COMPARAÇÃO DE MÉTODOS DE DETECÇÃO DE FALHAS EM PROCESSOS QUÍMICOS UTILIZANDO INTELIGÊNCIA ARTIFICIAL

FAILURES DETECTION METHODS IN CHEMICAL PROCESS USING ARTIFICIAL INTELLIGENCE

TELES, Athus Costa^{1*}; FREITAS Ana Clara de Pádua²; RODRIGUES, Antonio Cruz³;^{1,2,3} Universidade Federal dos Vales do Jequitinhonha e Mucuri, Instituto de Ciência e Tecnologia, Departamento de Engenharia Química

* Autor correspondente

e-mail: athusteles@hotmail.com

Received 13 January 2019; received in revised form 09 March 2019; accepted 13 March 2019

RESUMO

Qualquer mudança atípica em um procedimento pode ser caracterizada como uma “falha” e essa, consequentemente, pode levar a perdas econômicas e/ou aumento de custos operacionais, já que, muitas vezes, ao acontecer necessita-se de uma interrupção do processo. Então, a preocupação com a qualidade e segurança nos processos vêm estimulando estudos de monitoramento e diagnóstico de falhas em equipamentos industriais. Diante disso, o presente artigo tem como objetivo aplicar três diferentes métodos (Redes Neurais Artificiais (*Artificial Neural Networks* - ANN), a Lógica Nebulosa (*Fuzzy Logic* - FL) e a máquina de vetores de suporte (*Support Vector Machine* - SVM)) como sistemas de detecção e classificação de falhas no processo químico de produção do ciclopentenodiol, bem como diagnosticar essas técnicas de inteligência artificial a fim de compará-las quanto à eficácia. Toda a investigação foi realizada modelando um reator com cinética de Van der Vusse quando provocado quatro tipos de falhas, na concentração de um reagente (Falha 1), no sensor que mede a concentração do produto desejado e a temperatura (Falha 2 e 3) e no emperramento de uma válvula (Falha 4). Os dados utilizados na metodologia basearam-se em informações históricas quantitativas e qualitativas. Todos os métodos conseguem detectar as falhas, mas em tempos diferentes. O ANN é o que detecta mais rapidamente todas as falhas. O SVM detecta alguns minutos depois, mas com qualidade na precisão, mesmo apresentando um baixo esforço computacional em relação ao ANN. O fuzzy, na maioria dos casos estudados, demora horas para ser capaz de detectar a alteração no sistema o tornando o menos efetivo dentre as técnicas de inteligência artificial utilizadas.

Palavras-chave: Detecção de falhas, cinética de Van der Vusse, Redes Neurais Artificiais, Lógica Nebulosa, Máquina de Vetores de Suporte.

ABSTRACT

Any atypical change in a procedure can be characterized as a “failure”. Consequently, it may result in economic losses and/or a rise of the operational cost, because most of the times the process will need to be interrupted. Therefore, the concern with the quality and security of the processes has stimulating studies of diagnosis and monitoring failures in industrial equipments. In light of this, the present article has as purpose to apply three different methods (*Artificial Neural Networks* - ANN, *Fuzzy Logic* – FL and *Support Vector Machine* – SVM). All of those were applied as detection and classification systems of failure in the processes of a case study in order to diagnose these artificial intelligence techniques so that the efficiency of each method can be compared. All investigation is done by modeling a reactor of Van der Vusse’s kinetic causing four types of failures, in the concentration of a reagent (failure 1), in the sensor which measures the concentration of the interested product and temperature (failure 2 and 3), and in the valve locking (failure 4). The data used in this methodology is based in quantitative and qualitative historical information. All methods are able to detect failures, but in different times. ANN is the one which detects faster all the failures. SVM detects some minutes later, however with good precision, even though this method uses less computational effort compared to ANN. *Fuzzy*, in the most of the cases studied, takes hours to detect any change in the system, which makes this one the less effective among the ones studied.

Keywords: Failure detection, Van der Vusse’s kinetic, *Artificial Neural Networks*, *Support Vector Machine*, *Fuzzy Logic*.

1. INTRODUÇÃO

O monitoramento e o diagnóstico de falhas em equipamentos industriais em geral vêm sendo estimulado devido à crescente preocupação com a segurança, a confiabilidade e a qualidade durante os processos de produção, em que uma pequena parada no processo pode levar a perdas econômicas e aumento de custos operacionais (Bueno *et al.*, 2006).

