

DESENVOLVIMENTO DE UMA PLATAFORMA BASEADA EM PROTOCOLO OPC PARA O MONITORAMENTO DE FALHAS EM PROCESSOS INDUSTRIAIS

G. R. dos SANTOS¹ e T. V. da COSTA¹

¹ Universidade Federal de Itajubá, Instituto de Recursos Naturais, Engenharia Química
E-mail para contato: thiagocosta@unifei.edu.br

RESUMO – O desenvolvimento de métodos de detecção de falhas é uma área de pesquisa de grande interesse, uma vez que muitas falhas ocorrem em processos industriais e são mascaradas pela grande quantidade de dados gerados. Neste contexto, buscou-se com este trabalho desenvolver uma ferramenta para fins de detecção e análise de falhas em sistemas de interesse da engenharia química, considerando o uso do protocolo de comunicação OPC (*OLE for Process Control*) e a linguagem de programação Python. Os resultados obtidos demonstraram que a ferramenta é eficiente, permitindo observar graficamente a ocorrência de falhas a partir do processamento online do algoritmo de detecção no processo e poderá ser facilmente implementada em sistemas reais, uma vez que o padrão OPC é utilizado pela maioria das indústrias.

1. INTRODUÇÃO

Os sistemas industriais modernos, altamente instrumentados, proporcionam uma grande quantidade de informação disponível na forma de dados de processo (Bisarya e Puri, 2005). Deste modo, é possível, a partir de algoritmos de detecção de falhas, comparar a situação atual do processo com situações normais de operação verificando se um comportamento não esperado é proveniente de uma falha no processo.

O reconhecimento de situações de falhas em processos industriais, principalmente para pequenas falhas, é realizado por métodos de detecção e monitoramento de falhas. São ferramentas que auxiliam na análise e comparação do comportamento nominal da planta, permitindo que situações anormais de operação sejam detectadas e as providências para a contenção dos seus efeitos sejam tomadas com rapidez.

Portanto, grande foco tem sido dado ao desenvolvimento de ferramentas para detecção e correção de falhas, principalmente com a aplicação em processos da Engenharia Química. Pode-se observar no trabalho de Tulsian e Barton (2016) a aplicação da técnica denominada “método de detecção de falhas baseado no alcance”, através da proposta de uma transformação linear para mapear a dinâmica do sistema de um CSTR (*Continuous Stirred-Tank Reactor*, no inglês). Em Jiang *et al.* (2017) há a proposta de um novo algoritmo de detecção, derivado do método de análise de componentes principais (PCA, na sigla em inglês), denominado seleção de componente principal esparsa (SPCS, na abreviação em inglês). Através de um estudo de caso baseado no processo Tennessee Eastman, os autores analisam o desempenho do método SPCS. Os autores observaram que o método concentra a informação da falha em um subespaço, melhorando a identificação da falha.

Nesse contexto, buscou-se com este trabalho desenvolver um *software* capaz de adquirir dados *online* de um processo considerando o protocolo de comunicação industrial OPC, permitindo que a realização de cálculos computacionais complexos para a aplicação dos algoritmos de detecção e monitoramento de falhas fossem realizados. Para avaliação da plataforma de monitoramento, propôs-se um estudo de caso baseado em falhas simuladas em um reator CSTR a partir de um método quantitativo para detecção de falhas. O modelo interfaceado com um cliente OPC foi simulado em intervalos de tempo pré-determinados, possibilitando assim a consideração de um cenário próximo de uma situação real de monitoramento para a plataforma avaliada.

2. MATERIAIS E MÉTODOS

2.1. Análise de componentes principais

Neste trabalho, optou-se por utilizar a análise de componentes principais (PCA) como método de análise de falhas a ser implementado na plataforma de monitoramento proposta. Este método de detecção atua capturando a variância sob condições normais de operação. Segundo Braatz *et al* (2000), considerando-se uma matriz de dados $X \in \mathbb{R}^{m \times n}$, obtidos através de m medidas e n observações, pode-se definir a condição normal de operação de um sistema. Após os dados serem devidamente ordenados, o método PCA pode ser usado para dividir o espaço de medidas em dois subespaços: o componente principal e o subespaço residual, como mostra a equação (1):

$$X = TP^T + E = \sum_{i=1}^A t_i p_i^T + E \quad (1)$$

sendo X os dados sob condição normal, E a matriz de erro residual, P a matriz com os A (numero de componentes principais) maiores autovalores e T a matriz de componentes principais.

