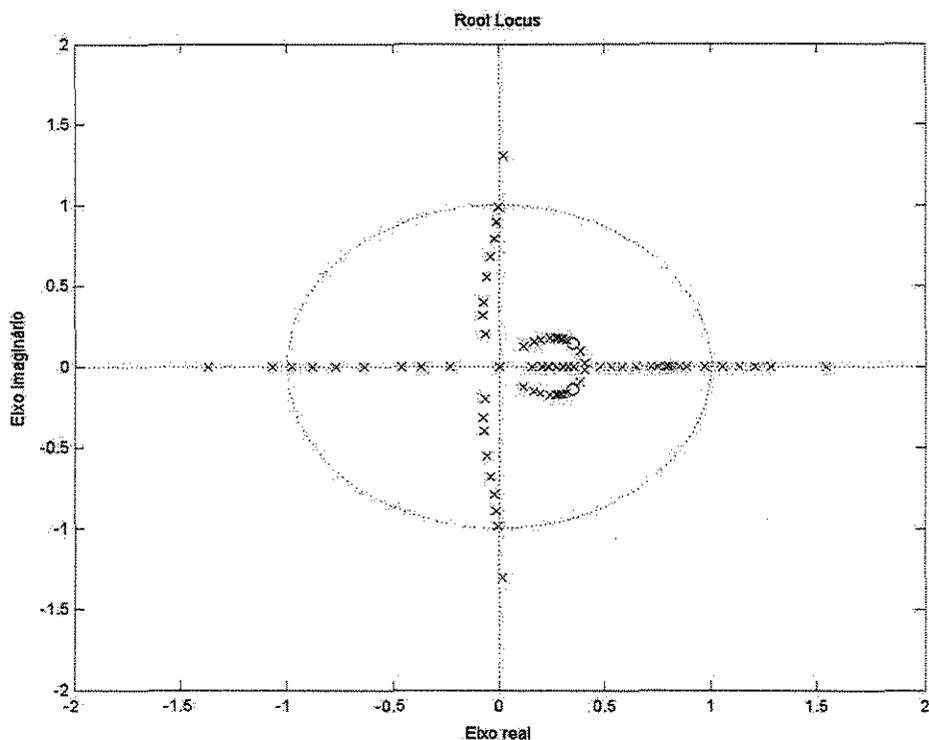


Figura 3.17: Gráfico *Root Locus* para a abertura da válvula no período final da reação (Tabela 3.15).

Como resultado, a Tabela 3.8 mostra os valores de  $K_c$  em função do *overshoot* resultante.

Tabela 3.8: *Overshoot* para Diferentes Valores de  $K_c$  (final da reação)

<b>Determinação do <i>overshoot</i> do controlador PID adaptativo</b>			
Fechamento da válvula		Abertura da válvula	
Overshoot (%)	$K_c$ (mL <sup>0</sup> C.s)	Overshoot (%)	$K_c$ (mL <sup>0</sup> C.s)
10	-2,4	10	-0,7
30	-12,0	30	-4,8
100	-39,4	100	-21,5

### 3.5) Controle *fuzzy*

O controlador *fuzzy* proposto é baseado na estrutura de LI (Figura 2.13), com regras de Mamdani e inferência do tipo *max-min*. As variáveis linguísticas envolvidas são

erro, primeira variação do erro ( $\Delta e$ ) e variação de vazão de água de resfriamento ( $\Delta u$ ). Em princípio, é utilizada cardinalidade, representada pela variável  $n$ , igual a 3 com os valores lingüísticos N (negativo), Z (zero) e P (positivo). Contudo, a cardinalidade pode ser aumentada para 5 ou 7 com a adição dos valores NM (negativo médio), NS (negativo pequeno), PM (positivo médio) e PS (positivo pequeno).

Não existe um método geral para obter as funções de pertinência, usadas nas interfaces entre os domínios. Medidas feitas em malha aberta serviram para “desenhar” as funções de pertinência das variáveis erro e  $\Delta e$ . A faixa de vazão de água resultaram nas funções de pertinência para a saída do controlador.

### 3.5.1) Sintonia *fuzzy*

O erro e sua primeira variação são as variáveis de entrada no controlador. Para determinar as funções de pertinência para os valores lingüísticos positivos e negativos, perturbações em degrau foram impostas à variável manipulada e o comportamento da temperatura foi analisado. Por questão de simplicidade são empregadas funções de pertinência triangulares. As Figuras 3.18 e 3.19 foram obtidas a partir de dois ensaios distintos.

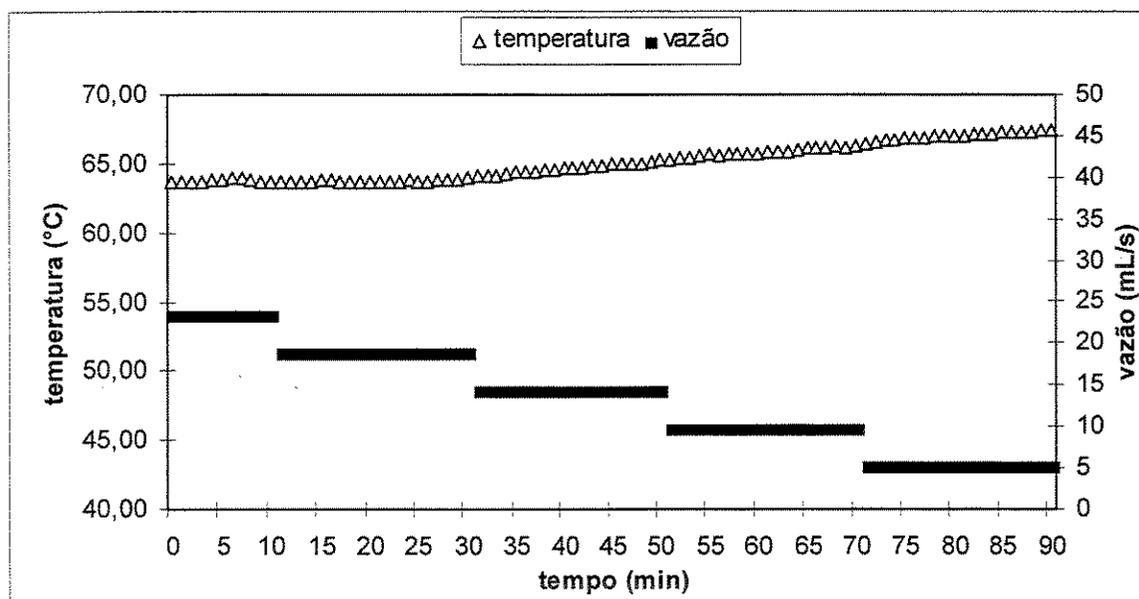


Figura 3.18: Ensaio em malha aberta.

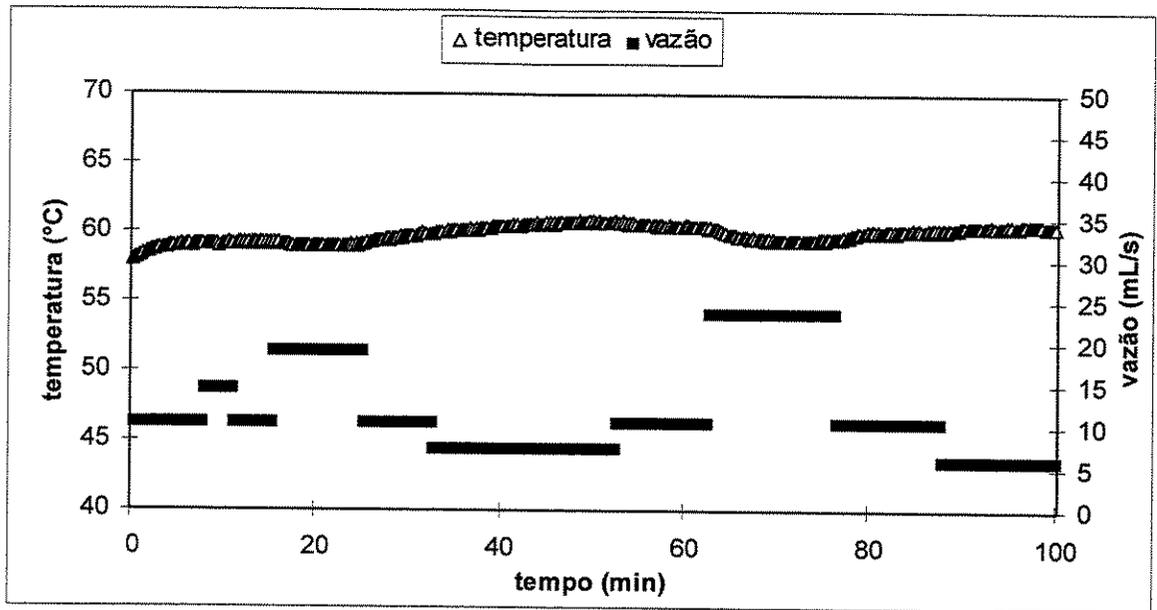


Figura 3.19: Ensaio em malha aberta.

Na Figura 3.18,  $64^{\circ}\text{C}$  é a temperatura usada como referência porque corresponde a um fluxo de  $14\text{ mL/s}$ , que é o ponto médio na faixa de vazão de água de resfriamento. Os valores para as funções de pertinência são obtidos da seguinte forma:

- Erro ( $e$ ):

Positivo (Figura 3.18):  $e = T_{\text{ref}} - T_{\text{reator}} (t = 0 \text{ min}) = 64 - 63,6 = 0,4^{\circ}\text{C}$

Negativo (Figura 3.18):  $e = T_{\text{ref}} - T_{\text{reator}} (t = 90 \text{ min}) = 64 - 67,43 = -3,4^{\circ}\text{C}$

- Variação do erro ( $\Delta e$ ):

$$\Delta e = e_2 - e_1 = (T_{\text{ref}} - T_{\text{reator2}}) - (T_{\text{ref}} - T_{\text{reator1}}) = T_{\text{reator1}} - T_{\text{reator2}}$$

Positivo (Figura 3.18):  $\Delta e = T_{\text{reator}} (t = 0 \text{ min}) - T_{\text{reator}} (t = 90 \text{ min}) = 63,64 - 67,43 = -3,79^{\circ}\text{C}$

Negativo (Figura 3.19):  $\Delta e = T_{\text{reator}} (t = 52,5 \text{ min}) - T_{\text{reator}} (t = 76 \text{ min}) = 60,79 - 59,55 = 1,24^{\circ}\text{C}$

Os resultados são colocados na Figura 3.20. A Figura 3.19 também possibilita estimar o período de amostragem. No instante 62 minutos, a variável manipulada é perturbada, praticamente, um degrau completo e o resultado é uma constante de tempo de 5 minutos.

De acordo com uma regra prática, pode-se usar 20% deste valor como período de amostragem (60 s).

Para a saída do controlador (variação no fluxo de água), o intervalo inicial foi limitado em [4; -9] (mL/s) porque a variação máxima é de aproximadamente 18 mL/s. Contudo, a faixa positiva da função de pertinência Z do erro é menor do que o lado negativo, o sistema tende a receber um erro completamente negativo (função de pertinência unitária) com maior frequência. Para corrigir esta distorção, a função de pertinência P tem um alcance menor (Figura 3.21).

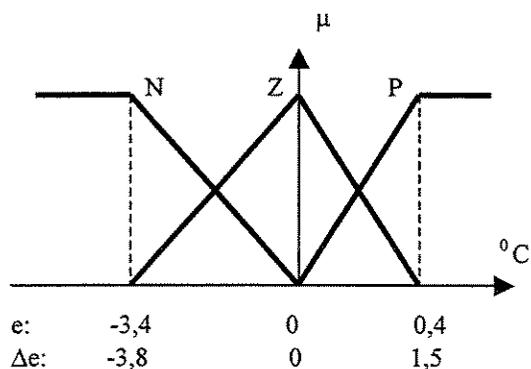


Figura 3.20: Funções de pertinência para erro e  $\Delta$ erro.

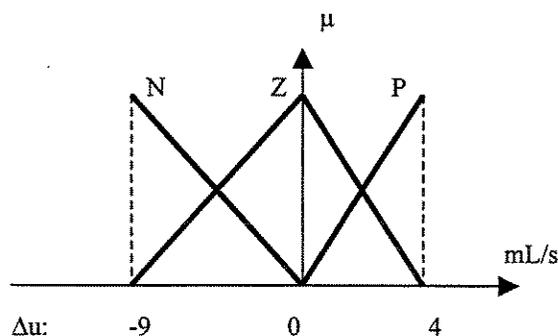


Figura 3.21: Funções de pertinência para  $\Delta u$ .

Como será observado mais adiante, no capítulo de resultados, haverá necessidade do uso de cardinalidades maiores que três. Fazendo uma simples interpolação linear nas Figuras 3.20 e 3.21, tem-se as funções de pertinência com cardinalidades 5 e 7 para as variáveis de entrada e de saída (Figuras 3.22 a 3.25). Na defuzificação, através do centro de massa da figura formada, observa-se que o valor escalar resultante é sempre menor os valores nas extremidades, do intervalo definido pelo universo de discurso. Este fato não parece ser um problema, uma vez que este valor extremo na resposta pode ser ajustado facilmente na sintonia do controlador.

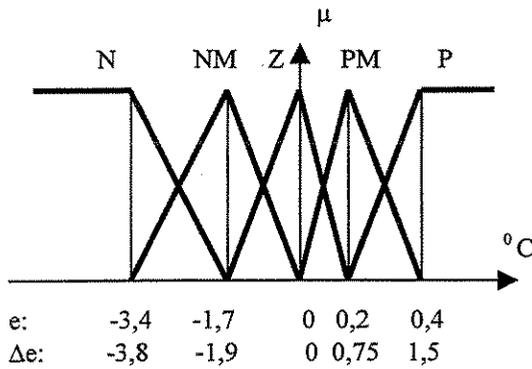


Figura 3.22: Funções de pertinência para erro e  $\Delta$ erro (card. 5).

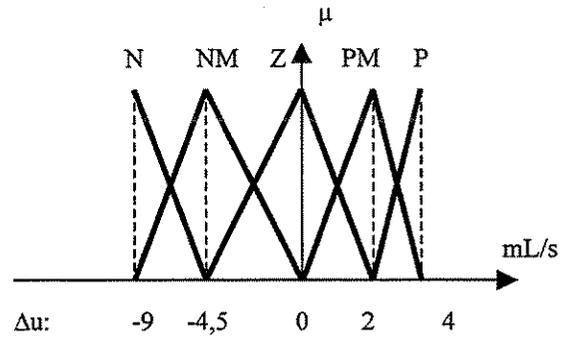


Figura 3.23: Funções de pertinência para  $\Delta u$  (card. 5).

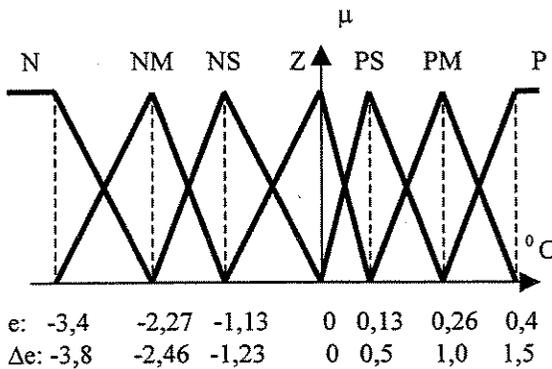


Figura 3.24: Funções de pertinência para erro e  $\Delta$ erro (card 7).

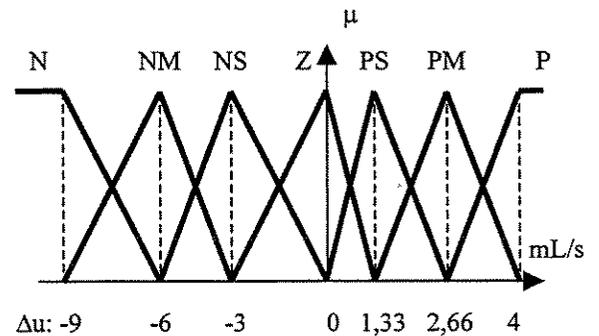


Figura 3.25: Funções de pertinência para  $\Delta u$  (card. 7).

O procedimento acima mostrou como o conhecimento a respeito do processo foi empregado no projeto do controle *fuzzy*. Outro modo de incluir o conhecimento de um especialista é aumentar o número de regras mostrado nas Tabelas 3.9 a 3.10.

#### Parâmetros de sintonia.

Os quatro parâmetros  $K_e$ ,  $K_d$ ,  $K_1$ ,  $K_2$  mostrados na Figura 2.13 são usados para fins de sintonia. Na entrada, os valores de  $K_d$  e  $K_e$  são igualados à unidade por dois motivos: simplicidade e evitar distorções entre os valores medidos de erro e  $\Delta$ erro e os valores representados nas funções de pertinência (Figuras 3.20, 3.22 e 3.23). O parâmetro  $K_2$ , escala do componente posição, também é igualada à unidade para evitar valores

desbalanceados, tendenciosos. Por outro lado,  $K_1$  pode ser variada com mais liberdade para a sintonia porque ela interfere na componente velocidade e não pode desbalancear a saída do controlador. Um dado valor atuaria igualmente no fechamento ou na abertura da válvula.

Para avaliar o comportamento do sistema de controle foram realizados ensaios com valores candidatos de  $K_1$ , uma vez que não se tem nenhuma informação sobre a ordem de grandeza do parâmetro correto. Utilizando a mesma mistura usada na reação, o material foi colocado no reator e submetido a um aquecimento com 45 W de modo que a temperatura ficasse abaixo do *set-point*, fixado em 40 °C. Esperou-se a temperatura estabilizar, anotando-se 4 pontos, um a cada minuto. O sistema de aquisição e controle foi ligado. Neste instante, é percebido uma redução abrupta da vazão, numa tentativa de elevar a temperatura. Depois de três minutos, os aquecedores são trocados, o de 45 W é substituído pelo mais potente (75 W). A medida que a temperatura aumenta, o perfil da curva de vazão é analisado para escolher o valor mais adequado. A primeira tentativa é feita com o valor unitário para  $K_1$  e o resultado é mostrado na Figura 3.26.

A Figura 3.26 mostra que o controlador *fuzzy* reagiu de forma muito abrupta para o aumento da temperatura. A ação foi praticamente do tipo liga/desliga, indo da vazão máxima à vazão mínima, sem pontos intermediários. Esse regime pode trazer instabilidade ao sistema reacional do mesmo tipo que um valor exagerado de  $K_c$  para o controlador PID. Para diminuir o efeito, é preciso reduzir o valor de  $K_1$  para atenuar o aumento de vazão. Assim, o valor de -0,5 é testado, conforme a Figura 3.27.