Qualquer mudança atípica durante um processo pode ser caracterizada como uma “falha” e esta pode processar-se de forma brusca, o que leva a interrupção do processo, ou de forma gradativa (Simani e Fantuzzi, 2000). Dessa forma, visando reduzir os custos de processos, monitora-se as possíveis falhas tendo em vista sua detecção e posterior reparo em um curto período de tempo.

Há diversos tipos de falhas, porém as mais comuns encontradas em equipamentos industriais são devido a impactos, fadiga, vibrações, atrito, mau funcionamento em sensores (na medição de temperatura, pressão, vazão, nível e outros), atuadores, perturbações indesejadas no processo, emperramento de válvulas dentre outras razões (Tebaldi *et al.*, 2006).

Nos últimos anos, as técnicas dedicadas a detecção de falhas têm tido uma grande evolução devido ao grande avanço nos equipamentos de processamento de sinais. Sendo que a detecção de falhas pode ser dividida em dois grupos: métodos qualitativos, onde não há necessidade de um modelo (Chen e Patton, 1993), e métodos matemáticos, também conhecidos como quantitativos (Kabeya, 1998).

No monitoramento e detecção de falhas, algumas técnicas de Inteligência Artificial como: as Redes Neurais Artificiais (*Artificial Neural Networks* - ANN), a Lógica Nebulosa (*Fuzzy Logic* - FL) e a máquina de vetores de suporte (*Support Vector Machine* - SVM) têm sido aplicadas.

Além das técnicas utilizadas neste trabalho (SVM, FL e ANN), existem outras como exemplo a Análise de componentes principais-PCA, Controle Estatístico de Processos- SPC, Algoritmo Genético e Filtro de Kalman para sistemas de detecção, diagnóstico e correção de falhas (Sartori *et al.*, 2012).

Neste trabalho, foi apresentado um estudo de caso para comparar as três promissoras técnicas de inteligência artificial para

a detecção e diagnóstico de falhas. Todas as metodologias envolvidas aprendem pela experiência, assim detectando as falhas apresentadas em um sistema.

2. METODOLOGIA E DESENVOLVIMENTO

As metodologias utilizadas para a detecção e diagnóstico de falhas foram: máquinas de vetores de suporte, redes neurais artificiais e lógica nebulosa.

Ao simular e obter os dados do sistema passando por uma operação normal e quando submetido a falha, foi realizado o treinamento do conjunto destes dados e, posteriormente, utilizaram-se as metodologias de detecção para detectar e classificar as ocorrências de falhas. A detecção deste comportamento indesejado aconteceu com a identificação e classificação de padrões que as três metodologias apresentam de maneiras distintas. O sistema foi submetido a quatro falhas comuns que ocorrem em diferentes variáveis, sendo estas: perturbação na entrada do processo (concentração de alimentação do reagente), falha no sensor de concentração do produto desejado e de temperatura, e falha no atuador (válvula de entrada do reagente). A representação do sistema que sofreu as quatro falhas pode ser observado na Figura 1.

Figura 1. Fluxograma de detecção de falhas.

2.1. Máquina de vetores de suporte (SVM)

O SVM possui dois métodos comuns de simplificação das tomadas de decisões na classificação multiclasse, estes são: *one-against-one* (OAO) e *one-against-all* (OAA).

O método “*one-against-one*” consiste em construir um SVM para cada par de classes. Assim, para um problema com *c* classes, os SVMs são treinados para distinguir as amostras de uma classe das amostras de uma outra classe. Normalmente, a classificação de um padrão não conhecido é feita de acordo com uma votação, onde cada SVM vota por uma classe.

O método *one-against-all* consiste na construção de um SVM por classe, que é treinado para distinguir as amostras de uma classe das amostras de todas as classes restantes. Normalmente, a classificação de um padrão desconhecido é feita de acordo com o máximo de saídas entre todos os SVMs (Milgram, J; Cheriet, M; Sabourin, R.).

