

**UNIVERSIDADE SANTA CECÍLIA**  
**PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA**  
**MESTRADO EM ENGENHARIA MECÂNICA**

**KELLY CRISTINA ABOU ARABI DE MENDONÇA**

**CONTROLE DE RAZÃO ADAPTATIVA COM ALGORITMOS DA LÓGICA  
PARACONSISTENTE CONSTRUÍDA EM MICROCONTROLADOR**

**SANTOS/SP**

**2023**

**KELLY CRISTINA ABOU ARABI DE MENDONÇA**

**CONTROLE DE RAZÃO ADAPTATIVA COM ALGORITMOS DA LÓGICA  
PARACONSISTENTE CONSTRUÍDA EM MICROCONTROLADOR**

Dissertação apresentada a Universidade Santa Cecília como parte dos requisitos para obtenção de título de mestre no Programa de Pós-Graduação em Engenharia Mecânica, sob a orientação do Prof. Dr. Mauricio Conceição Mário e coorientação do Prof. Dr. João Inácio da Silva Filho

**SANTOS/SP**

**2023**

Autorizo a reprodução parcial ou total deste trabalho, por qualquer que seja o processo, exclusivamente para fins acadêmicos e científicos.

511.31 Mendonça, Kelly Cristina Abou Arabi de.  
M495c           CONTROLE DE RAZÃO ADAPTATIVA COM ALGORITMOS DA LÓGICA  
PARACONSISTENTE CONSTRUÍDA EM MICROCONTROLADOR. /

Kelly Cristina Abou Arabi de Mendonça.  
2023.  
61 f.

Orientador: Prof. Dr. Mauricio Conceição Mário.  
Coorientador: Prof. Dr. João Inácio da Silva Filho.

Dissertação (Mestrado) -- Universidade Santa Cecília,  
Programa de Pós-Graduação em Engenharia Mecânica, Santos,  
SP, 2023.

1. Controle de Razão. 2. Lógica Paraconsistente. 3.  
Controlador Programável. 4. Algoritmo Paraconsistente.  
I. Mário, Mauricio Conceição. II. Filho, João Inácio da Silva.  
III. CONTROLE DE RAZÃO ADAPTATIVA COM ALGORITMOS DA LÓGICA  
PARACONSISTENTE CONSTRUÍDA EM MICROCONTROLADOR

Elaborada pelo SIBi – Sistema Integrado de Bibliotecas - Unisanta

## DEDICATÓRIA

*Dedico este trabalho aos meus pais, irmãos, ao meu amor e minha família, em especial ao meu tio Youssef (in memoriam) que me apoiaram de diversas maneiras durante esta importante etapa de minha vida.*

## **AGRADECIMENTOS**

Em um momento de importância a minha vida profissional meus sinceros agradecimentos ao Prof. Dr. João Inácio da Silva Filho, por toda ajuda, orientação e ensinamentos necessários para o desenvolvimento deste trabalho. Agradeço também ao Prof. Dr. Mauricio Conceição Mario, pela orientação e recomendações lógicas no estudo e rápido retorno em todas as dúvidas.

Aos professores e coordenadores da graduação UNISANTA – Universidade Santa Cecília representando os mesmos.

Prof. Me. Luís Fernando Pompeo Ferrara

Profa. Me. Nilene Janini de Oliveira Seixas

E o diretor da Engenharia, Dr. Antônio de Salles Penteado

A todo o corpo docente por todo ensinamento durante ao compartilhamento de vossos conhecimentos passado durante a obtenção dos créditos entre eles cito-os.

Prof. Dr. Marcos Tadeu Tavares Pacheco

Prof. Dr. Vitor da Silva Rosa

Aos professores e amigos que me apoiaram, incentivaram e se empenharam em me manter motivada durante todo o transcorrer desses anos.

Profa. Me. Irene Silva Farias

Profa. Me. Raquel Galhardo de Carvalho Lopes Araújo

Prof. Yuri Silva Cruz Storino

E finalmente agradeço a Deus, pela saúde e graça concebida a mim.

## RESUMO

Apresenta-se nesta dissertação um desenvolvimento de controle adaptativo de razão baseado em Lógica Paraconsistente com implementação em uma placa eletrônica *open-source* Arduino que possui um microcontrolador ATmega328P. A Lógica Paraconsistente (LP) é uma lógica não clássica capaz de suportar sinais contraditórios e, em algumas situações, considera condições vantajosas em relação a lógica clássica binária. O controle de razão é configurado com dois controladores atuando com ações PID-Proporcional+Integral+Derivativo e vem sendo muito utilizado nas plantas industriais em processos de misturas. Um dos principais problemas neste tipo de controle é que apresentam desequilíbrios entre a comunicação dos controladores quando ocorrem mudanças no ajuste de *set point* e na variável controlada, ocasionando diferenças percentuais entre as quantidades, o que pode inviabilizar a qualidade do produto final. Diante dessas condições, nesta pesquisa é proposta uma Estação de Controle Adaptativo de Razão Paraconsistente (ECAR-LPA), portanto configurada com algoritmos fundamentados em LP compondo uma rede de análises de sinais paraconsistentes. Foi desenvolvido um controle adaptativo de possível ajustes no set point, no fator multiplicador de razão, nos ganhos proporcionais ( $K_{p1}$  e  $K_{p2}$ ) e nos tempos interativos ( $T_{i1}$  e  $T_{i2}$ ) dos controladores. A ECAR-LPA foi simulada em laboratório e interligada a uma planta piloto experimental que atua no controle de processo para um sistema de transferência e de mistura de líquidos. Os resultados obtidos a partir do modelo construído com a Lógica Paraconsistente são promissores e satisfatórios quando comparados com o modelo convencional, o que permite concluir que a realização destes experimentos contribui de maneira significativa na investigação do controle de razão de fluidos.

**Palavras-chave:** lógica paraconsistente, controlador programável, algoritmo paraconsistente, controle de razão.

## ABSTRACT

This dissertation presents a development of adaptive ratio control based on Paraconsistent Logic with implementation on an open-source Arduino electronic board that has an ATmega328P microcontroller. Paraconsistent Logic (LP) is a non-classical logic capable of supporting contradictory signals and, in some situations, considers advantageous conditions in relation to classical binary logic. The ratio control is configured with two controllers acting with PID-Proportional+Integral+Derivative actions and has been widely used in industrial plants in mixing processes. One of the main problems in this type of control is that there are imbalances between the communication of the controllers when there are changes in the set point adjustment and in the controlled variable, causing percentage differences between the quantities, which can make the quality of the final product unfeasible. Given these conditions, this research proposes a Paraconsistent Ratio Adaptive Control Station (ECAR-LPA), therefore configured with algorithms based on LP, composing a network of paraconsistent signal analysis. An adaptive control was developed with possible adjustments to the set point, the ratio multiplier factor, the proportional gains ( $K_{p1}$  and  $K_{p2}$ ) and the interactive times ( $T_{i1}$  and  $T_{i2}$ ) of the controllers. The ECAR-LPA was simulated in the laboratory and connected to an experimental pilot plant that acts in the process control for a liquid transfer and mixing system. The results obtained from a logic constructed with an experimental and conventional logic, when compared with the model of consistent results in a consistent and consistent way in the attempt to control the ratio.

**Keywords:** Paraconsistent Logic, programmable controller, Paraconsistent Algorithm, Flow Rate Control.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Diagrama de um controle de razão típico com a Estação de Razão aplicada ao fluxo principal $y_1$ .....	17
Figura 2 – Diagrama de um controle de razão com a Estação de Razão aplicada ao ponto de ajuste $r_1$ .....	18
Figura 3 – Diagrama de um controle de razão usando a estação de mistura aplicada ao set point $r_1$ e ao fluxo principal $y_1$ .....	19
Figura 4 – Diagrama de um controle de razão aplicado ao set point $r_1$ , fluxo principal $y_1$ e fluxo $y_2$ .....	20
Figura 5 – Estação de controle de razão paraconsistente - ECR-LPA com ação a no ponto de ajuste $r_1$ .....	23
Figura 6 – Reticulado de quatro vértices associado à Lógica Anotada Paraconsistente com anotação de dois valores (LPA2v) com deduções de equações utilizadas no NAP. ....	29
Figura 7 – (a) Símbolo de um nó típico de análise paraconsistente - NAP. (B) Representação gráfica do algoritmo extrator de grau de evidência. (c) Tipo de configuração RAP onde as proposições secundárias $p_1$ , $p_2$ geram os graus de evidência para a análise da proporção final $p_0$ .....	31
Figura 8 – Controle adaptativo de razão paraconsistente - ECAR-LPA com sinal do fluxo secundário $y_2$ . ....	32
Figura 9 – Visão geral da planta piloto de processo industrial usada nos testes de validação da estação de razão paraconsistente. ....	33
Figura 10 – a) configuração do módulo ECAR-LPA2v. b) Diagrama e instalação de uma Estação de Controle de Razão Paraconsistente (ECAR-LPA2v) em uma planta típica de controle de mistura.....	35
Figura 11 – Diagrama da Rede de Análise Paraconsistente (RAP) que compõe a Estação de Controle Adaptativo de Razão Paraconsistente (ECAR-LPA). ....	39

Figura 12 – Portas e pinos de tensão do microcontrolador Atmega328P.....	43
Figura 13 – Potenciômetros de ajustes dos <i>set-point</i> remoto e fator de razão (A). .....	44
Figura 14 – Display LCD 16x2 .....	44
Figura 15 – Estratégia de comunicação paraconsistente do ECAR-LPA com o <i>Matlab Simulink OPC Tool Box</i> .....	46
Figura 16 - Bloco de tratamento dos dados favoráveis e desfavoráveis.....	47
Figura 17 - Gráfico dos resultados do comportamento das variáveis de controle utilizando a estação de razão convencional na planta piloto experimental. .....	49
Figura 18 - Gráfico dos resultados do comportamento das variáveis de controle utilizando a estação de razão convencional na planta piloto experimental. .....	49
Figura 19 - Gráfico dos resultados do comportamento das variáveis de controle utilizando a Estação de Razão Adaptativa Paraconsistente (ECAR-LPA) na planta piloto experimental. ....	50
Figura 20 - Gráfico dos resultados do comportamento das variáveis de controle utilizando a Estação de Razão Adaptativa Paraconsistente (ECAR-LPA) na planta piloto.....	50

## LISTA DE SIGLAS

CP	Controlador Programável.
ECAR-LPA	Estação de controle Adaptativo de razão Paraconsistente
ECR-LP	Estação de controle de razão Paraconsistente
IDE	Interface Development Enviroment
IMC	Modelamento Interno do Controlador.
LP	Lógica Paraconsistente.
LPA	Lógica Paraconsistente Anotada.
LPA2v	Lógica Paraconsistente Anotada com anotação de dois valores.
NAP	Nó de Análise Paraconsistente.
PID	Proporcional-Integral-Derivativo contínuo
PDT	Transmissor de pressão diferencial.
PV	Variável de proceso.
PVyn	Variavel do transmissor de vazão do processo.
RAP	Rede de Análise Paraconsistente
SDCD	Sistema Digital de Controle Distribuído.
SPL	<i>Set point</i> local.
SPR	<i>Set point</i> remoto.
USB	Universal Serial Bus

## LISTA DE SIMBOLOS

$N$	Coeficiente de Filtro derivativo
$C$	Complemento
$\mathbb{R}$	Conjunto dos números reais
$r$	Constante de tempo do processo
$e(t)$	Erro do controlador
$E(s)$	Erro (SP-PV) entrada PI (s) contínuo paralelo
$F$	Falso
$B(s)$	Função de transferência planta processo A (vazão produto tanque A)
$H(s)$	Função de transferência planta processo B (vazão produto tanque B)
$D$	Ganho derivativo do controlador
$K$	Ganho do Processo
$I$	Ganho integral do controlador
$P$	Ganho proporcional do controlador
$K_p$	Ganho proporcional no controlador
$G_c$	Grau de Certeza
$G_{cr}$	Grau de Certeza Real
$G_{ct}$	Grau de Contradição
$\mu_{ctr}$	Grau de Contradição Normalizado
$\mu_1$	Grau de crença ou grau de evidência favorável FIT-101
$\mu_2$	Grau de descrença ou grau de evidência desfavorável FIT-102
$\lambda$	Grau de evidência desfavorável
$\mu$	Grau de evidência favorável
$\mu_{ER}$	Grau de Evidência Resultante Real
$T$	Inconsistente

$\varphi$	Intervalo de certeza
PT	Sentença proposicional inconsistente
PF	Sentença proposicional Falsa
P $\perp$	Sentença proposicional indeterminada
Pv	Sentença proposicional verdadeira
Ti	Tempo integral do controlador
$t$	Tempo morto do processo
$C(s)$	Transmissor de vazão tanque A
$J(s)$	Transmissor de vazão tanque B
$I(s)$	Valor da saída do controlador PI(s) contínuo paralelo - vazão do produto tanque A no instante "t"
$y(s)$	Valor da saída do controlador PI(s) contínuo paralelo - vazão do produto tanque B no instante "t"
$y_0$	Valor inicial da saída do controlador
$I_0$	Valor inicial da saída do controlador
V	Verdadeiro

## SUMÁRIO

<b>1. INTRODUÇÃO .....</b>	<b>13</b>
1.1 Justificativa e Relevância do Tema .....	15
1.2 Objetivo.....	16
1.2.1 Objetivo Geral .....	16
1.2.2 Objetivos específicos .....	16
1.3 Fundamentação Teórica .....	16
1.3.1 Controle de Razão .....	17
1.3.2 Noções Iniciais da Estação Paraconsistente para Controle de Razão.....	22
1.3.3 Lógica Clássica .....	23
1.3.4 Lógica Não Clássica .....	24
1.3.5 Lógica Paraconsistente.....	25
1.3.6 Logica Paraconsistente Anotada com anotação de dois valores (LPA2v).....	25
1.3.7 Algoritmo NAP – Nó de Análise Paraconsistente com saídas $\mu ER$ e $\mu ctr$ .....	29
1.3.8 Extrações de grau de evidência .....	30
1.3.9 Rede de Análise Paraconsistente – RAP.....	31
<b>2. MATERIAIS E MÉTODOS.....</b>	<b>32</b>
2.1 Noções Iniciais da Estação Paraconsistente Adaptativa para Controle de Razão .....	32
2.2 Detalhes da planta piloto planta piloto experimental usada nos testes de controle de razão.....	33
2.3 Determinações de graus de evidência para os sinais de entrada na Estação de Controle de Razão Paraconsistente (ECAR-LPA).....	34
2.4 Estrutura Algorítmica de uma Estação de Controle Adaptativa de Razão Paraconsistente (ECAR-LPA2v).....	34
2.5 Funcionamento básico da Estação de Controle de Razão Paraconsistente (ECAR-LPA) .....	36
2.6 Configuração do Controle de Razão Adaptativa Paraconsistente ECAR-LPA utilizada neste estudo .....	39

2.7	Determinação dos Fatores de Tuning para as configurações das Redes de Análise Paraconsistente da ECAR-LPA.....	41
2.8	Construção da Estação de razão adaptativa microcontrolada .....	43
2.7.1	Protótipo – <i>Hardware</i> .....	43
2.7.2	Arquitetura da Comunicação Paraconsistente ECAR-LPA .....	44
2.7.3	Bloco de aquisição de dados e Análise NAP no protótipo da ECAR-LPA implementado na plataforma Arduino .....	47
<b>3.</b>	<b>RESULTADOS E DISCUSSÃO .....</b>	<b>48</b>
3.1	Comparações de Resultados.....	52
<b>4.</b>	<b>CONCLUSÃO .....</b>	<b>54</b>
4.1	Trabalhos Futuros .....	54
	<b>REFERÊNCIAS.....</b>	<b>56</b>
	<b>ANEXO I – Código fonte da Rede de Análise Paraconsistente desenvolvida no IDE Arduino .....</b>	<b>59</b>
	<b>ANEXO II – Placa Arduino UNO R3. ....</b>	<b>62</b>

## 1. INTRODUÇÃO

A informatização vem sendo aplicada em grande escala pelas indústrias, tornando um diferencial logístico e produtivo. A automação presente trazida pela informatização está revolucionando as ferramentas de tomadas de decisões, apresentando conceitos novos, melhorando resultados, reduzindo custos, aumentando a qualidade de produção ou serviços e apresentando tecnologias ágeis (JUNIOR, 2006) (ROZA, 2017).