A variância de cada componente principal é obtida pelos autovalores associados a cada componente principal. O teste estatístico T^2 é definido baseado nos componentes principais, como é possível observar na equação (2):

$$T_i^2 = t_i \Lambda^{-1} t_i^T = x_i P \Lambda^{-1} P^T x_i^T \quad (2)$$

A matriz Λ é a matriz diagonal que contém os autovalores associados com os A componentes principais retidas no modelo. O limite de confiança para o teste estatístico pode ser calculado através da equação (3):

$$T_{A,n,a}^2 = \frac{(n-1)A}{n-A} F_a(A, n-A) \quad (3)$$

na qual $F_a(A, n-A)$ é o ponto crítico superior da distribuição F de Fischer com A e $n-A$ graus de liberdade.

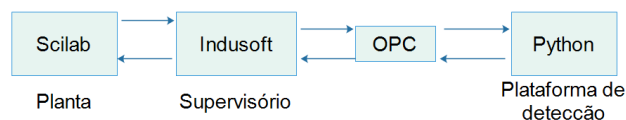
Outro teste estatístico, o teste SPE, baseado nos dados do subespaço residual, também é utilizado para análise dos dados. Segundo Mu e Venkatasubramanian (2003) o teste SPE assume que a variância é a mesma em todas as direções. Semelhantemente ao teste T^2 , o teste SPE também possui um limite de confiança.

Quando novos dados estão disponíveis, os métodos estatísticos podem ser calculados e comparados com os limites de confiança calculados anteriormente. Se os valores para os testes T^2 e SPE ultrapassarem os limites de confiança, a falha é detectada.

2.2. Plataforma de monitoramento de falhas

A estrutura proposta neste trabalho, representada na Figura 1, fundamentou-se em duas etapas. Primeiramente, implementou-se o *software* de monitoramento desenvolvido em *Python*, no qual seriam adicionados métodos de detecção de falha. Esses métodos, por sua vez, realizam cálculos avançados, como as rotinas estatísticas do método PCA. Em seguida, empregou-se um modelo para gerar dados em tempo real a partir do *software* Scilab para resolver e fornecer os dados do modelo para o sistema supervisor do processo. Para garantir o uso da ferramenta de detecção de falhas, optou-se por utilizar o sistema supervisor Indusoft, sistema SCADA (Supervisory Control and Data Acquisition, no inglês) comum em diversos sistemas de automação industrial.

Figura 1 – Esquema de aquisição de dados através de comunicação OPC.



Para realizar a comunicação entre a linguagem *Python* e o servidor OPC utilizou-se uma biblioteca denominada *OpenOPC* que disponibiliza os métodos e funções necessárias para conexão a um servidor OPC. Desta forma, foi possível conectar a plataforma de monitoramento de falhas com o servidor OPC disponível no sistema SCADA e realizar a leitura de dados do processo, permitindo o acompanhamento e análise de falhas *online*.

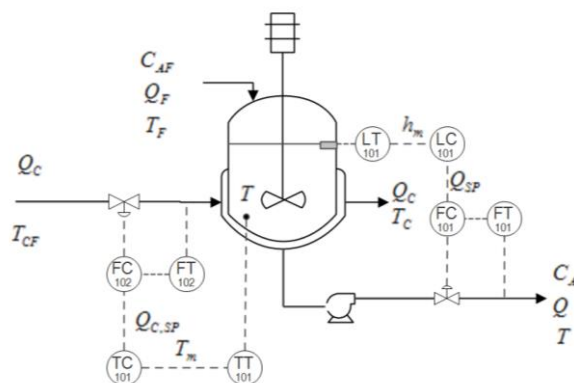
Salienta-se que a escolha do *software* Scilab para a simulação do processo foi realizada de forma a ressaltar as diferenças entre a plataforma de detecção implementada em *Python*, foco principal deste trabalho, e a geração de dados da planta por meio de simulação. Observa-se que se o modelo feito em Scilab for substituído por um sistema real, poderá se realizar a comunicação com o supervisor e com o algoritmo de detecção de falhas, através do padrão OPC. Deste modo, acredita-se que o projeto se torna muito prático, pois se tem a independência entre o monitoramento de processo e a plataforma de monitoramento de falhas.

2.3 Estudo de caso

Com o intuito de avaliar o PCA, método de detecção de falhas utilizado neste trabalho, foram simuladas quatro falhas no modelo em Scilab em um reator CSTR (*Continuous Stirred Tank Reactor*) não isotérmico, com reação de primeira ordem, representado na Figura 2. O

modelo consiste em um reator CSTR onde se processa uma reação do tipo $A \rightarrow B$, segundo uma cinética de reação de ordem 1.

