UMA ABORDAGEM EVOLUTIVA MULTIOBJETIVO BASEADA EM PONTO DE ATRAÇÃO PARA SELEÇÃO DE VARIÁVEIS EM PROBLEMAS DE CLASSIFICAÇÃO DE FALHAS

Fernando Marcos Souza Silva
Instituto Federal do Norte de Minas Gerais – IFNMG
Faz. do Meio Pé da Serra, S/N – Zona Rural, Araçuaí/MG, CEP: 39.600-000
Programa de Pós-Graduação em Modelagem Computacional e Sistemas – UNIMONTES
Av. Dr. Ruy Braga, S/N – Vila Mauriceia, Montes Claros/MG, CEP: 39.401-089
fernando.silva@ifnmg.edu.br

Jéssica Flaviane Ferreira
Bacharelado em Engenharia de Sistemas – UNIMONTES
Av. Dr. Ruy Braga, S/N – Vila Mauriceia, Montes Claros/MG, CEP: 39.401-089
jessicafferreira11@gmail.com

Reinaldo Mattinez Palhares
Departamento de Engenharia Eletrônica – UFMG
Av. Pres. Antônio Carlos, 6627 – Pampulha, Belo Horizonte/MG, CEP: 31.270-901
rpalhares@ufmg.br

Marcos Flávio Silveira Vasconcelos D'Angelo
Departamento de Ciência da Computação – UNIMONTES
Av. Dr. Ruy Braga, S/N – Vila Mauriceia, Montes Claros/MG, CEP: 39.401-089
marcos.dangelo@unimontes.br

RESUMO

Neste trabalho é proposto um método *wrapper* de seleção de variáveis denominado NSGA-II-GMM-AP que é baseado no algoritmo evolutivo NSGA-II e classificadores usando modelos de mistura gaussiana. Este algoritmo segue uma abordagem bi-objetivo e possui como principal característica o uso do conceito de ponto de atração, responsável pelo controle da complexidade dos indivíduos da população do NSGA-II durante o processo de otimização. Experimentos realizados em dados do processo petroquímico Tennessee Eastman para classificação de falhas mostraram que o NSGA-II-GMM-AP leva a soluções com menor erro de classificação que os demais métodos estudados, sendo uma abordagem promissora para o problema de seleção de variáveis.

PALAVRAS CHAVE. Seleção de Variáveis. NSGA-II. Classificação de Falhas. Ponto de Atração.

Tópico: OA – Outras aplicações em PO.

ABSTRACT

This paper presents a variable selection wrapper method called NSGA-II-GMM-AP that is based on the evolutionary algorithm NSGA-II and classifiers that uses Gaussian Mixture Models. This algorithm is a bi-objective approach that has as main characteristic the use of attraction point which is responsible for the complexity control of individuals in the NSGA-II population during the optimization process. Experiments carried out on Tennessee Eastman Petrochemical Process dataset for fault classification showed that NSGA-II-GMM-AP leads to solutions with lower classification error than the other methods applied, being a promising approach to the variable selection problem.

KEYWORDS. Variable Selection. NSGA-II. Fault Classification. Attraction Point.

Paper topic: OA – Other applications in OR.

Introdução

Nas últimas décadas, o uso de sistemas computacionais trouxe grandes melhorias para o monitoramento de eventos anormais em processos, embora ainda haja parte significativa destas tarefas feitas por operadores humanos. A complexidade dos processos e o grande número de dados que devem ser analisados para uma tomada de decisão adequada são características impeditivas para a atuação integral de humanos nestas tarefas, ao passo que tornam-se um convite para a automação das atividades desta área denominada Gerenciamento de Eventos Anormais (*Abnormal Event Management*, AEM) [Venkatasubramanian et al., 2003a]. Automatizar as tarefas de AEM buscando minimizar problemas causados por erros humanos ainda é um grande desafio para a indústria e comunidade acadêmica. Certas situações de tomada de decisão podem ser tão sensíveis de modo que uma escolha equivocada poderia levar a perdas de produtividade, materiais, equipamentos ou ainda levar à perda de vidas humanas.

Um dos componentes das atividades de AEM são os Sistemas de Detecção e Diagnóstico de Falhas (*Fault Detection and Diagnosis*, FDD), que a partir do monitoramento de um processo, apontam possíveis falhas e as diagnosticam, classificando-as conforme o seu tipo e apontando sua(s) causa(s) e origem(ns). Há uma vasta gama de propostas de sistemas de FDD na literatura utilizando métodos baseados em modelos quantitativos, modelos qualitativos e métodos baseados em dados históricos do processo [Venkatasubramanian et al., 2003a,b,c].

Os métodos baseados em dados históricos são especialmente interessantes por não necessitarem de nenhum conhecimento prévio, matemático ou físico, sobre o funcionamento do processo. Diferente dos métodos baseados em modelos, é necessário apenas haver a disponibilidade de dados passados capazes de descrever ao máximo os diversos estados do processo que deseja-se monitorar [Yin et al., 2014a]. O advento de algoritmos de aprendizagem de máquina e a melhoria contínua de tecnologias de persistência de dados também são fatores encorajadores para a aplicação de métodos baseados em dados históricos em FDD [Jämsä-Jounela, 2007]. Métodos baseados em dados históricos, como Análise de Componentes Principais (*Principal Component Analisys*, PCA) [Benaicha et al., 2010; Jing e Hou, 2015], Mínimos Quadrados Parciais (*Partial Least Squares*, PLS) [Harrou et al., 2015], Máquinas de Vetores de Suporte (*Suport Vector Machines*, SVM) [Jing e Hou, 2015] e Modelos de Mistura Gaussiana (*Gaussian Mixture Models*, GMM) [Tong et al., 2014] tem sido frequentemente aplicados com sucesso nas tarefas de sistemas de FDD. Um estudo comparativo incluindo alguns destes e outros métodos baseados em dados históricos foi feito por Yin et al. [2012].