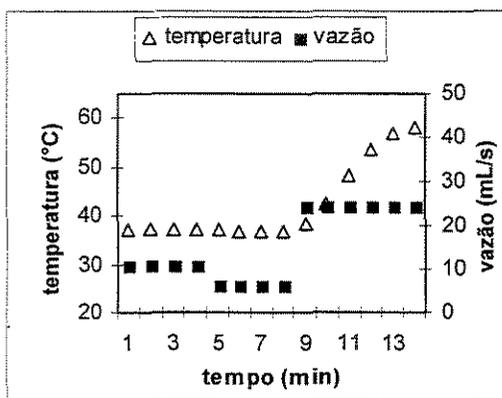


Figura 3.26: Teste de sintonia ( $K_1 = -1,0$ )

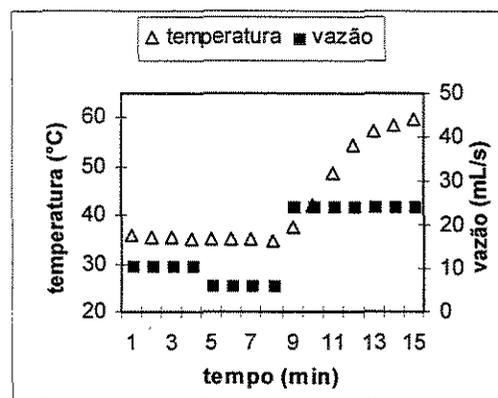


Figura 3.27: Teste de sintonia ( $K_1 = -0,5$ )

Novamente houve o problema de resposta escalar do controlador, indo rapidamente da vazão mínima para a vazão máxima. Reduzir-se o valor de  $K_1$  para -0,2 (Figura 3.28).

Apesar das reduções do parâmetro de sintonia, o problema ainda persistiu e foi realizado mais um teste para procurar um valor adequado. É feita uma nova tentativa, com um valor de -0,02.

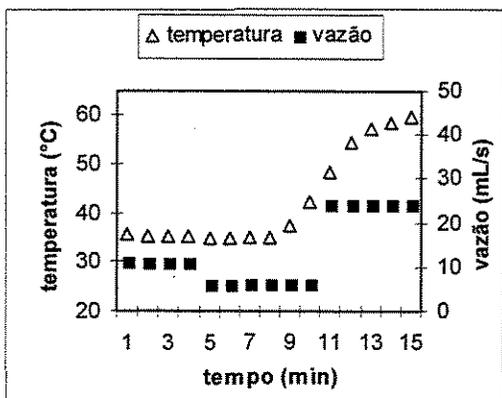


Figura 3.28: Teste de sintonia ( $K_1 = -0,2$ )

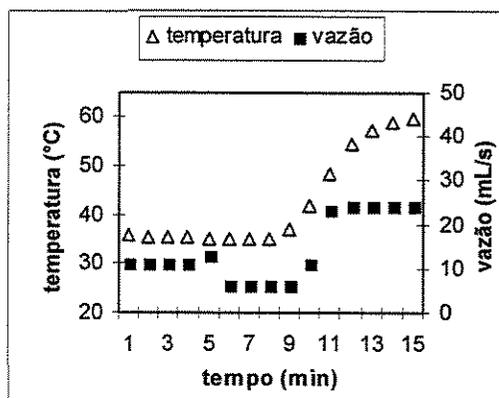


Figura 3.29: Teste de sintonia ( $K_1 = -0,02$ )

Finalmente, foi determinado um valor razoável para  $K_1$ . Submetido ao mesmo aquecimento que os demais ensaios, a vazão apresentou uma variação mais suave, com dois pontos intermediários entre a transição da vazão mínima para a máxima. Com isso, é esperado que o controlador *fuzzy* responda mais suavemente em relação ao erro. Usando um dos valores anteriores, a variação de vazão poderia ser exagerada para um erro pequeno, começando um ciclo que poderia levar o sistema a problemas de oscilações.

### 3.5.2) Implementação *fuzzy*

No presente trabalho, o conjunto de regras usado para cada componente é mostrado nas Tabelas 3.9 e 3.10.

Tabela 3.9: Conjunto de regras para *fuzzy-PI* (card. 3)

		erro		
		N	Z	P
$\Delta$ erro	N	N	N	Z
	Z	N	Z	P
	P	Z	P	P

Tabela 3.10: Conjunto de regras para *fuzzy-PD* (card. 3)

		erro		
		N	Z	P
$\Delta$ erro	N	P	P	N
	Z	P	Z	N
	P	P	N	N

Para se utilizar cardinalidade 5 e 7, mostradas nas Figuras 3.22 até 3.25, é necessário montar novas Tabelas para determinar quais ações de controle devem ser selecionadas nestes casos. O resultado é mostrado nas Tabelas 3.11 até 3.14.

Tabela 3.11: Conjunto de regras para *fuzzy-PI* (card. 5).

		erro				
		N	NM	Z	PM	P
$\Delta$ erro	N	N	N	N	NM	Z
	NM	N	N	NM	Z	PM
	Z	N	NM	Z	PM	P
	PM	NM	Z	PM	P	P
	P	Z	PM	P	P	P

Tabela 3.12: Conjunto de regras para *fuzzy-PD* (card. 5).

		erro				
		N	NM	Z	PM	P
$\Delta$ erro	N	P	P	PM	NM	NM
	NM	P	PM	PM	NM	N
	Z	P	PM	Z	NM	N
	PM	P	PM	NM	NM	N
	P	PM	PM	NM	N	N

Tabela 3.13: Conjunto de regras para *fuzzy-PI* (card. 7).

		erro						
		N	NM	NS	Z	PS	PM	P
$\Delta$ erro	N	N	N	N	N	NM	NS	Z
	NM	N	N	N	NM	NS	Z	PS
	NS	N	N	NM	NS	Z	PS	PM
	Z	N	NM	NS	Z	PS	PM	P
	PS	NM	NS	Z	PS	PM	P	P
	PM	NS	Z	PS	PM	P	P	P
	P	Z	PS	PM	P	P	P	P

Tabela 3.14: Conjunto de regras para *fuzzy*-PD (card. 7).

		erro						
		N	NM	NS	Z	PS	PM	P
$\Delta$ erro	N	P	PM	PS	N	NS	NM	NM
	NM	P	P	P	NM	NS	NS	NS
	NS	P	P	PM	NS	NS	NS	NM
	Z	P	PM	PS	Z	NS	NM	N
	PS	PM	PS	PS	NS	NM	N	N
	PM	PS	PS	PS	NS	N	N	N
	P	PS	PS	PS	NS	N	N	N

### 3.5.3) Resultados preliminares

Enquanto que os aspectos mais teóricos foram discutidos no capítulo anterior, o presente capítulo apresentou questões práticas que foram efetivamente usadas no trabalho. Sendo um conceito relativamente recente para grande parte da indústria de processos, o sistema híbrido digital/analógico, com interface padrão *fieldbus* foi usado para a comunicação de dados.

O presente trabalho não consta apenas do desenvolvimento do programa que implementa os algoritmos de controle. Várias melhorias foram feitas no sistema de

controle, no sentido global, para que o processo de polimerização fosse viável. Primeiramente, observou-se que a reação ocorria somente num vidro de *erlenmeyer* e exibia um rendimento praticamente nulo quando realizada no reator. Havia problemas de perda de calor, sendo que a temperatura do meio reacional dificilmente ultrapassava os 55 °C, e de perda de material, havendo necessidade de repor acetato de etila para evitar entupimento no tubo do densímetro. A vedação adequada do reator era dificultada pela presença de um agitador. Além disso, durante a amostragem para o densímetro era usada uma bomba centrífuga que contribuía para o baixo rendimento quebrando as cadeias poliméricas. O controle de aquecedor auxiliar era feito manualmente.

Abandonando o uso do agitador, a vedação do reator pôde ser aprimorada, obtendo-se temperaturas maiores e manutenção no nível de reagentes ao longo da reação. A agitação do meio era feita através do bombeamento e do borbulhamento de nitrogênio. A bomba centrífuga foi trocada por uma de deslocamento positivo. Contudo, mesmo com a troca de bombas, o funcionamento contínuo da bomba ainda prejudicava o rendimento através da quebra das cadeias. Por isso, foi introduzido, no programa de controle, um acionamento automático da bomba somente a cada três minutos, durante dez segundos. Deste modo, a bomba também passou a ser controlada por um bloco funcional, correspondente a um relé normalmente aberto na interface *fieldbus*. Da mesma forma, o estado do aquecedor auxiliar passou a ser controlado pelo programa a cada amostragem, de acordo com o desvio da temperatura em relação ao *set-point*: o relé é fechado se a temperatura está 4 °C abaixo e é aberto se a temperatura estiver 4 °C acima. Com todas essas mudanças, foi possível se ter uma polimerização dentro do reator com controle automático e com rendimento razoável.

Seguinte à descrição do aparato experimental, foram descritos aspectos teóricos e de implementação de uma das alternativas de controle, o modo PID (forma velocidade) convencional e adaptativo, mencionando também a sintonia e análise de estabilidade.

Finalmente, é mostrado o modo, em grande parte empírico, de implementação do controle *fuzzy*, através de um algoritmo híbrido. Também foi mostrado como se determinaram os fatores de escala e as funções de pertinência.



## **Capítulo 4**

### **Resultados**

## Introdução

Neste capítulo serão apresentados os resultados da implementação dos controladores PID, PID adaptativo e PID-*fuzzy*, seguindo as propostas de sintonia descritas no capítulo 3. Uma comparação final entre os resultados dos controladores será também apresentada.

Conforme descrito no capítulo anterior, o controle de temperatura é feito através de duas atuações, uma na válvula que regula a vazão de água de resfriamento, operando em modo PID ou *fuzzy*, e outra através de um controle *on/off* sobre um dos aquecedores (45 W). Assim, um dos critérios de performance do controle sobre a vazão, sua influência em manter o erro reduzido, pode ser o número de vezes que o aquecedor foi acionado. Ou seja, um acionamento freqüente significa que o controle de vazão não está sendo suficiente para manter a temperatura no *set-point* desejado. Este valor de *set-point* foi determinado empiricamente através do rendimento da reação, que era muito baixo em temperaturas de 55 ou 58 °C, mas que foi satisfatório a 63 °C que foi fixado como *set-point*. Da literatura era sabido que a polimerização do MMA ocorria numa temperatura próxima de 60 °C. Quanto menor o número de acionamentos, melhor. Portanto, para cada reação feita, serão apresentados dois gráficos como resultados: um deles contendo a temperatura do reator e a vazão em cada ponto de amostragem e outro mostrando o acionamento do aquecedor. Critérios como integral do erro absoluto (IAE) e integral do quadrado do erro (ISE) são os principais avaliadores de desempenho empregados nas comparações.

### 4.1) Controlador PID

Primeiramente, é mostrada a importância na escolha correta dos parâmetros de sintonia. Para um conjunto determinado empiricamente ( $K_c = -1,1 \text{ ml}/(\text{s} \cdot ^\circ\text{C})$ ,  $\tau_I = 300 \text{ s}$  e  $\tau_D = 30 \text{ s}$ ), o comportamento da reação, mostrado na Figura 4.1, exibe a dinâmica num caso em que os parâmetros foram escolhidos de forma indevida. Pode-se observar o aquecedor auxiliar em funcionamento freqüente, mostrado na Figura 4.2.

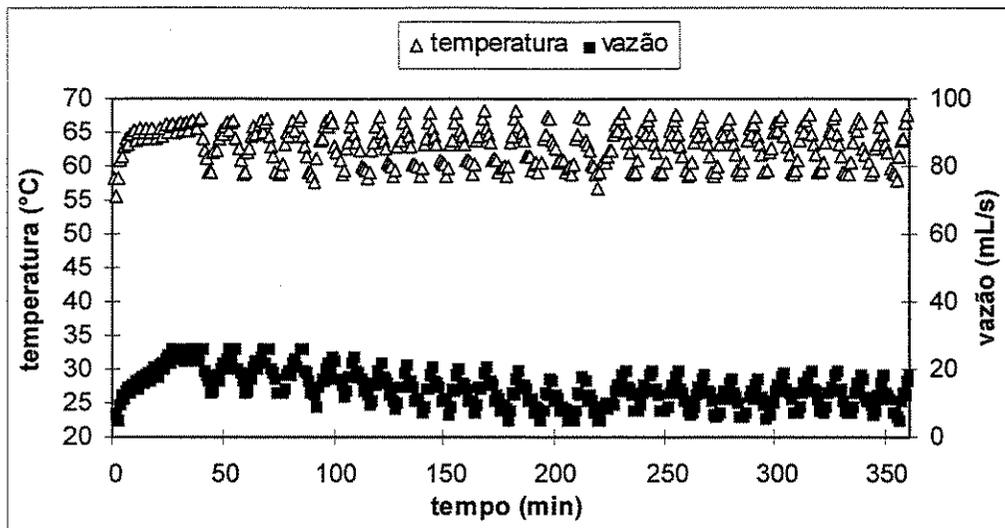


Figura 4.1: Reação com controle PID mal sintonizado.

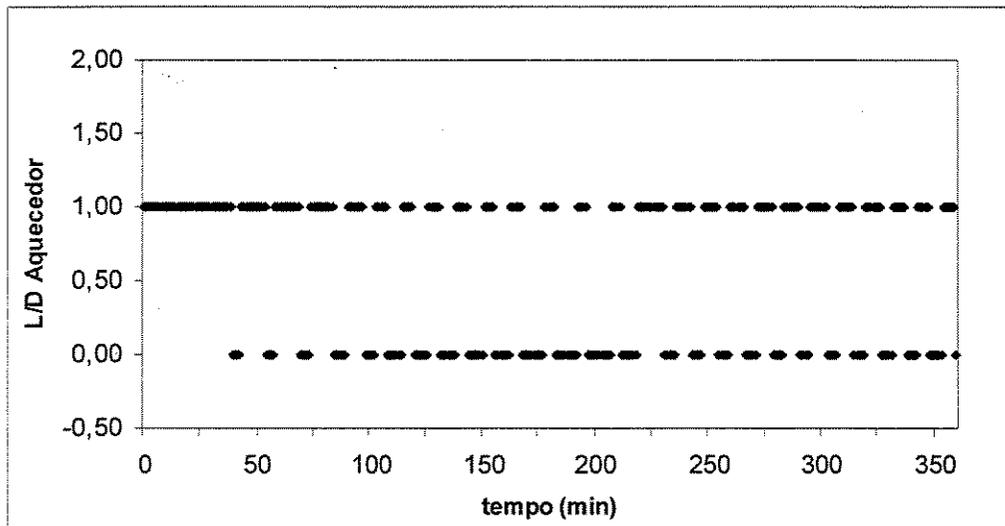


Figura 4.2: Acionamento do aquecedor.

A Figura 4.3 mostra um ensaio com controle digital PID (forma velocidade). Os parâmetros de sintonia foram obtidos através do procedimento de Cohen-Coon e análise de estabilidade (Tabelas 3.3 e 3.6:  $K_c = -1,1 \text{ mL}/(\text{s} \cdot ^\circ\text{C})$ ,  $\tau_I = 130 \text{ s}$  e  $\tau_D = 20 \text{ s}$ ). O comportamento da temperatura é claramente oscilatório, produzindo um frequente liga/desliga no aquecedor de 45 W, que deve ser acionado somente em casos de emergência.

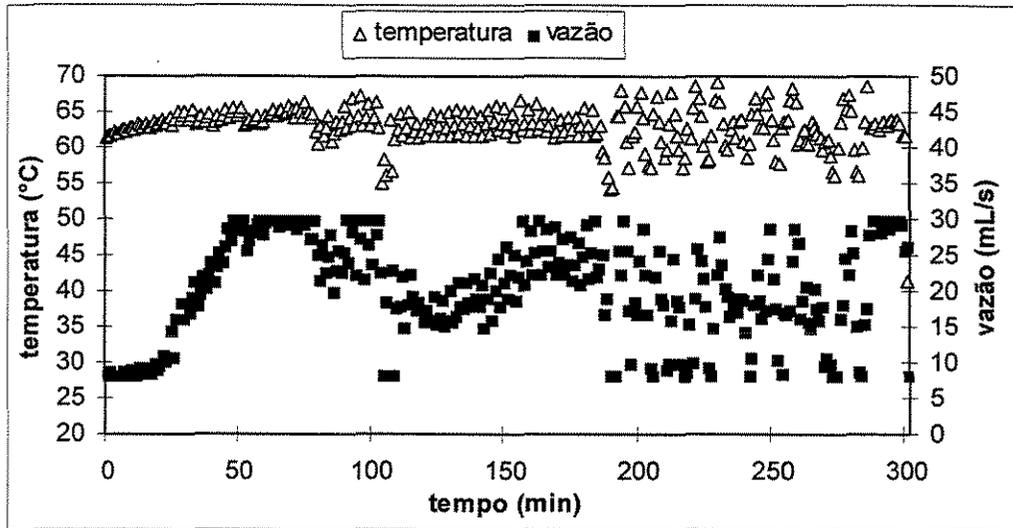


Figura 4.3: Reação com controle PID.

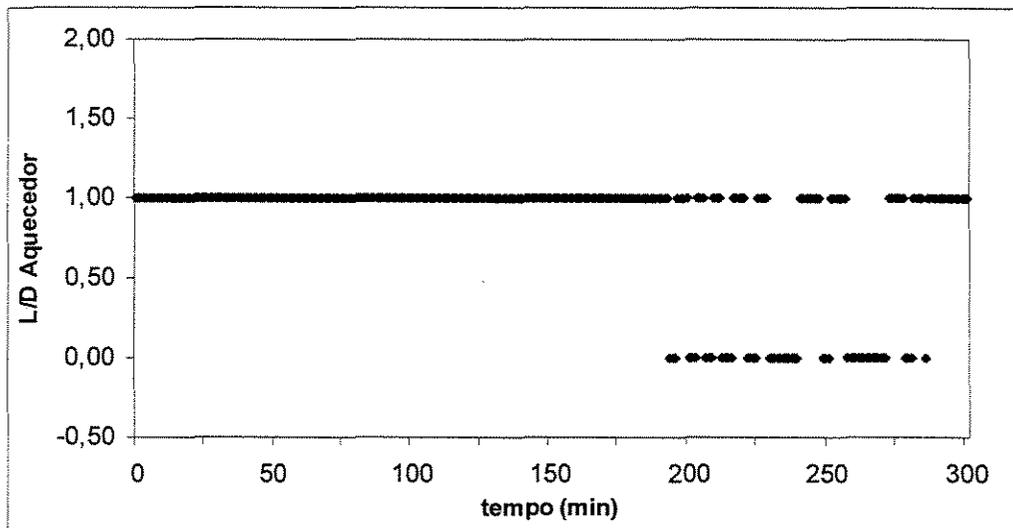


Figura 4.4: Acionamento do aquecedor.

Para melhorar o controle da temperatura da reação, foi adicionado um filtro de temperatura descrito pela equação

$$T_f = 0,4T(k) + 0,2T_f(k-1) + 0,1T_f(k-2) + 0,1T_f(k-3) + 0,1T_f(k-4) + 0,1T_f(k-5)$$

onde T: temperatura do reator; k: ponto de amostragem;  $T_f$ : temperatura filtrada do reator.