A teoria do aprendizado estatístico oferece uma base para estudar o problema de prever comportamentos, adquirir conhecimento e determinar com base no conjunto de dados o comportamento de determinado processo. O método gera um hiperplano que representa perfeitamente a função subjacente no espaço destino. Pode-se formular o problema de aprendizado supervisionado partindo de um conjunto de dados binários (x_i, y_i) em que o problema consiste em encontrar uma função que reduza ao mínimo possível a distância (erro) em novos dados, detectando a classe a que estes pertencem (Jakkula, 2018).

A Equação 1, que representa o hiperplano, possui um produto escalar entre os vetores w e x , sendo que $\frac{b}{\|w\|}$ corresponde a distância do hiperplano em relação ao seu ponto de partida. É possível observar na Figura 2 os dados de treinamento, os hiperplanos de separação e os vetores descritos (Boswell, 2002).

Figura 2. Distância entre os hiperplanos H_1 e H_2 .

O hiperplano (w, b) que separa os dados em duas áreas H_1 e H_2 , podendo ser observados na Figura 2, nos dá uma função demonstrada na Equação 2.

$$f(x) = \text{sign}(w * x + b) \quad (\text{Eq.2})$$

Espera-se que esta função classifique corretamente estes dados de treinamento e outros dados que ainda não tenham sido observados ainda. Pode-se definir o hiperplano canônico como sendo aquele que separa os dados do hiperplano por uma distância de, no mínimo, 1 unidade (Boswell, 2002). Então, representa-se as Equações 3 e 4:

$$x_i * w + b \geq +1 \text{ em que } y_i = +1 \quad (\text{Eq.3})$$

$$x_i * w + b \leq -1 \text{ em que } y_i = -1 \quad (\text{Eq.4})$$

Podendo ser reescrita de forma mais compacta de acordo com a Equação 5:

$$y_i(x_i * w + b) \geq 1 \quad (\text{Eq.5})$$

Considerando as restrições, a independência de b e fazendo uso dos multiplicadores de Lagrange, pode-se obter um classificador que interpole a margem, representado na Equação 6 (Lorena e Carvaho, 2003).

$$f(x) = h(\langle w, x \rangle + b); h(z) = \begin{cases} -1 & : z < -1 \\ z & : -1 \leq z \leq 1 \\ +1 & : z > 1 \end{cases} \quad (\text{Eq.6})$$

2.2. Redes Neurais

Pode-se definir um neurônio artificial como uma unidade de processamento matemático simples, sendo que esta unidade recebe no mínimo uma entrada e a transforma em saída. Existe um peso associado a cada entrada, o que determina a sua intensidade (Tubb, 1993). A representação do neurônio artificial é dada pela Figura 3.

Figura 3. Representação não-linear de um neurônio.

A partir da representação não-linear do neurônio define-se alguns elementos chaves:

- **sinapses** – são determinadas por um peso (w), podendo representar sua

intensidade. A função do peso w_{kj} é multiplicar o sinal X_j da entrada da sinapse j , interligada a um neurônio k .

- **somatório** – o somatório adiciona as entradas ponderadas por seus respectivos pesos, dado pela seguinte equação:

$$u_k = \sum_{i=1}^n w_{ij} X_j \quad (\text{Eq. 7})$$

- **limiar (threshold), q_k** – esta variável tem um papel fundamental na saída do neurônio. Quando o valor de u_k for menor que este limiar, a saída será inibida, do contrário o neurônio ficará ativo.
- **função de ativação, $j(\cdot)$** – a função de ativação limita a amplitude da saída do neurônio, sendo que a entrada é normalizada em um intervalo fechado, $[0,1]$ ou $[-1,1]$.
- **saída** – é a saída do sinal do neurônio y_k , sendo (Equação 8):

$$y_k = j(u_k - q_k) \quad (\text{Eq. 8})$$

onde j é a função de ativação.

2.3. Lógica Nebulosa

O primeiro passo na representação de conjunto *fuzzy* é a escolha da função de pertinência. Tal escolha depende do problema a ser modelado e da capacidade disponível para se processar o que se deseja. Funções não lineares podem ser mais eficientes para problemas mais complicados, porém demandam um poder computacional muito maior (Sanchez, 2018).