Classificação de falhas é uma etapa relacionada ao diagnóstico em sistemas de FDD que consiste em discriminar os tipos das falhas detectadas pelo sistema, i.e., distinguir as causas de uma condição anormal. Esta é uma tarefa importante, pois a partir da correta identificação de uma falha em tempo hábil é possível dar início a atividades corretivas evitando o comprometimento da produtividade e segurança de um processo. Uma falha não detectada e/ou não diagnosticada (classificada) em tempo hábil pode ter efeitos catastróficos. Não são raros os acidentes em indústrias envolvendo vidas humanas. Um caso recente ocorrido em fevereiro de 2015 no estado do Espírito Santo no Brasil, em um navio-plataforma da Petrobrás[1] deixou 9 mortos e 26 feridos após uma explosão ocasionada por vazamento de gás e outros erros humanos. A mesma empresa, antes disso, havia experimentado um acidente ainda maior em março de 2001, quando três explosões em tanques de óleo em uma platarforma semi-submersa mataram 11 pessoas e levaram a estrutura ao fundo do mar.

Os processos petroquímicos, bem como outros processos industriais modernos estão se tornando cada vez mais complexos, tanto em níveis estruturais quanto de automação, para aplicações de FDD, podendo ter muitas variáveis redundantes e/ou irrelevantes lidas diretamente de sensores dispersos pelas plantas. A eliminação destas variáveis do conjunto de variáveis monitoradas pode

[1]Empresa estatal brasileira que atua no segmento de energia, prioritariamente nas áreas de exploração, produção, refino, comercialização e transporte de petróleo, gás natural e seus derivados.

levar a uma redução de custos de monitoramento e também a sistemas de FDD mais eficientes e robustos. Entretanto, a escolha do conjunto de variáveis mais adequado para o monitoramento não é uma tarefa fácil [Foster et al., 2014]. O problema aqui apresentado, abordado na literatura na área de mineração de dados como seleção de variáveis, por ser *NP-hard* de complexidade $O(2^n)$, tem sido frequentemente tratado por meio de métodos heurísticos ou metaheurísticos, que são largamente classificados como *filter* e *wrapper* [Chandrashekar e Sahin, 2014].

Métodos de *filter* são independentes de algoritmos de aprendizagem e fazem a seleção de variáveis usando características intrínsecas dos dados como uma etapa de preprocessamento, calculando índices baseados em critérios de relevância e/ou similaridade entre variáveis, como o coeficiente de correlação de Pearson ou Informação Mútua [Yu e Liu, 2004]. Métodos de *wrapper* utilizam o algoritmo de aprendizagem como uma função de "caixa-preta"embutida em um algoritmo de otimização, com a finalidade de extrair métricas de performance guiando o processo de seleção de variáveis [Chandrashekar e Sahin, 2014; Guyon e Elisseeff, 2003; Kohavi e John, 1997]. Há ainda métodos híbridos que combinam características destes dois métodos, como em Li et al. [2011], onde é utilizado um método de *filter* para gerar um conjunto de soluções e em seguida um método *wrapper* baseado em algoritmos genéticos para refiná-las. Já em Zhu et al. [2007] um método de *filter* é utilizado como mecanismo de busca local de um *wrapper* também baseado em algoritmos genéticos.

As estratégias adotadas por métodos *wrapper* podem ser determinísticas usando algoritmos de seleção sequencial, como o *Sequential Forward Selection* (SFS) [Whitney, 1971] ou *Sequential Backward Selection* (SBS) [Marill e Green, 1963]. Também podem ser utilizadas heurísticas ou metaheurísticas de busca global capazes de explorar o espaço de buscas de forma mais abrangente, como algoritmos evolutivos, *Particle Swarm Optimization* (PSO) [Ahmad, 2015; Xue et al., 2012], *Ant Colony Optimization* (ACO) [Allegrini e Olivieri, 2011], *Differential Evolution* [Al-Ani et al., 2013] e *Simulated Annealing* [Brusco, 2014], dentre outros métodos. As estratégias mais utilizadas podem ser mono-objetivo (quando busca-se minimizar o erro de classificação) ou multiobjetivo, que normalmente buscam minimizar o erro de classificação ao mesmo tempo que minimizam a complexidade das soluções encontradas, i.e., o número de variáveis selecionadas.

Métodos multiobjetivo têm vantagem sobre métodos mono-objetivo por conseguirem eliminar melhor as variáveis pouco informativas, pois reduzir a complexidade das soluções é um objetivo explícito. Estes métodos retornam um conjunto de soluções não dominadas, das quais pode-se escolher uma única solução para aplicação levando em consideração o custo-benefício de cada uma das soluções obtidas [Mukhopadhyay et al., 2014a]. Entretanto, em sistemas de FDD críticos, onde uma falha não corretamente tratada pode comprometer a segurança do processo e de seres humanos envolvidos, é natural pensar que a escolha da solução a ser utilizada deve se balizar na qualidade do sistema de FDD (ou seja, em sua acurácia).