A origem da automação começou como controle manual, e conforme as implementações tecnológicas foram acontecendo, passou a ser implementadas em máquinas para auxiliar no controle manual. Passados os anos, o processo de controle automático avançou muito rápido e atualmente existe uma enorme variedade de equipamentos de medidas primárias, transmissores, controles pneumáticos, elétricos e eletrônicos, registradores de computação (relés analógicos), digitais com microprocessador, PLC's, SDCD's entre outros (FERNANDES, 2012).

Os computadores dedicados aos processos de automação são denominados de controladores industriais e no início surgiram dois tipos de controladores, o Controlador Lógico Programável (CLP), que é mais dedicado ao sistema de manufatura, e o Sistema Digital de Controle Distribuído (SDCD), sendo este último mais utilizado na indústria de processos contínuos. A rápida evolução da informática permitiu que logo houvesse uma convergência das duas tecnologias onde um mesmo computador pode ser configurado para reunir as mesmas funcionalidades, e este passou a ser denominado de CP – Controlador Programável.(FERNANDES, 2012).

A lógica clássica é binária e seus modelos compõem a base para os conceitos de toda a tecnologia que hoje conhecemos. Construídos com Lógica clássica, se pode citar: computadores, equipamentos eletrônicos, sistemas digitais, dentre outros. No entanto, a lógica clássica por ser binária, só trata informações considerando-as como verdadeiras ou falsas, não admitindo tratamentos de parcialidades verdadeiras ou falsas. Quando é necessária alta precisão para descrever modelos de sistemas reais fica mais difícil se estabelecer afirmativas ou negativas a respeito de qualidade das coisas utilizando a lógica clássica. No entanto, sabe-se que uma descrição precisa dos sistemas físicos leva a algoritmos mais eficientes. Diante disso, surgiram as Lógicas não clássicas para oferecer uma resposta mais adequada a problemas nos quais a lógica clássica é inoperante (DA SILVA FILHO, ABE e TORRES, 2008).

Segundo (ABE, 1992), as Lógicas não clássicas são aquelas que divergem em seus fundamentos de alguma das leis binárias que sustentam a lógica clássica.

No campo de pesquisas, o uso de Lógicas não clássicas tais como: lógica Fuzzy, lógica Modal e a lógica Paraconsistente, vem sendo aplicado com o intuito de promover avanços tecnológicos na indústria. Tendo em vista este caminho, é importante uma tecnologia paraconsistente envolvendo automação e controle com implementação de novos *softwares* de simulação, monitoramento de plantas industriais, controle e acionamento de processos (DA SILVA FILHO, ABE e TORRES, 2008).

A implementação da lógica não clássica paraconsistente requer alguns desafios pois é necessário lidar com a contradição. No processo lógico racional, a contradição existente nos bancos de dados do conhecimento incerto é imprescindível para extração de dados relevantes a serem tratados no processo de tomada de decisão. Diante disso, a Lógica Paraconsistente, suporta a contradição em seus fundamentos, e é uma boa ferramenta para tratar as contradições que antes, com os sistemas clássicos eram insignificantes, mas de sinais contraditórios pode-se extrair informações que tornam as análises mais eficientes (DA SILVA FILHO, ABE e TORRES, 2008).

Os controladores PID com ação proporcional-integral-derivativa são tradicionalmente utilizados para atuar no controle de processos e apresentam bom custo/benefício, sendo utilizados em uma ampla gama de processos industriais. Em alguns processos de controle que requerem maior complexidade, os controladores PID são conectados por meio de acoplamentos, resultando em tipos especiais de *loops*, como o controle de proporção (ou razão) (MARLIN, 2000) (OGATA, 2014) (ÅSTRÖM e TORE, 2006), que é o tipo de controle que trataremos nesta pesquisa.

O controle de proporção ou razão é classificado como um tipo de controle complexo onde o acoplamento é feito entre sistemas de controle simples de forma a manter uma razão constante entre duas variáveis de processo (ÅSTRÖM e TORE, 2006).

Na prática, o controle de razão é amplamente aplicado em processos industriais onde é necessário obter controle de dosagem para dois produtos. Como exemplo, podemos citar seu uso mais comum em sistemas de tratamento de água, onde a cloração é feita em recipientes de mistura. Também é amplamente utilizado nas indústrias química e farmacêutica que usam o controle de proporção em

procedimentos de mistura de produtos (MORARI, M. ZAFIRIOU, E., 1989) (HÄGGLUND, 2017).

Com as demandas do mercado, que deseja elevados níveis de qualidade no produto final, pesquisas têm sido desenvolvidas com o objetivo de trazer novas técnicas que visam aumentar a eficiência dos controladores e seu desempenho nas diferentes malhas de controle. Essas pesquisas incluem estudos e desenvolvimentos de novas técnicas que possam trazer inovações nos controles que utilizam acoplamentos básicos; como o controle de razão.

Neste trabalho apresenta-se a aplicação da Lógica Paraconsistente (LP) sendo implementada em um microcontrolador, onde um algoritmo, denominado Nó de Análise Paraconsistente (NAP), foi usado criando uma rede de análise paraconsistente RAP. A RAP, composta de algoritmos NAPs interligados, foi utilizada como ferramenta da tomada de decisão do controle de razão entre dois produtos em um sistema de automação de uma malha de processo industrial.

### **1.1 Justificativa e Relevância do Tema**

O controle da vazão é usado como um medidor de vazão. Se trata de um instrumento feito para visualizar e controlar diferentes condições do processo em que é ofertado o sinal analógico ou o sinal de frequência, é um processo muito importante em processos industriais, principalmente em fatores de segurança evitando acidentes, melhorando a qualidade do produto acabado, como também interagir nos cálculos de custos e no controle da demanda. Para Campos e Teixeira (2010) um dos controles mais importantes são os de níveis, pois para manter o nível do tanque ou vaso constante e proporcional é necessário que a vazão de entrada seja proporcionalmente igual na saída dando estabilidade no sistema. O controle de razão é utilizado em processos industriais quando requer manter a manutenção da proporção entre duas ou mais variáveis em um processo. Porém é comum haver inconsistências nesse processo e muitas vezes elas não são tratadas em sua lógica do processo adotado. Conforme descrito anteriormente pesquisas estão sendo desenvolvidas para se adotar uma técnica que mostre eficiente no controle de razão em malhas que controle vazão em dois produtos que se misturam nos tanques. Este trabalho apresenta uma forma diferente de tratamento de sinais desse tipo de controle fazendo com que o cálculo de razão seja efetuado através de sistemas microprocessados implementados

com algoritmos fundamentados em Lógica Paraconsistente. Portanto foi construída uma Estação de Controle Adaptativo de Razão Paraconsistente (ECAR-LPA) que pode atuar de forma mais eficiente com as incertezas, imprecisões e contradições no processo automatizado do controle de mistura.

## **1.2 Objetivo**

### **1.2.1 Objetivo Geral**

O objetivo geral da pesquisa é a construção com sistemas microcontrolados de uma Estação de Controle Adaptativo de Razão Paraconsistente (ECAR-LPA) a partir de uma rede de análise baseada em uma lógica não clássica, a Lógica Paraconsistente Anotada de Anotação de 2 valores – LPA2v, e melhorar a razão e sintonia entre dois fluxos em uma malha de processo industrial.

### **1.2.2 Objetivos específicos**

Os objetivos específicos abrangem a implementação de algoritmos fundamentados em Lógica Paraconsistente Anotada com Anotação de 2 valores (LPA2v), criando uma rede de análise paraconsistente capaz de tratar sinais de incertezas por meio de um controle PI(s) de razão em malha fechada conectado ao *Hardware* do controlador de processo. Para isto as seguintes etapas devem ser cumpridas:

- a) Aplicar o algoritmo da Lógica Paraconsistente, efetuar testes e simulações para obter o maior grau de precisão aos controladores que atuam diretamente na produção;
- b) Tratar os dados provenientes dos transmissores de uma estação experimental e desenvolver um controle capaz de oferecer resultados sob incertezas.
- c) Comparar os valores comportamentais das variáveis de controle em um processo convencional e utilizando o controle de razão adaptativo com a técnica LPA2v obtidos através da planta piloto experimental.

## **1.3 Fundamentação Teórica**

### 1.3.1 Controle de Razão

No processo de controle de razão, existem duas malhas de controle onde o objetivo é manter um valor constante da razão entre os valores de duas variáveis de processo  $y_1$  e  $y_2$ , mesmo sob condições transitórias, como aquelas que acontecem ao alterar o valor do ponto de ajuste ou distúrbios de origem externa (HÄGGLUND., 2001) (VISIOLI., 2005) (OU, YU e ZANG, 2007).

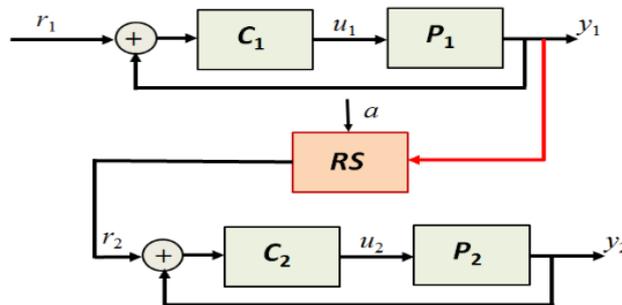


Figura 1 – Diagrama de um controle de razão típico com a Estação de Razão aplicada ao fluxo principal  $y_1$

Fonte: (HÄGGLUND., 2001)

A Figura 1 mostra o diagrama de um típico controle de razão, onde o loop principal consiste no processo  $P_1$  e no controlador  $C_1$ . A saída ( $y_1$ ) é o fluxo principal e o ponto de ajuste externo ( $r_1$ ) define o fluxo principal desejado. No segundo *loop*, que consiste no processo  $P_2$  e no controlador  $C_2$ , o controle de Razão tenta controlar o fluxo ( $y_2$ ) de forma que a fração  $y_2/y_1$  seja igual a um fator  $a$ . Para isso, existe uma Estação Razão (RS) onde o *set point* ( $r_2$ ) é representado pela equação 1 é determinado por (HÄGGLUND., 2001) (VISIOLI., 2005):

$$r_2(t) = ay_1(t) \quad (1)$$

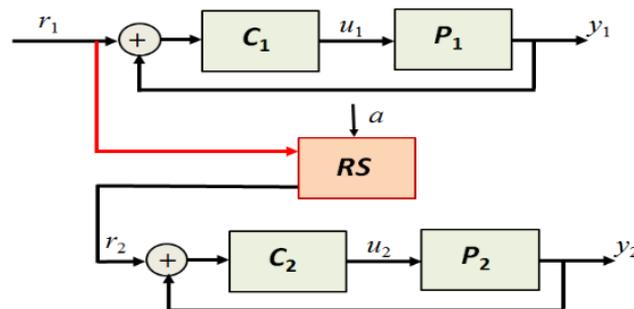
Enquanto o ganho do segundo *loop* for igual à unidade, a proporção satisfeita no estado estacionário é

$$\frac{y_1(t)}{y_2(t)} = a \quad (2)$$

Nessa configuração, como a saída  $y_2$  está necessariamente atrasada em relação a  $y_1$ , então sempre haverá um atraso na resposta transiente quando houver alteração no *Set-point*  $r_1$  (HÄGGLUND., 2001).

A duração desse atraso é determinada pela dinâmica do segundo *loop*. Quando o *Set Point* ( $r_1$ ) está aumentando, o atraso provoca fornecimento insuficiente do produto correspondente à vazão ( $y_2$ ); e, quando  $r_1$  está diminuindo, há excesso de oferta do produto correspondente ao fluxo ( $y_2$ ). O objetivo do controle de razão é manter a proporção entre duas variáveis, geralmente fluxos, em uma razão estabelecida pelo fator  $a$  (HÄGGLUND., 2001) (VISIOLI., 2005). Na prática, este controle não apresenta problemas quanto à proporção entre os dois produtos mantidos durante uma operação em regime permanente. Entretanto, um problema sério ocorre durante os transientes onde a Estação de Razão perde eficiência e não consegue manter a razão desejada no valor estabelecido pelo fator  $a$ . Este problema tem sérias consequências para o nível de qualidade do produto, uma vez que o controle de razão é normalmente aplicado a sistemas de produção onde os fluxos devem variar e as condições de regime permanente são incomuns (VISIOLI., 2005).

A abordagem sugerida em (HÄGGLUND., 2001) para superar problemas transientes representada pela figura 2, é usar o ponto de ajuste ( $r_1$ ) na estação de Razão ao invés do sinal de medição relacionado ao fluxo ( $y_1$ ).



**Figura 2 – Diagrama de um controle de razão com a Estação de Razão aplicada ao ponto de ajuste  $r_1$ .**

Fonte: (HÄGGLUND., 2001)

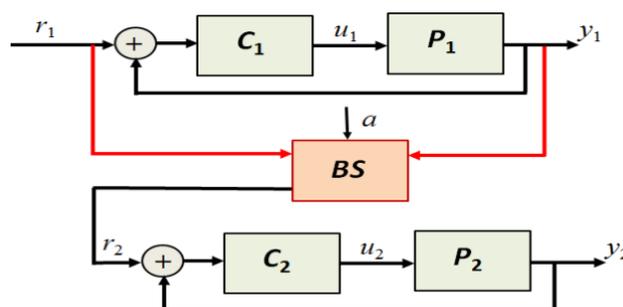
Nesta configuração mostrada na figura 2, o ponto de ajuste ( $r_2$ ) é determinado pela equação 3 (HÄGGLUND., 2001):

$$r_2(t) = ar_1(t) \quad (3)$$

Nesta configuração, o segundo fluxo não é necessariamente atrasado em relação ao fluxo principal, como na abordagem anterior, e o comportamento transiente é determinado pela dinâmica em ambos os loops. Portanto, ajustando os controladores para que as malhas obtenham a mesma dinâmica da malha fechada, a relação  $y_2 / y_1$  pode ser mantida conforme estabelecido pelo fator  $a$ , mesmo durante as mudanças do *Set Point* (HÄGGLUND, 2017) (HÄGGLUND., 2001).

Seguindo essa linha de raciocínio, os autores em (HÄGGLUND., 2001) propuseram uma mudança na estrutura da Estação de Razão (RS) a fim de melhorar o controle da razão nas condições transitórias, que acontecem com as variações do *set point*  $r_1$ , e nas variações de fluxos que acontecem nos processos.

A Figura 3 mostra o diagrama dessa nova abordagem de controle de razão que é apresentada com uma modificação por meio de um bloco multiplicativo especial denominado Estação de Mistura (BS).



**Figura 3 – Diagrama de um controle de razão usando a estação de mistura aplicada ao set point  $r_1$  e ao fluxo principal  $y_1$**

Fonte: (HÄGGLUND., 2001)

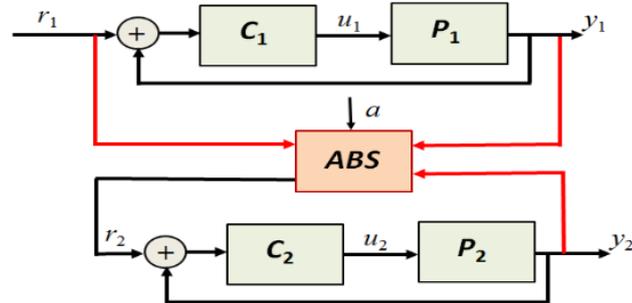
Como pode ser visto na figura 3 o bloco da estação de mistura apresentado em (HÄGGLUND., 2001) tinha como característica principal a modificação da estrutura de forma que o valor do *set point*  $r_2$  fosse dependente tanto do valor da saída do processo  $y_1$ , e quanto do ponto de ajuste  $r_1$ . Com isso, a expressão da Equação (3) passou a ser:

$$r_2(t) = a[\gamma r_1(t) + (1 - \gamma)y_1(t)] \quad (4)$$

Onde  $\gamma$  é um parâmetro constante que pesa a influência relativa do set point  $r_1$  em  $r_2$  em relação a  $y_1$ .