A influência deste filtro é mostrada nas Figuras 4.5 e 4.6. Nesta reação, a presença do filtro não trouxe melhoria significativa. O comportamento da temperatura ainda mantivesse excessivamente oscilatório num período da batelada.

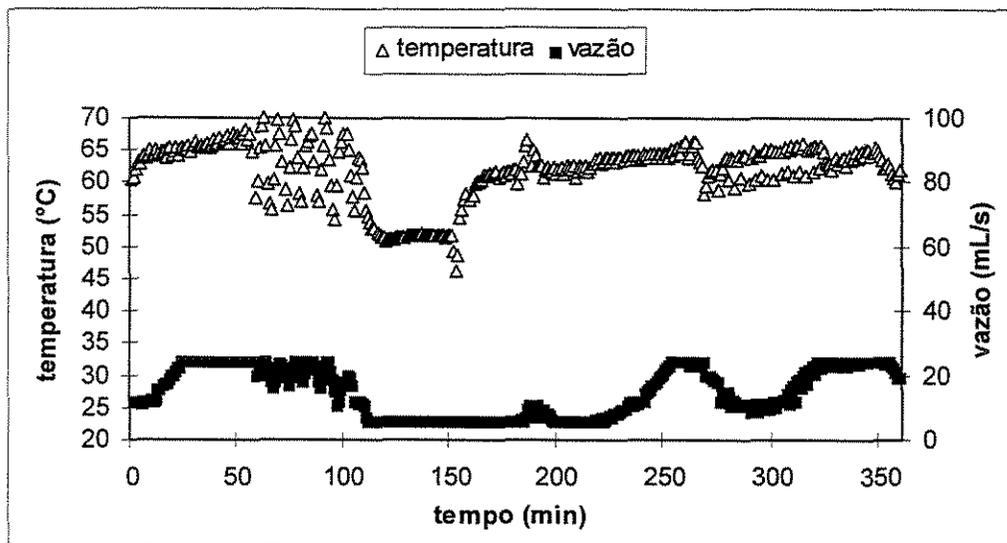


Figura 4.5: Reação com controle PID com filtro de temperatura.

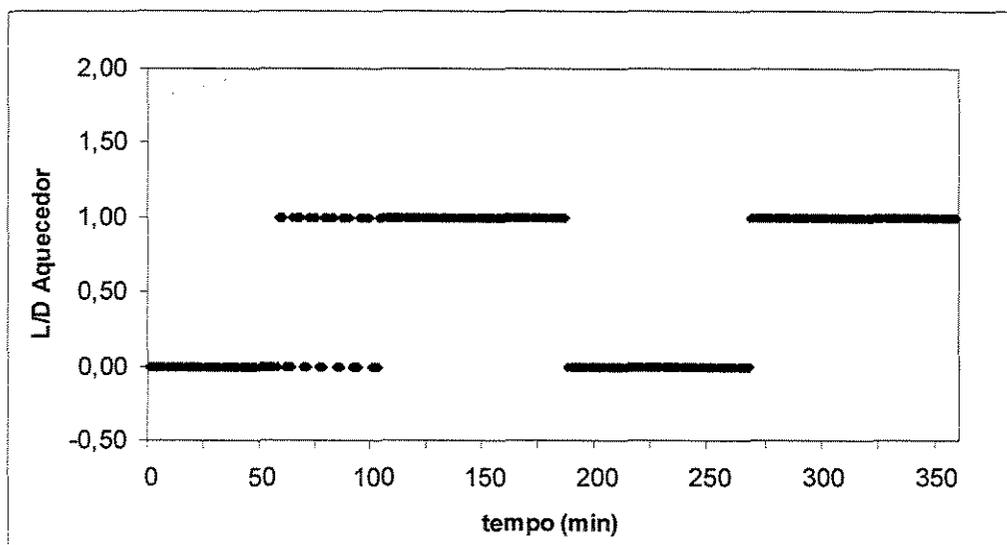


Figura 4.6: Acionamento do aquecedor.

## 4.2) Controlador PID adaptativo

A partir dos dados produzidos pelos ensaios em malha aberta feitos no início e o no final da reação (Tabelas 3.4 e 3.5) e considerando também os resultados da análise de estabilidade (Tabelas 3.7 e 3.8) foram realizadas duas reações com controle PID adaptativo com filtro de temperatura. Na reação mostrada pelas Figuras 4.7 e 4.8, foram usados três valores de  $K_c$ , cada um vigorando a cada 2 horas de polimerização. Na primeira etapa foi usado o menor valor correspondente a um *overshoot* de 10 %, ou seja  $K_c = -1,6 \text{ ml/s} \cdot ^\circ\text{C}$ . Na

etapa final foi empregado o menor valor correspondente a um *overshoot* de 10 %, ou seja  $K_c = -0,7 \text{ ml/s} \cdot ^\circ\text{C}$ . Durante a etapa intermediária de 2 horas foi usado o valor médio:  $K_c = -1,15 \text{ ml/s} \cdot ^\circ\text{C}$ . Os outros parâmetros de sintonia foram mantidos nos valores originais ( $\tau_i = 130 \text{ s}$  e  $\tau_D = 20 \text{ s}$ ) devidos aos resultados consistentes mostrados na Tabelas 3.4 e 3.5.

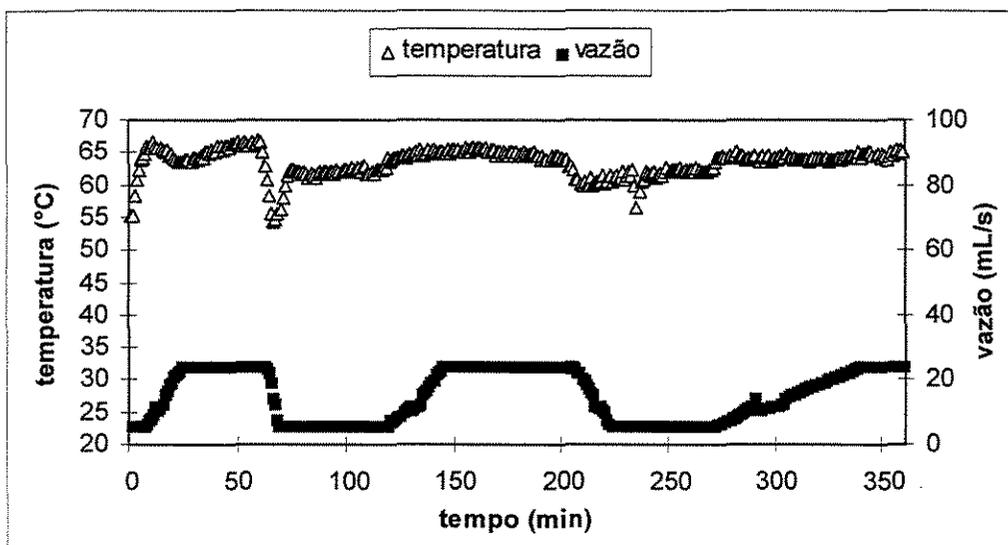


Figura 4.7: Reação com controle PID adaptativo com filtro de temperatura.

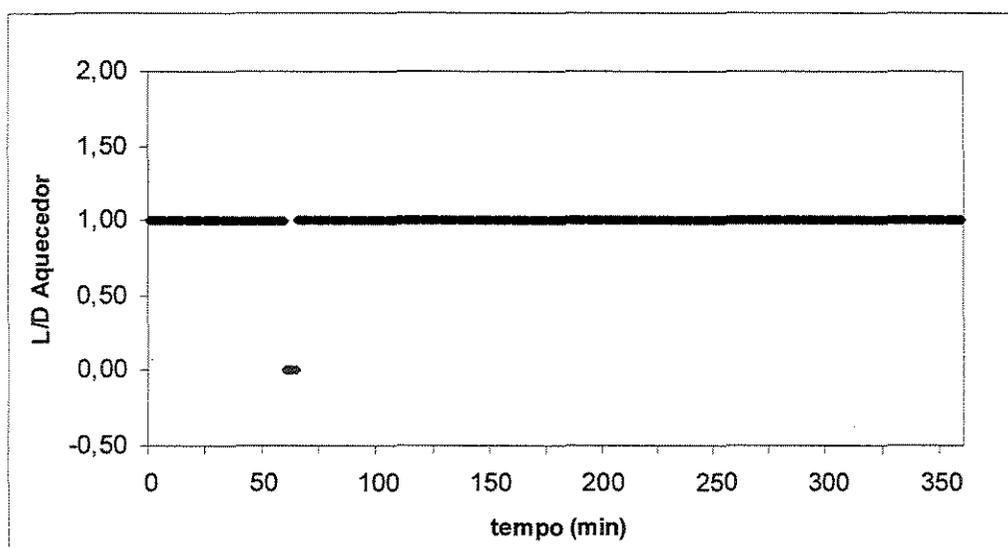


Figura 4.8: Acionamento do aquecedor.

Apesar de não haver um acionamento freqüente do aquecedor auxiliar, que é um indicador positivo do desempenho do sistema de controle, a temperatura ainda teve muita oscilação, de modo que foi decidido reduzir o valor de  $K_c$ . As Figuras 4.9 e 4.10, mostram

uma reação em que foram empregados os valores de  $K_c = -0,7 \text{ ml/s.}^\circ\text{C}$  na primeira etapa,  $K_c = -0,6 \text{ ml/s.}^\circ\text{C}$  no período intermediário e  $K_c = -0,5 \text{ ml/s.}^\circ\text{C}$  no final.

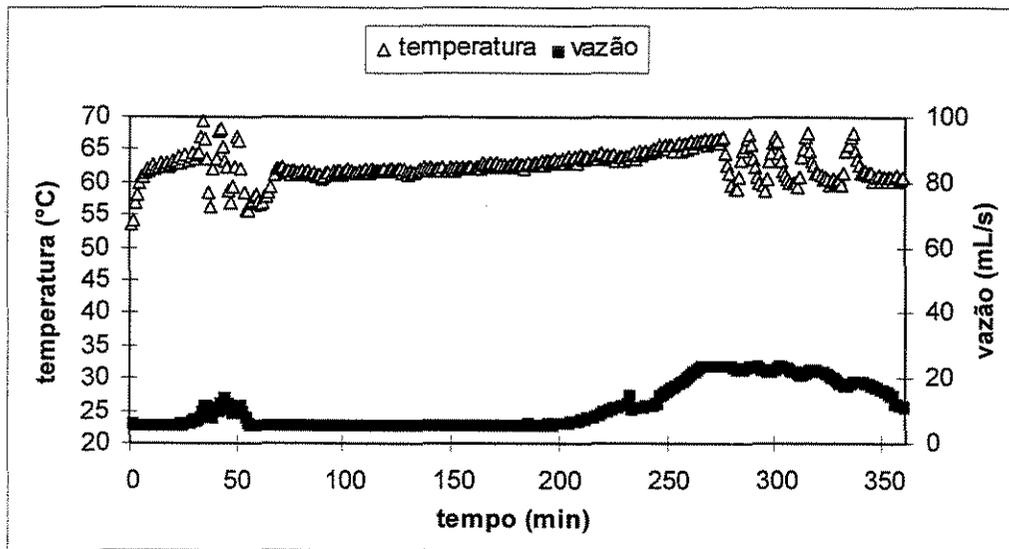


Figura 4.9: Reação com controle PID adaptativo com filtro de temperatura.

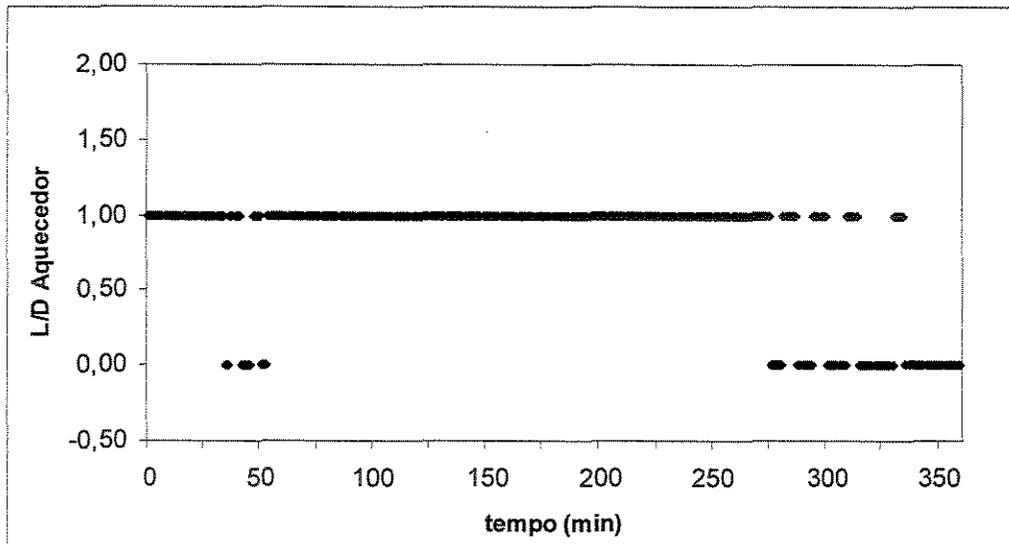


Figura 4.10: Acionamento do aquecedor.

Mesmo que o resultado obtido não tenha sido satisfatório em termos do controle de temperatura, esta foi a reação no modo PID com o melhor desempenho, mantendo a temperatura próxima do *set-point* por um período maior que 120 minutos.

### 4.3) Controlador *fuzzy*

A implementação dos controladores *fuzzy* visa a obtenção de um conjunto de parâmetros que resultem num controle mais suave em suas ações de controle e menos oscilatório em relação ao valor de referência da variável controlada. Para entender a necessidade das diferentes configurações utilizadas, as reações realizadas em malha fechada com o controlador *fuzzy* são colocadas em ordem cronológica. Dessa forma, a cada resultado, é possível entender as motivações para o uso da configuração seguinte. A primeira configuração é a mais simples, onde as funções de pertinência são distribuídas de maneira uniforme. Em relação ao erro e sua variação, as funções de pertinência são colocadas entre -1 e +1. Para a variação na ação de controle, é usada a faixa de -9 a +9 mL/s. Para compensar as prováveis deficiências de uma configuração genérica como esta, usa-se o filtro de temperatura (Eq. 4.1) e cardinalidade 7. Os resultados são mostrados nas Figuras 4.11 e 4.12.

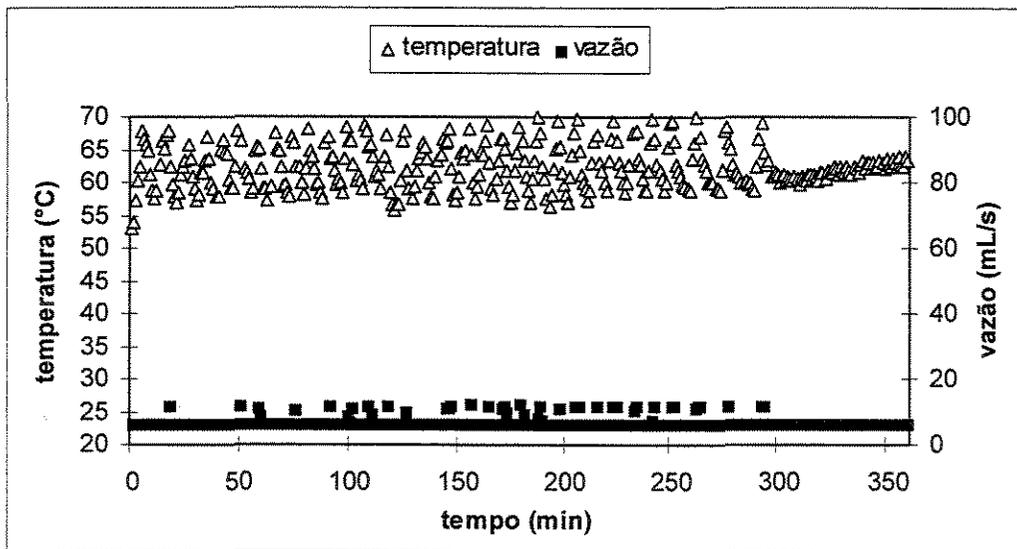


Figura 4.11: Reação com controle *fuzzy* com funções de pertinência igualmente espaçadas ( $n=7$ )

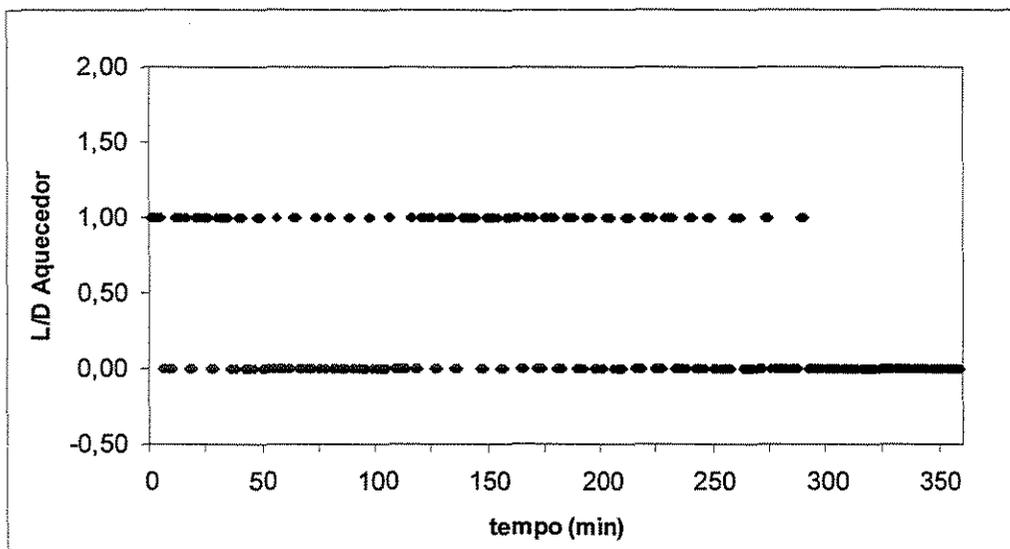


Figura 4.12: Acionamento do aquecedor.

As Figuras 4.11 e 4.12 mostram claramente a necessidade do emprego de funções de pertinência determinadas a partir das características dinâmicas particulares do processo. Praticamente, não houve atuação no processo através da vazão de resfriamento e o sistema de controle ficou reduzido à atuação on/off no aquecimento. Evidentemente, seria possível encontrar uma sintonia adequada alterando os quatro fatores de escala, mas, observando os resultados, é possível avaliar a dificuldade deste procedimento. No caso em que se utiliza as funções de pertinência derivadas dos ensaios em malha aberta (Figuras 3.20 a 3.25), uma sintonia adequada dos fatores de escala foi encontrada com poucos ensaios (Figuras 3.26 a 3.29). Dessa forma, na reação com controle *fuzzy* mostrada nas Figuras 4.13 e 4.14 utilizou-se as funções de pertinência determinadas no Capítulo 3 (Figuras 3.20 e 3.21). Esta é chamada de configuração padrão, no controle *fuzzy* desenvolvido, as especificações da primeira reação feita como neste modo, com funções de pertinência específicas para o processo. As reações seguintes são, basicamente, variações a partir desta configuração padrão, para verificar a influência das várias opções de sintonia. Por questão de simplicidade, a mesma cardinalidade foi usada para as três variáveis linguísticas (erro, variação do erro e variação da vazão).