Neste trabalho é feita uma proposta de um método *wrapper* multiobjetivo de seleção de variáveis para aplicação em classificação de falhas baseada em dados históricos de um processo petroquímico usando um algoritmo genético multiobjetivo e classificadores baseados em GMM. No método proposto, denominado NSGA-II-GMM-AP, é utilizada uma variação do *Non-domineted Sorting Gennetic Algorithm II* (NSGA-II) [Deb et al., 2002] como mecanismo de busca devido ao fato de este já ter sido aplicado com sucesso em problemas de seleção de variáveis [de Lucena et al., 2013a; Li et al., 2016; de Almeida Ribeiro et al., 2015].

A principal característica do método proposto é que ele utiliza o conceito de Ponto de Atração visando controlar a complexidade das soluções da população mantida pelo NSGA-II durante o processo de otimização, mantendo-a próxima à da melhor solução obtida em cada geração. Isto é feito buscando tratar o problema de que os operadores de cruzamento tradicionais em algoritmos evolutivos tendem a explorar soluções de complexidade média em problemas de seleção de variáveis [Emmanouilidis et al., 2000]. Outra característica do método é que apesar de seguir uma estratégia multiobjetivo de busca por ser baseado no NSGA-II, o NSGA-II-GMM-AP retorna ape-

nas uma única solução de melhor acurácia, por que nós argumentamos que este é o objetivo mais importante a ser otimizado em um sistema crítico de FDD.

Na próxima seção (Seção 1) será descrita a metodologia do trabalho contendo os detalhes sobre o funcionamento do algoritmo proposto e cada um dos algoritmos utilizados para comparação: o NSGA-II sem modificações, um algoritmo genético mono-objetivo, o SFS e o SBS. Em seguida, na Seção 2 são apresentados os resultados dos experimentos realizados utilizando os algoritmos desenvolvidos no conjunto de dados de uma planta petroquímica: o *Tennessee Eastman Process* (TEP). Finalmente, na Seção 2, são apresentadas as conclusões com indicações de possíveis trabalhos futuros.

1. Metodologia

Foram implementados 5 métodos *wrapper* de seleção de variáveis (Tabela 1). Todos os métodos medem a performance de classificação de falhas usando GMM e foram aplicados ao conjunto de dados do TEP. Portanto, as próximas duas subseções serão dedicadas a estes dois tópicos e, na sequência, serão tratadas as características de cada um dos algoritmos desenvolvidos.

Tabela 1: Algoritmos desenvolvidos.

Algoritmo	**Descrição**
NSGA-II-GMM	NSGA-II com GMM
NSGA-II-GMM-AP	NSGA-II com GMM e Ponto de Atração
MONO-GA-GMM	Algoritmo genético mono-objetivo com GMM
SFS-GMM	SFS com GMM
SBS-GMM	SBS com GMM

1.1. Classificadores baseados em Modelos de Mistura Gaussiana

O método de classificação supervisionada de falhas baseado em Modelos de Mistura Gaussiana (*Gaussian Mixture Models*, GMM) é um processo relativamente simples. Ele é composto por uma etapa de treinamento (*off-line*), utilizando dados históricos e uma etapa de predição para aplicação (*on-line*) na classificação das falhas.

O processo de treinamento consiste na otimização dos parâmetros dos GMM para cada tipo de falha do processo. A construção dos GMM para cada tipo de falha é feita utilizando o algoritmo iterativo de Estimação de Máxima verossimilhança (*Expectation Maximization*, EM) [Dempster et al., 1977], que faz os ajustes dos parâmetros dos GMM até a convergência, i.e., até que o erro não seja reduzido após um número determinado de iterações.

Uma vez feito o treinamento dos GMM e apresentada uma nova instância de dados para classificação, a predição da classe de falha é feita calculando-se o valor da função de densidade de probabilidade (*Probability Density Function*, PDF) para cada um dos GMM construídos anteriormente, ou seja, calcula-se a probabilidade de um novo dado pertencer a cada tipo de falha. A escolha da classe de falha é feita baseada no maior valor da PDF.

1.2. Conjunto de dados do Processo *Tennessee Eastman*

O TEP é um processo industrial petroquímico real muito utilizado para avaliar métodos de controle e monitoramento de processos. Ele é inteiramente descrito no trabalho de Downs e Vogel [1993] e revisado por Bathelt et al. [2015], onde é apresentado um simulador de código aberto. Este processo tem sido utilizado sistematicamente para aplicação e comparação de métodos de FDD, como em Yin et al. [2012], Yin et al. [2014a] e Jing e Hou [2015].

Para este trabalho foram utilizados dados do TEP disponíveis no site[2] do *Massachusetts Institute of Technology* (MIT). Neste conjunto de dados existem 52 variáveis e 21 tipos de falhas. Estão disponíveis dados para treinamento e testes das condições de operação normal (*Normal Operation Conditions*, NOC) e de todos os tipos de Falha do processo. Dos dados disponíveis, foram

[2]Disponível em http://web.mit.edu/braatzgroup/links.html, acessado em 15/07/2016.

utilizadas 480 instâncias de para treinamento e 800 para testes de cada tipo de falha. Todos os dados de NOC foram descartados, pois neste trabalho será tratado o problema de classificação de falhas usando GMM, partindo do pressuposto que as falhas já foram detectadas pelo sistema de FDD e devem ser classificadas. Uma explicação detalhada sobre este conjunto de dados pode ser obtida em Yin et al. [2014b].