Diferente da lógica convencional, a lógica *fuzzy* pressupõe que qualquer tipo de variável (altura, temperatura, velocidade, etc) admite graus de pertinência e com isso tenta modelar o senso comum do ser humano. A pertinência de um dado elemento com relação a um conjunto refere-se ao fato de tal elemento pertencer ou não a esse conjunto. (Ortega, 2001). Tal lógica também trata de valores que variam entre 0 e 1, e não só do verdadeiro ou falso como a lógica Booleana. Assim, uma pertinência de 0,5 pode representar meio verdade, logo 0,9 e 0,1, representam uma quase verdade e um quase falso, respectivamente (Silva, 2005).

Um conjunto pode ser representado como (Equação 9):

$$A = \{(x, \mu_A(x)) | x \in X\} \quad (\text{Eq. 9})$$

E suas operações podem ser assim apresentadas:

- O conjunto *fuzzy* A é um subconjunto de um conjunto *fuzzy* B se o grau de pertinência de cada elemento do conjunto universo U no conjunto A é menor ou igual que seu grau de pertinência no conjunto B ; ou seja, para todo $x \in U$, $\mu_A(x) \leq \mu_B(x)$ e indicamos $A \subseteq B$;
- Os conjuntos *fuzzy* A e B são iguais se $\mu_A(x) = \mu_B(x)$ para todo elemento $x \in U$ e indicamos: $A = B$;
- Os conjuntos *fuzzy* A e B são diferentes se $\mu_A(x) \neq \mu_B(x)$ para no mínimo um $x \in U$ e indicamos $A \neq B$;
- O conjunto *fuzzy* A é um subconjunto próprio do conjunto *fuzzy* B quando A é um sub-conjunto de B e $A \neq B$, isto é, $\mu_A(x) \leq \mu_B(x)$ para todo $x \in U$ e $\mu_A(x) < \mu_B(x)$ para no mínimo um $x \in U$ e indicamos $A \subset B$ se e somente se $A \subseteq B$ e $A \neq B$;
- O complemento de um conjunto *fuzzy* A em relação ao conjunto universo U é indicado por A' e a função de pertinência é definida como: $\mu_{A'}(x) = 1 - \mu_A(x)$ para todo $\mu_A(x) \in U$;
- A união de dois conjuntos *fuzzy* A e B é um conjunto *fuzzy* $A \cup B$ tal que para todo $x \in U$, $\mu_{A \cup B}(x) = \max[\mu_A(x), \mu_B(x)]$;
- A intersecção de dois conjuntos *fuzzy* A e B é um conjunto *fuzzy* $A \cap B$ tal que para todo $x \in U$, $\mu_{A \cap B}(x) = \min[\mu_A(x), \mu_B(x)]$ (Abar, 2004).

Um sistema *fuzzy* pode ser usado para modelar um componente ou um módulo de um sistema maior. Depois de construídos, os modelos obtidos são usados em um simulador voltado para detecção de falhas. Há diversos pacotes para auxílio de projetos e implementações desse método como: *Fuzzy toolbox* do matlab, *sciFLT* do scilab, *X-Fuzzy*, *fuzzyClips*, FLIE (*Fuzzy Logic Inference Engine*) do Fabro. O processo de classificação *fuzzy* pode ser dividido em três estágios, que são *fuzzyficação*, *inferência* e *desfuzzyficação*.

(Sanchez, 2018), como representa a Figura 4.

Figura 4. Sistema de controle fuzzy.

Na *fuzzificação* são definidas as variáveis linguísticas e as funções de pertinência para cada variável de entrada do processo. A etapa de inferência é onde as regras são analisadas, de acordo com as variáveis de entrada e saída. E por fim, na *defuzzificação*, as regiões resultantes são convertidas em valores para a variável de saída do sistema (Albertos, 2002).

2.4. Estudo de caso

Um reator para produção de ciclopentanol com cinética de Van der Vusse é investigado quando provocado quatro tipos de falhas a fim de comparar e selecionar o melhor método para detectá-las. A reação principal é a transformação do reagente ciclopentadieno (A) no produto ciclopentenodiol (B), que, numa reação em série, produz diciticlopentanodiol (C). Além disso, numa reação paralela, o ciclopentadieno forma um produto indesejado, diciticlopentadieno (D). Todas as reações são exotérmicas irreversíveis e as taxas k_1, k_2 e k_3 dependem exponencialmente da temperatura do reator de acordo com a lei de Arrhenius, descritas pelo esquema reacional abaixo (Equação 10) (Praxedes e Ravagnani, 2008).