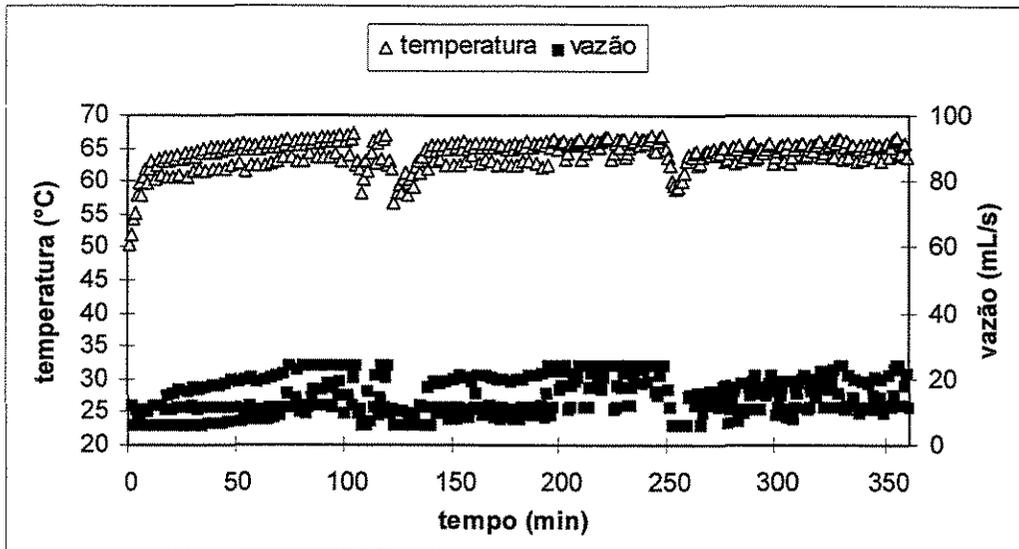


Figura 4.13: Reação com controle *fuzzy* ( $K_1 = -0,02$  e  $n=3$ ).

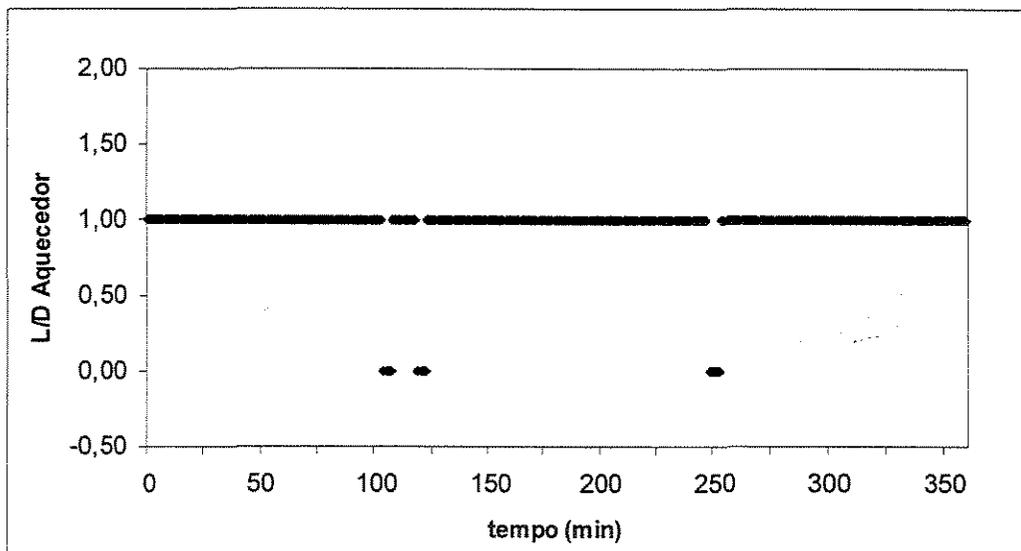


Figura 4.14: Acionamento do aquecedor.

De acordo com os resultados experimentais, o algoritmo PID-fuzzy mostrou melhor performance, mantendo a temperatura mais próxima do *set-point*, apesar da ocorrência de uma diferença persistente - chamada de *offset*. A vazão de resfriamento, sob controle *fuzzy*, foi capaz de estabelecer um comportamento mais suave de temperatura, principalmente no período final da batelada. Através da Figura 4.14 observa-se que o sistema on/off praticamente não foi acionado durante a batelada.

Conforme discussão apresentada no Capítulo 3, o parâmetro que pode ser modificado mais facilmente é o  $K_1$ . Com base nos testes de sintonia apresentados nas

Figuras 3.26 a 3.29, adotou-se preliminarmente o valor de -0,02. Nas reações seguintes, valores próximos, acima e abaixo de -0,02, serão usados para mostrar a influência deste parâmetro de sintonia, mantendo-se a cardinalidade igual a 3. Na reação representada na Figura 4.15, o valor utilizado foi de -0,01.

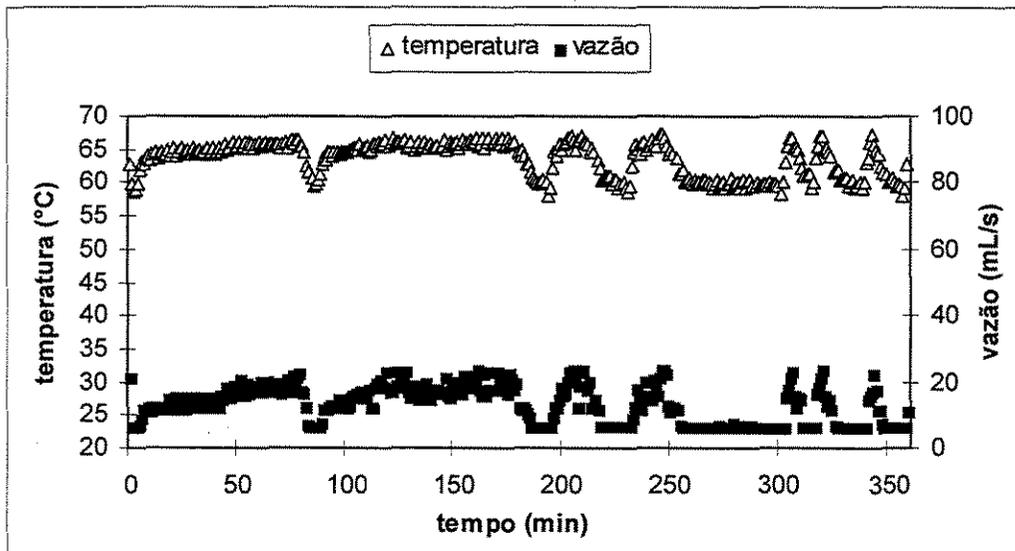


Figura 4.15: Reação com controle *fuzzy* ( $K_1 = -0,01$  e  $n=3$ ).

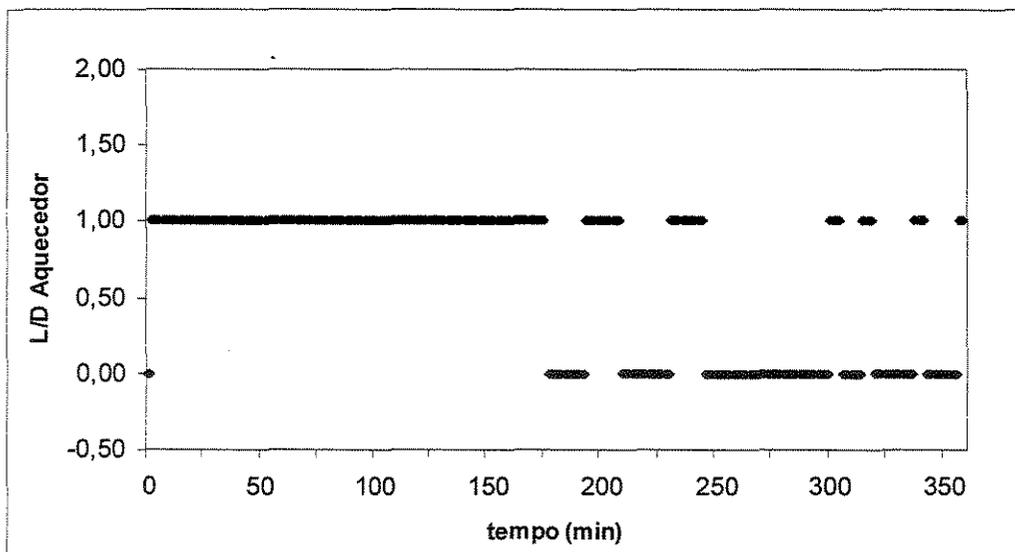


Figura 4.16: Acionamento do aquecedor.

O resultado desta implementação não se apresentou com qualidade do obtido anteriormente, mostrando comportamento mais oscilatório da temperatura e um maior número de acionamentos do segundo aquecedor. Principalmente no período final da reação,

talvez pelo aumento de viscosidade, o controle de temperatura foi insatisfatório. A Figura 4.17 mostra uma configuração análoga, mas com o valor de  $K_1$  menor, em termos absolutos, que a da configuração padrão.

Observa-se que, para um período da reação (até 180 minutos), houve um comportamento menos oscilatório do que no primeiro caso (Figura 4.13). Contudo, considerando o tempo total das reações, o desempenho se deteriorou, ou seja, o valor de  $-0,02$  foi mais adequado até o momento.

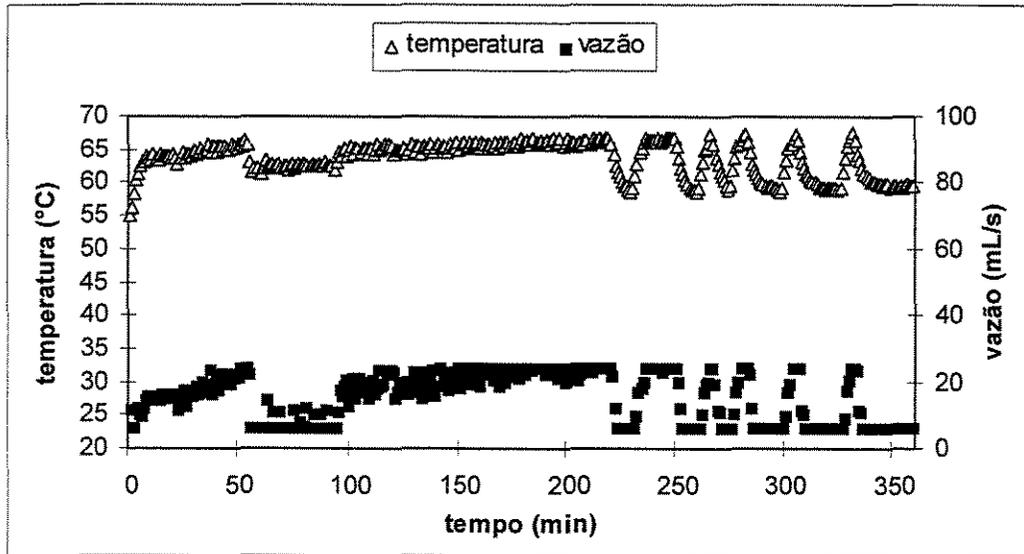


Figura 4.17: Reação com controle *fuzzy* ( $K_1 = -0,03$  e  $n=3$ ).

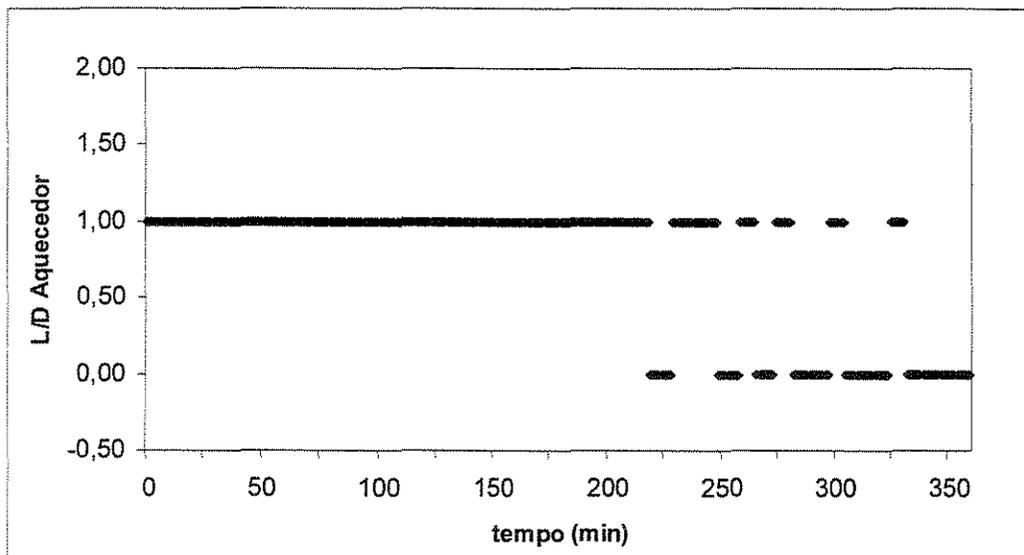


Figura 4.18: Acionamento do aquecedor.

Como a oscilação de temperatura ainda persiste, foi empregado o filtro de temperatura (Eq. 4.1) na configuração padrão para amenizar este problema. A Figura 4.19 mostra esta batelada.

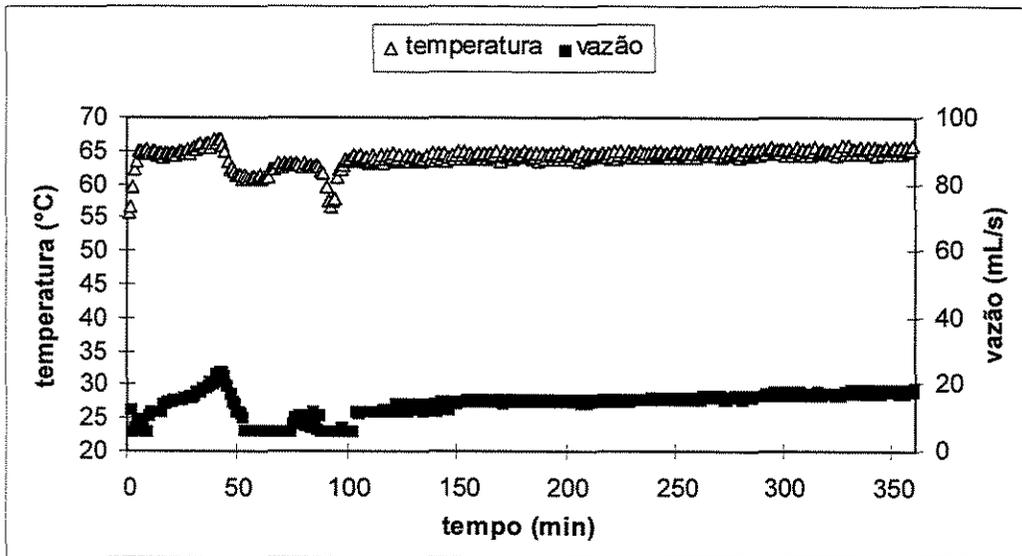


Figura 4.19: Reação com controle *fuzzy* (com filtro).

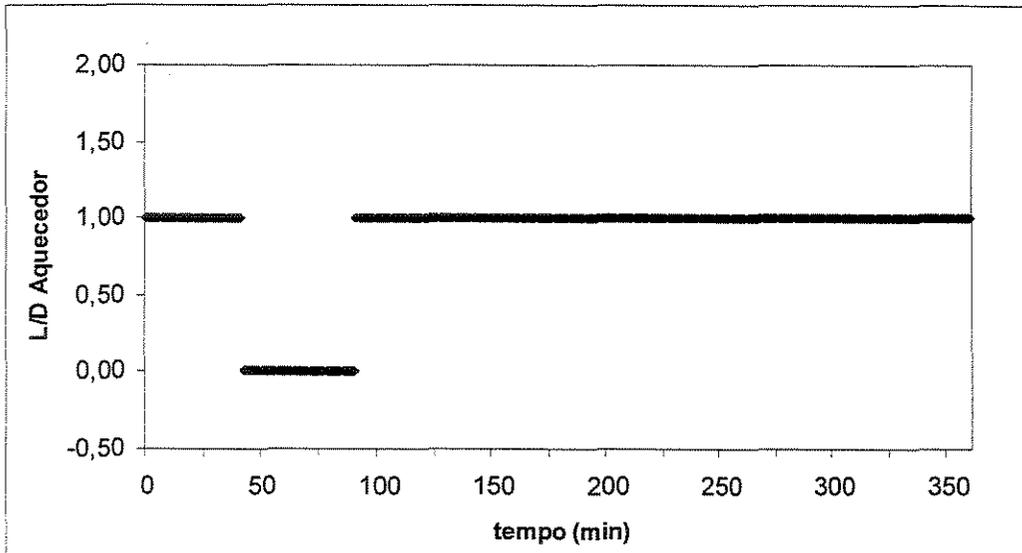


Figura 4.20: Acionamento do aquecedor.

O comportamento da temperatura é semelhante ao da configuração padrão (Figura 4.13), mas com menor oscilação. No entanto, *offset* ainda é presente e para tentar eliminá-lo, é adicionado um valor constante (2 mL/s) ao valor de vazão de saída calculado pelo algoritmo *fuzzy*. O resultado é mostrado nas Figuras 4.21 e 4.22.

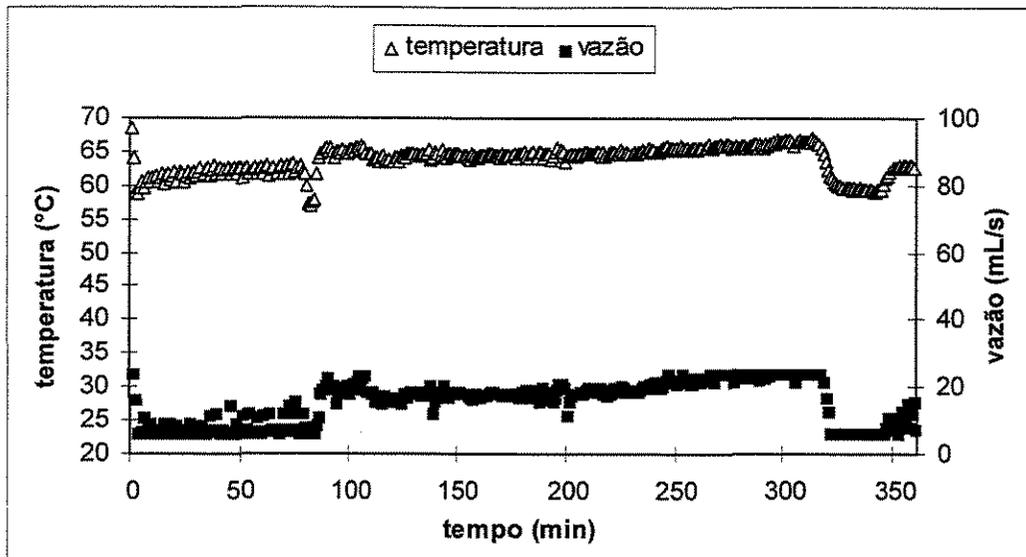


Figura 4.21: Reação com controle *fuzzy* (com + 2 mL/s).