1.3. NSGA-II-GMM

O NSGA-II proposto por Deb et al. [2002] foi adaptado para atuar como método *wrapper* de seleção de variáveis usando classificadores baseados em GMM, denominado NSGA-II-GMM (Algoritmo 1). Neste algoritmo (i) os indivíduos tiveram representação binária, sendo cada um composto por um vetor de tamanho 52 (quantidade de variáveis do TEP), onde cada gene de índice i possuía a informação sobre a seleção (valor 1) ou não (valor 0) da i-ézima variável; (ii) a população de 400 indivíduos foi inicializada de forma aleatória, seguindo uma distribuição uniforme da quantidade de variáveis que compunham os indivíduos, buscando partir de uma população com uma quantidade aproximadamente igual de amostras de soluções de todas as complexidades (garantido explicitamente pelo algoritmo de inicialização da população); (iii) foi utilizado o operador de cruzamento de 2 pontos, com probabilidade de 90%, gerando duas soluções filhas; (iv) foi utilizada a mutação um bit, com probabilidade de 10%; (v) o método de seleção utilizado foi o de torneio binário; e (vi) foram utilizadas 50 gerações como condição de parada.

A escolha dos parâmetros deste algoritmo se deu após a realização de testes verificando populações de 100, 200, 300, 400 e 500 indivíduos; probabilidades de cruzamento de 70%, 80%, 90% e 100%; probabilidades de mutação de 5%, 10%, 15%, 20%; e quantidades de gerações de 30, 50, 100 e 200. Os operadores genéticos utilizados foram os clássicos para representação binária e o mecanismo de seleção foi o padrão utilizado por Deb et al. [2002].

Algoritmo 1: NSGA-II-GMM

Entrada: N: tamanho da população; Gen_{max}: máximo de gerações; $Cross_p$: probabilidade de cruzamento; Mut_p: probabilidade de mutação
Saída: Conjunto de soluções não-dominadas

1 **início**
2 Inicializa a população *pop*;
3 Avalia os objetivos $f1$ (eq. 1) e $f2$ (eq. 2) de *pop*;
4 $gen \leftarrow 1$
5 **repita**
6 Seleciona os indivíduos por torneio binário;
7 Cria uma população descendente (*offspring*) aplicando-se os operadores genéticos conforme as probabilidades $Cross_p$ e Mut_p;
8 Avalia os objetivos $f1$ (eq. 1) e $f2$ (eq. 2) de *offspring*;
9 $intermediate \leftarrow pop \cup offspring$;
10 Ordena a população *intermediate* conforme não-dominância de Pareto e *crowding distance*;
11 Substitui *pop* por *intermediate* até o limite de N indivíduos;
12 $gen \leftarrow gen + 1$;
13 **até** $gen > Gen_{max}$;
14 **retorna** Soluções não-dominadas de *pop*;
15 **fim**

Este algoritmo busca minimizar dois objetivos $f1$ (eq. 1) e $f2$ (eq. 2). O objetivo $f1$ se refere à quantidade de variáveis de uma solução. O segundo objetivo refere-se à taxa de erro de classificação dos dados utilizando os classificadores baseados em GMM, onde FP são os Falsos

Positivos, FN são os Falsos Negativos, TP são os Verdadeiros Positivos e TN são os Verdadeiros Negativos.

$$f1 : \sum_{i=1}^{n} I_i \tag{1}$$

$$f2 : \frac{FP + FN}{TP + FP + TN + FN} \tag{2}$$

1.4. NSGA-II-GMM-AP

O NSGA-II-GMM citado anteriormente é uma aplicação direta do NSGA-II tradicional e classificadores GMM ao problema de seleção de variáveis. Emmanouilidis et al. [2000] relatam um problema que ocorre em algoritmos evolutivos quando se utiliza os operadores tradicionais de cruzamento (como o de 2 pontos utilizado), que levam à exploração de soluções de complexidade média. Ou seja, o cruzamento de uma solução de n variáveis com uma solução de m variáveis tende a criar uma solução de $(n + m)/2$ variáveis. Os autores tratam este problema propondo um novo operador de cruzamento. O NSGA-II-GMM-AP busca contornar este problema por meio do controle das complexidades das soluções da população, mantendo-as próximas à complexidade da solução de melhor performance, utilizando a ideia de Ponto de Atração (AP).

O processo de substituição da população do NSGA-II utiliza o procedimento de ordenação por não-dominância em fronteiras de Pareto e o cálculo da *Crowding Distance*, que tendem a eliminar as soluções mais distantes das não-dominadas ao mesmo tempo que preserva sua diversidade em relação aos dois objetivos do problema de seleção de variáveis. Este mecanismo é interessante quando busca-se construir uma fronteira de Pareto diversificada em relação aos dois objetivos considerados, porém, tende a acentuar o problema relatado dos operadores tradicionais de cruzamento. Por exemplo, ao se observar uma população durante a 8ª geração de uma execução do NSGA-II-GMM (Figura 1), é possível perceber que, como as soluções encontradas possuem entre 0 e 35 variáveis, o operador de cruzamento utilizado tende a explorar soluções de complexidade inferior a 35, pois há várias soluções com poucas variáveis presentes na população, deixando outras áreas do espaço de busca sub-exploradas. Além disso há uma tendência das soluções mais distantes das não-dominadas serem eliminadas durante as próximas gerações (como no caso das soluções com 34 e 35 variáveis).