Figura 2 – Esquema do reator CSTR (Conner e Seborg, 2005)



Legenda:

Q_c - Vazão na camisa (L/min); T_{CF} - Temperatura de entrada na camisa (K); Q_F - Vazão de entrada no reator (L/min); T_c - Temperatura de saída na camisa (K); T - Temperatura de saída do reator (K); Q - Vazão de saída do reator L/min); C_{AF} - Concentração de entrada no reator (mol/L); C_A - Concentração de saída do reator (mol/L).

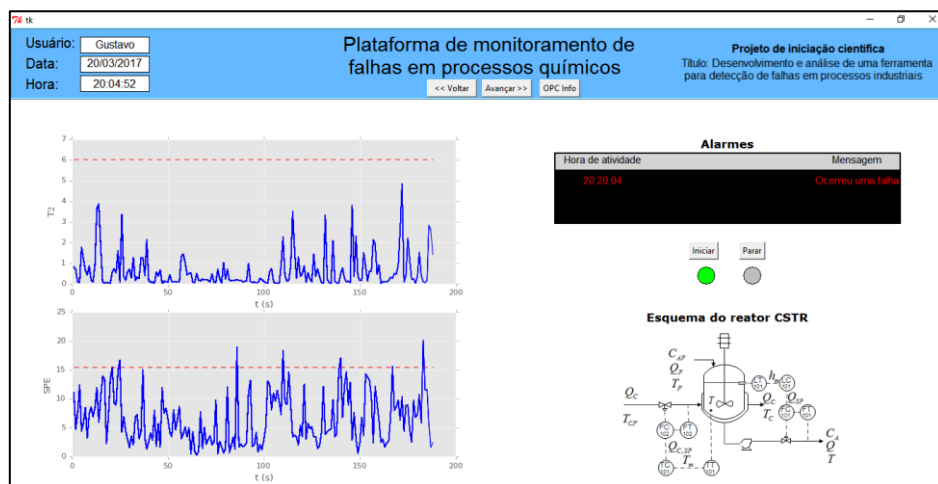
O sistema de controle é representado por uma estrutura de controladores em cascata. A malha de temperatura possui um controlador que atua a partir da informação do sensor de temperatura. Assim, conforme a temperatura aumente ou diminua, o controlador de vazão (FC-101) atua diminuindo ou aumentando a vazão da camisa. A malha de nível possui um sensor de nível que por sua vez envia o sinal da variável medida até o controlador de nível (LC-101), que atua aumentando ou diminuindo a vazão da corrente de entrada do reator.

3. RESULTADOS E DISCUSSÃO

No estudo de caso foram analisados quatro cenários de falhas, considerando-se os dados das variáveis de saída e das variáveis manipuladas (Q e Q_c) para aplicação do algoritmo de detecção. Para este trabalho optou-se por demonstrar a falha de incrustação no reator, pois é uma falha muito comum na indústria química. Desse modo, simulou-se a falha de incrustação no reator, caracterizada pela diminuição do parâmetro que engloba a área de troca térmica e o coeficiente global de transferência de calor no reator. Sendo assim, o parâmetro passa a diminuir em 125 J/min/K. Os dados do reator, fornecidos durante a simulação ao sistema supervisor, são compartilhados com a plataforma de monitoramento de falhas. Desta forma, as falhas que não aparecem no painel de alarme do supervisor, devido a sua pequena magnitude, aparecerão na tela de monitoramento de falhas, pois serão reconhecidas pelo algoritmo de detecção. Assim, a falha se tornará visível, sendo que a diferença entre os dados do processo em estado normal e anormal é evidenciada.

A seguir está representada a interface gráfica da plataforma de monitoramento de falhas (Figura 3) desenvolvida em *Python*. A plataforma é composta por uma área de visualização gráfica, onde é feito o monitoramento em tempo real da situação do processo. Além disso, possui um painel destinado a alertar o usuário sobre a ocorrência de falha.