O valor de  $\gamma$  pode ser selecionado como a proporção da constante de tempo para os dois sistemas de malha fechada ou, alternativamente, aplicando um

procedimento adaptativo apropriado. Considerando esta última possibilidade, ainda no trabalho (HÄGGLUND., 2001) os autores introduziram uma configuração adaptativa para a estação de mistura, conforme mostrado na figura 4.



**Figura 4 – Diagrama de um controle de razão aplicado ao set point  $r_1$ , fluxo principal  $y_1$  e fluxo  $y_2$**

Fonte: (HÄGGLUND., 2001)

Nesta configuração, é proposto um procedimento adaptativo para obtenção do parâmetro  $\gamma$ , que deixa de ser ajustado pelo usuário. Para isso, a Estação adaptativa de Mistura recebeu outro sinal do fluxo ( $y_2$ ), portanto do processo  $P_2$ . Assim, na estação de mistura adaptativa, o fator  $\gamma$  é ajustado em linha com os valores reais dos dois fluxos  $y_1$  e  $y_2$ . Os autores também sugerem que o mecanismo de adaptação seja baseado na fórmula:

$$\frac{d\gamma}{dt} = \frac{S}{T_a} (a y_1 - y_2) \quad (5)$$

Onde  $S \in (-1,0,1)$  é um parâmetro de sinal que leva em consideração se a etapa do ponto de ajuste é positiva ou negativa.  $T_a$  é o valor da taxa de adaptação considerada como um fator vezes o tempo completo mais longo dos dois *loops*. O tempo integral  $T_i$  determina a taxa de adaptação. O seu valor é encontrado automaticamente como um fator multiplicativo do tempo integral mais longo dos dois *loops*. Desta forma, o fator  $\gamma$  é ajustado com base na integral da diferença entre os dois fluxos, devidamente dimensionados com a relação desejada  $a$ . O equilíbrio ocorre quando  $a y_1 = y_2$  ou  $S = 0$ .

Com o objetivo de obter melhores respostas transitórias, em (VISIOLI., 2005) uma nova arquitetura para controle de razão foi proposta baseada na estação de mistura. Nessa arquitetura, é adotada uma variação no parâmetro  $\gamma$  em relação ao

tempo, de forma que  $\gamma$  seja representado em função do tempo  $\gamma(t)$ . Para esta nova arquitetura, os processos P1 e P2 devem ser modelados com funções de transferência de primeira ordem mais tempo morto (FOPDT)

$$P_1(s) = \frac{K_1}{T_1s+1} e^{-L_1s} \quad (6)$$

$$P_2(s) = \frac{K_2}{T_2s+1} e^{-L_2s} \quad (7)$$

Nesta nova adaptação em (VISIOLI., 2005), os dois controladores, C1 e C2, são selecionados como controladores PI com ponderação de *set point*, ou seja, as variáveis manipuladas  $u_1$  e  $u_2$  da (figura 3) podem ser expressas como

$$u_1(t) = K_{p1} \left( b_1 r_1(t) - y_1(t) + \frac{1}{T_{i1}} \int_0^t (r_1(\tau) - y_1(\tau)) d\tau \right) \quad (8)$$

$$u_2(t) = K_{p2} \left( b_2 r_2(t) - y_2(t) + \frac{1}{T_{i2}} \int_0^t (r_2(\tau) - y_2(\tau)) d\tau \right) \quad (9)$$

Onde  $b_1$  e  $b_2$  são os valores do *Set Points*.

O valor do fator  $\gamma$  é escolhido como a saída de um controlador PI, cuja entrada é o erro de relação no instante atual mais um valor constante  $\gamma^*$ .

Uma condição adicional deve ser definida para evitar que no início da resposta transitória a condição  $y_2(t) > y_1(t)$  permaneça, ou seja, a saída  $y_2$  inicie seu transiente antes de  $y_1$ . Formalmente, é

$$\gamma(t) = \begin{cases} 0 & \text{if } L_1 > L_2 \text{ and } t < t_o + L_1 - L_2, \\ \gamma^* + K_p(e_r(t) + \frac{1}{T_{i2}} \int_0^t e_r(\tau) d\tau) & \text{elsewhere} \end{cases} \quad (10)$$

onde

$$e_r(t) = y_2(t) - ay_1(t) \quad (11)$$

A adoção de um parâmetro variável no tempo  $\gamma$  tem a intenção de realmente "moldar" a função de referência  $r_2(t)$  de modo que a resposta do segundo sistema em malha fechada seja tão igual quanto possível à do primeiro, apesar da possibilidade de este ter uma dinâmica diferente.

A adoção desta estratégia de controle obviamente implica que um esforço extra de sintonia é necessário em relação à Estação de Mistura. Os autores sugerem que  $\gamma^*$  seja escolhido como  $T_i2/T_i1$  e os ganhos do controlador PI, que fornece o valor atual de  $\gamma$ , sejam selecionados de acordo com a seguinte fórmula:

$$K_p = 0.5 \frac{L_2 T_1}{T_2 L_1}, \quad T_i = \frac{T_1}{L_1} \quad (12)$$

No estudo da Estação de Mistura (RS), seguiram-se outras publicações importantes, como os trabalhos vistos em (HÄGGLUND, 2017) e (OU, YU e ZANG, 2007). Porém, para esta pesquisa, que trata da aplicação de lógicas não clássicas, será usada como base as configurações e equações que vêm das referências (HÄGGLUND., 2001) e (VISIOLI., 2005). Assim, apresenta-se uma nova abordagem para a estação de controle de razão utilizando algoritmos baseados em uma lógica não clássica denominada Lógica Paraconsistente Anotada com anotação de dois valores (LPA2v) (DA SILVA FILHO, LAMBERT-TORRES e ABE, 2010).

### 1.3.2 Noções Iniciais da Estação Paraconsistente para Controle de Razão

Nesta pesquisa, serão utilizados os principais conceitos fundamentais da Lógica Paraconsistente Anotada com anotação de dois valores (LPA2v) (DA SILVA FILHO, ABE e TORRES, 2008) e com base nas referências citadas, junto com a rede de algoritmos formando uma Estação de Controle de Razão Paraconsistente (ECR-LP) (MATOS, 2018) para oferecerem melhores condições no processo de controle de relação. Esta primeira configuração ECR-LP, como visto na figura 5, é uma versão construída com base na figura 1 e figura 2, e seu objetivo é mostrar as diferenças estruturais e funcionais entre uma estação de razão convencional e a estação de razão que usa configurações de algoritmos paraconsistentes (DA SILVA FILHO, ABE e TORRES, 2008) (COELHO, 2019).

Comparando as Estações de Razão convencionais (ver figuras 1 e 2) com a primeira configuração paraconsistente mostrada na figura 5, pode-se verificar que para estabilizar as malhas nas ocorrências das variações do Set Point, a ECR-LP recebe o sinal do ponto de ajuste e o sinal de feedback do fluxo  $y_1$ . Após as análises paraconsistentes, a ECR-LP gera o sinal do set point secundário  $r_2$  e mais quatro



conclusões obtidas. Os quatro princípios que formam a lógica clássica são representados por:

**Princípio da Identidade ( $p = p$ )**

Onde toda proposição ou objeto é idêntico a si mesmo.

**Princípio da Identidade Proposicional ( $p \rightarrow p$ )**

Toda proposição implica nela mesma.

**Princípio do Terceiro Excluído ( $p \vee \neg p$ )**

De duas proposições contraditórias, uma delas é verdadeira.

**Princípio da Não-Contradição  $\neg(p \wedge \neg p)$**

Entre duas proposições contraditórias, uma delas é falsa.

Diante dos avanços tecnológicos, as adequações na implementação física no desenvolvimento dos processos, vêm requisitando o tratamento de informações com maior nível de precisão visto que, no mundo real nem todas as situações podem ser classificadas como verdadeiro ou falso (DA SILVA FILHO, ABE e TORRES, 2008).

#### 1.3.4 Lógica Não Clássica

Os limites entre o falso e o verdadeiro são indefinidos, incertos, e até contraditórios e, para processos de automatização, verifica-se que a lógica clássica é limitada para estes recursos já que seus algoritmos são incapazes de tratar situações que não foram consideradas de acordo com seu fundamento binário. Diante dessa situação, a tecnologia exigiu que novas teorias fossem criadas para tratar situações que a lógica clássica não é capaz de tratar, originando assim a Lógicas não clássicas (DA SILVA FILHO, ABE e TORRES, 2008).

No século XX, lógicos, matemáticos e filósofos trouxeram relevantes contribuições para transformar a lógica em uma ciência de extrema importância, culminando assim na criação das Lógicas não clássica. Na época foi uma transformação de natureza profunda no âmbito científico, cujas consequências de caráter filosófico ainda não se instigaram de maneira sistemática e integral. As Lógicas não clássicas foram inseridas com a finalidade de interpretar situações que no mundo real não podemos classificar como sendo puramente verdadeira ou falsa como acontece na Lógica clássica. Estudos da Lógica não clássica são aplicados quando requer obter precisão, ou obter soluções mais complexas para tomadas de decisões

onde o sistema é responsável em decidir e solucionar problemas onde implicam grandes quantidades de variáveis de diversos tipos.

### **1.3.5 Lógica Paraconsistente**

Nesta pesquisa os estudos voltados a Lógica paraconsistente (LP) é implementado através de sistemas microcontrolados no controle de razão adaptativa, para tratar e solucionar casos não triviais, ou seja, casos onde a lógica clássica ou tradicional, não pode ser empregada (COELHO, 2019).

Os algoritmos aplicados nesta pesquisa utilizam uma extensão da LP denominada de Lógica Paraconsistente Anotada (LPA) que se aplica a diversos campos. A LPA pode ser representada de modo particular, através de um reticulado de Hasse em que, intuitivamente, as constantes de anotação nos vértices do reticulado associado vão dar conotações de estados lógicos extremos às proposições que serão analisadas (DA SILVA FILHO, ABE e TORRES, 2008) (COELHO, 2019).

### **1.3.6 Logica Paraconsistente Anotada com anotação de dois valores (LPA2v)**

Conforme descrito anteriormente, a Lógica Paraconsistente (LP) pertence ao grupo das lógicas não clássicas e sua principal característica é se opor ao princípio da não contradição (DA COSTA, 1974). Uma extensão de LP é a Logica Paraconsistente Anotada (LPA), que tem um Reticulado de Hasse (Reticulado QUATRO) associado a estados lógicos representados em seus vértices. Desta forma, sentenças podem ser obtidas onde proposições podem ser analisadas com base em evidências (COELHO, 2019). Nesta representação, os quatro estados lógicos extremos representados nos vértices do Reticulado associado à LPA são: verdadeiro (V), Falso (F), Paracompleto ( $\perp$ ) e Inconsistente (T) (DA SILVA FILHO, ABE e TORRES, 2008).

Se dois valores de evidência relacionados a uma determinada proposição são considerados, então é possível associar um novo tipo de LPA, denominado Lógica Paraconsistente Anotada com a anotação de dois valores (LPA2v). Com os valores de evidência representados por meio de anotação ou graus normalizados de evidência, pode-se obter equações envolvendo os estados lógicos. Nessa configuração, a evidência é representada como graus de evidência que podem ser

obtidos a partir de medidas realizadas no ambiente físico (DA SILVA FILHO, ABE e TORRES, 2008) (COELHO, 2019).

A fórmula atômica da evidência da forma  $p(\mu, \lambda)$  pode ser vista como uma anotação para a proposição  $p$ , onde  $\mu, \lambda \in [0,1]$  (intervalo de unidade real). Assim, o grau de evidência ( $\mu$ ) é um valor que representa a evidência favorável à proposição  $p$ , e o grau de evidência ( $\lambda$ ) é um valor que representa a evidência desfavorável à proposição  $p$ . Na LPA2v a associação de um par  $(\mu, \lambda)$  com uma proposição  $p$  significa que o grau de evidência favorável em  $p$  é  $\mu$ , e o grau de evidência desfavorável em  $p$  é  $\lambda$  (COELHO, 2019).

Considerando os graus de evidência representados em um quadrado unitário no plano cartesiano (QUPC) onde  $\mu$  é exposto no eixo  $x$  e  $\lambda$  exposto no eixo  $y$ , pode-se fazer um mapeamento que leva seus valores a um Reticulado de quatro vértices (k) semelhante ao Reticulado teórico representativo da LPA2v.

Três ações são necessárias para este mapeamento, que são definidos como:

1. Expansão do QUPC de raiz de 2.
2. Rotação de  $45^\circ$  em torno da origem.
3. translação no eixo  $y$  (valor -1).

Essas três ações do USCP resultam em uma transformação  $T$ , representada na equação (3), a partir da qual equações paraconsistentes podem ser obtidas por interpretações no Reticulado de quatro vértices (k) (DA SILVA FILHO, ABE e TORRES, 2008).

$$T(X, Y) = (x - y, x + y - 1) \quad (3)$$

Pode-se relacionar os componentes da transformação  $T(X, Y)$  de acordo com a nomenclatura usual de LPA2v, onde:  $x = \mu \rightarrow$  Grau de evidência favorável, com  $0 \leq \mu \leq 1$  e  $y = \lambda \rightarrow$  grau de evidência desfavorável, com  $0 \leq \lambda \leq 1$ , onde:

(a) o primeiro termo obtido no par ordenado da equação de transformação é:  $X = x - y = \mu - \lambda \rightarrow$  que será denominado grau de certeza -  $Gc$ . Portanto, o grau de certeza é obtido por:

$$Gc = \mu - \lambda \quad (4)$$

(b) o segundo termo obtido no par ordenado da equação de transformação é:

$Y = x + y - 1 = \mu + \lambda - 1 \rightarrow$  que será denominado Grau de Contradição ( $G_{ct}$ ) (DA SILVA FILHO, ABE e TORRES, 2008). Portanto, o grau de contradição é obtido por:

$$G_{ct} = \mu + \lambda - 1 \quad (5)$$

As equações de LPA2v permitem que os estados lógicos paraconsistentes sejam encontrados na rede. Estes são pontos de interpolação dados por:

$$\mathcal{E}\tau = (G_c, G_{ct}) \quad (6)$$

Pode-se, portanto, tomar decisões com base na proximidade do estado lógico  $\mathcal{E}\tau$  aos estados lógicos extremos Verdadeiro (V) ou Falso (F), localizados nos vértices da rede LPA2v.