Objetivando sanar estes problemas observados, foi feita a proposta do NSGA-II-GMM-AP, descrito no Algoritmo 2. Este algoritmo foi desenvolvido com as mesmas características do NSGA-II-GMM, mas com algumas modifiçãoes. A mais evidente é em relação ao objetivo de minimização $f1$, que utiliza o elemento AP (eq. 3). Este objetivo representa a distância entre a complexidade de um indivíduo I que possui n genes e a complexidade do indivíduo com menor erro de classificação presente na população, denotada por AP. No exemplo da Figura 2, que representa o retrato de uma população do NSGA-II-GMM-AP ao término de sua 8ª geração, como $AP = 37$, estão presentes soluções de 31 a 43 variáveis ($AP \pm 6$). Nas próximas gerações, caso seja encontrada alguma solução com menor erro de classificação, o valor de AP será atualizado e o mecanismo de substituição da população do NSGA-II naturalmente se encarregará de excluir as soluções mais distantes de AP.

$$f1 : \left| \sum_{i=1}^{n} I_i - AP \right| \tag{3}$$

A escolha do AP neste algoritmo se dá de forma gulosa após a geração da população inicial e ao término de cada geração. Para minimizar a possibilidade desta escolha levar o algoritmo convergir para ótimos locais, foram incluídas ainda duas características no NSGA-II-GMM-AP visando preservar a diversidade da população. Uma pode ser observada nas linhas 10 e 11 do

Algoritmo 2: NSGA-II-GMM-AP

Entrada: N: tamanho da população; Gen_{max}: máximo de gerações; $Cross_p$: probabilidade de cruzamento; Mut_p: probabilidade de mutação
Saída: Solução de melhor performance

1 **início**
2 Inicializa a população *pop*;
3 Avalia o objetivo $f2$ (eq. 2) de *pop*;
4 $AP \leftarrow$ quantidade de variáveis da solução com menor valor de $f2$ em *pop*;
5 Avalia o objetivo $f1$ (eq. 3) de *pop*;
6 $gen \leftarrow 1$
7 **repita**
8 Seleciona os indivíduos por torneio binário;
9 Cria uma população descendente (*offspring*) aplicando-se os operadores genéticos conforme as probabilidades $Cross_p$ e Mut_p;
10 **se** *offspring possui alguma solução que já foi explorada anteriormente* **então**
11 Force mutações nestas soluções já exploradas de *offspring*;
12 **fim**
13 Avalia os objetivos $f1$ (eq. 3) e $f2$ (eq. 2) de *offspring*;
14 $intermediate \leftarrow pop \cup offspring$;
15 Remove as soluções repetidas de *intermediate*, se possível;
16 Ordena a população *intermediate* conforme não-dominância de Pareto e *crowding distance*;
17 Substitui *pop* por *intermediate* até o limite de N indivíduos;
18 $AP \leftarrow$ quantidade de variáveis da solução com menor valor de $f2$ em *pop*;
19 Avalia o objetivo $f1$ (eq. 3) de *pop*;
20 $gen \leftarrow gen + 1$;
21 **até** $gen > Gen_{max}$;
22 **retorna** Solução de melhor performance em *pop* (única não-dominada);
23 **fim**

Algoritmo 2, onde são forçadas mutações nas soluções que por ventura já tenham sido exploradas em gerações passadas. Uma estrutura de dados de busca eficiente (tabela *hash*) foi utilizada para fazer este controle. A outra característica, da linha 15, faz a remoção de soluções repetidas da população intermediária antes de iniciar o processo de substituição. Estas duas modificações vão no mesmo sentido das propostas por Yuen e Chow [2009] e Vachhani et al. [2016].

1.5. MONO-GA-GMM

O NSGA-II-GMM-AP, como visto, apesar de seguir uma estratégia de otimização bi-objetivo, é um método que retorna apenas uma única solução: a de menor erro de classificação. Então foi desenvolvido um método *wrapper* de seleção de variáveis baseado em algoritmo evolutivo mono-objetivo e classificadores usando GMM (MONO-GA-GMM) para fins de comparação. Buscou-se manter ao máximo as características dos algoritmos anteriores, i.e., representação dos indivíduos, operadores genéticos, probabilidades de cruzamento e mutação, tamanho da população e número de gerações. A função objetivo utilizada foi o erro de classificação (eq. 2), além disso utilizou-se uma taxa de elitismo de 5% do tamanho da população.

1.6. SFS-GMM

O SFS-GMM utiliza o processo de busca baseado no SFS, que é um algoritmo guloso de seleção de variáveis sequencial. Este algoritmo parte de uma solução vazia (sem nenhuma variável)

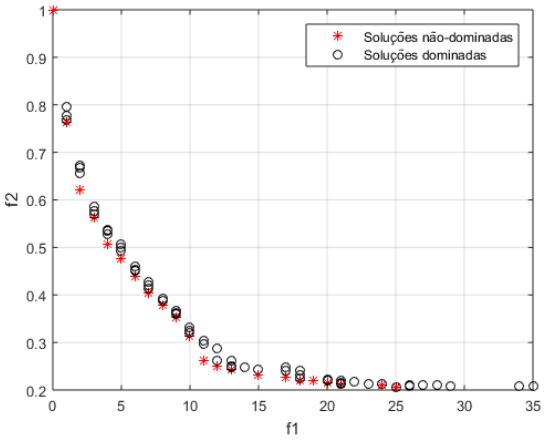

Figura 1: Uma população do NSGA-II-GMM

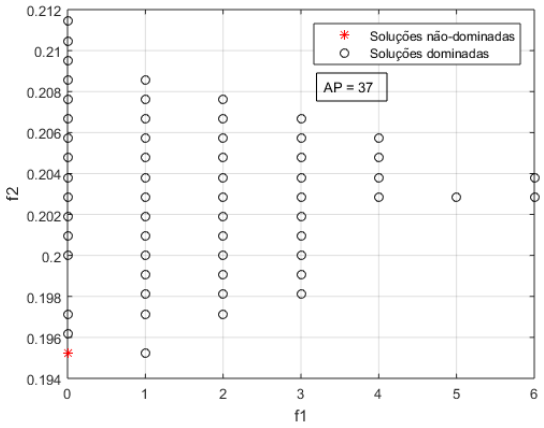

Figura 2: Uma população do NSGA-II-GMM-AP

e em um processo iterativo vai incluindo, uma a uma, as variáveis mais relevantes para classificação usando GMM. O processo é feito até que não seja possível adicionar alguma variável sem piorar a performance de classificação.