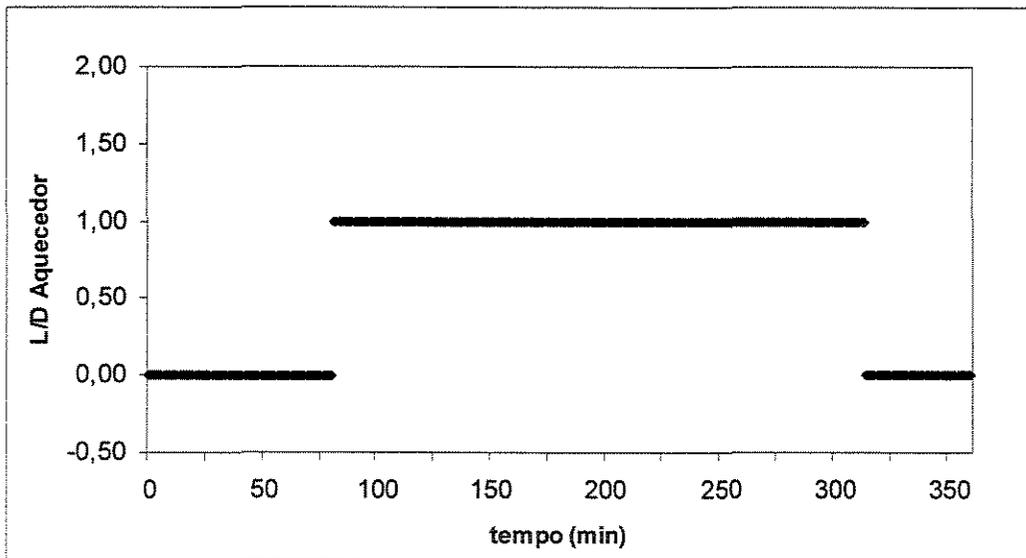


Figura 4.22: Acionamento do aquecedor.

Observa-se que o *offset* de temperatura não foi eliminado. Uma razão que poderia justificar sua existência é a assimetria nas funções de pertinência e a reduzida cardinalidade utilizada. Para corrigir esta deficiência, o programa computacional do algoritmo *fuzzy* foi modificado de forma a possibilitar a escolha de outras cardinalidades. As Figura 4.23 e 4.24 mostram o resultado da reação na configuração padrão, sem o filtro de temperatura e com  $K_1 = -0,02$ , mas com cardinalidade 5.

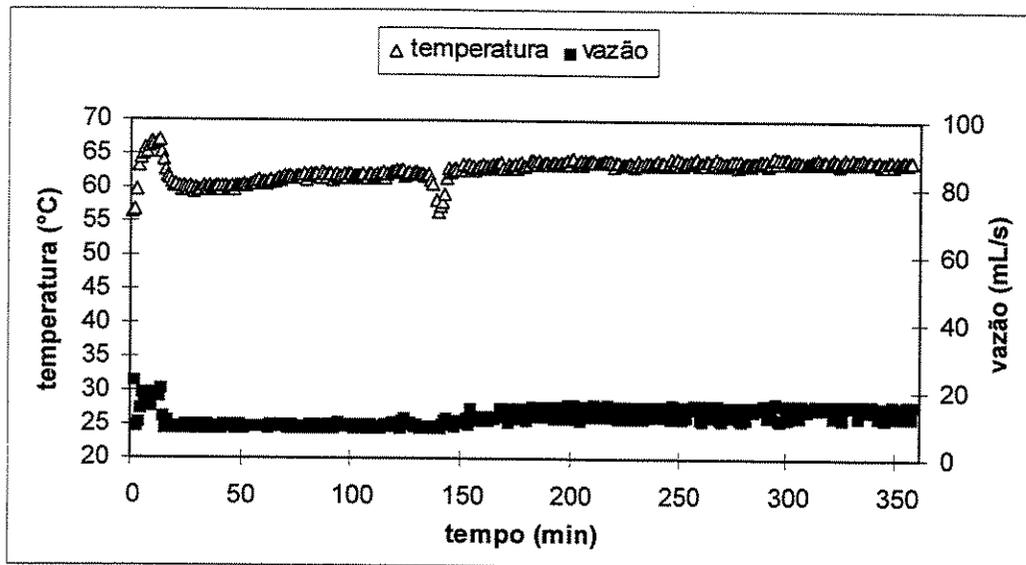


Figura 4.23: Reação com controle *fuzzy* (n=5).

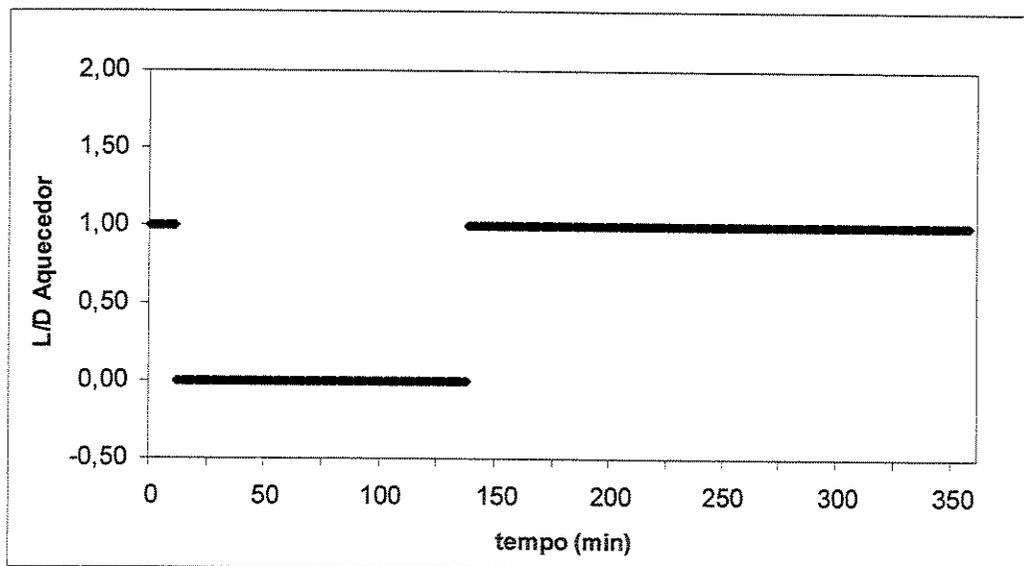


Figura 4.24: Acionamento do aquecedor.

As Figuras 4.25 e 4.26 mostram o resultado usando-se as mesmas condições, mas com cardinalidade 7.

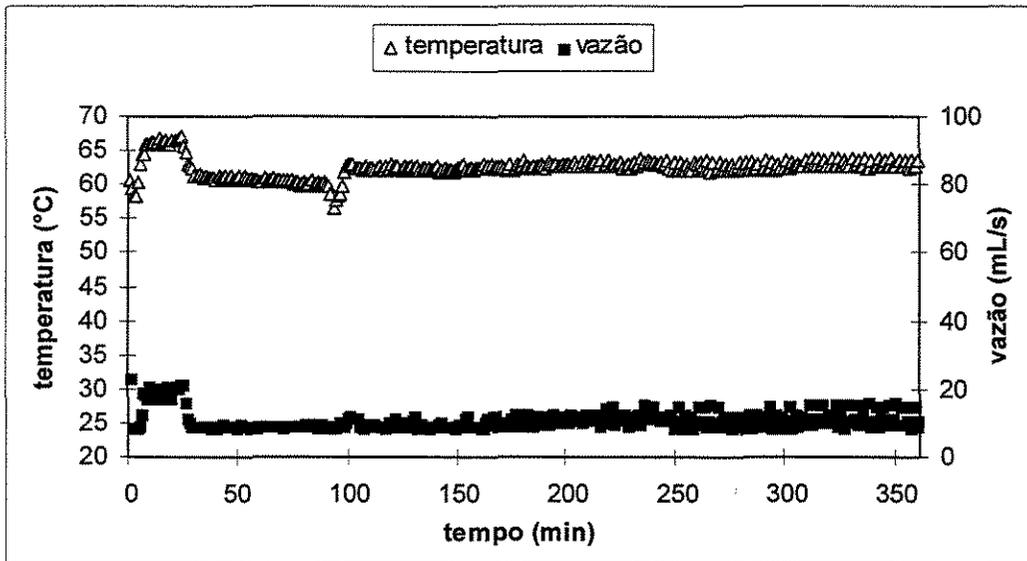


Figura 4.25: Reação com controle *fuzzy* (n=7).

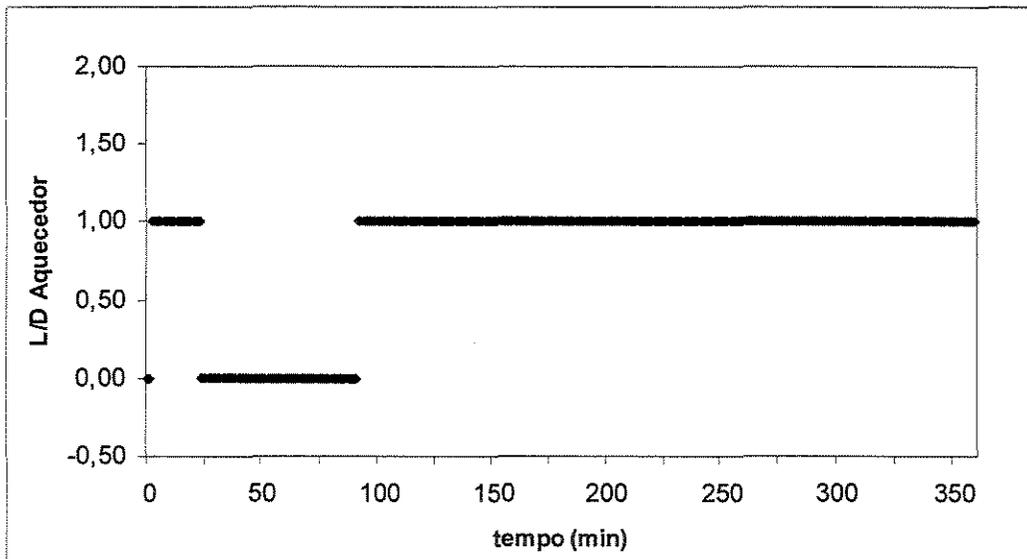


Figura 4.26: Acionamento do aquecedor.

Os resultados obtidos com o aumento de cardinalidade foram significativamente melhores entre todas bateladas realizadas com o controle *fuzzy*. Observando as Figuras de 3.22 a 3.25 pode-se concluir que essa melhoria pode ser atribuída à maior resolução das funções de pertinência tanto na etapa de fuzificação quanto na de defuzificação. O uso de funções de pertinência intermediárias suaviza a ação de controle.

Para ilustrar a variação das outras variáveis secundárias medidas durante a polimerização, é mostrada a variação da massa específica (Figura 4.27) e das temperaturas de entrada e saída da água de resfriamento (Figura 4.28) para a reação com controle *fuzzy* e cardinalidade 7.

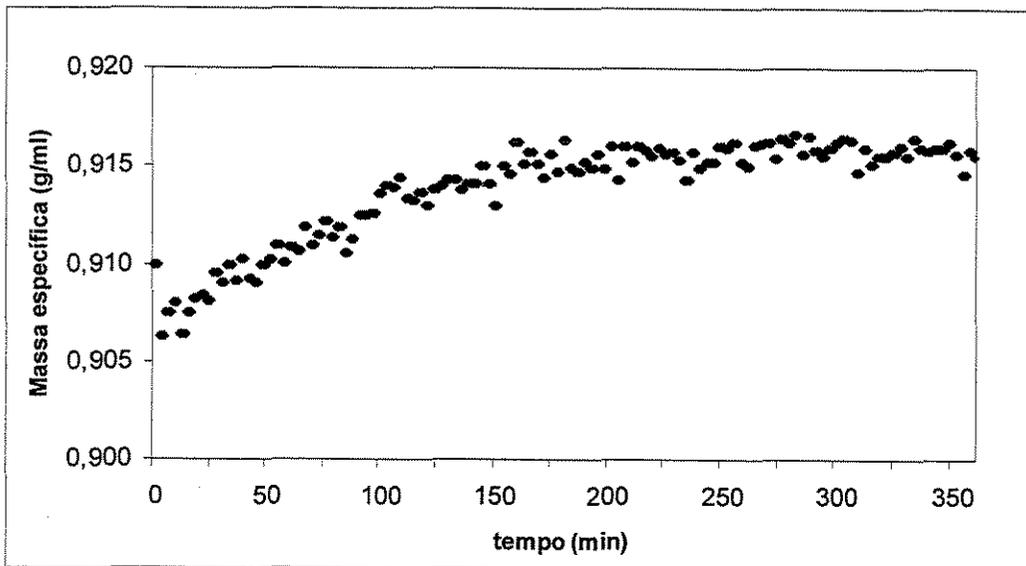


Figura 4.27: Massa específica.

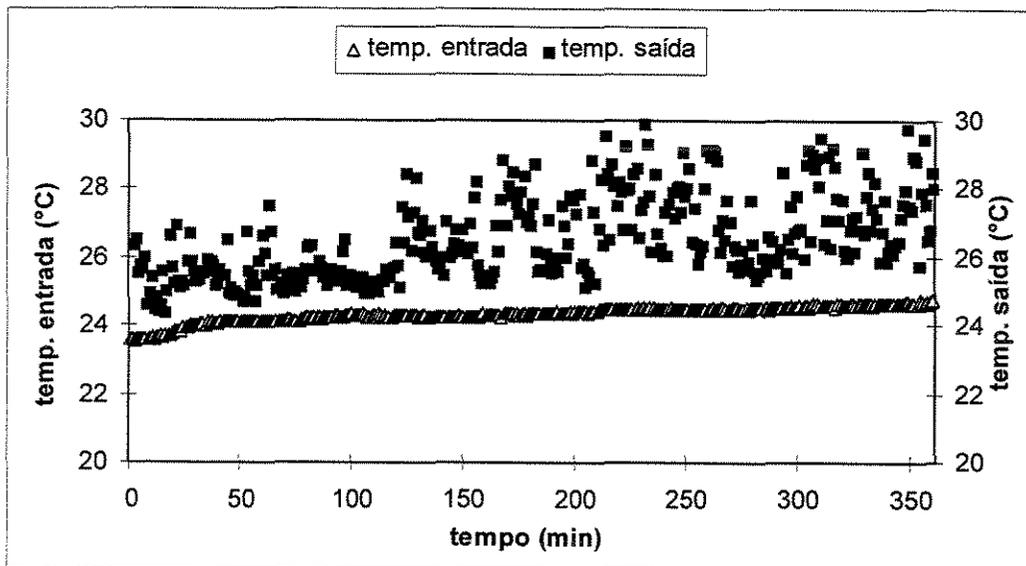


Figura 4.28: Temperaturas de entrada e saída.

Além desses experimentos de controle regulatório, mantendo o *setpoint* a 63 °C, também aplicou-se o sistema de controle *fuzzy* em um problema de controle servo. Para isso, fixou-se o *setpoint* em 64 °C na primeira metade da reação e em 62 °C no período

restante. Como é uma forma de controle mais difícil, a configuração usada é a de cardinalidade 7 e com o filtro. O resultado é apresentado nas Figuras 4.29 e 4.30. Considerando a maior dificuldade desta reação, o resultado foi satisfatório.

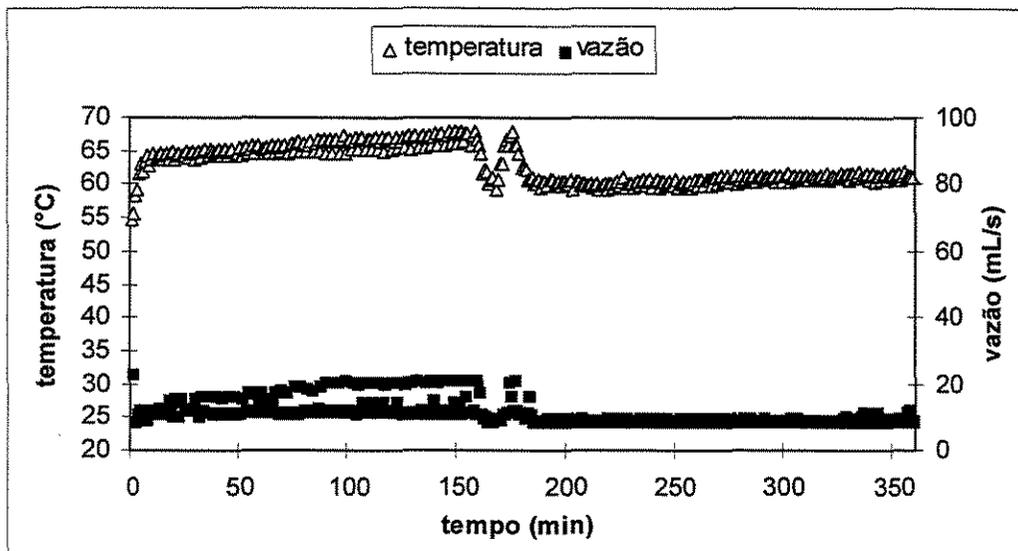


Figura 4.29: Reação com controle servo (n=7 e filtro)

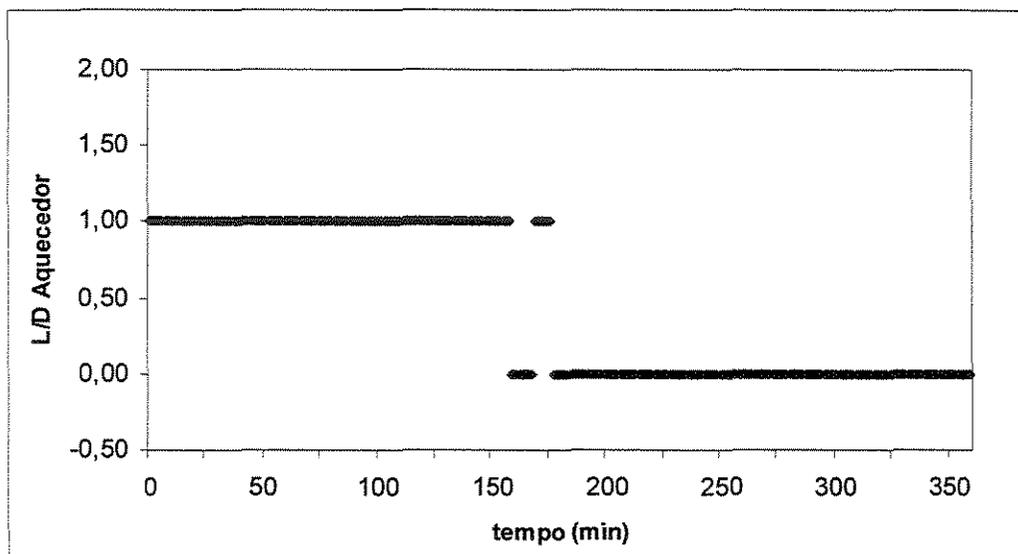


Figura 4.30: Acionamento do aquecedor.