O grau de Contradição normalizado ( $\mu_{ctr}$ ), onde uma variação entre 0 e 1 é obtida, pode ser calculado através da equação:

$$\mu_{ctr} = \frac{\mu + \lambda}{2} \quad (7)$$

O grau de certeza real ( $G_{cr}$ ) é obtido pela determinação da distância  $d$  no Reticulado LPA2v como mostrado (Figura 6)

$$d = \sqrt{(1 - |G_c|)^2 + G_{ct}^2} \quad (8)$$

Os valores ( $G_{cr}$ ) são calculados de acordo com as condições mostradas abaixo (DA SILVA FILHO, ABE e TORRES, 2008):

$$\text{Se } G_c > 0 \quad G_{cr} = 1 - \sqrt{(1 - |G_c|)^2 + G_{ct}^2} \quad (9)$$

$$\text{Se } G_c < 0 \quad G_{cr} = \sqrt{(1 - |G_c|)^2 + G_{ct}^2} - 1 \quad (10)$$

Em sistema prático, o valor do grau de Evidência real resultante ( $\mu_{ER}$ ), que é obtido pelo valor normalizado do ( $G_{cr}$ ) (DA SILVA FILHO, LAMBERT-TORRES e ABE, 2010), é utilizado como saída que representa o resultado trazendo informação da proximidade ao verdadeiro estado lógico V. O valor  $\mu_{ER}$  é calculado a partir das equações (9) e (10) por:

$$\mu_{ER} = \frac{G_{cr} + 1}{2} \quad (11)$$

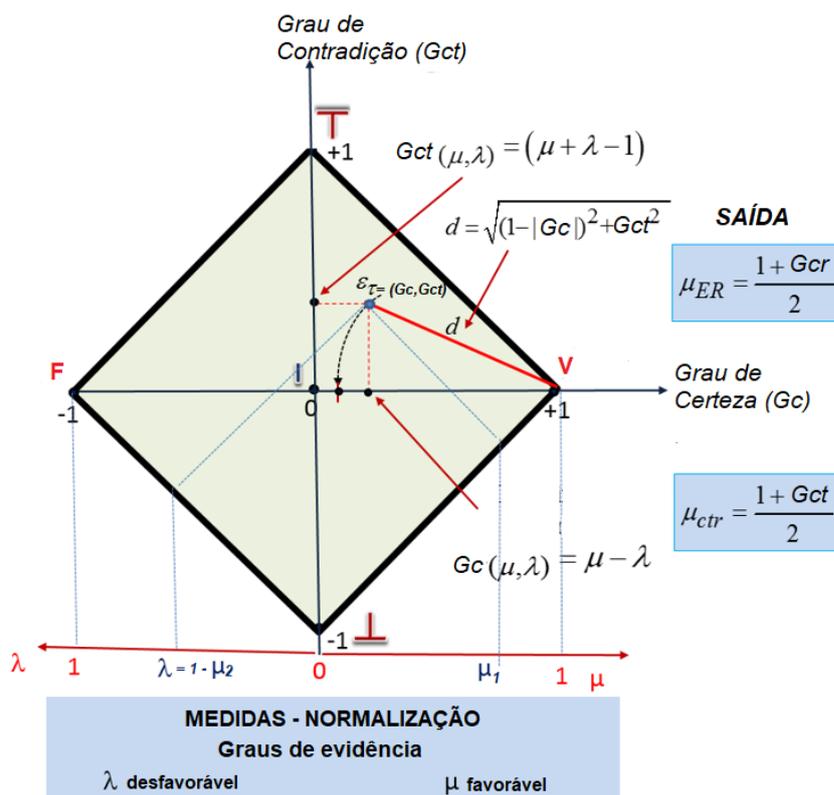
Com as equações obtidas pelas interpretações feitas no Reticulado associado ao LPA2v, são construídos algoritmos utilizados na análise e tratamento lógico de sinais de informação.

Neste trabalho é utilizado o algoritmo denominado Nó de Análise Paraconsistente (NAP), que pode computar informações incompletas e contraditórias (DA SILVA FILHO, ABE e TORRES, 2008) (COELHO, 2019).

Em sua versão completa, o algoritmo Nó de Análise Paraconsistente (NAP) recebe dois sinais de informação representados por graus de evidência, equaciona seus valores de acordo com as equações anteriores e apresenta um único valor do grau de evidência resultante. Essa saída, chamada de grau de evidência real resultante, é um valor que expressa uma representação única da análise paraconsistente feita nos dois valores de entrada. A ação do algoritmo resulta na anulação do efeito da contradição (DA SILVA FILHO, ABE e TORRES, 2008) (COELHO, 2019).

Dependendo da aplicação, o NAP pode oferecer outros sinais de informação sobre as contradições da análise.

A descrição de um NAP genérico segue os procedimentos estabelecidos pelas equações (4) a (11). A Figura 6 mostra a representação das equações obtidas por interpretações no Reticulado resultando no NAP dedicado ao tratamento de sinais baseado na Lógica Paraconsistente.



**Figura 6 – Reticulado de quatro vértices associado à Lógica Anotada Paraconsistente com anotação de dois valores (LPA2v) com deduções de equações utilizadas no NAP.**

Fonte: (DA SILVA FILHO, LAMBERT-TORRES e ABE, 2010)

O algoritmo NAP em seu modelo descritivo é apresentado a seguir e (DA SILVA FILHO, 2016):

### 1.3.7 Algoritmo NAP – Nó de Análise Paraconsistente com saídas $\mu_{ER}$ e $\mu_{ctr}$

1. Apresente os dois valores normalizados provenientes das fontes de informação:  $\mu_1$  e  $\mu_2$
2. Calcule o grau de evidência desfavorável:  $\lambda_1 = 1 - \mu_2$ .
3. Calcule o grau de certeza:  $G_c = \mu_1 - \lambda_1$ .
4. Calcule o grau de contradição:  $G_{ct} = (\mu_1 + \lambda_1) - 1$
5. Calcule a distância d:  $d = \sqrt{(1 - |G_c|)^2 + G_{ct}^2}$
6. Considere a restrição:
  - Se  $d > 1$ , então faça  $S_1 = 0,5 \rightarrow$  *Considerar indefinição na saída!*
  - Se não, vá para o próximo item.
7. Determine o Grau de Certeza real:

Se  $G_C > 0$ , Calcule:  $G_{CR} = (1 - d)$

Se  $G_C < 0$ , Calcule:  $G_{CR} = (d - 1)$

8. Calcule o Grau de Evidência resultante:

$$\mu_{ER} = \frac{G_{CR} + 1}{2}$$

9. Calcule o Grau de Contradição normalizado resultante:

$$\mu_{ctr} = \frac{G_{ct} + 1}{2}$$

10. Apresente os resultados nas saídas  $S_1$  e  $S_2$ :

Faça  $S_1 = \mu_{ER}$  e  $S_2 = \mu_{ctr}$

10. Fim.

Os resultados obtidos nas equações indicam que o NAP pode ser utilizado como gerador de graus de evidencia (figura 2)  $\mu_{ER}$  para outras proposições, portanto como sinais de entradas de análises por outros NAPS, formando assim uma rede de NAPs interligados.

### 1.3.8 Extrações de grau de evidência

Na extração do grau de evidência, é utilizado um algoritmo que tem a função de normalizar a magnitude física que está sendo considerada para análises paraconsistentes dentro de um intervalo de interesse ou universo de discurso. A função matemática utilizada para fazer a normalização, define no Intervalo de Interesse, como haverá a variação do valor do grau de evidência que representa a grandeza medida. Esta função matemática pode ser linear, exponencial ou qualquer outra tipo de função selecionada no projeto de acordo com a especificação que melhor se adequa às propriedades da malha e ao tipo de controle utilizado (COELHO, 2019).

Uma representação gráfica do algoritmo extrator de grau de evidência, cuja função matemática selecionada é linear, é mostrada na Fig. 7(b). Nesse caso, o grau de evidência  $\mu$  é calculado pela equação da reta (COELHO, 2019) e (DA SILVA FILHO, 2016)

### 1.3.9 Rede de Análise Paraconsistente – RAP

O algoritmo NAP pode ser interligado com diferentes configurações formando uma rede de análise paraconsistente - RAP (DA SILVA FILHO, ABE e TORRES, 2008) (COELHO, 2019).

Na RAP, cada algoritmo NAP analisa uma proposição  $p$  por meio dos valores dos graus de evidência  $\mu$  e  $\lambda$ , que representam a anotação LPA2v. Na análise de dados com uma configuração de 3 NAPs, por exemplo, duas proposições secundárias  $p_1, p_2$  fornecerão os graus de evidência para uma proposição objetiva ( $p_0$ ). Este tipo de configuração pode ser visto na figura 7(c) onde as proposições secundárias  $p_1, p_2$  recebem os valores de evidência das fontes de informação  $S_1, S_2, S_3$  e  $S_4$  e, através dos algoritmos NAP1 e NAP2, fornecem os graus de evidência ( $\mu_1, \lambda_2$ ) para o NAP0 que, por sua vez analisa a proposição de objetivo final (COELHO, 2019) (GARCIA, 2019) e (DA SILVA FILHO, 2016).

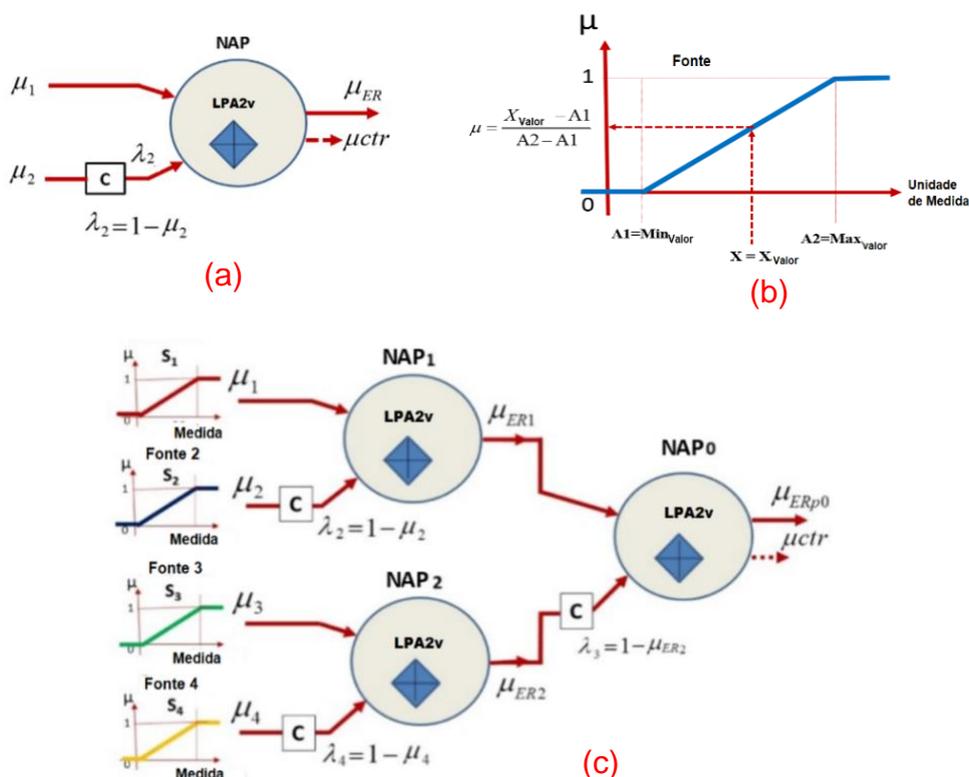


Figura 7 – (a) Símbolo de um nó típico de análise paraconsistente - NAP. (B) Representação gráfica do algoritmo extrator de grau de evidência. (c) Tipo de configuração RAP onde as proposições secundárias  $p_1, p_2$  geram os graus de evidência para a análise da proporção final  $p_0$ .

## 2. MATERIAIS E MÉTODOS

Neste trabalho, a estação de controle de razão paraconsistente que será construída terá seus testes efetuados em uma malha de controle de fluxo de uma planta Piloto experimental composto por três tanques de armazenamento de líquidos. Os detalhes da planta Piloto experimental utilizada na implementação do controle de razão com a (ECAR-LPA) serão apresentados a seguir:

### 2.1 Noções Iniciais da Estação Paraconsistente Adaptativa para Controle de Razão

Para essa pesquisa uma configuração da Estação de Razão Paraconsistente foi criada com características de controle adaptativo e neste caso, a configuração Paraconsistente construída (ECAR-LPA) foi estruturada com base na Estação Adaptativa de Mistura (ABS) mostrada na figura 4.

A Figura 8 mostra um diagrama de blocos da estrutura construída para a Estação de Razão Adaptativa Paraconsistente (ECAR-LPA).

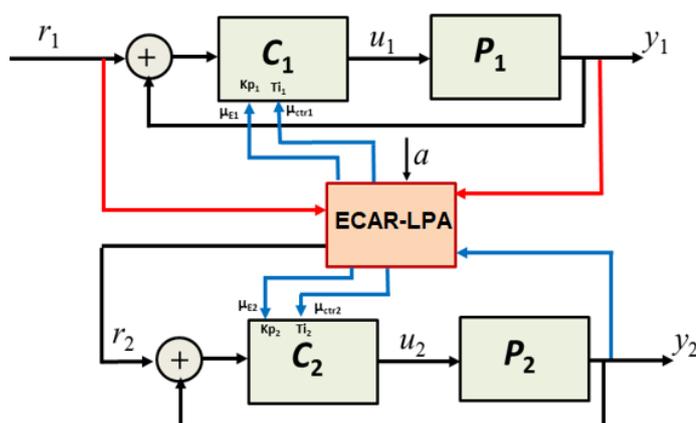


Figura 8 – Controle adaptativo de razão paraconsistente - ECAR-LPA com sinal do fluxo secundário  $y_2$ .

Comparando a Estação de Mistura (HÄGGLUND., 2001), mostrada na figura 4, com o ECAR-LPA na figura 8, pode-se observar que para estabilizar as malhas na ocorrência de variações no *Set Point* e nos fluxos, a ECAR-LPA recebe mais dois sinais de *feedback* que vêm dos fluxos  $y_1$  e  $y_2$ . Além do sinal do ponto de ajuste secundário  $r_2$ , gerando mais quatro sinais que atuam diretamente no controle PI nos

ganhos proporcionais ( $K_p1$ ) e ( $K_p2$ ) e nos tempos integrativos ( $T_i1$ ) e ( $T_i2$ ) dos dois controladores C1 e C2 que atuam nos processos P1 e P2, respectivamente.

## 2.2 Detalhes da planta piloto planta piloto experimental usada nos testes de controle de razão

A planta Piloto experimental utilizada nos testes e estudos da Estação de Controle Adaptativo de Razão Paraconsistente (ECAR-LPA) é mostrada na Figura 9.



Figura 9 – Visão geral da planta piloto de processo industrial usada nos testes de validação da estação de razão paraconsistente.

Observa-se na figura 9 que existe na planta piloto dois medidores do tipo Vórtex que possuem um sensor piezoelétrico denominado transmissores de vazão FIT-101 e FIT-102, dois tanques de líquidos TQ-101 e TQ-102 e um tanque maior TQ-103 no qual os líquidos serão misturados. Os controladores estão acoplados no painel instalado na plataforma metálica da planta.

Para o sistema com controle de razão, a saída do transmissor de vazão FIT-101 é a entrada do bloco multiplicador “a” e do controlador de fluxo FIC-101 associado ao tanque TQ-101. Portanto, para o controle de razão, o especialista de processo define o fator de taxa de fluxo do bloco de ganho “a”. O SPR - *Set Point*, que é a saída do bloco com ganho “a”, será, portanto, o ponto de ajuste (*set point*) do controlador

FIC-102 de vazão de produto para o tanque TQ-102. O controle será feito para que o terceiro tanque TQ-103 receba os produtos dos tanques TQ-101 e TQ-102.

Os transmissores de vazão FIT-101 e FIT-102 são do tipo placa de orifício associada ao transmissor de pressão diferencial SMAR LD301, em escala de 4 a 20 mA e constante de tempo de 0,2 seg.

### 2.3 Determinações de graus de evidência para os sinais de entrada na Estação de Controle de Razão Paraconsistente (ECAR-LPA)

O sinal de saída do transmissor de vazão, FIT-101, é a fonte de evidência PVy1 que fornece o grau favorável de evidência  $\mu y1$ . Por meio da complementação do *Set Point*, esse sinal é transformado em um grau de evidência desfavorável  $\lambda y1$  formando a anotação com o sinal de ponto de ajuste do Controlador Fy-101, que é a segunda fonte de evidência  $\mu SPr1$  do primeiro nó de análise paraconsistente NAP1.

O sinal de saída do transmissor de vazão, FIT-102, é a fonte de evidência PVy2 que fornece o grau favorável de evidência  $\mu y2$  formado pela anotação com o sinal  $\mu y1$  multiplicando o fator de razão ajustado complementando denominado  $\lambda y1$  do Controlador FY-102.

O valor lógico paraconsistente de evidências favoráveis ou desfavoráveis é um número real entre 0 e 1. Assim, os sinais de medição  $PV1 = y\mu1$  e  $PV2 = y\mu2$  fornecidos pelos transmissores de vazão FIT-101 e FIT-102 foram convertidos para seus valores lógicos paraconsistentes, conforme mostrado pelas seguintes correspondências lineares:

- $PV1 \in \mathbb{R} \mid 0 \leq PV1 \leq 1 \blacktriangleright 0 \leq \mu1 \leq 1 \blacktriangleright 0 \text{ m}^3/\text{h} (0\%) \leq y1 \leq 2.5 \text{ m}^3/\text{h} (100\%)$
- $PV2 \in \mathbb{R} \mid 0 \leq PV2 \leq 1 \blacktriangleright 0 \leq \mu3 \leq 1 \blacktriangleright 0 \text{ m}^3/\text{h} (0\%) \leq y2 \leq 2.5 \text{ m}^3/\text{h} (100\%)$

Os valores lógicos paraconsistentes das vazões entre 0 e 1 estão em correspondência linear com a vazão desejada entre 0% e 100% correspondendo à faixa de 0 a 2,5 m<sup>3</sup>/h.