1.7. SBS-GMM

O SBS-GMM também é um algoritmo guloso que faz a seleção de variáveis de forma sequencial, baseado no SBS. A estratégia de busca deste algoritmo é exatamente contrária ao SFS, partindo de um conjunto cheio do qual são removidas, uma a uma, as variáveis que menos contribuem para a performance preditiva do classificador baseado em GMM. O processo é interrompido quando não é possível remover alguma variável do conjunto sem piorar a performance de classificação.

2. Resultados Experimentais e Discussões

Métodos *wrapper* demandam um grande custo computacional, que na maioria das vezes está associado à avaliação de uma solução para extração de métricas de performance. Em todos os algoritmos desenvolvidos a performance dos classificadores GMM é medida pelo erro de classificação no conjunto de dados de testes do TEP. Utilizar mecanismos de validação cruzada do tipo *k-fold* ou todos os dados disponíveis do TEP para treinamento e testes mostrou-se uma estratégia inviável devido ao grande tempo demandado. Logo, reduzimos o conjunto de dados de testes do TEP por meio de uma amostragem aleatória de tamanho 1.050 (50 dados para cada um dos 21 tipos de falha). Para treinamento foram utilizados todos os 10.080 dados disponíveis (480 para cada tipo de falha).

Os critérios utilizados para comparação entre os métodos de seleção de variáveis foram o erro de classificação e a quantidade de variáveis das soluções. Nenhuma comparação específica para métodos de otimização multiobjetivo foi feita entre os métodos NSGA-II-GMM e o NSGA-II-GMM-AP, pelo fato de que o método proposto além de retornar apenas uma única solução não dominada, ainda possui objetivos de minimização diferentes do NSGA-II-GMM.

Todos os algoritmos foram executados 30 vezes, portanto, foram feitas 30 amostragens diferentes dos dados de testes, conforme explicado anteriormente. Os métodos de seleção de variáveis foram aplicados utilizando os dados de treinamento e cada subconjunto de dados amostrais de testes. Ao término de cada execução de cada algoritmo, foi selecionada a solução com o menor erro de classificação, por considerarmos que este seja o principal critério para seleção de variáveis em sistemas críticos de FDD. Destas soluções analisou-se o erro de classificação e a quantidade de variáveis selecionadas. A Tabela 2 trás informações sobre as médias e os erros padrões (EP) do número de

variáveis e do erro de classificação dos algoritmos, cuja diferença entre as menores médias (destacadas em negrito) e as demais mostrou-se relevante conforme o teste *t* de *Student* realizado ao nível de confiança de 95% ($\alpha = 5\%$). As Figuras 3 e 4 ilustram a distribuição das soluções.

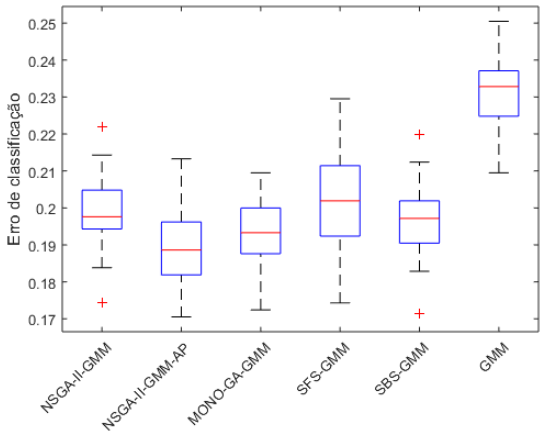

Figura 3: Comparação entre os erros de classificação das soluções obtidas pelos algoritmos.

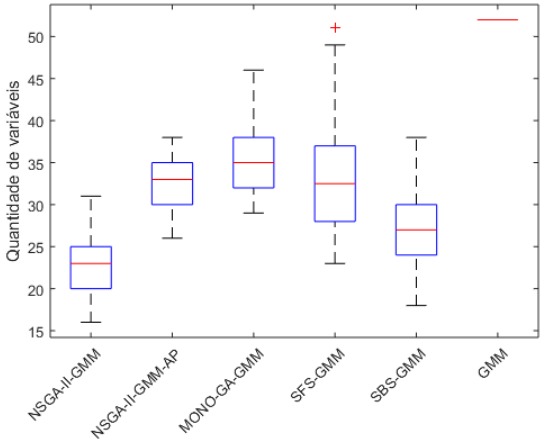

Figura 4: Comparação entre a quantidade de variáveis das soluções obtidas pelos algoritmos.

Tabela 2: Estatísticas das 30 execuções dos algoritmos.