O reator é contínuo de tanque agitado e não adiabático e não isotérmico (CSTR encamisado) e foi modelado considerando o volume (V) e propriedades físicas constantes. A corrente de entrada do reator contém somente o reagente A, cuja concentração é C_{A0} . A temperatura do reator (T) e da camisa de resfriamento (T_c), a vazão de calor removido ($\dot{Q}_J$) as concentrações dos produtos ciclopentanodiol

(C_B), diciticlopentanodiol (C_C), diciticlopentadieno (C_D) e a final do ciclopentadieno (C_A) foram medidos através de sensores, e utilizados pelas metodologias de detecção de falhas, Fuzzy, SVM e ANN. Na Figura 5, um reator de Van der Vusse é esquematizado por Reis e Oliveira-Lopes (2007).

Figura 5. Reator de Van der Vusse

Os balanços de massa e energia do processo para cada componente é dado pelas Equações 11 a 16.

$$\frac{dC_A(t)}{dt} = \frac{(C_{A0} - C_A(t))\dot{F}}{V} - K_1 C_A(t) - K_3 C_A(t)^2 \quad (\text{Eq. 11})$$

$$\frac{dC_B(t)}{dt} = \frac{(-C_B(t))\dot{F}}{V} + K_1 C_A(t) - K_2 C_B(t) \quad (\text{Eq. 12})$$

$$\frac{dC_C(t)}{dt} = \frac{(-C_C(t))\dot{F}}{V} + K_2 C_B(t) \quad (\text{Eq. 13})$$

$$\frac{dC_D(t)}{dt} = \frac{(-C_D(t))\dot{F}}{V} + \frac{K_3 C_A(t)^2}{2} \quad (\text{Eq. 14})$$

$$\frac{dT(t)}{dt} = \frac{-1}{\rho \cdot c_p} ((K_1 C_A(t) H_1) + (K_2 C_B(t) H_2) + (K_3 C_A(t)^2 H_3)) + \frac{(T_0 - T)\dot{F}}{V} + \frac{k_w A (T_c - T)}{\rho \cdot c_p \cdot V} \quad (\text{Eq. 15})$$

$$\frac{dT_c(t)}{dt} = \frac{1}{m_k \cdot c_{pk}} (\dot{Q}_J + k_w A (T_c - T)) \quad (\text{Eq. 16})$$

Para a simulação foi adicionado um ruído Gaussiano de média zero e variância 1×10^{-5} para concentração e 1×10^{-3} para as outras medidas. As variáveis manipuladas foram a vazão de entrada do reagente A ($\dot{F}$) e a quantidade de calor removido pelo fluido refrigerante ($\dot{Q}_J$) sujeitas às restrições na forma de limite superior e inferior apresentadas na Equação 17.

$$50 \frac{L}{h} \leq \dot{F} = u_1 \leq 350 \frac{L}{h} \quad (Eq. 17)$$

$$-8500 \frac{KJ}{h} \leq \dot{Q}_J = u_2 \leq 0 \frac{KJ}{h}$$

Os valores dos parâmetros utilizados foram os mesmos do artigo de Klatt e Engell (1998) e se encontram na Tabela 1.

Tabela 1. Parâmetros do modelo

$k_{20} = 1,287 \times 10^{12} / h$
$k_{30} = 9,043 \times 10^9 L / molA \cdot h$
$\frac{E_1}{R} = -9758,3 K$
$\frac{E_2}{R} = -8560 K$
$H_1 = 4,4 kJ / molA$
$H_2 = -11 kJ / molA$
$H_3 = -41,85 kJ / molA$
$\rho = 0,9342 kg / L$
$c_p = 3,01 kJ / kg \cdot K$
$c_{pk} = 2 kJ / kg \cdot K$
$k_w = 4032 kJ / m^2 \cdot h \cdot K$
$A = 0,215 m^2$
$m_k = 5 kg$
$V = 10 L$
$T_0 = 403,15 K$
$c_{A0} = 5,1 molA / L$
$u_1 = 112 L / h$
$u_2 = -2856,9052 kJ / h$