### 2.4 Estrutura Algorítmica de uma Estação de Controle Adaptativa de Razão Paraconsistente (ECAR-LPA2v)

O controle dos fluxos em caso de alteração do *Set Point* e outras perturbações nas variáveis controladas em controle de mistura pode ser desenvolvido através de um Algoritmo para tratar as inconsistências formando um bloco que é denominado

Estação de Controle Adaptativa de Razão Paraconsistente (ECAR-LPA2v). O diagrama em bloco de uma ECAR-LPA2v está mostrado na Figura 10.

Figura 10 (a) mostra a configuração do módulo ECAR-LPA2v.

A Figura 10 (b) mostra a instalação da ECAR-LPA2v em um controle processo típico de controle de Taxa de Fluxo da Planta Piloto de Processo Industrial.

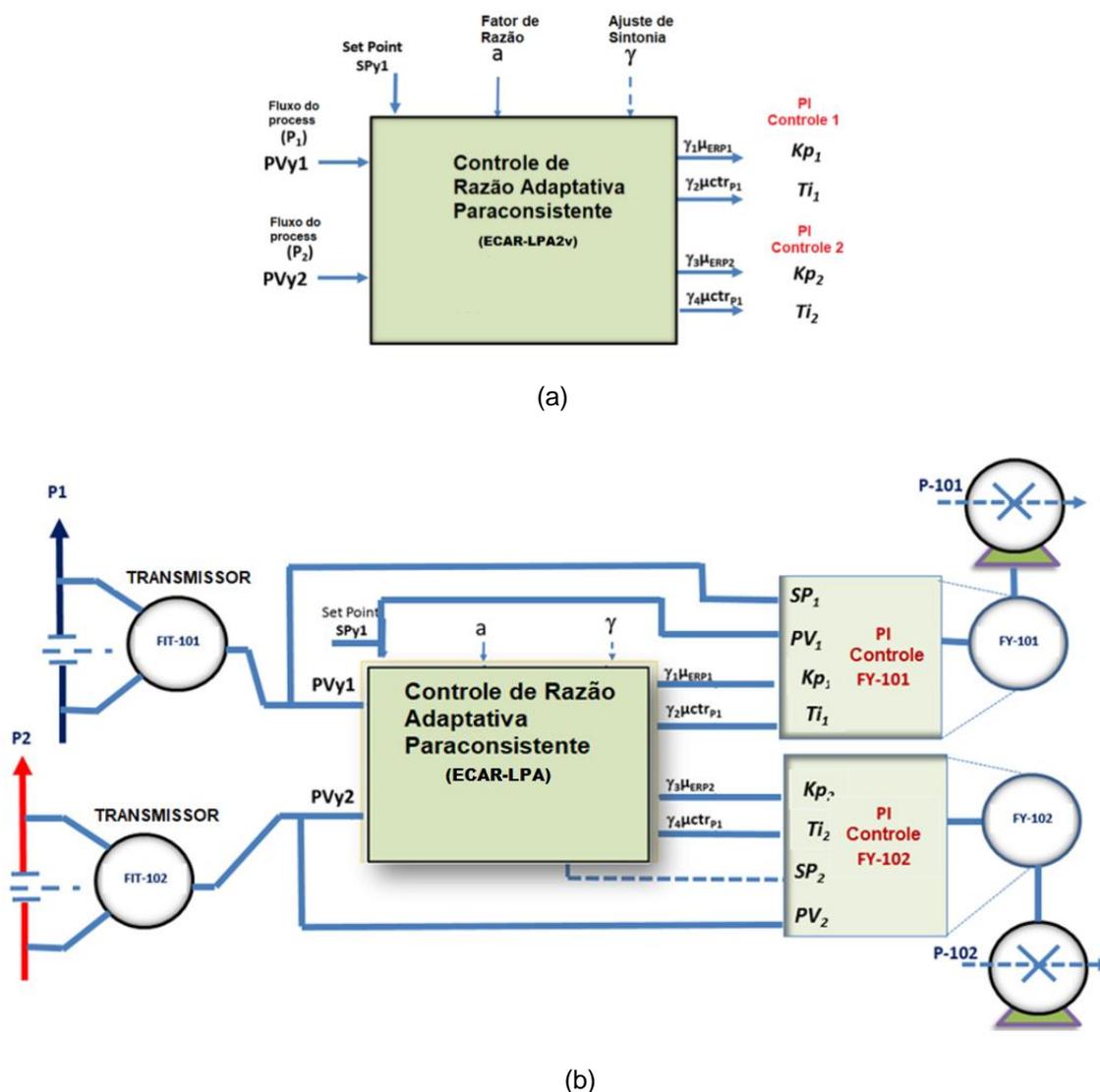


Figura 10 – a) configuração do módulo ECAR-LPA2v. b) Diagrama e instalação de uma Estação de Controle de Razão Paraconsistente (ECAR-LPA2v) em uma planta típica de controle de mistura.

Conforme visto na figura 10 (b), uma a ECAR-LPA típica tem duas entradas do *feedback* de processos que funcionam dinamicamente, e duas entradas que são viáveis de ajustar. Essas quatro entradas são descritas como:

- a) PVy1 - Entrada da variável de processo, que é o sinal de *feedback* do fluxo principal  $y_1$ , simbolizado por  $\mu y_1$ .

- b) PVy2 - Entrada da variável de processo, que é o sinal secundário de feedback de fluxo controlado  $y_2$ , simbolizado por  $\mu y_2$ .
- c) SPr1 - Ajuste do set point do controlador PI do processo mestre.
- d) a - Ajuste do valor multiplicativo referente ao índice que será utilizado.

Pode-se observar na figura 6a que cada uma das quatro saídas da ECAR-LPA é multiplicada pelo seu respectivo fator representado pela letra  $\gamma$ .

Chamaremos  $\gamma$  de fator de sintonia.

A representação dos fatores de sintonia  $\gamma$  nas saídas da ECAR-LPA refere-se à inserção de valores obtidos por cálculos e análises das redes de algoritmos paraconsistentes fazendo o processamento dos dados. Esses resultados finais dependem da configuração utilizada e serão valores constantes após suas determinações, porém, na prática, podem ser ajustados. Portanto, a ECAR-LPA tem quatro valores de saída, conforme abaixo:

a)  $\gamma_1 \mu_{ER1}$  e  $\gamma_3 \mu_{ER2}$  que são dois graus de evidência resultante multiplicados pelos respectivos fatores de sintonia cuja função do sinal resultante é atuar nos ganhos proporcionais  $K_{p1}$  e  $K_{p2}$  dos controladores PI.

b)  $\gamma_2 \mu_{crt1}$  e  $\gamma_4 \mu_{ctr2}$  que são dois graus normalizados de contradição multiplicados pelos respectivos fatores de sintonia, cujos valores resultantes atuam diretamente nos tempos integrais  $T_{i1}$  e  $T_{i2}$  dos controladores PI.

O método de inserção com os procedimentos para calcular esses valores de ajuste  $\gamma$  será descrito a seguir.

## **2.5 Funcionamento básico da Estação de Controle de Razão Paraconsistente (ECAR-LPA)**

O funcionamento da ECAR-LPA baseia-se na atuação nas equações (8) e (9), a partir das quais são retirados os dois pesos  $b_1$  e  $b_2$  e inseridos os valores resultantes do tratamento dos dados realizado pelos algoritmos paraconsistentes. Para satisfazer este controle representado nas equações (8) e (9), o RAP que compõe uma ECAR-LPA possui dois algoritmos principais (NAP) na saída.

Depois que os sinais de entrada são processados pelos primeiros NAPs, esses dois algoritmos executam os últimos tratamentos de dados e geram dois sinais de saída cada.

O primeiro NAP-Output gera o grau de evidência resultante  $\mu_{ERP1}(t)$  e o grau normalizado de evidência  $\mu_{ctrP1}(t)$  que atua no Controlador de Processo 1.

O segundo NAP-Output gera o grau de evidência resultante  $\mu_{ERP2}(t)$  e o normalizado grau de evidência  $\mu_{ctrP2}(t)$  atuando no Controlador de Processo 2. Com este procedimento, pode-se obter as duas novas equações de controle de razão, conforme mostrado abaixo:

$$u_1(t) = \gamma_1 \mu_{ERP1}(t) K_{p1} \left( r_1(t) - y_1(t) + \frac{1}{\gamma_2 \mu_{ctrP1}(t) T_{i1}} \int_0^t (r_1(\tau) - y_1(\tau)) d\tau \right) \quad (22)$$

$$u_2(t) = \gamma_3 \mu_{ERP2}(t) K_{p2} \left( r_2(t) - y_2(t) + \frac{1}{\gamma_4 \mu_{ctrP2}(t) T_{i2}} \int_0^t (r_2(\tau) - y_2(\tau)) d\tau \right) \quad (23)$$

Onde

$\mu_{ERP1}(t)$  é o Grau de Evidência resultante do NAP na saída da ECAR-LPA que direciona os sinais para o ganho proporcional ( $K_{p1}$ ) do Controlador PI (FY-101) que atua no processo P1.

$\mu_{ctrP1}(t)$  é o grau normalizado de contradição resultante do NAP da saída da ECAR-LPA2v que direciona os sinais para o tempo integrativo do Controlador PI ( $T_{i1}$ ) do Controlador PI (FY-101) que atua no processo P1.

$\mu_{ERP2}(t)$  é o Grau de Evidência resultante do NAP na saída da ECAR-LPA que direciona os sinais para o ganho proporcional ( $K_{p2}$ ) do Controlador PI (FY-102) que atua no processo P2.

$\mu_{ctrP2}(t)$  é o grau normalizado de contradição resultante do NAP da saída da ECAR-LPA que direciona os sinais para o tempo integrativo do Controlador PI ( $T_{i2}$ ) do Controlador PI (FY-102) que atua no processo P2.

#### 2.4.1 Equações dos valores de ajuste $\gamma$

Verifica-se que  $\gamma_1, \gamma_2, \gamma_3$  e  $\gamma_4$  são os fatores de ajuste que se adequam às condições de processo estático para as quatro saídas da ECAR-LPA. Os valores dos fatores de ajuste  $\gamma_1, \gamma_2, \gamma_3$  e  $\gamma_4$  são calculados usando as seguintes equações:

$$\gamma_1 = \frac{1}{\mu_{ERP1(Balance)}} , \quad (24)$$

$$\gamma_2 = \frac{1}{\mu_{ctrP1(Balance)}} , \quad (25)$$

$$\gamma_3 = \frac{1}{\mu_{ERP2(Balance)}} , \quad (26)$$

$$\gamma_4 = \frac{1}{\mu_{ctrP2(Balance)}} . \quad (27)$$

Os valores de equilíbrio, e que definem os fatores de sintonia ( $\gamma$ ) são obtidos a partir da análise na RAP que forma a ECAR-LPA considerado no estado ideal para a configuração utilizada. Uma condição ideal que ocorre em equilíbrio para o controle de razão paraconsistente é considerada aquela em que os valores de entrada (sinais de *feedback* e ajuste) e os valores de saída (graus de evidência e graus normalizados de contradição) são iguais ao valor ajustado pelo especialista. Portanto, para cada configuração esta condição especial de equilíbrio deve ser simulada. A partir da simulação os respectivos valores dos graus normalizados de evidência e contradição, que atuarão diretamente nos ganhos proporcionais (Kp1 e Kp2), são calculados nos dois NAPs da saída, e em os tempos (Ti1 e Ti2) dos controladores.

Conforme visto nas equações (22) e (23), cada fator de ajuste com seu valor definido na análise da configuração ECAR-LPA é multiplicado pelos graus de evidência dinâmica  $\mu_{ERP}(t)$  e graus de contradição dinâmica normalizados  $\mu_{ctrP}(t)$ . Assim, quando os valores dinâmicos de  $\mu_{ERP}(t)$  e  $\mu_{ctrP}(t)$  são iguais aos obtidos na condição de Equilíbrio, tem-se: e.  $\gamma_1 \mu_{ERP1}(t) = 1$   $\gamma_2 \mu_{ctrP2}(t) = 1$  ,  $\gamma_3 \mu_{ERP2}(t) = 1$  e  $\gamma_4 \mu_{ctrP2}(t) = 1$

Quando a multiplicação dos fatores de sintonia com os valores dinâmicos resulta na unidade, o estado dinâmico controlado dos *loops* do processo é estabelecido pela condição de “nenhuma ação” da ECAR-LPA nos ganhos (Kp1 e Kp2) e integração tempos (Ti1 e Ti2) dos controladores. Portanto, com o processo neste estado significa que não há transição, causada por mudanças no *set point*, e não há distúrbios de fluxos no sistema de malha de controle. Nesta condição os sinais

dinâmicos tornam-se iguais e os de equilíbrio, tais que:  $\gamma_1 \mu_{ERP1}(t) K_{p1} = K_{p1(Tuning)}$ ,  $\gamma_3 \mu_{ERP1}(t) K_{p2} = K_{p2(Tuning)}$  e  $\gamma_2 \mu_{ctr_{P1}}(t) T_{i2} = T_{i1(Tuning)}$ ,  $\gamma_4 \mu_{ctr_{P2}}(t) T_{i2} = T_{i2(Tuning)}$ .

De outro modo, quando esta condição ocorre, a ECAR-LPA fica praticamente inativo e os únicos valores que atuam nos controladores (FY-101 e FY-102) dos processos são os dos ganhos proporcionais e os dos tempos integrais que foram obtidos pelo procedimento usual de ajuste para malhas (IMC).

## 2.6 Configuração do Controle de Razão Adaptativa Paraconsistente ECAR-LPA utilizada neste estudo

A configuração de NAPs da ECAR-LPA utilizada neste estudo é mostrada na figura 11 e a sua operação e o método de determinação dos fatores de ajuste serão detalhados a seguir.

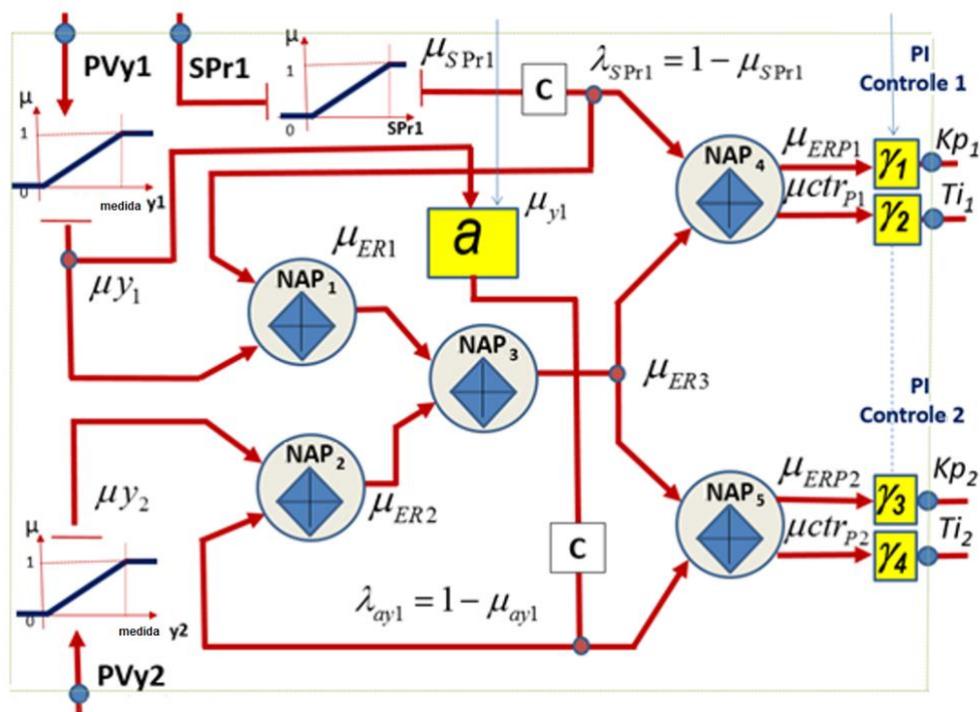


Figura 11 – Diagrama da Rede de Análise Paraconsistente (RAP) que compõe a Estação de Controle Adaptativo de Razão Paraconsistente (ECAR-LPA).