## 4.4) Comparações

Os resultados de reações com diferentes parâmetros de sintonia são apresentados nas Tabelas 4.1 e 4.2. Apesar de ter sido obtida uma grande melhoria no controle de temperatura, é observado que a conversão aumenta a medida que a temperatura se aproxima de 70 °C (experimentos com controle inferior). Porém, por razões de segurança, a temperatura do *set-point* foi mantida em 63 °C.

Tabela 4.1: Comparação de desempenho (Controle PID).

Kc (mL/s.°C)	Filtro	ISE	IAE	Acionamentos	Conversão (%)
-1,1 (*)	não	175621	53470	51	16,74
-1,1	não	337437	1145782	19	33,40
-1,1	sim	485407	70415	2	21,41
-1,7 ; -1,15 ; -0,6	sim	96267	37258	2	18,61
-0,6 ; -0,5 ; -0,4	sim	121608	39188	15	16,29

(\*) neste caso foi empregado  $\tau_I = 300$  s e  $\tau_D = 30$  s

Em todas as reações foi usado  $\tau_I = 130$  s e  $\tau_D = 20$  s

Tabela 4.2: Comparação de desempenho (Controle *fuzzy*).

K <sub>I</sub>	Filtro	ISE	IAE	Acionamentos	Card.	Conversão (%)
-0,02 (*)	sim	259917	61546	63	7	22,7
-0,02	não	137792	44459	6	3	35,5
-0,02	sim	70687	33927	2	3	26,5
-0,01	não	148436	52819	13	3	29,6
-0,03	não	151202	50855	11	3	25,5
-0,02	não	57458	27541	2	5	25,9
-0,02	não	50454	22515	3	7	23,3
-0,02 (**)	sim	81699	34829	3	7	23,8

(\*) reação com funções de pertinência uniformemente distribuídas

(\*\*) reação das Figuras 4.29 e 4.30, com *set-point* variável

Apesar de nenhuma reação com controle PID ter apresentado um controle de temperatura plenamente satisfatório, a introdução do modo adaptativo resultou em melhoria significativa, flexibilizando a resposta do controlador PID e tornando-o mais apto ao processo em batelada. No controlador *fuzzy*, a alteração nos fatores de escala, que não têm nenhuma relação com a lógica *fuzzy* em si, não foi efetivo na melhoria do desempenho. O primeiro ganho real de desempenho foi devido ao emprego do filtro de temperatura, diminuindo sua oscilação. Apesar disso, a eliminação do *off-set* só foi possível com o incremento de cardinalidade, com um aumento na sensibilidade do sistema de controle.

#### 4.4.2) Qualidade do polímero produzido

Apesar de manter as mesmas condições em todas as reações realizadas, alguns polímeros foram produzidos com aspecto muito diferente da maioria. Para um processo considerado como bem sucedido, o produto final deveria ter aspecto visual uniforme, conforme mostrado na Figura 4.31.

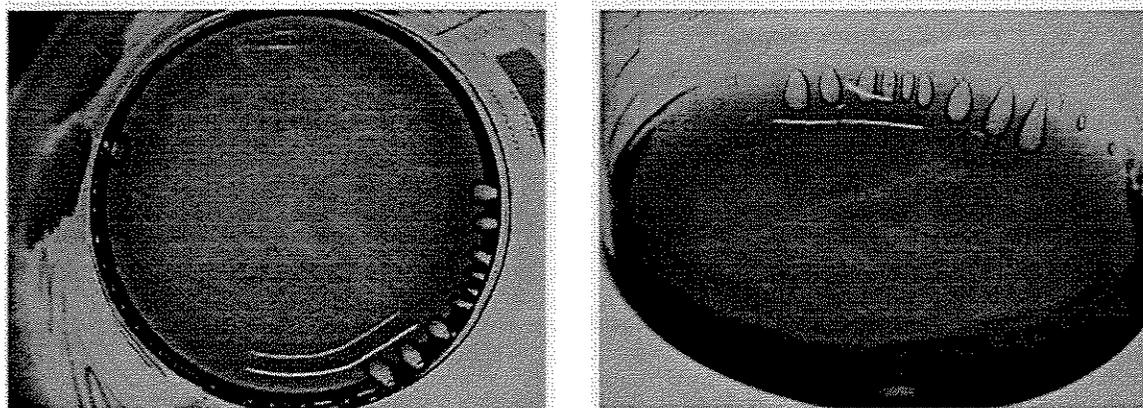


Figura 4.31: Polímero com aspecto normal (ensaio da Figura 4.15).

No entanto, em algumas reações, o resultado obtido foi muito diferente, tendo o aspecto de algum sólido cristalizado dentro do polímero. Este resultado é diferente dos padrões e é mostrado nas Figuras 4.32 e 4.33.

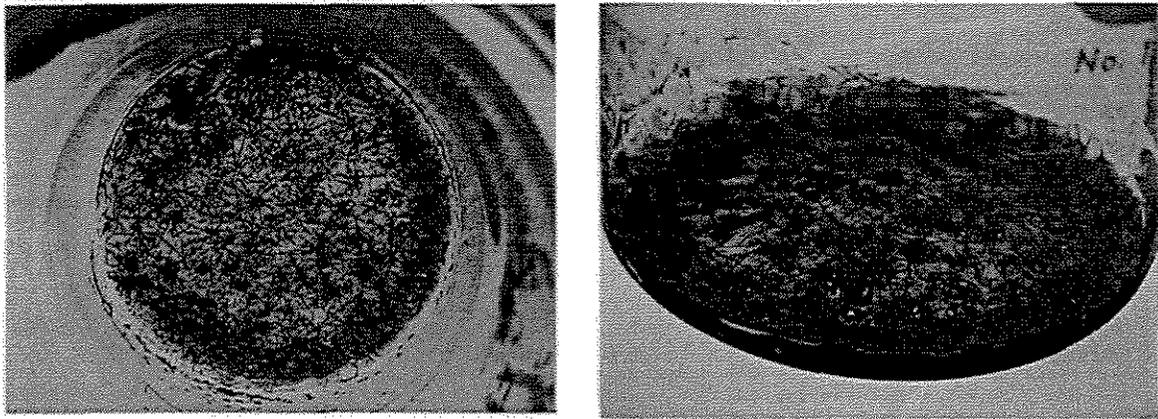


Figura 4.32: Polímero com aspecto fora dos padrões (ensaio da Figura 4.19).

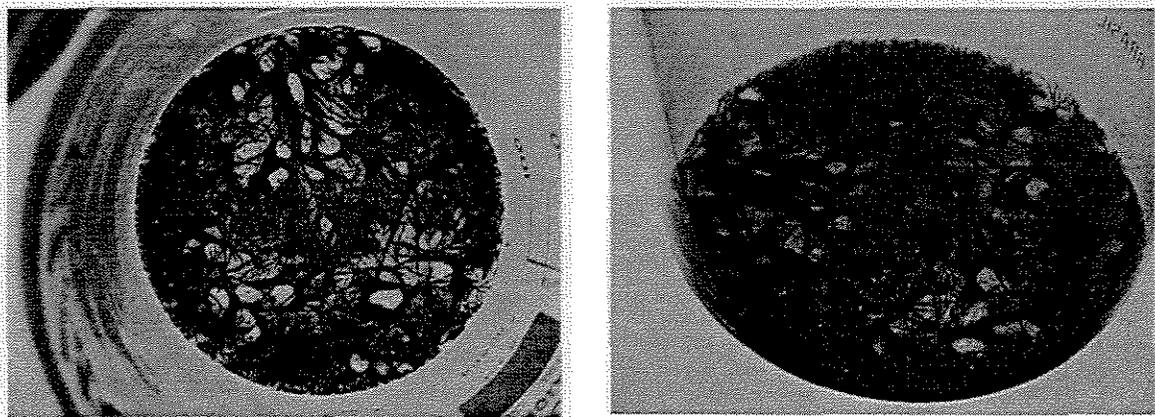


Figura 4.33: Polímero com aspecto fora dos padrões (ensaio da Figura 4.21).

Em princípio, não houve uma hipótese que explicasse uma diferença tão acentuada, uma vez que não houve diferença significativa no procedimento experimental. Segundo FELISBERTI (2004), a diferença entre os procedimentos não estaria na condução da reação em si, como se pensava, mas no que é feito depois da reação. Depois de finalizada a polimerização, os aquecedores são desligados, e, depois da temperatura do meio aproximar-se da temperatura ambiente, o produto é recolhido e nele é injetado ar e adicionado hidroquinona, parando a reação. A seguir o acetato é evaporado com o uso de um exaustor ou de maneira natural, quando o laboratório permanece fechado. Assim, o

tempo de evaporação pode ter influenciado no resultado, ou seja, numa evaporação mais lenta, com pouco uso do exaustor, houve tempo de cristalização de um excesso de iniciador.

Análises de Cromatografia por Permeação em Gel (GPC) foram realizadas para verificar a massa molar:

Tabela 4.3: Resultados das análises de GPC.

Massa molar média (Daltons)	Polidispersão	Condições de experimento
76000	1,27	<i>Fuzzy</i> ( $K_1 = -0,01$ e $n=3$ )
74200	1,78	<i>Fuzzy</i> ( $K_1 = -0,02$ e $n=7$ )
68000	1,76	<i>Fuzzy</i> ( $K_1 = -0,02$ e $n=3$ )

#### 4.4.2) Conclusões

Neste capítulo são mostradas as várias reações em batelada controladas por PID, propondo algum aprimoramento a cada reação, mas não chegando a um resultado plenamente satisfatório, embora tenham sido observadas melhoras significativas. Contando com um maior número de parâmetros de sintonia, foram realizadas reações com o controlador *fuzzy* tendo alterações, a cada ensaio, nos parâmetros de sintonia visando um melhor resultado na batelada seguinte, reforçando o caráter empírico da sintonia do controlador *fuzzy*. As melhorias mais visíveis foram conseguidas com o emprego de um filtro de temperatura e com o aumento da cardinalidade. Este último fator eliminou um *offset* persistente, que mantinha a temperatura acima do *set-point*. Como resultado, uma melhoria significativa foi observada nos dois critérios de desempenho utilizados: proximidade em relação ao *set-point* e frequência de acionamentos do aquecedor auxiliar.

## **Capítulo 5**

### **Conclusões e sugestões para trabalhos futuros**

O problema de seleção de controladores para lidar com um processo particular é complexo. A principal relação de custo e benefício que deve ser analisada é a sofisticação do controlador em relação aos ganhos de desempenho. Sistemas de controle mais simples, como o PID convencional, são mais conhecidos, menos custosos e mais fáceis de sintonizar. Contudo, não se consegue um desempenho satisfatório para qualquer processo. Para processos mais complexos, como os de alta não-linearidade, são necessários controladores mais elaborados como o *fuzzy*.

A literatura mostra um número relativamente pequeno de trabalhos experimentais com aplicação de metodologias avançadas de controle em processos químicos. O trabalho presente oferece um resultado mais amplo. Estabelece um estudo comparativo entre metodologias diferentes de controle para responder, com base experimental, a questão sobre a seleção de controladores para o processo particular de polimerização do MMA.

Este processo tem características que dificultam uma previsão sobre qual metodologia será mais adequada. O modo batelada do processo já dificulta a escolha de parâmetro de sintonia para todo o intervalo de batelada. Conforme comprovado pelos testes realizados em malha aberta, a polimerização é não-linear, indicando um problema para o PID. A polimerização também está sujeita a sofrer as conseqüências do efeito gel.

É interessante notar que o enfoque experimental do trabalho foi importante para o ganho de experiência na solução de problemas de implementação de um sistema real. Trabalhos com simuladores têm grande relevância, mas não são capazes de ensinar a lidar com vários problemas que surgem num equipamento. Entre os principais problemas práticos que foram solucionados, no reator, pode-se destacar a melhoria na vedação para reduzir a perda de calor e a troca da bomba de engrenagens pela de deslocamento positivo para incrementar a produção de polímero. Quanto ao programa, além da implementação dos algoritmos de controle, foram resolvidos problemas como a sincronização do acionamento da bomba de modo que, no momento da amostragem, a bomba estivesse desligada e que o tubo do densímetro estivesse com material reagente. Essa medida foi necessária porque o movimento pulsante da bomba dificulta a leitura do densímetro.

O controle PID (forma velocidade) de um modo geral não se mostrou adequado para o controle da polimerização. Como o uso do filtro de temperatura não amenizou o

problema de forma satisfatória, foi aplicada uma forma adaptativa, com melhores resultados (Tabela 4.1).

Quanto ao controle *fuzzy* (estrutura de LI, 1996), o trabalho apresentou a dificuldade em se determinar a sintonia conveniente para se atingir os resultados desejados. Para a escolha mais simples de funções de pertinência, ou seja, funções uniformemente espaçadas, a ação de controle foi bastante deficiente, pela falta de aspectos particulares da dinâmica do processo. Em relação às funções de pertinência determinadas a partir dos ensaios em malha aberta, entre as diversas opções de sintonia, a alteração do parâmetro de escala mostrou um resultado melhor que o PID, mas ainda com dois aspectos indesejáveis: o *off-set* persistente de temperatura e oscilação demasiada. A aplicação de um filtro digital obteve sucesso para resolver o problema do comportamento oscilatório da temperatura da reação, mas a temperatura ainda mantinha-se distante do valor desejado. Outro parâmetro de sintonia alterado foi a cardinalidade. Nos casos de erro pequeno, o objetivo seria ativar funções de pertinência intermediárias e suavizar a ação de controle. Finalmente, o objetivo foi atingido. Portanto, as várias opções de sintonia do controlador *fuzzy* garantiram maior flexibilidade para resolver o problema de controle de temperatura. Ou seja, se um parâmetro não conseguia produzir o efeito desejado, há vários outros que podem ajustar a saída do controlador. Pela quantidade de configurações diferentes testadas, pode-se perceber que o controle *fuzzy* conta com uma grande “caixa de ferramentas” para resolver problemas.

Observando-se as Tabelas 4.1 e 4.2 é possível verificar que, de um modo geral, os melhores resultados foram obtidos com pequenos ajustes na ação de controle. No caso do controle PID, o melhor desempenho foi resultado de um valor de  $K_c$  decrescente. Em relação ao *fuzzy*, a temperatura se manteve próxima ao *set-point*, com eliminação do *off-set*, somente com o aumento da cardinalidade. Com este aumento, pequenos desvios de temperatura ativavam funções de pertinência intermediárias, aumentando o ajuste fino na ação de controle. Isto foi capaz de estabelecer uma variação de temperatura suave, principalmente no período final da batelada. Também foi observado que a frequência de acionamentos do segundo aquecedor foi significativamente diminuída.

Como conclusão, o PID-*fuzzy* foi considerado mais adequado para o processo de polimerização porque mostrou desempenho muito superior ao PID convencional tanto em problemas regulatórios como num experimento com controle supervísório.

Como sugestões para trabalhos futuros, notam-se dois aspectos importantes que podem explorados no sentido de melhorar os objetivos do sistema de controle, extrapolando o critério de controle de temperatura. Primeiramente, a qualidade do polímero deve ser levada em consideração como um critério de desempenho da polimerização. Para isso, deve ser melhorado o critério de obtenção do *set-point*. Ao invés de escolher uma condição operacional onde a reação ocorra com um rendimento razoável, é proposto determinar uma certa trajetória que resulte num polímero de alta qualidade, medida de algum modo quantitativo como o peso molecular e sua distribuição.

A segunda sugestão para trabalhos futuros está, na verdade, relacionada também com a qualidade do polímero. Ao invés de se ter a temperatura do meio reacional como variável controlada, deve ser desenvolvida uma rede neural que estabeleça um mapeamento entre as condições operacionais da polimerização e a massa molecular média do produto. Ou seja, é sugerida implementação de um *soft-sensor* para controle de massa molecular média, sendo um evolução em relação ao mero controle de temperatura.

## Referências

**AHN, S., CHANG, S. e RHEE, H.**, Journal of Applied Polym. Science, 69, pág. 59-68, 1998.

**BERNARD, J. A.**, Use of a rule-based system for process control, IEEE Control Systems magazine, Outubro, pp. 3-13, 1988.

**BRANDÃO, D.**, Bloco funcional para controle Fieldbus por variáveis de estado. EESC/USP, Dissertação (Mestrado), São Carlos, 2000.

**BÜHLER, D., KÜCHLIN, W.**, Remote fieldbus system management with java and XML, International Symposium on Industrial Electronics (Proceedings), vol. 1, 2000, pág. 1-6.

**BÜHLER, D., NUSSER, G., KÜCHLIN, W. e GRUHLER, G.**, The java fieldbus control framework - Object oriented control of fieldbus devices, Fourth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (Proceedings), 2001, pág. 153-160.

**CARVALHO, M. T. e DURÃO, F.**, Control of a flotation column using fuzzy logic inference, Fuzzy Sets and Systems, 125, pp. 121-135, 2002.

**CHANG, J. S. e LAI, J. L.**, Ind. Chem. Res., 31, 861, 1992.

**CHANG, J. e LIAO, P.**, Molecular weight control of a batch polymerization reactor: Experimental study, Industrial & Engineering Chemistry Research, v. 38, n. 1, pág. 144-153, 1999.

**CHEN, C., PENG, S.**, A nonlinear control scheme for imprecisely known processes using the sliding mode and neural fuzzy techniques, Journal of Process Control, vol. 14, n. 5, 2004, pág. 501-515.

**CROWLEY, T. J. e CHOI, K.**, On-line monitoring and control of a batch polymerization reactor, J. Proc. Cont., vol. 6, n. 2/3, pp. 119-127, 1996.

**DRIANKOV, D.**, An introduction to fuzzy control, Imprenta Berlin: Springer, 1996.

**ELLIS, M. F., TAYLOR, T. W. e JENSEN, K. F.**, On-line molecular weight distribution estimation and control in batch polymerization, *AICHE Journal*, vol. 40, n. 3, pp. 445-462, 1994.

**FELISBERTI, M. I.**, Comunicação verbal, 2004.

**FODIL, M. S., et al.**, A fuzzy rule base for the improved control of a pressurized water nuclear reator, *IEEE TRANSACTIONS ON FUZZY SYSTEMS*, vol. 8, n. 1, 2000, pp. 1-10.

**FOULLY, L. e GALICHET, S.**, Fuzzy control with fuzzy inputs, *IEEE Transactions on Fuzzy Systems*, vol. 11, n. 4, 2003, pág. 437-449.

**GHOSH, P., GUPTA, S. K. e SARAF, D. N.**, An experimental study on bulky and solution polymerization of methyl methacrylate with response to step changes in temperature, *Chemical Engineering Journal*, 70, pág. 25-35, 1998.