3. RESULTADOS E DISCUSSÃO:

Foi realizado o treinamento das Redes Neurais (ANN) utilizando os dados obtidos através da simulação de operação normal e operações faltosas, com diferentes estruturas de camadas e nós. Primeiramente, na detecção realizada através do modelo obtido do treinamento de rede neural com duas camadas ocultas com dez nós cada para identificação da perturbação da concentração do reagente A na entrada, C_{Af} (Falha 1), observou-se alguns erros na identificação falha, como a identificação de uma perturbação na temperatura. Posteriormente, foi realizado o treinamento de rede neural com três camadas ocultas, tal que a primeira e a terceira camada possuem dez nós e a camada intermediária é composta por vinte nós. Neste caso, porém, não foi possível a detecção do emperramento da válvula (Falha 4). Então, foi escolhido para comparação com os outros métodos o modelo ANN de uma camada oculta com vinte nós já que melhor detectou as falhas, representando o estudo de caso. Onde

uma possível explicação para resultado é que o modelo com mais camadas pode estar com poucos nós, assim, necessitando um melhor estudo quanto ao número de nós nas camadas.

Dois tipos de SVM também foram testados, o modelo OAO (*one-against-one*) e o modelo OAA (*one-against-all*). A detecção de todas as falhas foi mais rápida com o modelo OAO, pois este é mais modular e consegue agilizar o processo de detecção e de tomada de decisão. Logo, ele foi o selecionado para posterior comparação com os demais métodos.

Foram construídos os gráficos da detecção e diagnóstico através das diferentes metodologias (ANN, SVM e *fuzzy*), que serão representados, respectivamente, para as diversas falhas.

Para a perturbação em C_{Af} , é possível observar que o ANN detectou essa falha mais rapidamente que as outras metodologias, antes de 2,1 horas (2 intervalos de amostragem). O SVM, apesar de apresentar um *outlier*, levou poucos instantes para detectar a falha. O *fuzzy* foi o que apresentou o maior tempo para detecção, como apresentado na Figura 6.

Figura 6. Detecção da Falha 1 utilizando as metodologias (ANN, SVM e fuzzy).

Na detecção da falha 2 pelos métodos descritos representados pelos gráficos da Figura 7, nota-se que a modelagem por Redes Neurais ainda se mostrou mais eficiente entre os métodos e o *fuzzy*, novamente, apresentou a detecção mais demorada. O SVM levou poucos instantes a mais que o ANN para detectar a falha.

Figura 7. Detecção da Falha 2 utilizando as metodologias (ANN, SVM e fuzzy).

Apesar do outlier apresentado nos dados de detecção e diagnóstico para a falha na temperatura (falha 3) gerado no gráfico do modelo ANN, o método detectou a falha mais rapidamente quando comparado aos outros. O SVM, por sua vez, demorou aproximadamente três minutos a mais para detectar, porém foi mais preciso na sua análise. Já o fuzzy levou mais de uma hora para perceber a detecção. Os comportamentos descritos estão demonstrados na Figura 8.

Figura 8. Detecção da Falha 3 utilizando as metodologias (ANN, SVM e fuzzy).

No emperramento da válvula (Falha 4), todos os métodos demonstraram um bom desempenho. O fuzzy, neste caso, foi o que reconheceu mais rapidamente a falha. O ANN manteve o padrão de alguns instantes para a detecção e, o SVM apresentou um tempo levemente maior de detecção em relação aos

demais. Esses comportamentos estão representados na Figura 9.

Figura 9. Detecção da Falha 4 utilizando as metodologias (ANN, SVM e fuzzy).

4. CONCLUSÕES:

O ANN foi considerado o melhor modelo na detecção de falhas, porém ele tem um esforço computacional muito maior quando comparado aos outros métodos. O SVM, apesar de levar alguns minutos a mais na detecção, pode ser considerado uma boa alternativa, visto que possui um desenvolvimento e aplicação mais simples e não há necessidade de criação de rótulos de saída. Quando se necessita de qualquer modificação no processo de detecção, o ANN leva horas para realizar o treinamento dos dados, já o SVM realiza em questão de segundos. O fuzzy, apesar de também diagnosticar todas as falhas, apresenta um delay de horas na detecção, tornando-se inviável neste estudo em questão.

Os métodos apresentam suas vantagens e desvantagens. Assim, na dúvida sobre qual método utilizar, deve-se antes saber em qual tipo de processo este está será aplicado, se há ou não a necessidade da detecção da falha de forma extremamente rápida e do investimento financeiro em relação aos equipamentos onde serão executados os métodos de detecção de falhas.