Algoritmo	**Média de Variáveis ± EP(%)**	**Média do Erro ± EP(%)**
NSGA-II-GMM	**23,17 ± 0,70**	19,89 ± 0,17
NSGA-II-GMM-AP	32,47 ± 0,55	**18,83 ± 0,18**
MONO-GA-GMM	35,17 ± 0,70	19,27 ± 0,17
SFS-GMM	33,30 ± 1,32	20,27 ± 0,26
SBS-GMM	26,97 ± 0,89	19,67 ± 0,18
GMM	52,00 ± 0,00	23,17 ± 0,19

O NSGA-II-GMM, que utiliza uma estratégia bi-objetivo tradicional, levou a soluções de tamanho médio em torno de 23,17 – a menor quantidade de variáveis entre os métodos analisados – pois, minimizar o número de variáveis é um objetivo explícito deste algoritmo, cujo processo de substituição da população aliado ao operador de cruzamento utilizado pelo NSGA-II tendem a sub-explorar áreas do espaço de busca com mais variáveis. Por outro lado, é possível perceber que o NSGA-II-GMM-AP foi o algoritmo que levou a soluções com menor erro de classificação dentre os métodos considerados na comparação. O controle da complexidade das soluções feito pelo NSGA-II-GMM-AP levou a uma exploração de soluções com aproximadamente 32,47 variáveis e erro de classificação médio de 18,83%. Estes resultados também são melhores que o método mono-objetivo (MONO-GA-GMM), que busca minimizar o erro de classificação sem se preocupar com o controle da complexidade das soluções, o que reforça a importância deste controle quando se utiliza algoritmos genéticos com operadores de cruzamento tradicionais de n pontos.

Os métodos de seleção sequencial (SFS-GMM e SBS-GMM), apesar de levarem a um maior erro de classificação que o NSGA-II-GMM-AP, obtiveram bons resultados, se levarmos em consideração o menor esforço computacional feito por estes métodos, que avaliaram apenas 6,75% da quantidade de soluções avaliadas pelos algoritmos genéticos nos experimentos realizados.

Ao se comparar a performance dos métodos de seleção de variáveis com os classificadores GMM quando não se faz nenhuma seleção de variáveis (indicados na Tabela 2 e Figuras 3 e 4 como GMM) é possível perceber o benefício que seleção de variáveis pode trazer para sistemas de FDD, pois é significante a diminuição do erro de classificação e do número de variáveis selecionadas pelos

algoritmos.

Conclusões

Este trabalho teve como objetivo a proposta e estudo de um método *wrapper* evolutivo multiobjetivo de seleção de variáveis baseado no algoritmo NSGA-II, classificadores usando GMM e o conceito de ponto de atração, denominado NSGA-II-GMM-AP, aplicado ao problema de classificação de falhas no TEP. O método proposto tem a característica de controlar as complexidades das soluções da população, mantendo-as próximas à complexidade da solução de melhor performance preditiva (menor erro de classificação), além de outras características para evitar ótimos locais por meio da manutenção da diversidade da população.

O NSGA-II-GMM-AP foi comparado a outros métodos de seleção de variáveis: um baseado no NSGA-II tradicional, outro em um algoritmo evolutivo mono-objetivo e outros algoritmos de seleção sequencial baseados no SFS e SBS. Os resultados obtidos mostraram que o NSGA-II-GMM-AP leva a soluções com menores erros de classificação que os algoritmos usados para comparação no conjunto de dados analisado.

Trabalhos futuros podem explorar a aplicação do NSGA-II-GMM-AP a problemas de classificação em outros conjuntos de dados com características distintas de sensibilidade à seleção de variáveis. Percebemos ainda que o conceito de ponto de atração utilizado neste trabalho adiciona uma informação global que guia a busca feita pelo NSGA-II, assim como o elemento *gbest* do PSO. Por isso acreditamos que uma comparação de nossa abordagem com um método de seleção de variáveis baseado no PSO mereça alguma atenção.

Referências

Ahmad, I. (2015). Feature selection using particle swarm optimization in intrusion detection. *International Journal of Distributed Sensor Networks*, 2015:8. URL http://dx.doi.org/10.1155/2015/806954.

Al-Ani, A., Alsukker, A., e Khushaba, R. N. (2013). Feature subset selection using differential evolution and a wheel based search strategy. *Swarm and Evolutionary Computation*, 9:15–26.

Allegrini, F. e Olivieri, A. C. (2011). A new and efficient variable selection algorithm based on ant colony optimization. applications to near infrared spectroscopy/partial least-squares analysis. *Analytica Chimica Acta*, 699(1):18 – 25. ISSN 0003-2670. URL http://www.sciencedirect.com/science/article/pii/S0003267011006209.

Bathelt, A., Ricker, N. L., e Jelali, M. (2015). Revision of the tennessee eastman process model. *IFAC-PapersOnLine*, 48(8):309–314.

Benaicha, A., Guerfel, M., Bouguila, N., e Benothman, K. (2010). New pca-based methodology for sensor fault detection and localization. In *International Conference of Modeling Simulation, MOSIM'10*. Hammamel Tunisia.

Brusco, M. J. (2014). A comparison of simulated annealing algorithms for variable selection in principal component analysis and discriminant analysis. *Computational Statistics & Data Analysis*, 77:38 – 53. ISSN 0167-9473. URL http://www.sciencedirect.com/science/article/pii/S0167947314000668.

Chandrashekar, G. e Sahin, F. (2014). A survey on feature selection methods. *Computers and Electrical Engineering*, 40(1):16–28. ISSN 0045-7906. URL http://dx.doi.org/10.1016/j.compeleceng.2013.11.024.

de Almeida Ribeiro, L., da Silva Soares, A., de Lima, T. W., Jorge, C. A. C., da Costa, R. M., Salvini, R. L., Coelho, C. J., Federson, F. M., e Gabriel, P. H. R. (2015). Multi-objective genetic algorithm for variable selection in multivariate classification problems: A case study in verification of biodiesel adulteration. *Procedia Computer Science*, 51:346–355.

de Lucena, D. V., de Lima, T. W., da Silva Soares, A., Delbem, A. C., Galvao Filho, A. R., Coelho, C. J., e Laureano, G. T. (2013a). Multi-objective evolutionary algorithm for variable selection in calibration problems: A case study for protein concentration prediction. In *2013 IEEE Congress on Evolutionary Computation*, p. 1053–1059. IEEE.