Conforme visto na figura 11, na configuração da Rede de algoritmos paraconsistentes que compõem a ECAR-LPA o NAP2 recebe como grau de evidência favorável ( $\mu_{y2}$ ) o sinal de *feedback* originado do fluxo y2 (PVy2) que corresponde à variável de processo controlada P2. Através da análise deste sinal, a configuração

funciona de forma que o controle da relação se torna adaptativo, portanto com uma estrutura muito semelhante à da figura 4. O sinal referente ao fluxo  $y_1$ , que corresponde à variável do processo mestre (P1), é multiplicado pelo fator de razão  $a$  e entra na análise paraconsistente de NAP2 como um grau de evidência desfavorável ( $\lambda_{ay1}$ ). Aplicando a notação LPA2v, o sinal lógico paraconsistente é simbolizado por  $p_2$  ( $\mu_{ER1}, \lambda_{ay1}$ ) onde  $p_2$  é a proposição “Existem anomalias no fluxo do processo 1 ou no fluxo do processo 2”; ( $\mu_{y2}, \lambda_{ay1}$ ) é a anotação composta pelos graus de evidência relacionados a esta proposição. O NAP3 faz uma análise comparativa entre os dois sinais obtidos nos NAPs anteriores, recebendo o grau de evidência 1 ( $\mu_{ER1}$ ) e o grau de evidência 2 ( $\mu_{ER2}$ ) nas entradas. O sinal ( $\mu_{ER1}$ ) é a informação entre o ponto de ajuste e a variável de processo do *loop* de controle de processo P1 (mestre). O sinal ( $\mu_{ER2}$ ) é a informação sobre o fluxo controlado do processo P2 (escravo 2) e o valor de *feedback* do fluxo controlado do processo P1 multiplicado pelo fator da razão ( $a$ ). Portanto, verifica-se que o NAP3 recebe a informação onde são intrínsecos os comportamentos do *Set Point* e dos dois fluxos que compõem a malha de controle de mistura. Desta forma, seu resultado comparativo servirá de referência para os outros dois NAPs finais (NAP4 e NAP5) que atuarão diretamente nos ganhos proporcionais e tempos integrativos dos controladores PY-101 e PY-102. As informações resultantes do NAP3 são recebidas pelos dois NAPs finais (NAP4 e NAP5), como um grau de evidência favorável ( $\mu_{ER3}$ ). Para esses dois NAPs finais (NAP4 e NAP5), os graus desfavoráveis de evidência estão relacionados ao ponto de ajuste e ao fluxo de  $y_1$  multiplicado pelo fator ( $a$ ). Portanto, os dois algoritmos NAP4 e NAP5 são os que finalizam a análise paraconsistente considerando as informações sobre as anomalias nos fluxos  $y_1$  e  $y_2$  e as possíveis variações no valor do *Set Point*. No algoritmo NAP4, o valor resultante de NAP3 é considerado como um grau de evidência favorável ( $\mu_{ER3}$ ) e o valor do *Set Point* principal é considerado como o grau de evidência desfavorável ( $\lambda_{SPr1}$ ). Em NAP5, o valor resultante de NAP3 é considerado como um grau de evidência favorável ( $\mu_{ER3}$ ) e o valor do *Setpoint* principal multiplicado pelo fator de Razão ( $a$ ) é considerado como o grau de evidência desfavorável ( $\lambda_{ay1}$ ). Com essa anotação composta por esses sinais as quatro saídas de NAP4 e NAP5 enviam as informações representadas por seus graus de evidência resultantes e graus normalizados de contradição para atuar nos ganhos proporcionais e tempos de integração dos controladores FY-101 e FY-102. Desta forma, as quatro saídas correspondem a valores dinâmicos que atuam em conjunto com os fatores de

sintonia  $\gamma$  e, assim, promovem continuamente o monitoramento e a análise das contradições nos dois controladores.

## 2.7 Determinação dos Fatores de Tuning para as configurações das Redes de Análise Paraconsistente da ECAR-LPA

O método para determinação dos fatores de ajuste para a ECAR-LPA segue alguns procedimentos baseados nas análises e funcionamento da RAP. Os procedimentos efetuados estão descritos a seguir.

Inicialmente, considerou-se as condições ideais de controle com o fator multiplicativo da razão ( $a$ ) ajustado para 0,5. Nesta condição, o *feedback* é caracterizado onde as medições de fluxo (PVy1) no processo de controle mestre P1 resultam em  $\mu_{y1} = 0,5$  e no fluxo (PVy2) do processo controlado 2 resulta em  $\mu_{y2} = 0,5$ .

Com as condições de estado ideais estabelecidas, foram feitas as simulações na RAP para obter os quatro valores de equilíbrio nas saídas da ECAR-LPA para vários valores de Set point variando de 0 a 1.

Em seguida, para cada grupo de valores do *Output*, foram obtidos os valores médios e determinados os fatores de equilíbrio. Os resultados desses procedimentos são apresentados na tabela 1.

**Tabela 1. Valores característicos da configuração NAP usados no Controle Adaptativo de Razão Paraconsistente (ECAR-LPA).**

PVy1 $\mu_{y1}$	PVy2 $\mu_{y2}$	Pr1 SPr1	a	NAP <sub>4</sub> $\mu_{ERP1}$	NAP <sub>4</sub> $\mu_{ctrP1}$	NAP <sub>5</sub> $\mu_{ERP2}$	NAP <sub>5</sub> $\mu_{ctrP2}$
0.5	0.5	1.0000	0.5	0.734834957	0.312500000	0.475985819	0.687500000
0.5	0.5	0.9375	0.5	0.730270679	0.343100085	0.475132561	0.686850084
0.5	0.5	0.8751	0.5	0.717046812	0.372367779	0.472597143	0.684917779
0.5	0.5	0.7500	0.5	0.669281086	0.427357646	0.462698922	0.677357646
0.5	0.5	0.6250	0.5	0.602252435	0.477868627	0.447077039	0.665368627
0.5	0.5	0.5625	0.5	0.564177322	0.501680562	0.437435031	0.657930562
0.5	0.5	0.5468	0.5	0.554263082	0.507526633	<b>0.434844278</b>	<b>0.655926633</b>
0.5	0.5	0.5312	0.5	<b>0.544290772</b>	<b>0.513284441</b>	0.432207200	0.653884442
<b>0.5</b>	<b>0.5</b>	<b>0.5000</b>	<b>0.5</b>	0.524014180	0.524653045	0.426753547	0.649653045
0.5	0.5	0.4685	0.5	0.503268558	0.536080672	0.421197513	0.645330672
0.5	0.5	0.4375	0.5	0.487244647	0.547430598	0.415877652	0.641180598

0.5	0.5	0.3750	0.5	0.451290043	0.570715223	0.405709506	0.633215223
0.5	0.5	0.2500	0.5	0.388439216	0.619576189	0.388439216	0.619576189
0.5	0.5	0.1250	0.5	0.363917258	0.687122986	0.394807891	0.624622986
0.5	0.5	0.0625	0.5	0.353711114	0.716901594	0.392948377	0.623151594
0.5	0.5	0.0000	0.5	0.350242981	0.747659187	0.392326581	0.622659187

Com os cálculos dos valores considerados fatores de equilíbrio, conforme evidenciado na tabela 1, mostrados em negrito. Com estes valores os Fatores de Tuning para as saídas da ECAR-LPA utilizados na sua construção por sistemas microprocessados foram obtidos através das equações (24) a (27), conforme mostrado a seguir:

Usando a equação (24)

$$\gamma_1 = \frac{1}{Kp_{C1(Tuning)}} = \frac{1}{0.544290772} = 1.837$$

Usando a equação (25)

$$\gamma_2 = \frac{1}{Ti_{1(Tuning)}} = \frac{1}{0.513284441} = 1.948$$

Usando a equação (26)

$$\gamma_3 = \frac{1}{Kp_{C2(Tuning)}} = \frac{1}{0.434844278} = 2.304$$

Usando a equação (27)

$$\gamma_4 = \frac{1}{Ti_{P2(Tuning)}} = \frac{1}{0.655926633} = 1.526$$

Assim, usando as equações (22) e (23) as equações de controle para a configuração do ECAR-LPA ficaram definidas, como se segue:

Para PI Controle (FY-101) do processo P1

$$u_1(t) = 1.837 \mu_{ERP1}(t) K_{p1} \left( r_1(t) - y_1(t) + \frac{1}{1.948 \mu_{ctr_{P1}}(t) T_{i1}} \int_0^t (r_1(\tau) - y_1(\tau)) d\tau \right)$$

Para PI Controle (FY-102) do processo P2

$$u_2(t) = 2.304\mu_{ERP2}(t)K_{p2} \left( r_2(t) - y_2(t) + \frac{1}{1.526\mu_{ctrP2}(t)T_{i2}} \int_0^t (r_2(\tau) - y_2(\tau))d\tau \right)$$

## 2.8 Construção da Estação de razão adaptativa microcontrolada

A partir das equações e parâmetros obtidos nas simulações a ECAR-LPA foi implementada no sistema microcontrolado conforme mostrado a seguir.

### 2.7.1 Protótipo – Hardware

Definida o diagrama para a construção do controle adaptativo o sistema de efetuação da coleta dos dados e implementação da ECAR-LPA foi feita com o uso da plataforma open-source de *software Arduino IDE* a placa de prototipagem *Hardware Arduino*, modelo UNO (anexo 2) que possui um microcontrador ATMEGA328P figura 12, composta por 14 entradas e saídas digitais (até 40 mA por pino) e 6 portas de entrada analógica (até 50mA por pino). O Hardware que compõem o protótipo da ECAR-LPA é visto na figura 8,

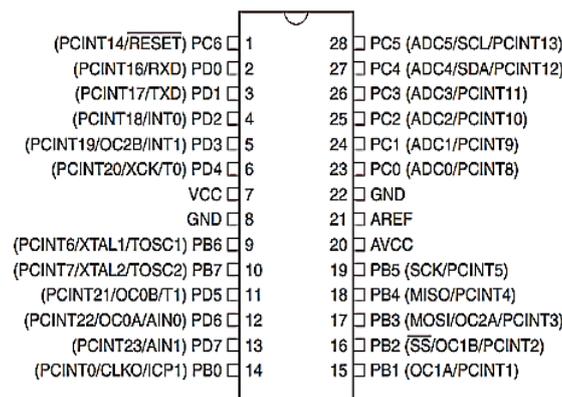


Figura 12 – Portas e pinos de tensão do microcontrolador Atmega328P.

Para efetuação das simulações em laboratório foram instalados na placa dois potenciômetros de 10KΩ, sendo, responsáveis pelo ajuste realizado pelo especialista do set-point remoto (SP<sub>r</sub>) e o outro pelo fator multiplicador de razão (a), figura (13).

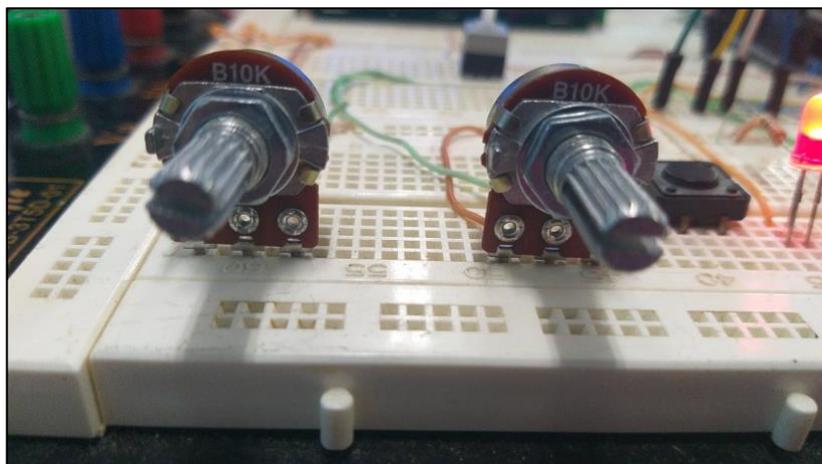


Figura 13 – Potenciômetros de ajustes dos *set-point* remoto e fator de razão (A).

Um display LCD 16x2 (colunas x linha) foi inserido para visualizar quatros dados, sendo compostos por valores de ajuste do *set-point* (SP) e fator de razão (FA), e os valores de entradas das transmissões de vazão (T1 e T2), figura 14.



Figura 14 – Display LCD 16x2

### 2.7.2 Arquitetura da Comunicação Paraconsistente ECAR-LPA

A figura 15 representa a estratégia do controle para comunicação dos transmissores de fluxo das malhas PVy1 e PVy2 provenientes da ER implementado sobre o controle adaptativo de razão desenvolvido. A interação acontece com o OPC Server Tool Box existente na implementação base da estação piloto, sendo responsável pela comunicação da leitura dos transmissores de vazão FIT-101 e FIT-

102 variam entre 4 e 20mA de forma proporcional a variação da vazão volumétrica com o hardware analógico CD-600 SMAR. Os valores são inseridos ao algoritmo de forma dinâmica e em valores reais sendo parametrizados no início do algoritmo de análise paraconsistente LPA2v, conforme figura 12.

A entrega ao sistema acontece nas saídas do NAP4 e NAP5 grau de evidencia resultante real e grau de contradição normalizados, irão atuar nas entradas do controlador PI (proporcional e integral) do bloco PI contínuo paralelo FIC-101 e FIC-102.

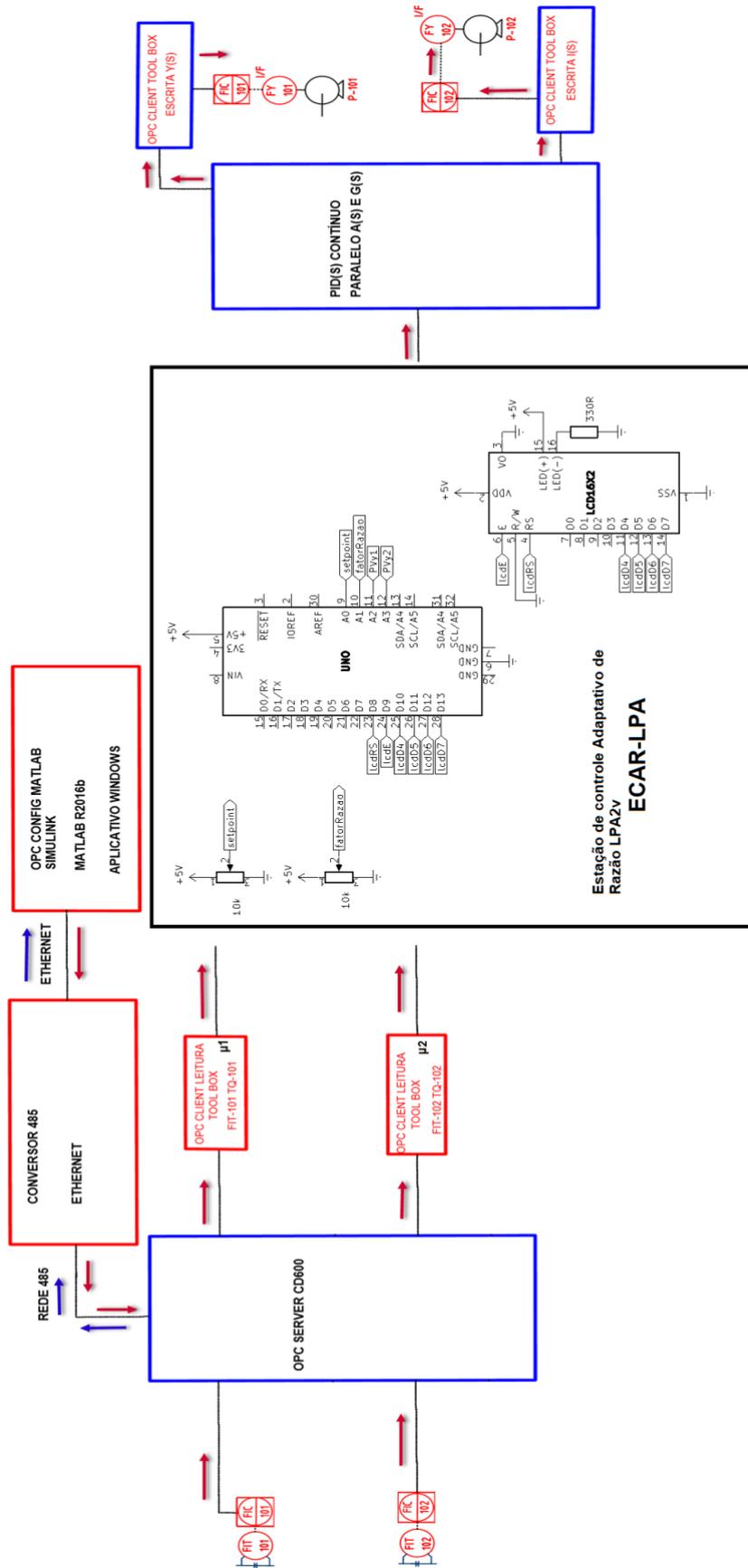


Figura 15 – Estratégia de comunicação para consistente do ECAR-LPA com o *Matlab Simulink OPC Tool Box*.