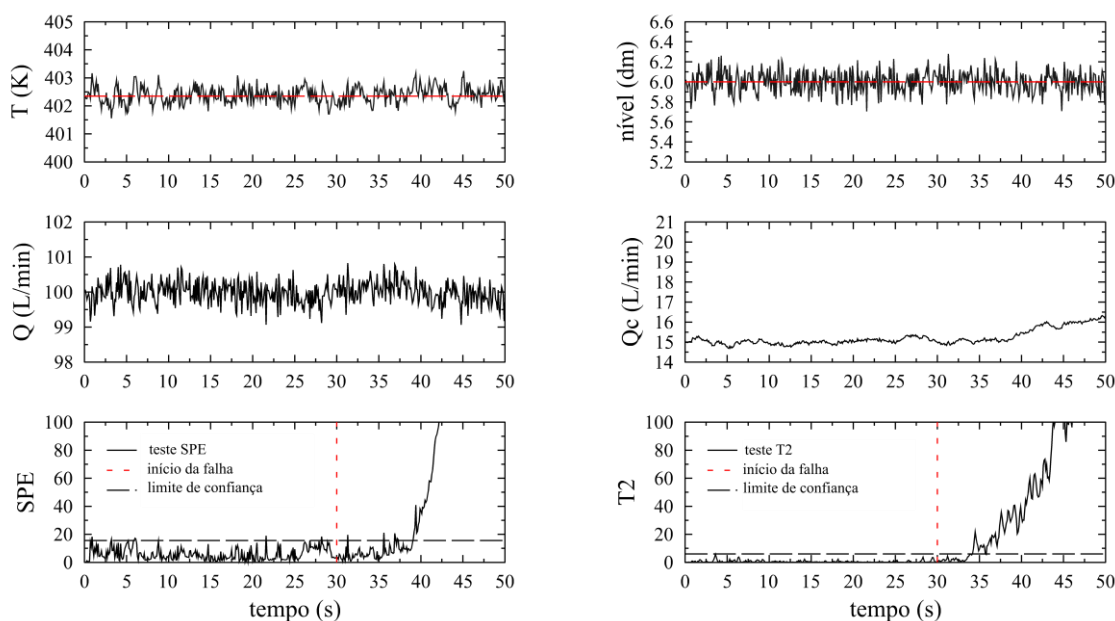
Figura 3 – Plataforma de monitoramento de falhas em tempo real implementada em *Python*.



A plataforma desenvolvida se comportou de forma eficiente, detectando a falha através do algoritmo utilizado. É importante ressaltar que a plataforma permite facilmente que outros algoritmos de detecção sejam utilizados. Dessa forma, futuramente, algoritmos mais complexos e eficientes poderão ser implementados para detecção de falhas. Além disso, destaca-se a utilização do padrão OPC, que possibilita a comunicação entre o software criado e sistemas industriais.

A seguir, na Figura 4, estão representados os dados da simulação da falha de incrustação no reator. Nessa falha é possível verificar a eficiência da plataforma:

Figura 4 – Detecção da falha de incrustação no reator.



Através da análise gráfica, percebe-se que houve atraso na detecção da falha. Esse fato acontece pois a falha é gradual e de ordem muito pequena. Desse modo, no início a variância nos dados provocada pela falha é tão pequena que o método PCA não consegue identificar a ocorrência de anomalias. Somente depois de dez segundos após a aplicação da falha é possível realizar a detecção.

4. CONCLUSÕES

A aplicação do método PCA para as falhas simuladas no reator CSTR propiciou a detecção da falha de forma eficiente, como é possível observar nos resultados apresentados nos testes T2 e SPE, em que os dados ultrapassam o limite quando a falha ocorre. Apesar disso, existiram alguns falsos alarmes, ou seja, dados que ultrapassaram o limite quando o reator operava em regime normal. No entanto, a porcentagem de falsos positivos foi de cerca de 3%, sendo uma quantia aceitável devido às características da aplicação do método que se baseia na obtenção da variância dos dados.

Os resultados obtidos com a plataforma mostram que ela funcionou corretamente, adquirindo dados por meio de conexão OPC e aplicando o método PCA em tempo real. Observa-se que futuramente a plataforma poderá ser implementada em um sistema industrial, uma vez que o padrão OPC é utilizado pelas estruturas de automação da maioria das indústrias. Deste modo, pretende-se utilizar a ferramenta desenvolvida em estudos futuros no desenvolvimento e na análise de novos algoritmos de detecção de falhas com aplicação em sistema reais.

5. REFERÊNCIAS

- BISARYA, R. K.; PURI, S. The Bhopal gas tragedy – A perspective. *Journal of Loss Prevention in the Process Industries*, v. 18, p. 209-212, 2005.
- BRAATZ, R. D.; CHIANG, L. H.; RUSSEL, E. L. *Fault detection in industrial processes using canonical variate analysis and dynamic principal component analysis*. Chemometrics and intelligent laboratory systems. Illinois. p. 81-93, 2000.
- JIANG, X.; ZHAO, H.; LEUNG, H. H. Fault Detection and Diagnosis in Chemical Processes Using Sparse Principal Component Selection. *Journal of Chemical Engineering of Japan*, vol. 50, p. 31- 44, 2017.
- TULSIAN, A.; BARTON, P. I. *Reachability-based fault detection method for uncertain chemical flow reactors*. IFAC (International Federation of Automatic Control). Cambridge, Massachusetts-USA, 2016.
- VENKATASUBRAMANIAN, V.; MU, F. *PCA with Efficient Statistical Testing Method for Process Monitoring*. Laboratory for Intelligent Process Systems, School of Chemical Engineering Purdue University. West Lafayette, 2003.