**GOMIDE, F. A. C. e GUDWIN, R. R.**, Modelagem, controle, sistemas e lógica fuzzy, *SBA Controle & Automação*, vol. 4, n. 3, pp. 97-115, 1994.

**HIROTA, K. e SUGENO, M.**, Industrial applications of fuzzy technology in the world, *World Scientific Publishing*, 1995.

**JONAS, B.**, Usando Foundation Fieldbus em aplicações híbridas e de batelada, *InTech*, n. 49, pág. 12-17.

**JEONG, B. e RHEE, H.**, Experimental validation of the optimal trajectory of initiator concentration in batch PMMA polymerization, *Journal of Applied Polymer Science*, v. 78, n. 6, pág. 1256-1266, 2000.

**KIPARISSIDES, C.**, Polymerization reactor modeling: a review of recent developments and future directions, *Chemical Engineering Science*, vol. 51, n. 10, pp. 1637-1659, 1996.

**KUMES, M. e SAUTER, T.**, Fieldbus-internet connectivity: the SMNP approach, *IEEE Transactions on Industrial Electronics*, vol. 48, n. 6, 2001, pág. 1248-1256.

**LEE, C. C.**, Fuzzy logic in control system: fuzzy logic controller - Part I, *IEEE Trans. on Sys., Man and Cybernetics*, vol. 20, n. 2, pp. 404-418, 1990a.

**LEE, C. C.**, Fuzzy logic in control system: fuzzy logic controller - Part II, IEEE Trans. on Sys., Man and Cybernetics, vol. 20, n. 2, pp. 419-435, 1990b.

**LI, H. e GATLAND, H. B.**, Conventional fuzzy control and its enhancement, IEEE Transaction os systems, Man and Cybernetics - Part B: Cybernetics, vol. 26, n. 5, pp. 791-797, 1996.

**LUIZ, C. C., SILVA, A. C., COELHA, A. A. R., BRUCIAPAGLIA, A. H.**, Controle adaptativo versus controle fuzzy: um estudo de caso em um processo de nível, SBA Controle & Automação, vol. 8, n. 2, pág. 43-50, 1997.

**LUMPP, T., GRUHLER, G., KÜCHLIN, W.**, Virtual java devices - Integration of fieldbus based systems in the internet, 24th Annual Conference of the IEEE Industrial Electronics Society (Proceedings), vol. 1, 1998, pág. 176-181.

**MANGANO, J. M. e DUNDIE, D. P.**, Foundation Fieldbus has arrived and is here to stay. In Tech, v. 45(2) Feb., p. 45-9, 1998.

**MANO, E. B.**, Introdução a polímeros, Ed. Edgard Blücher, 1985a, pag. 32.

**MANO, E. B.**, Introdução a polímeros, Ed. Edgard Blücher, 1985b, pag. 39.

**NUNES, R. L.**, Supervisão e controle "on line" de um reator de poimerização em escala piloto, UNICAMP, Dissertação (Mestrado), Campinas, 2003.

**O'DRISCOLL, K. F. e PANNUSWAMY, S. R.**, J. Appl. Polym Sci., 93, 1299, 1990.

**ODENTUNJI, O. A., KEHINDE, O. O.**, Computer simulation of fuzzy control systems fot gari fermentation plant, Journal of Food Engineering, versão corrigida, disponível on-line julho 2004.

**OLIVEIRA Jr., H. A.**, Lógica difusa: aspectos práticos e aplicações, Ed. Interciência, 1999, pág. 95.

**PAN, Y. e LEE, H. J.**, Recursive data-based prediction and control of product quality for a PMMA batch process, Chemical Engineering Science, 58, 2003, pág. 3215-3221.

**PEDRYCZ, W.**, Fuzzy control and fuzzy systems, J.Wiley, 1989.

**PINCETI, P.**, Fieldbus: more than a communication link, IEEE Instrumentation & Measurement Magazine, março, 2004, pág. 17-23.

**PRZYBYLSKI, F. J.**, Foundation Fieldbus enables the realization of the DCS full potencial. Advances in Instrumentation and Control, v. 51(1), p. 791-802, 1996.

**ROFFEL, B. e CHIN, P. A.**, Fuzzy control of a poymerization reactor, Hydrocarbon Processing, pp. 47-49, Junho, 1991.

**SHAW, I. S. e SIMÕES, M. G.**, Controle e modelagem fuzzy, Editora Edgard Blucher, 1999.

**SHAW, I. S.**, Fuzzy control of industrial systems: theory and applications, Imprenta Boston, 1998

**SOROUGH M., e KRAVARIS, C.**, Optimal design and operation of batch reactors. 2. A case study, Eng. Chem. Res., 32, pp. 882-893, 1993.

**SUGENO, M.** (editor), Industrial applications of fuzzy control, Elsevier Science Publishers, 1985.

**THOMAS, I. M. e KIPARISSIDES, C.**, Can. J. Chem. Eng., 62, 284, 1984.

**TONG, S., LI, H. e WANG, W.**, Observer-based adaptative fuzzyontrol for SISO nonlinear systems, Fuzzy Sets and Systems, versão orrigida, disponível on-line abril, 2004.

**VOGEL, A. I.**, Química orgânica qualitativa, Ao Livro Técnico, 1986.

**ZHOU, S. FENG, G. e FENG, C.**, Robust control for a class of uncertain nonlinear systems: adaptive fuzzy approach based on backstepping, Fuzzy Sets and Systems, versão corrigida, disponível on-line julho 2004.

# Apêndice A

## Materiais usados

### Produtos químicos:

- Solvente acetato de etila p.a., 99,9 % de pureza, marca Mallinckrodt;
- Iniciador termossensível peróxido de benzofila, 70 % de pureza, marca Aldrich Chemical Company, Inc.;
- Monômero metacrilato de metila (MMA) com 10 a 100 ppm de Metilhidroquinona-MeHQ (inibidor de polimerização para estocagem), 99,0 % de pureza, marca Aldrich Chemical Company, Inc.;
- Hidroquinona p.a., 99,0 % de pureza, marca Labsynth Produtos para Laboratório Ltda;
- Água (fluido de resfriamento);
- Nitrogênio gasoso com 99,9999 % de pureza, marca White Martins.

### Reator:

- Reator batelada de vidro encamisado, com capacidade para 1 litro de mistura reacional e 1 litro de fluido de resfriamento em sua jaqueta, descarga pelo fundo e alimentação pelo topo (Figura 3.1);
- Três termopares (0 - 150 °C, sinal de 4 a 20 mA);
- Reservatório de água de 50 litros, nível constante para normalizar a vazão de água de resfriamento para VC;
- Válvula de Controle (VC) analógica, do tipo agulha com entrada de sinal de 0 a 5 V para regular a abertura de 0 a 100 %;

- Densímetro "on-line" Modelo Promass 63 marca Endress+Hauser, cujo princípio de funcionamento baseia-se na geração controlada de forças de tração e rotação, saída analógica de 4 a 20 mA para 0,8200 a 1,200 kg/L;
- Duas resistências de aquecimento de 75 e 45 Watts cada uma, com sistema de segurança para desarme via software;
- Válvulas, reguladores de pressão e tubulação para aplicação do nitrogênio ao reator;
- Balança digital e equipamentos comuns de laboratório;
- Bomba de deslocamento.

### **Protocolo de comunicação Fieldbus Foundation:**

- DFI 302 Smar(R) controlador e sistema host Fieldbus Foundation, é um equipamento multifunção modular montado em um backplane, conectado em um trilho DIN, onde todos os componentes são instalados;
- Módulo Fonte DF 50 - Fonte padrão de 24 VDC de alta performance;
- Módulo Processador DF 51 - Baseado em um processador 32-bit RISC e programa armazenado em Flash, este módulo manipula comunicação e tarefas de controle;
- Módulo DF 44 - 8 Entradas analógicas (analog inputs-AI);
- Módulo DF 46 - 4 Saídas analógicas (analog outputs-AO);
- Módulo DF 28 - 16 Saídas digitais (digital outputs-DO);
- Rede Windows NT(R) 4.0;
- Hub de 8 portas;
- Softwares: Syscon(R), TagViewer(R) configuração física do sistema e leituras das tags da rede, interface entre hardware e software.
- Software OPC control 1.0 – programado em Visual Basic(R) - software desenvolvido durante este trabalho para processar rotinas de aquisição de dados e controle a partir de informações das variáveis disponibilizadas na rede (tags).

# Apêndice B

## Código fonte

```
Dim TempIn As Double, TempReac As Double, TempOut As Double, _  
VoltValv As Double  
  
Dim densidade As Double, instante As Integer, p As Integer, _  
Cod1 As String  
  
Dim Erro(3) As Single, Vazao As Single, Conv As Single  
  
Dim LD_Aquec As Integer, LD_Bomba As Integer, TempReacFilt As  
Double, _  
TempReacFiltAnt(5) As Double  
  
Dim cont_bomba As Integer  
  
Dim saida_1_ant As Single  
  
  
Const saida_2_ee = 12  
  
  
'Parâmetros para o controle  
  
Const VazaoSP = 15  
  
Const TempSP = 63  
  
Const Kc = -1.5  
  
Const Taul = 130  
  
Const TauD = 20  
  
Const T = 60 'Período de amostragem
```

```
Private Sub Command1_Click()  
Opodata1.OPCProgID = Text2.Text  
Opodata1.NodeName = Text1.Text  
Text3.Text = "Acessando..."  
Opodata1.Connect  
Text3.Text = Opodata1.ConnectStatus  
Text4.Text = Opodata1.ConnectHR  
End Sub
```

```
Private Sub Command2_Click()  
Opodata1.Disconnect  
Text3.Text = Opodata1.ConnectStatus  
End Sub
```

```
Private Sub Command3_Click()  
Cod1 = "D:\Saida\" + Text26.Text + ".txt" 'Abre arquivo de saida  
Open Cod1 For Output As #1  
Opodata1.ItemID(0) = Text5.Text  
Opodata1.ItemID(1) = Text6.Text  
Opodata1.ItemID(2) = Text7.Text  
Opodata1.ItemID(3) = Text8.Text  
Opodata1.ItemID(4) = Text32.Text  
Opodata1.ItemID(5) = Text29.Text  
Opodata1.ItemID(6) = Text36.Text
```

```
Text13.Text = "Acessando..."  
Opccdata1.CreateGroup  
Text13.Text = Opccdata1.GroupStatus  
Text14.Text = Opccdata1.GroupHR  
Text9.Text = Opccdata1.AddError(0)  
Text10.Text = Opccdata1.AddError(1)  
Text11.Text = Opccdata1.AddError(2)  
Text12.Text = Opccdata1.AddError(3)  
Text31.Text = Opccdata1.AddError(4)  
Text33.Text = Opccdata1.AddError(5)  
Text37.Text = Opccdata1.AddError(6)
```

```
Opccdata1.Update  
tempo$ = Time$  
TempIn = Opccdata1.Value(0)  
Text15.Text = TempIn  
TempReac = Opccdata1.Value(1)  
Text16.Text = TempReac  
TempOut = Opccdata1.Value(2)  
Text17.Text = TempOut  
densidade = Opccdata1.Value(3)  
Text18.Text = densidade  
VoltValv = Opccdata1.Value(4)  
Text28.Text = VoltValv  
LD_Bomba = Opccdata1.Value(5)
```

```
Text34.Text = LD_Bomba
LD_Aquec = Opccdata1.Value(6)
Text38.Text = LD_Aquec
Text23.Text = Opccdata1.UpdateStatus
Text24.Text = Opccdata1.UpdateHR
Text19.Text = Opccdata1.ReadError(0)
Text20.Text = Opccdata1.ReadError(1)
Text21.Text = Opccdata1.ReadError(2)
Text22.Text = Opccdata1.ReadError(3)
Text27.Text = Opccdata1.ReadError(4)
Text35.Text = Opccdata1.ReadError(5)
Text39.Text = Opccdata1.ReadError(6)
Text25.Text = instante
```

```
Timer1.Enabled = True
Timer2.Enabled = True
cont_bomba = 0
Timer3.Enabled = True
```

```
End Sub
```

```
Private Sub Command4_Click()
Opccdata1.DeleteGroup
Text13.Text = Opccdata1.GroupStatus
End Sub
```

```
Private Sub Command5_Click()
    Opccdata1.Update
    tempo$ = Time$
    instante = 0
    Text15.Text = Opccdata1.Value(0)
    Text16.Text = Opccdata1.Value(1)
    Text17.Text = Opccdata1.Value(2)
    Text18.Text = Opccdata1.Value(3)
    Text23.Text = Opccdata1.UpdateStatus
    Text24.Text = Opccdata1.UpdateHR
    Text19.Text = Opccdata1.ReadError(0)
    Text20.Text = Opccdata1.ReadError(1)
    Text21.Text = Opccdata1.ReadError(2)
    Text22.Text = Opccdata1.ReadError(3)
    Text25.Text = instante
    Timer1.Enabled = True
End Sub
```

```
Private Sub Command6_Click()
    Close #1
    Close #3
End
End Sub
```

```
Private Sub Form_Load()
```

```
Timer1.Enabled = False
```

```
Timer2.Enabled = False
```

```
Timer3.Enabled = False
```

```
instante = 0
```

```
End Sub
```

```
Private Sub Timer1_Timer()
```

```
Opodata1.Update
```

```
tempo$ = Time$
```

```
If (instante = 0) Then
```

```
    Vazao = 1.897 + 5.6315 * VoltValv
```

```
    For p = 1 To 3
```

```
        Erro(p) = 0
```

```
    Next p
```

```
saida_1_ant = 12
```

```
End If
```

```
TempIn = Opodata1.Value(0)
```

```
Text15.Text = TempIn
```

```
TempReac = Opodata1.Value(1)
```

```
Text16.Text = TempReac
```

```
TempOut = Opodata1.Value(2)
```

```
Text17.Text = TempOut
```

```

'Ler densidade somente quando bomba é ligada
If ((instante Mod 3) = 0) Then
    densidade = Opccdata1.Value(3)
    Text18.Text = densidade
End If

'Inicializacao do vetor de temperaturas filtradas
If ((instante = 0) And (OptSim = True)) Then
    For p = instante To 4
        TempReacFiltAnt(p + 1) = TempReac
    Next p
End If

'Atualização do vetor das temperaturas filtradas e cálculo da
'temperatura filtrada
If (OptSim = True) Then
    TempReacFiltAnt(5) = TempReacFiltAnt(4)
    TempReacFiltAnt(4) = TempReacFiltAnt(3)
    TempReacFiltAnt(3) = TempReacFiltAnt(2)
    TempReacFiltAnt(2) = TempReacFiltAnt(1)
    If (instante > 0) Then
        TempReacFiltAnt(1) = TempReacFilt
    End If
    TempReacFilt = 0.4 * TempReac + 0.2 * TempReacFiltAnt(1) + _
0.1 * TempReacFiltAnt(2) + 0.1 * TempReacFiltAnt(3) + _
0.1 * TempReacFiltAnt(4) + 0.1 * TempReacFiltAnt(5)
End If

```

```

Erro(3) = Erro(2)
Erro(2) = Erro(1)
If (OptSim = True) Then
    Erro(1) = TempSP - TempReacFilt
Else
    Erro(1) = TempSP - TempReac
End If
If (OptPID.Value = True) Then
    'Controle PID na forma velocidade
    Vazao = Vazao + Kc * (1 + (T / Taul) + (TauD / T)) * Erro(1) _
    - Kc * (1 + 2 * (TauD / T)) * Erro(2) + Kc * (TauD / T) * Erro(3)
End If

If (OptFuzzy.Value = True) Then
    Call fuzzy(Erro(1), Erro(2))
End If

If (Vazao < 5.55) Then
    Vazao = 5.5
End If

If (Vazao > 24.3) Then
    Vazao = 24.4
End If

```

```

If (Vazao < 5.55) Then
    VoltValv = 1.25
Else
    If (5.55 <= Vazao And Vazao <= 11.1) Then
        VoltValv = -0.563 + 0.325 * Vazao
    Else
        If (11.1 <= Vazao And Vazao <= 24.3) Then
            VoltValv = 1.133 + 0.125 * Vazao
        Else
            VoltValv = 4.2
        End If
    End If
End If

If ((1.25 <= VoltValv) And (VoltValv <= 4.2)) Then
    Opdata1.Value(4) = VoltValv
Else
    If (VoltValv < 1.25) Then
        VoltValv = 1.25
        Opdata1.Value(4) = VoltValv
    Else
        VoltValv = 4.2
        Opdata1.Value(4) = VoltValv
    End If

```

End If

Opcdata1.Update

Text28.Text = Opcdata1.Value(4)

Text23.Text = Opcdata1.UpdateStatus

Text24.Text = Opcdata1.UpdateHR

Text19.Text = Opcdata1.ReadError(0)

Text20.Text = Opcdata1.ReadError(1)

Text21.Text = Opcdata1.ReadError(2)

Text22.Text = Opcdata1.ReadError(3)

Text27.Text = Opcdata1.ReadError(4)

If (OptSim = True) Then

Print #1, tempo\$, instante \* T, Format(densidade, "#.####"), \_  
Format(TempReac, "#.###"), Format(TempIn, "#.###"), \_  
Format(TempOut, "#.###"), Format(VoltValv, "#.###"), LD\_Aquec, \_  
Format(TempReacFilt, "#.###")

Else

Print #1, tempo\$, instante \* T, Format(densidade, "#.####"), \_  
Format(TempReac, "#.###"), Format(TempIn, "#.###"), \_  
Format(TempOut, "#.###"), Format(VoltValv, "#.###"), LD\_Aquec

End If

instante = instante + 1

Text25.Text = instante

End Sub

Private Sub Timer2\_Timer() ' acionamento da bomba de 3 em 3 minutos

If ((cont\_bomba Mod 2) = 0) Then

    If (((cont\_bomba + 6) Mod 6) = 0) Then

        LD\_Bomba = 1

    End If

    Timer2.Interval = 10000

Else

    LD\_Bomba = 0

    Timer2.Interval = 50000

End If

cont\_bomba = cont\_bomba + 1

Opodata1.Value(5) = LD\_Bomba

Opodata1.Update

Text34.Text = Opodata1.Value(5)

Text35.Text = Opodata1.ReadError(5)

End Sub

Private Sub Timer3\_Timer()

    If (instante = 0) Then

        Timer3.Interval = 60000

    End If

    ' NOVO

    Opodata1.Update

```

Static d_erro_n(3) As Single, d_erro_z(5, 4) As Single, _
d_erro_p(3) As Single
Static acao_n(2, 3) As Single, acao_z(2, 5, 4) As Single, _
acao_p(2, 3) As Single
Dim cont As Integer, total_pontos As Integer, _
x_pontos(40) As Single, y_pontos(40) As Single
Dim x_inters(6) As Single, y_inters(6) As Single
Dim dx As Single, x As Single, y As Single, soma_num As Double, _
soma_dem As Double
Dim saida(2) As Single
Dim ke As Single, kd As Single, k1 As Single, k2 As Single

If (OptCard3.Value = True) Then
    NumCard = 3
    Open "C:\Arnaldo\fig_iniciais23.txt" For Input As #4
Else
    If (OptCard5.Value = True) Then
        NumCard = 5
        Open "C:\Arnaldo\fig_iniciais25.txt" For Input As #4
    Else
        NumCard = 7
        Open "C:\Arnaldo\fig_iniciais27.txt" For Input As #4
    End If
End If
End If

```

```

Input #4, erro_n(1), erro_n(2), erro_n(3)
For n = 1 To (NumCard - 2)
    Input #4, erro_z(n, 1), erro_z(n, 2), erro_z(n, 3), erro_z(n, 4)
Next n
Input #4, erro_p(1), erro_p(2), erro_p(3)

Input #4, d_erro_n(1), d_erro_n(2), d_erro_n(3)
For n = 1 To (NumCard - 2)
    Input #4, d_erro_z(n, 1), d_erro_z(n, 2), d_erro_z(n, 3), _
d_erro_z(n, 4)
Next n
Input #4, d_erro_p(1), d_erro_p(2), d_erro_p(3)

Input #4, acao_n(1, 1), acao_n(1, 2), acao_n(1, 3)
For n = 1 To (NumCard - 2)
    Input #4, acao_z(1, n, 1), acao_z(1, n, 2), acao_z(1, n, 3), _
acao_z(1, n, 4)
Next n
Input #4, acao_p(1, 1), acao_p(1, 2), acao_p(1, 3)

Input #4, acao_n(2, 1), acao_n(2, 2), acao_n(2, 3)
For n = 1 To (NumCard - 2)
    Input #4, acao_z(2, n, 1), acao_z(2, n, 2), acao_z(2, n, 3), _
acao_z(2, n, 4)
Next n

```

Input #4, acao\_p(2, 1), acao\_p(2, 2), acao\_p(2, 3)

For n = 1 To (NumCard \* NumCard)

Input #4, tabela(1, n)

Next n

For n = 1 To (NumCard \* NumCard)

Input #4, tabela(2, n)

Next n

Close #4

ke = 1

kd = 1

k1 = -0.03

k2 = 1

e = ke \* erro1

d\_e = kd \* (erro1 - erro2)

' os dois modos do loop correspondem ao pi, forma velocidade,(modo = 1)

' e pd(modo = 2), forma posição.