### 2.7.3 Bloco de aquisição de dados e Análise NAP no protótipo da ECAR-LPA implementado na plataforma Arduino

A IDE do Arduino utiliza como base da programação a linguagem C++. Inicialmente foram definidas as variáveis responsáveis pela leitura de entrada analógica das portas do Arduino Uno vindas dos transmissores FIT101 e FIT102 e dos potenciômetros utilizados para configuração dos ajustes de SET POINT e fator de razão *a*, definidos pelo especialista no processo. A seguir foi definida dentro da função `setup()` a inicialização da Comunicação Serial para transferência de dados a uma taxa de transmissão padronizada a 9600bit/segundo. Dentro da função `loop` que é a principal de repetição da programação onde está sendo executado de forma contínua, a parametrização da leitura analógica dos dados de entrada entre 0 e 1 através da informação de entrada dividido pelo valor máximo do nível lógico analógico 1023 (bloco. tratamento dos dados). Figura 16

Realizados a parametrização, sinais como *SET POINT* e o Fator de razão *a*, foram necessários serem complementados para servirem como finais de entrada desfavoráveis nos algoritmos nós paraconsistentes NAP1 e NAP2 identificados nos (bl. SETPOINT COMPLENETADO e FATOR DE RAZAO COMPLEMENTADO).

```

////////////////////////////////leitura das portas analogicas
leituraSP = analogRead(A0);
leituraFatorA = analogRead(A1);
leituraPV1 = analogRead(A2);
leituraPV2 = analogRead(A3);

////////////////////////////////TRATAMENTO DOS DADOS////////////////////////////////
setPoint = leituraSP/1023;//SETPOINT 0~1
pv1 = leituraPV1/1023; // sinal do transmissor FT101
pv2 = leituraPV2/1023; // sinal do transmissor FT102
ajusteFatorA =leituraFatorA/1023; //FATOR DE RAZÃO A 0~1

```

Figura 16 - Bloco de tratamento dos dados favoráveis e desfavoráveis.

Definidos os sinais *mi* e *lambda*, que no NAP1 são chamados de *pv1* e *lamb\_SP* e no NAP2 chamados *pv2* e *lamb\_pv2*, é dado início aos cálculos das grandezas dos NAP's. Grau de Evidência – *me*, Grau de Certeza – *gc*, Grau de Contradição – *gct*, Grau de Contradição Normalizado – *mctr* e o Intervalo de Evidência Resultante – *fie*, exceto ao cálculo do Grau de Certeza Real – *gcr*. Dessa forma, cada NAP possui uma função específica para a definição das grandezas de cada Nó de Analise Paraconsistente.

O bloco das funções do código desenvolvido a partir do algoritmo NAP para cálculo das grandezas é descrito no Anexo I.

### 3. RESULTADOS E DISCUSSÃO

Neste capítulo, são apresentados gráficos e tabelas comparativos como meio de validação. Os resultados obtidos pela planta piloto experimental de processo industrial (figura 9) utilizando o processo de controle de razão convencional comparando com o controle adaptativo da razão, na leitura e comportamento tanto do fluxo primário quanto fluxo secundário.

Conforme foi descrito, a definição de como seria a rede análise paraconsistente adaptativa partiu de estudos realizados na configuração do controle de razão convencional e com a necessidade de adicionar ao processo o comportamento de ambas variáveis de fluxo tratando os sinais. Diante das análises para o desenvolvimento do diagrama (figura 11) foram obtidos valores de sintonia através de testes realizados no controle de fluxo do piloto da planta industrial para verificar o comportamento das estruturas computacionais e eletronicamente microcontrolada desenvolvida.

As variações de 0% a 90% e de 90% a 50% foram feitas para os valores do ponto de ajuste em diferentes etapas. Para cada etapa de variação do *Set Point*, foram observados graficamente o comportamento das respostas dos fluxos ( $y_1$  e  $y_2$ ) sinais desfavoráveis, as variáveis manipuladas ( $u_1$  e  $u_2$ ) sinais favoráveis. Ainda no desempenho da ECAR-LPA, as variações nos valores dos graus de evidência e os graus normalizados de contradição que atuam nos ganhos proporcionais ( $K_p$ ) e tempos integrativos ( $T_i$ ) dos controladores FY-101 e FY-102 da Planta Piloto de Processo Industrial utilizados nos testes foram observados.

Foram feitos inúmeros testes e dentre estes apresenta-se a seguir os resultados gráficos que mostram com mais clareza a diferença de atuação entre uma Estação de Razão Convencional e a Estação de Razão Adaptativa Paraconsistente - ECAR-LPA.

Nas figuras 17 e 18 são mostrados os resultados do comportamento das variáveis de controle na ocorrência de variações no valor do *Set Point* de 0% a 90% e de 90% a 50% utilizando uma Estação de Razão Convencional na planta piloto da estrutura Industrial utilizada nos testes. Portanto, na Figura 17 é mostrado o comportamento das variáveis do fluxo  $y_1$ , juntamente com a variável manipulada  $\mu_1$  do controle principal. Na Figura 18 é mostrado o comportamento das variáveis de fluxo  $y_2$ , juntamente com a variável manipulada  $\mu_2$  do controle secundário.

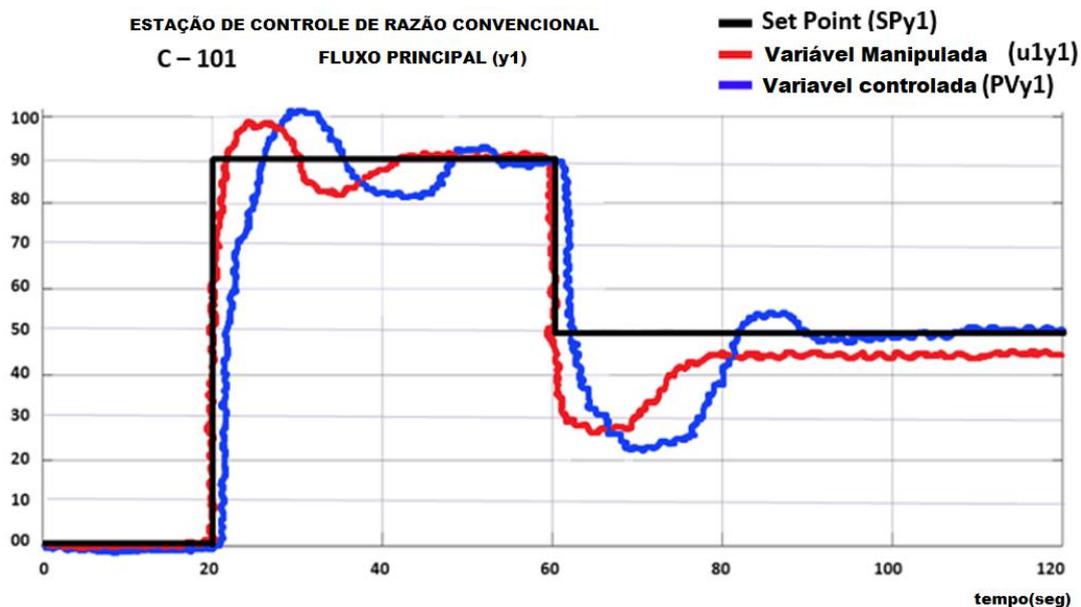


Figura 17 - Gráfico dos resultados do comportamento das variáveis de controle utilizando a estação de razão convencional na planta piloto experimental.

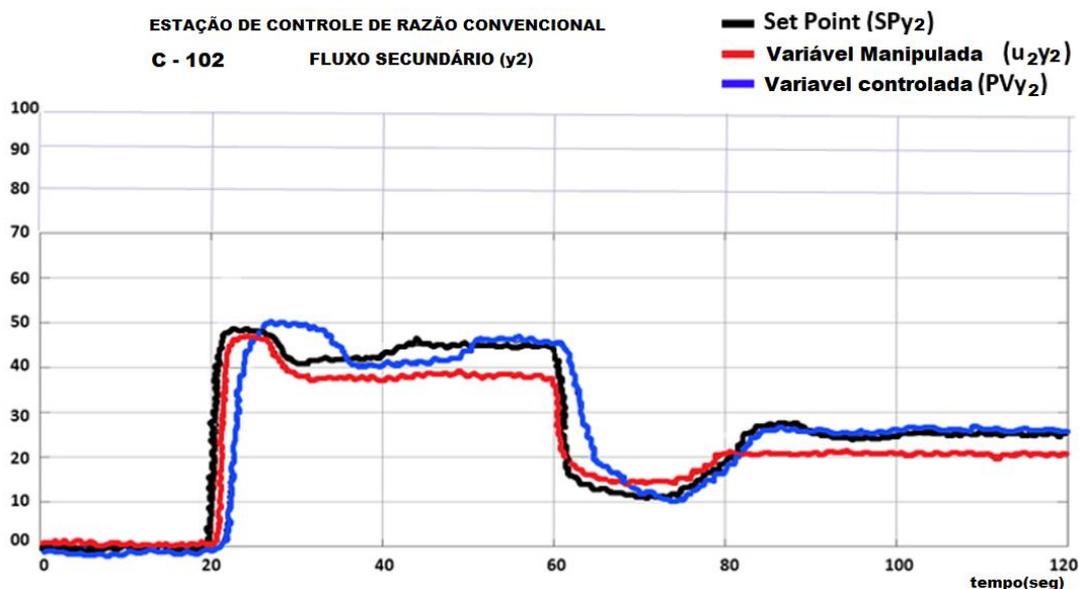


Figura 18 - Gráfico dos resultados do comportamento das variáveis de controle utilizando a estação de razão convencional na planta piloto experimental.

As Figuras 19 e 20 mostram os resultados das mesmas variáveis nas respostas às mesmas variações dos valores do set point quando a configuração anterior foi usada.

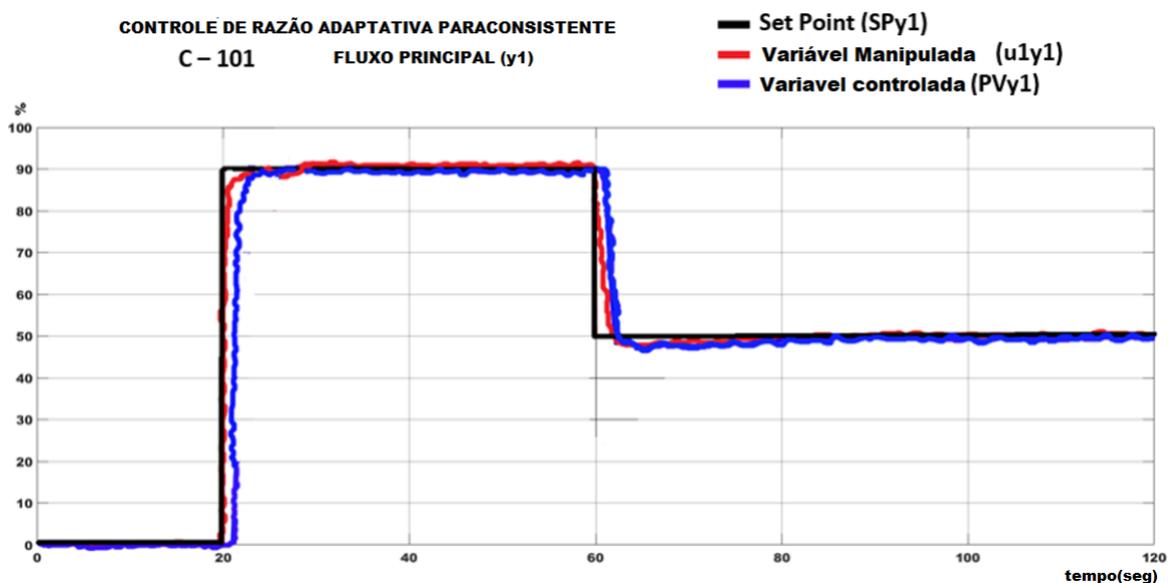


Figura 19 - Gráfico dos resultados do comportamento das variáveis de controle utilizando a Estação de Razão Adaptativa Paraconsistente (ECAR-LPA) na planta piloto experimental.

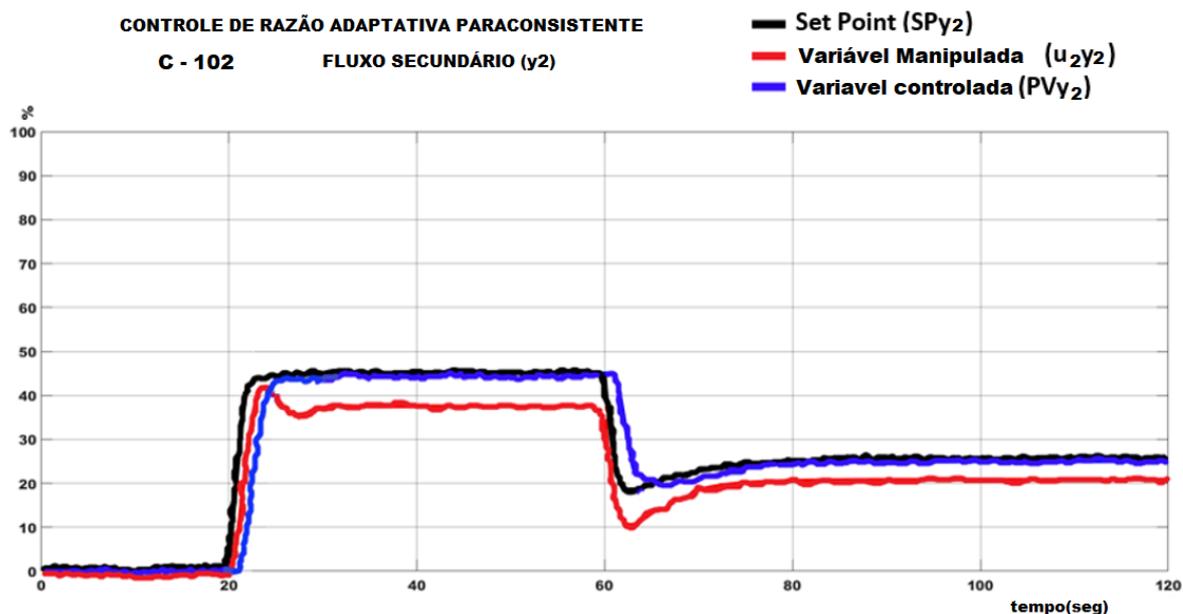


Figura 20 - Gráfico dos resultados do comportamento das variáveis de controle utilizando a Estação de Razão Adaptativa Paraconsistente (ECAR-LPA) na planta piloto

Nas figuras 19 e 20 os resultados dos comportamentos das variáveis na ocorrência de variações no valor do *Set Point* de 0% a 90% e de 90% a 50% utilizando o controle de Razão Adaptativa Paraconsistente (ECAR-LPA) na Planta Industrial utilizada nos testes. Na figura 19 é apresentado o comportamento da variável de

processo de fluxo  $y_1$  e variável manipulada  $u_1$  a partir do *feedback* do controle principal P1. Na figura 20 é apresentado o comportamento da variável de processo do fluxo  $y_2$  e variável manipulada  $u_2$  a partir do *feedback* do controle secundário P2.