5. REFERÊNCIAS:

1. Abar, C. "O Conceito Fuzzy". Pontifícia Universidade Católica de São Paulo, 2004. Disponível em: <http://www.pucsp.br/~logica/Fuzzy.htm>.

- Acessado em 02/10/2018 às 3h.
2. Albertos, P. *Fuzzy Controllers - AI Techniques in Control*. Pergamon Press, 2002.
 3. Boswell, D. *Introduction to Support Vector Machines*. 2002. Disponível em: <http://dustwell.com/PastWork/IntroToSV M.pdf>>. Acessado em: 19 set. 2018.
 4. Bueno, E. I. *Utilização de Redes Neurais Artificiais na Monitoração e Detecção de Falhas em sensores do reator IEA-R1*. Diss. Universidade de São Paulo, 2006.
 5. Chen, J.; Patton, R. J. (1993). *Robust model-based fault diagnosis for dynamic systems*, Kluwer Academic Publishers, Boston, MA, USA.
 6. Dormehl, Luke. What is an artificial neural network?: Here's everything you need to know. 2019. Disponível em: <https://www.digitaltrends.com/cool-tech/what-is-an-artificial-neural-network/>>. Acesso em: 05 jan. 2019.
 7. Jakkula, V. Tutorial on Support Vector Machine (SVM). Washington State University, Pullman, 15 ago. 2018.
 8. Kabeya, K. (1998). *Structural health monitoring using multiple piezoelectric sensors and actuators*, Master thesis, Virginia Polytechnic Institute and State University – CIMSS, USA.
 9. Klatt, K. U., Engell, S. Gain-Scheduling trajectory control of a continuous stirred tank reactor. *Computers & Chemical Engineering*, Vol. 22, 491-502, 1998.
 10. Lorena, A. C., Carvaho, A. C. P. L. F. de. *Introdução as Máquinas de Vetores Suporte*. 192. ed. São Carlos: Universidade de São Paulo (usp), 2003. 66 p.
 11. Mathworks. What Is Fuzzy Logic?: Description of Fuzzy Logic. 2018. Disponível em: <https://www.mathworks.com/help/fuzzy/what-is-fuzzy-logic.html>>. Acesso em: 13 out. 2018.
 12. Milgram, J; Cheriet, M; Sabourin, R. "One Against One" or "One Against All": Which One is Better for Handwriting Recognition with SVMs?. *Hal: Inria, La Baule*, v. 1, n. 3, p.1-7, 05 out. 2006. Disponível em: <https://hal.inria.fr/inria-00103955>>. Acesso em: 01 jan. 2019.
 13. Ortega, N. R. S. *Aplicação da Teoria de Conjuntos Fuzzy a Problemas da Biomedicina*. 2001. 166 f. Tese de Doutorado em Ciências - Instituto de Física. Universidade São Paulo, São Paulo.
 14. Praxedes, M. A., Junior, A. T., Ravagnani, M. A. da S. S. Otimização de sistema de reatores utilizando programação matemática. *Revista Tecnológica*, v.17, n. 1, 19-28, 2008.
 15. Reis, L. L. G., Oliveira-Lopes, L. C. Controle PID Tolerante a Falhas de um CSTR. Em *V Seminário Nacional de Controle e Automação Industrial, Elétrica e de Telecomunicações*, Salvador. V SNCA, 1-7, 2007.
 16. Sanchez, E. C. M. Controle por aprendizado acelerado e neuro-fuzzy de sistemas servo-hidráulicos de alta frequência. Sistema Maxwell. https://www.maxwell.vrac.puc-rio.br/32823/32823_3.PDF Acessado em 02/10/2018 às 5h.
 17. Silva, R. A. C. *Inteligência artificial aplicada à ambientes de Engenharia de Software: Uma visão geral*. Universidade Federal de Viçosa, 2005.
 18. Sartori, Isabel, et al. "Detecção, Diagnóstico e Correção de Falhas: Uma Proposição Consistente de Definições e Terminologias." *Ciência & Engenharia* 21.2 (2012): 41-53.
 19. Simani, S., Fantuzzi, C. Fault diagnosis in power plant using neural networks. *Information Sciences*, v. 117, p. 125-136, 2000.
 20. Tubb, N. R. "A development path to success in neural computing." *Expert Systems Applications* 9.5 (1993): 5-9.