For modo = 1 To 2

For n = 1 To NumCard

ativacao(n) = 0

Next n

'cálculo da função de pertinência do erro

```

fp_e(1) = trap_esq(e, erro_n())
For n = 2 To (NumCard - 1)
    For i = 1 To 4
        vetor_aux(i) = erro_z(n - 1, i)
    Next i
    fp_e(n) = triang(e, vetor_aux())
Next n
fp_e(NumCard) = trap_dir(e, erro_p())

```

'cálculo da função de pertinência do delta erro

```

fp_d_e(1) = trap_esq(d_e, d_erro_n())
For n = 2 To (NumCard - 1)
    For i = 1 To 4
        vetor_aux(i) = d_erro_z(n - 1, i)
    Next i
    fp_d_e(n) = triang(d_e, vetor_aux())
Next n
fp_d_e(NumCard) = trap_dir(d_e, d_erro_p())

```

'verificação das regras (Mandani) que são válidas

```

For i = 1 To NumCard
    For j = 1 To NumCard
        If (fp_e(j) > 0.001 And fp_d_e(i) > 0.001) Then
            ativacao(tabela(modos, j + NumCard * (i - 1))) = _
max(ativacao(tabela(modos, j + NumCard * (i - 1))), _

```

```
* acao_z(modo, n - 1, 4) / (acao_z(modo, n - 1, 2) - _  
acao_z(modo, n - 1, 3))
```

```
Next n
```

```
End If
```

```
x_inters(NumCard - 1) = (acao_z(modo, NumCard - 2, 3) - _  
acao_p(modo, 1) * (acao_p(modo, 3) / (acao_p(modo, 2) - _  
acao_p(modo, 1))) * (acao_z(modo, NumCard - 2, 2) - _  
acao_z(modo, NumCard - 2, 3)) / acao_z(modo, NumCard - 2, 4))) / _  
(1 - (acao_p(modo, 3) / (acao_p(modo, 2) - acao_p(modo, 1))) * _  
(acao_z(modo, NumCard - 2, 2) - acao_z(modo, NumCard - 2, 3)) / _  
acao_z(modo, NumCard - 2, 4)))
```

```
y_inters(NumCard - 1) = (x_inters(NumCard - 1) - _  
acao_z(modo, NumCard - 2, 3)) * acao_z(modo, NumCard - 2, 4) / _  
(acao_z(modo, NumCard - 2, 2) - acao_z(modo, NumCard - 2, 3))
```

```
'determinação das coordenadas dos pontos que formaram o perfil de  
'defuzificação
```

```
cont = 1
```

```
'ponto inicial
```

```
x_pontos(cont) = acao_n(modo, 1)
```

```
y_pontos(cont) = ativacao(1)
```

```
cont = cont + 1
```

'determinação dos pontos entre as regiões N e Z

If (ativacao(1) > y\_inters(1)) Then

    x\_pontos(cont) = acao\_n(modos, 2) + ativacao(1) \* \_  
(acao\_n(modos, 1) - acao\_n(modos, 2)) / acao\_n(modos, 3)

    y\_pontos(cont) = ativacao(1)

    cont = cont + 1

If (ativacao(2) > y\_inters(1)) Then

    x\_pontos(cont) = x\_inters(1)

    y\_pontos(cont) = y\_inters(1)

    cont = cont + 1

    x\_pontos(cont) = acao\_z(modos, 1, 1) + ativacao(2) \* \_  
(acao\_z(modos, 1, 2) - acao\_z(modos, 1, 1)) / acao\_z(modos, 1, 4)

    y\_pontos(cont) = ativacao(2)

    cont = cont + 1

Else

    x\_pontos(cont) = acao\_n(modos, 2) + ativacao(2) \* \_  
(acao\_n(modos, 1) - acao\_n(modos, 2)) / acao\_n(modos, 3)

    y\_pontos(cont) = ativacao(2)

    cont = cont + 1

End If

End If

If (ativacao(1) <= y\_inters(1)) Then

    If (ativacao(1) <= ativacao(2)) Then

```
    x_pontos(cont) = acao_z(modos, 1, 1) + ativacao(1) * _  
(acao_z(modos, 1, 2) - acao_z(modos, 1, 1)) / acao_z(modos, 1, 4)
```

```
    y_pontos(cont) = ativacao(1)
```

```
    cont = cont + 1
```

```
    x_pontos(cont) = acao_z(modos, 1, 1) + ativacao(2) * _  
(acao_z(modos, 1, 2) - acao_z(modos, 1, 1)) / acao_z(modos, 1, 4)
```

```
    y_pontos(cont) = ativacao(2)
```

```
    cont = cont + 1
```

```
Else
```

```
    x_pontos(cont) = acao_n(modos, 2) + ativacao(1) * _  
(acao_n(modos, 1) - acao_n(modos, 2)) / acao_n(modos, 3)
```

```
    y_pontos(cont) = ativacao(1)
```

```
    cont = cont + 1
```

```
    x_pontos(cont) = acao_n(modos, 2) + ativacao(2) * _  
(acao_n(modos, 1) - acao_n(modos, 2)) / acao_n(modos, 3)
```

```
    y_pontos(cont) = ativacao(2)
```

```
    cont = cont + 1
```

```
End If
```

```
End If
```

```
'determinação dos pontos entre as regiões intermediárias
```

```
If (NumCard > 3) Then
```

```
    For n = 2 To (NumCard - 2)
```

```

If (ativacao(n) > y_inters(n)) Then
    x_pontos(cont) = acao_z(modos, n - 1, 3) + _
ativacao(n) * (acao_z(modos, n - 1, 2) - acao_z(modos, n - 1, 3)) / _
acao_z(modos, n - 1, 4)
    y_pontos(cont) = ativacao(n)
    cont = cont + 1
If (ativacao(n + 1) > y_inters(n)) Then
    x_pontos(cont) = x_inters(n)
    y_pontos(cont) = y_inters(n)
    cont = cont + 1

    x_pontos(cont) = acao_z(modos, n, 1) + _
ativacao(n + 1) * (acao_z(modos, n, 2) - acao_z(modos, n, 1)) / _
acao_z(modos, n, 4)
    y_pontos(cont) = ativacao(n + 1)
    cont = cont + 1
Else
    x_pontos(cont) = acao_z(modos, n - 1, 3) + _
ativacao(n + 1) * (acao_z(modos, n - 1, 2) - acao_z(modos, n - 1, 3)) / _
acao_z(modos, n - 1, 4)
    y_pontos(cont) = ativacao(n + 1)
    cont = cont + 1
End If
End If

```

```

If (ativacao(n) <= y_inters(n)) Then
    If (ativacao(n) <= ativacao(n + 1)) Then
        x_pontos(cont) = acao_z(modos, n, 1) + _
ativacao(n) * (acao_z(modos, n, 2) - acao_z(modos, n, 1)) / _
acao_z(modos, n, 4)
        y_pontos(cont) = ativacao(n)
        cont = cont + 1

        x_pontos(cont) = acao_z(modos, n, 1) + _
ativacao(n + 1) * (acao_z(modos, n, 2) - acao_z(modos, n, 1)) / _
acao_z(modos, n, 4)
        y_pontos(cont) = ativacao(n + 1)
        cont = cont + 1
    Else
        x_pontos(cont) = acao_z(modos, n - 1, 3) + _
ativacao(n) * (acao_z(modos, n - 1, 2) - acao_z(modos, n - 1, 3)) / _
acao_z(modos, n - 1, 4)
        y_pontos(cont) = ativacao(n)
        cont = cont + 1

        x_pontos(cont) = acao_z(modos, n - 1, 3) + _
ativacao(n + 1) * (acao_z(modos, n - 1, 2) - acao_z(modos, n - 1, 3)) / _
acao_z(modos, n - 1, 4)
        y_pontos(cont) = ativacao(n + 1)

```

```

        cont = cont + 1
    End If
End If

Next n
End If

'determinação dos pontos entre as regiões Z e P
If (ativacao(NumCard - 1) > y_inters(NumCard - 1)) Then
    x_pontos(cont) = acao_z(modulo, NumCard - 2, 3) + _
ativacao(NumCard - 1) * (acao_z(modulo, NumCard - 2, 2) - _
acao_z(modulo, NumCard - 2, 3)) / acao_z(modulo, NumCard - 2, 4)
    y_pontos(cont) = ativacao(NumCard - 1)
    cont = cont + 1
    If (ativacao(NumCard) > y_inters(NumCard - 1)) Then
        x_pontos(cont) = x_inters(NumCard - 1)
        y_pontos(cont) = y_inters(NumCard - 1)
        cont = cont + 1
        x_pontos(cont) = acao_p(modulo, 1) + ativacao(NumCard) * _
(acao_p(modulo, 2) - acao_p(modulo, 1)) / acao_p(modulo, 3)
        y_pontos(cont) = ativacao(NumCard)
        cont = cont + 1
    Else
        x_pontos(cont) = acao_z(modulo, NumCard - 2, 3) + _
ativacao(NumCard) * (acao_z(modulo, NumCard - 2, 2) - _

```

acao\_z(modo, NumCard - 2, 3)) / acao\_z(modo, NumCard - 2, 4)

y\_pontos(cont) = ativacao(NumCard)

cont = cont + 1

End If

End If

If (ativacao(NumCard - 1) <= y\_inters(NumCard - 1)) Then

If (ativacao(NumCard - 1) <= ativacao(NumCard)) Then

x\_pontos(cont) = acao\_p(modo, 1) + ativacao(NumCard - 1) \* \_  
(acao\_p(modo, 2) - acao\_p(modo, 1)) / acao\_p(modo, 3)

y\_pontos(cont) = ativacao(NumCard - 1)

cont = cont + 1

x\_pontos(cont) = acao\_p(modo, 1) + ativacao(NumCard) \* \_  
(acao\_p(modo, 2) - acao\_p(modo, 1)) / acao\_p(modo, 3)

y\_pontos(cont) = ativacao(NumCard)

cont = cont + 1

Else

x\_pontos(cont) = acao\_z(modo, NumCard - 2, 3) + \_  
ativacao(NumCard - 1) \* (acao\_z(modo, NumCard - 2, 2) - \_  
acao\_z(modo, NumCard - 2, 3)) / acao\_z(modo, NumCard - 2, 4)

y\_pontos(cont) = ativacao(NumCard - 1)

cont = cont + 1

x\_pontos(cont) = acao\_z(modo, NumCard - 2, 3) + \_

```

ativacao(NumCard) * (acao_z(modos, NumCard - 2, 2) - _
acao_z(modos, NumCard - 2, 3)) / acao_z(modos, NumCard - 2, 4)
    y_pontos(cont) = ativacao(NumCard)
    cont = cont + 1
End If
End If

'ponto final
x_pontos(cont) = acao_p(modos, 2)
y_pontos(cont) = ativacao(NumCard)

total_pontos = cont

'inicializar variáveis para o cálculo de vazao
dx = 0.05
soma_num = 0
soma_dem = 0

For cont = 1 To (total_pontos - 1)
    x = x_pontos(cont)
    If (Abs(x_pontos(cont) - x_pontos(cont + 1)) > 0.0001) Then
'evita divisão por zero em pontos coincidentes
        Do While (x <= x_pontos(cont + 1))
            y = y_pontos(cont) + (x - x_pontos(cont)) * _
(y_pontos(cont + 1) - y_pontos(cont)) / (x_pontos(cont + 1) - _

```

```
x_pontos(cont))
```

```
    soma_num = soma_num + y * x
```

```
    soma_dem = soma_dem + y
```

```
    x = x + dx
```

```
    Loop
```

```
End If
```

```
Next cont
```

```
    saida(modos) = soma_num / soma_dem
```

```
Next modos
```

```
Vazao = saida_1_ant + k1 * saida(1) * T + k2 * (saida_2_ee + saida(2))
```

```
saida_1_ant = -saida(1)
```

```
End Sub
```

```
Public Function trap_esq(x As Single, pontos() As Single)
```

```
    If (x <= pontos(1)) Then
```

```
        trap_esq = pontos(3)
```

```
    Else
```

```
        If (pontos(1) < x And x < pontos(2)) Then
```

```
            trap_esq = (x - pontos(2)) * pontos(3) / (pontos(1) - _
```

```
pontos(2))
```

```
        Else
```

```
            trap_esq = 0
```

```

    End If
End If
End Function

Public Function trap_dir(x As Single, pontos() As Single)
    If (x <= pontos(1)) Then
        trap_dir = 0
    Else
        If (pontos(1) < x And x < pontos(2)) Then
            trap_dir = (x - pontos(1)) * pontos(3) / (pontos(2) - _
pontos(1))
        Else
            trap_dir = pontos(3)
        End If
    End If
End Function

Public Function trap_centro(x As Single, pontos() As Single)
    If (x <= pontos(1)) Then
        trap_centro = 0
    Else
        If (pontos(1) < x And x < pontos(2)) Then
            trap_centro = (x - pontos(1)) * pontos(5) / (pontos(2) - _
pontos(1))
        Else

```

```

    If (pontos(2) <= x And x <= pontos(3)) Then
        trap_centro = pontos(5)
    Else
        If (pontos(3) < x And x < pontos(4)) Then
            trap_centro = (x - pontos(4)) * pontos(5) / _
(pontos(3) - pontos(4))
        Else
            trap_centro = 0
        End If
    End If
End If
End If
End Function

```

```

Public Function triang(x As Single, pontos() As Single)
    If (x <= pontos(1)) Then
        triang = 0
    Else
        If (pontos(1) < x And x < pontos(2)) Then
            triang = (x - pontos(1)) * pontos(4) / (pontos(2) - _
pontos(1))
        Else
            If (pontos(2) <= x And x < pontos(3)) Then
                triang = (x - pontos(3)) * pontos(4) / (pontos(2) - _
pontos(3))
            End If
        End If
    End If
End Function

```

```
Else
    triang = 0
End If
End If
End If
End Function
```

```
Public Function max(a As Single, b As Single)
```

```
    If (a >= b) Then
```

```
        max = a
```

```
    Else
```

```
        max = b
```

```
    End If
```

```
End Function
```

```
Public Function min(a As Single, b As Single)
```

```
    If (a <= b) Then
```

```
        min = a
```

```
    Else
```

```
        min = b
```

```
    End If
```

```
End Function
```



## Apêndice C

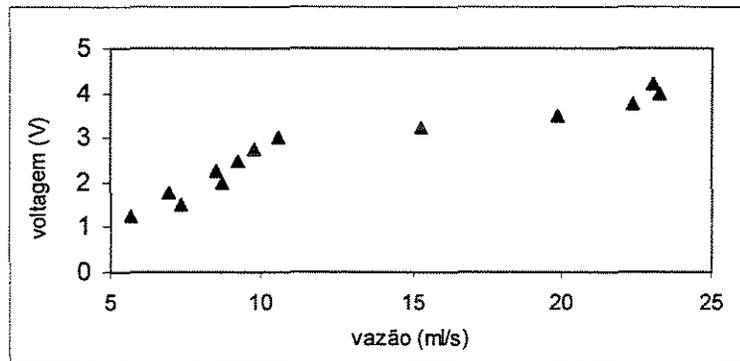
### Teste de estabilidade (MatLab) e Calibração da válvula

#### Teste de estabilidade

```
%  
% dados do processo  
%  
t = 60;      % período de amostragem  
taui = 187.21; % constante de tempo integral  
taud = 32.12; % constante de tempo derivativa  
kp = -0.033; % constante de ganho em estado estacionário  
tp = 93.3;   % constante de tempo do processo  
td = 106.7;  % atraso  
  
kc = [-50 -30 30 50 -25 -20 -15 -10 -5 -3 -1 0 1 3 5 10 15 20 25 30 50];  
  
num = [kp*((1-exp(-t/tp))*(1+(t/taui)+(td/t))) -kp*((1-exp(-  
t/tp))*(1+(2*(taud/t)))) kp*(1-exp(-t/tp))*(taud/t)];  
den = [1 -exp(-t/tp) 0 0 0];  
sys = tf(num, den, t);  
Rlocus(sys,kc);  
axis([-2 2 -2 2])
```

## Calibração da válvula

A calibração da válvula de controle é mostrada na figura abaixo.



Por questão de simplicidade, a equação de calibração é composta por duas faixas de variação aproximadamente linear, produzindo as equações abaixo:

Se Vazao < 5,55 mL/s: VoltValv = 1,25 V

Se 5,55 mL/s <= Vazao <= 11,1 mL/s: VoltValv = -0,563 +  
+ 0,325 \* Vazao (V)

Se 11,1 mL/s <= Vazao <= 24,3 mL/s: VoltValv = 1,133 +  
+ 0,125 \* Vazao (V)

Se Vazao > 24,3 mL/s: VoltValv = 4,2 V