Conforme descrito, um dos problemas no controle da razão é a perda de proporcionalidade entre as vazões causada pela modificação dos valores do *set point*, bem como pela ocorrência de distúrbios nas variáveis de saída. Nesta dissertação, foi apresentado um controle de razão adaptativo paraconsistente capaz minimizar estas problemáticas e realizar testes para fazer uma comparação na condição transitória causada pela modificação dos valores de set point entre uma Estação de razão construída de forma convencional e aquela que utiliza técnicas baseadas em lógica paraconsistente.

Os resultados gráficos das figuras 17 e 18 mostram o comportamento dos sinais de controle de razão usando uma estação de razão convencional. Os testes foram realizados em ambas as configurações nas mesmas condições das variações do Set Point e do fator de multiplicação da Razão ( $\alpha$ ), utilizando os mesmos controladores atuando no sistema que compõe a malha de controle de fluxo da planta piloto industrial.

Foram feitas variações nos valores do Set Point, cujo procedimento foi iniciado com valores zero no tempo de referência  $t = 0s$  e após 20 segundos deu-se um passo de 0% a 90%. Após 20 segundos do primeiro evento, uma nova variação do valor do Set point foi feita, passando de 90% para 50%. Os gráficos mostram as variações do Set Point ( $SP_{yn}$ ) e as respostas a essas variações com os sinais correspondentes da Variável Manipulada ( $PV_{yn}$ ) e da Variável Controlada ( $u_{yn}$ ) nos dois loops de controle de fluxo.

Pode-se observar através dos gráficos das figuras 17 e 18 que quando ocorrem as variações do Set point, o problema de desconexão de valores entre os dois processos P1-principal e P2-controlados aparece de forma muito agressiva na aplicação da Estação Razão convencional. Os gráficos mostrados nas figuras 19 e 20 indicam que na aplicação do controle de razão paraconsistente, quando ocorrem as variações do Set point, as discrepâncias são menores e o sistema obedece de forma mais equilibrada e controlada a proporcionalidade entre os dois fluxos.

### 3.1 Comparações de Resultados

Para comparar os valores dos fluxos 1 e 2, obtendo o erro relativo em cada estação de Razão, foi considerado o mesmo fator multiplicativo da razão ( $\alpha = 0,5$ ) e como referência, o instante de início da variação do set point valor é considerado.

Em relação ao controle Razão Convencional, apresentou menores valores de *overshooting* e *undershooting*. Pode-se observar nos gráficos das figuras 17 e 18 que a variável controlada na estação convencional apresentou um tempo de aproximadamente 52 segundos para entrar em equilíbrio com um *overshooting* de 10% e 8% de *undershooting* na Vazão P1. O controle de Razão Paraconsistente teve um tempo de aproximadamente 12 segundos para controlar os dois fluxos com um *overshooting* de 1% e *undershooting* de 0% no fluxo P1, como visto nas figuras 19 e 20.

Para a verificação dos valores de discrepância representados pela porcentagem do erro relativo entre as duas configurações, foi estabelecido um tempo de análise de 40 segundos com verificações realizadas em intervalos de 5 segundos. A Tabela 2 mostra os valores comparativos com a média da porcentagem de erro relativo encontrado entre os fluxos da Estação de Razão Convencional (CRS) e do controle de Razão Paraconsistente (ECAR-LPA).

Tabela 2. Valores comparativos e identificação do erro de proporcionalidade entre os fluxos da estação de razão convencional e da estação de Razão Adaptativa Paraconsistente (ECAR-LPA).

Estação de Razão Convencional

tempo (seg)	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100	115	120
PVy1 (%)	80	100	90	82	81	91	89	90	40	22	23	40	54	50	48	50	50	50
PVy2 (%)	45	50	40	40	40	45	45	45	30	10	11	18	25	22	25	25	25	25
Erro (%)	12	0.0	12	2.1	1.2	0.1	0.1	0.0	50	1.2	0.4	0.1	7.4	12	4.1	0.0	0.0	0.0

Percentual médio de erro relativo       $\Delta$ erro =5.760 %

Controle de Razão Adaptativa Paraconsistente (ECAR-LPA)

tempo (seg)	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100	115	120
PVy1 (%)	92	90	90	90	90	90	90	90	90	90	70	50	50	50	50	50	50	50
PVy2 (%)	45	44	45	45	45	45	45	45	45	45	35	25	25	25	25	25	25	25
Erro (%)	2.1	2.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Percentual médio de erro relativo       $\Delta$ erro =0.233 %

Os gráficos das figuras 19 e 20 mostram o comportamento das variáveis de processo ao alterar o set point com a aplicação do Controle Razão Adaptativa Paraconsistente - ECAR-LPA. Verifica-se que o ECAR-LPA apresentou os melhores resultados, com pequena sobreposição e maior tempo de similaridade de valores entre os fluxos quando o sistema estava no estado de transição devido à modificação dos valores do Set Point.

Com a inclusão do sinal de feedback do Fluxo Secundário  $y_2$  em sua configuração de algoritmos LPA2v, o ECAR-LPA apresenta maior estabilidade nos sinais com resposta a estabilidade mais rápida quando comparado ao ECAR-LPA. Com esta configuração especial, o desempenho do ECAR-LPA tornou-se adaptativo e, portanto, uma perturbação tanto no Set Point  $r_1$  e nos fluxos  $y_1$ -Main ou  $y_2$ -Controlados terá sua reação para fazer alterações nos valores de ganhos proporcionais e na integração dos tempos de controladores.

## 4. CONCLUSÃO

De acordo com a literatura, o maior problema em sistemas que utilizam controle de razão são as dissimilaridades que acontecem com a proporção dos valores dos fluxos na ocorrência das variações do *Set point* e de perturbações inesperadas nas variáveis de saída. Com o objetivo de propor novas técnicas para amenizar este problema, foi desenvolvido nesta pesquisa uma estrutura computacional que utiliza redes de análise construídas com algoritmos baseados em Lógica Paraconsistente e implementada em plataforma de sistemas microcontrolados. Os resultados do protótipo microprocessado, que foi denominado de Estação de Controle Adaptativo de Razão Paraconsistente - ECAR-LPA apresentou resultados significativos e satisfatórios no controle da malha do sistema de controle de mistura quando comparado ao modelo de controle de razão convencional, apresentando a qualidade de atuar diretamente nos ganhos proporcionais ( $K_p$ ) e nos tempos de integração ( $T_i$ ) dos dois controladores de malha. Verificou-se que ECAR-LPA formou um controle de razão adaptativo que permitiu o monitoramento contínuo de anomalias e distúrbios de mudança de ponto de ajuste que podem ter ocorrido nos fluxos. Em geral, a ECAR-LPA mostrou-se bastante eficiente e permitiu detectar mudanças no set point e nos fluxos, ajustando a sintonia rapidamente as diferenças com boa uniformidade entre os dois processos.

### 4.1 Trabalhos Futuros

Considerando que nestes primeiros testes a ECAR-LPA respondeu apropriadamente a quaisquer mudanças de valor ocorrendo no sistema, ajustando fluxos de correção para manter a razão adequada, novas pesquisas deverão ser desenvolvidas no futuro. Entre estas pode-se destacar:

- Investigação de novas configurações da RAP da ECAR-LPA que possam responder com maior eficiência ruídos indesejáveis nos sinais da planta de controle da mistura.
- Desenvolvimento de ECAR-LPA com algoritmos paraconsistentes implementados em modernas linguagens de programação tais como, Sharp, C++ e Python.

- Desenvolvimento de tipos de ECAR-LPA com algoritmos paraconsistentes implementados sistemas residentes de controladores funcionando com microcontroladores mais potentes que o utilizado nesta pesquisa.
- Desenvolvimento de técnicas de sistemas de informação que possam melhorar o tratamento de dados e protocolos de transmissão de sinais de informação que permitam a ação da ECAR-LPA em sistemas remotos e assim ajustando-a a revolução 4g.

## REFERÊNCIAS

ABE, J. M. **Fundamentos da Lógica Anotada**. Orientador: Prof Dr. Newton Carneiro Affonso da Costa, Tese (Doutorado) - Filosofia. Universidade de São Paulo. [S.l.]. p.139, 1992.

ÅSTRÖM, K. J.; TORE, H. **PID Controllers: Theory, Design, and Tuning, ISA-The Instrumentation, Systems, and Automation Society**...: Research Triangle Park, 2nd (livro), p.343. ISBN 1-55617-516-7. 1995.

COELHO, M. S. Hybrid PI controller constructed with paraconsistent annotated logic., **Control Engineering Practice**, 84 artigo, 112-124. Março, 2019. ISSN 112-124. Disponível em: <<https://doi.org/10.1016/j.conengprac.2018.11.007>>.

DA COSTA, N. C. A. On the theory of inconsistent annotated logic for applications in expert systems. **Notre Dame Journal Form Log**, artigo 497-510, 1974. ISSN 497-510.

DA COSTA, N. C. A. The Paraconsistent Logic of Quantum Superpositions. **Foundations of Physics**, 43 (artigo), 845-858, Maio, 2013 Disponível em: <<http://dx.doi.org/10.1007/s10701-013-9721-9>>.

DA SILVA FILHO, J. I. Paraconsistent Analysis Network applied in the treatment of Raman spectroscopy data to support medical diagnosis of skin cancer. **Medical Biological. Engineering & Computing (artigo)**, p. 1453-1467, Março, 2016.

DA SILVA FILHO, J. I. Undulatory Theory with Paraconsistent Logic: Quantum Logical Model with Two Wave Functions. **Journal of Quantum Information Science**, v. 06, p. 143-180, 2016.

DA SILVA FILHO, J. I. **A Importância das Lógicas Não-Clássicas**. Santos, p. 2.

DA SILVA FILHO, J. I.; ABE, J. M.; TORRES, G. L. **Inteligencia Artificial**. 1ª. ed. Rio de Janeiro: **LTC-Livros Técnicos e Científicos Editora S.A**, ISBN8521616317, p.328, 2008.

DA SILVA FILHO, J. I.; LAMBERT-TORRES, G.; ABE, J. M. Uncertainty treatment using paraconsistent logic introducing paraconsistent artificial neural networks, **ed. los Pr Inc**. ISBN-10 1607505576, p. 311, Junho,2010.

FERNANDES, C. L. M. **LÓGICA PARACONSISTENTE APLICADA EM SISTEMAS DE AUTOMAÇÃO E CONTROLE**. Orientador: Prof.Dr. João Inácio da

Silva Filho, Dissertação (Mestrado) - Engenharia Mecânica - Universidade Santa Cecília. Santos, p. 109. 2012.

GARCIA, D. V. Analysis of Raman spectroscopy data with algorithms based on paraconsistent logic for characterization of skin cancer. **Vibrational Spectroscopy (artigo)**, n. 103, Julho, 2019.

HÄGGLUND, T. The Tracking Ratio Station. **Control Engineering Practice (artigo)**, n.69, 122-130, Dezembro, 2017. ISSN 0967-0661. Disponível em: <<https://doi.org/10.1016/j.conengprac.2017.09.016>>.

HÄGGLUND., T. The Blend Station - A new ratio control structure. **Control Engineering Practice (artigo)**, n.9, 1215-1220, Novembro, 2001.

JUNIOR, M. M. **Uma Ferramenta de Apoio à Decisão na escolha de Fornecedores de Tecnologia de Informação Baseado na Lógica Paraconsistente**. Orientador: Jair Minoro Abe, 118p, Dissertação (Mestrado) - Engenharia de Produção - Universidade Paulista. São Paulo, 2006.

MARLIN, T. E. Process Control: Designing processes and control systems for dynamic performance. 2nd International. ed. Singapore: McGraw-Hill Book Companies, Inc., 2000.

MATOS, S. L. D. C. **Rede de Análise Paraconsistente Aplicada em Malha de Controle de Razão para um Processo Industrial**, Orientador: Dr. João Inácio da Silva Filho, 130p, Dissertação (Mestrado) - Engenharia Mecânica - Universidade Santa Cecília, Santos, 2018.

MORARI, M. ZAFIRIOU, E. Robust Process Control. ed. **Prentice Hall Australia (livro)**, p.512, ISBN-10 0137819560, Dezembro, 1988.

MURRILL, P. W. Fundamentals of process control theory, **ISA: The Instrumentation, Systems, and Automation Society (livro)**, p. 328, 3rd, ISBN-10 155617683X, Fevereiro, 2000.

OGATA, K. **Modern control engineering**. 5nd. ed. Victoria: Prentice Hall Inc., 2014. ISBN 0-13-615673-8.

OU, L.; YU, L.; ZANG, W. A ratio control scheme decoupling disturbance response from set-point response. **ISA Transactions (artigo)**, Janeiro, 2007.

PRIEST, G. **An Introduction to Non-Classical Logic**. 2nd. **Graham Priest (livro)**: University of Melbourne, Cambridge, ISBN 9780511801174, 2008.

ROZA, R. H. Revolução Informal e os Avanços Tecnológicos da Informática e das Telecomunicações. **Ciência da Informação - Revista**, Maceió, v. IV, n. 3, p. 3-11, DEZ 2017.

VISIOLI., A. Design and tuning of ratio controller. **Control Engineering Practice**, 2005.

## ***ANEXO I – Código fonte da Rede de Análise Paraconsistente desenvolvida no IDE Arduino***

```

float grau_evidencia(float a, float b ){
  float c;
  c = ((a - b)+1)/2;
  return c; }

////////////////////////////////////
float grau_certeza(float c, float d ){
  float e;
  e = c - d;
  return e;
}

////////////////////////////////////
float grau_contradicao(float f, float g ){
  float h;
  h = (f + g) - 1;
  return h;
}

////////////////////////////////////
float grau_certeza_real_1(float i, float j){
  double k,x1;
  if (i >= 0){
    x1 = 1-gc_pan1;
    k = 1 - sqrt( pow(x1,2) + pow(gct_pan1,2) );
  }
  else{
    x1 = 1-(gc_pan1*(-1));
    k = sqrt( pow(x1,2)+ pow(gct_pan1,2))- 1;
  }
  return k;
}

float grau_certeza_real_2(float i, float j){
  double k,x2;
  if (i >= 0){

```

```

    x2 = 1-gc_pan2;
    k = 1 - sqrt( pow(x2,2) + pow(gct_pan2,2) );
}
else{
    x2 = 1-(gc_pan2*(-1));
    k = sqrt( pow(x2,2) + pow(gct_pan2,2) ) - 1;
}
return k;
}
float grau_certeza_real_3(float i, float j){
    double k,x3;
    x3 = 1-gc_pan3;
    if (i >= 0){
        k = 1 - sqrt( pow(x3,2) + pow(gct_pan3,2) );
    }
    else{
        x3 = 1-(gc_pan3*(-1));
        k = sqrt( pow(x3,2) + pow(gct_pan3,2) ) - 1;
    }
    return k;
}
float grau_certeza_real_4(float i, float j){
    double k,x4;
    if (i >= 0){
        x4 = 1-gc_pan4;
        k = 1 - sqrt( pow(x4,2) + pow(gct_pan4,2) );
    }
    else{
        x4 = 1-(gc_pan4*(-1));
        k = sqrt( pow(x4,2) + pow(gct_pan4,2) ) - 1;
    }
    return k;
}
float grau_certeza_real_5(float i, float j){
    double k,x5;

```

```
if (i >= 0){
    x5 = 1-gc_pan5;
    k = 1 - sqrt( pow(x5,2) + pow(gct_pan5,2) );
}
else{
    x5 = 1-(gc_pan5*(-1));
    k = sqrt( pow(x5,2) + pow(gct_pan5,2) ) - 1;
}
return k;
}
////////////////////////////////////////////////////////////////
float grau_evidencia_real(float m){
    return (m + 1)/2;
}
////////////////////////////////////////////////////////////////
float grau_contradicao_normalizado(float n ){
    return (n + 1)/2;
}
```

## ANEXO II – Placa Arduino UNO R3.