Deb, K., Pratap, A., Agarwal, S., e Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197.

Dempster, A. P., Laird, N. M., e Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, p. 1–38.

Downs, J. J. e Vogel, E. F. (1993). A plant-wide industrial process control problem. *Computers & chemical engineering*, 17(3):245–255.

Emmanouilidis, C., Hunter, A., e MacIntyre, J. (2000). A multiobjective evolutionary setting for feature selection and a commonality-based crossover operator. In *Evolutionary Computation, 2000. Proceedings of the 2000 Congress on*, volume 1, p. 309–316. IEEE.

Foster, D. P., Karloff, H. J., e Thaler, J. (2014). Variable selection is hard. *CoRR*, abs/1412.4832. URL `http://arxiv.org/abs/1412.4832`.

Guyon, I. e Elisseeff, A. (2003). An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3:1157–1182. ISSN 1532-4435. URL `http://dl.acm.org/citation.cfm?id=944919.944968`.

Harrou, F., Nounou, M. N., Nounou, H. N., e Madakyaru, M. (2015). Pls-based ewma fault detection strategy for process monitoring. *Journal of Loss Prevention in the Process Industries*, 36: 108–119.

Jämsä-Jounela, S.-L. (2007). Future trends in process automation. *Annual Reviews in Control*, 31 (2):211–220.

Jing, C. e Hou, J. (2015). Svm and pca based fault classification approaches for complicated industrial process. *Neurocomputing*, 167:636–642.

Kohavi, R. e John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, 97 (1–2):273 – 324. ISSN 0004-3702. URL `http://www.sciencedirect.com/science/article/pii/S000437029700043X`. Relevance.

Li, A.-D., He, Z., e Zhang, Y. (2016). Bi-objective variable selection for key quality characteristics selection based on a modified nsga-ii and the ideal point method. *Computers in Industry*, 82: 95–103.

Li, B., Zhang, P.-l., Tian, H., Mi, S.-s., Liu, D.-S., e Ren, G.-q. (2011). A new feature extraction and selection scheme for hybrid fault diagnosis of gearbox. *Expert Systems with Applications*, 38(8):10000–10009.

Marill, T. e Green, D. (1963). On the effectiveness of receptors in recognition systems. *IEEE Transactions on Information Theory*, 9(1):11–17. ISSN 0018-9448.

Mukhopadhyay, A., Maulik, U., Bandyopadhyay, S., e Coello, C. A. C. (2014a). A survey of multiobjective evolutionary algorithms for data mining: Part i. *IEEE Transactions on Evolutionary Computation*, 18(1):4–19.

Tong, C., El-Farra, N. H., Palazoglu, A., e Yan, X. (2014). Fault detection and isolation in hybrid process systems using a combined data-driven and observer-design methodology. *AIChE Journal*, 60(8):2805–2814.

Vachhani, V. L., Dabhi, V. K., e Prajapati, H. B. (2016). Improving nsga-ii for solving multi objective function optimization problems. In *Computer Communication and Informatics (ICCCI), 2016 International Conference on*, p. 1–6. IEEE.

Venkatasubramanian, V., Rengaswamy, R., e Kavuri, S. N. (2003b). A review of process fault detection and diagnosis: Part ii: Qualitative models and search strategies. *Computers & Chemical Engineering*, 27(3):313–326.

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S. N., e Yin, K. (2003c). A review of process fault detection and diagnosis: Part iii: Process history based methods. *Computers & chemical engineering*, 27(3):327–346.

Venkatasubramanian, V., Rengaswamy, R., Yin, K., e Kavuri, S. N. (2003a). A review of process fault detection and diagnosis: Part i: Quantitative model-based methods. *Computers & chemical engineering*, 27(3):293–311.

Whitney, A. W. (1971). A direct method of nonparametric measurement selection. *IEEE Transactions on Computers*, C-20(9):1100–1103. ISSN 0018-9340.

Xue, B., Zhang, M., e Browne, W. N. (2012). Multi-objective particle swarm optimisation (pso) for feature selection. In *Proceedings of the 14th annual conference on Genetic and evolutionary computation*, p. 81–88. ACM.

Yin, S., Ding, S. X., Haghani, A., Hao, H., e Zhang, P. (2012). A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark tennessee eastman process. *Journal of Process Control*, 22(9):1567–1581.

Yin, S., Ding, S. X., Xie, X., e Luo, H. (2014a). A review on basic data-driven approaches for industrial process monitoring. *IEEE Transactions on Industrial Electronics*, 61(11):6418–6428.

Yin, S., Gao, X., Karimi, H. R., e Zhu, X. (2014b). Study on support vector machine-based fault detection in tennessee eastman process. In *Abstract and Applied Analysis*, volume 2014. Hindawi Publishing Corporation.

Yu, L. e Liu, H. (2004). Efficient feature selection via analysis of relevance and redundancy. *Journal of machine learning research*, 5(Oct):1205–1224.

Yuen, S. Y. e Chow, C. K. (2009). A genetic algorithm that adaptively mutates and never revisits. *IEEE transactions on evolutionary computation*, 13(2):454–472.

Zhu, Z., Ong, Y.-S., e Dash, M. (2007). Wrapper–filter feature selection algorithm using a memetic framework. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 37(1): 70–76.