

# UMA ABORDAGEM BASEADA EM GRANULAÇÃO *FUZZY* E MÁQUINAS DE VETORES SUPORTE PARA PROGNÓSTICO DE FALHAS

MURILO CESAR O. CAMARGOS FILHO\*, REINALDO M. PALHARES†, MARCOS FLÁVIO S. V. D'ANGELO‡

\*Programa de Pós-Graduação em Engenharia Elétrica - Universidade Federal de Minas Gerais - Av. Antônio Carlos 6627, 31270-901, Belo Horizonte, MG, Brasil

†Departamento de Engenharia Eletrônica - Universidade Federal de Minas Gerais - Av. Antônio Carlos 6627, 31270-901, Belo Horizonte, MG, Brasil

‡Departamento de Ciência da Computação - Universidade Estadual de Montes Claros - Av. Rui Braga s/n, Vila Mauricéia, 39401-089, Montes Claros, MG, Brasil

Emails: murilo.camargosf@gmail.com, rpalhares@ufmg.br, marcos.dangelo@unimontes.br

**Abstract**— This paper proposes a data-based approach for fault prognosis. The proposed method uses an information granulation technique based on fuzzy forms to represent a machine's wear time series; then the granules are modeled with the least squares support vector machines (LSSVM) which parameters are optimized by the Nelder–Mead simplex algorithm. The approach was successfully applied for cutter's wear prognosis of a computer numerical control (CNC) machine. The results indicate that the use of Nelder–Mead's simplex can considerably improve the prediction accuracy and reduce the time spent at the training step.

**Keywords**— Prognosis, Fuzzy Granulation, Support Vector Machines (SVM), Nelder–Mead.

**Resumo**— Este artigo propõe uma abordagem baseada em dados históricos para prognóstico de falha. A abordagem proposta utiliza uma técnica para granulação da informação baseada em formas *fuzzy* para representar a série temporal dos dados de desgaste em uma máquina; em seguida, os grânulos são modelados através de máquinas de vetores suporte por mínimos quadrados (LSSVM), cujos parâmetros são otimizados através do algoritmo simplex de Nelder–Mead. A abordagem foi aplicada com sucesso no prognóstico do desgaste da fresa em máquinas de comando numérico computadorizado (CNC). Os resultados mostram que a utilização do simplex de Nelder–Mead, quando comparados com outros algoritmos de otimização, pode melhorar a acurácia da previsão e diminuir o tempo gasto na fase de treinamento.

**Palavras-chave**— Prognóstico, Granulação *Fuzzy*, Máquinas de Vetores Suporte (SVM), Nelder–Mead.

## 1 Introdução

Confiabilidade é uma aspecto muito importante em produtos e equipamentos, principalmente na indústria. Estes equipamentos, com o passar do tempo, necessitam de manutenção tendo em vista sua deterioração ou envelhecimento. Esta manutenção pode ser feita quando um equipamento quebra, de forma programada ou com base na condição do equipamento, sendo a última capaz de reduzir custos envolvidos neste processo (Jardine et al., 2006).

Por questões econômicas, é desejável que o processo de detecção, diagnóstico e prognóstico de falhas seja baseado em métodos computacionais que possam processar um grande número de variáveis. Essas tarefas podem ser abordadas com base em modelos matemáticos ou dados históricos (Venkatasubramanian et al., 2003).

O prognóstico baseado em modelos utiliza a física do processo para descrever a previsão de falhas (Sikorska et al., 2011). Em (Liu et al., 2017), o prognóstico em circuitos analógicos é feito através de um modelo baseado em sinais, enquanto que em (Poon et al., 2017), um modelo para estimação de estados em conversores de potência é usado para detecção e identificação de falhas. Em

(Wang et al., 2016), o filtro de Kalman é utilizado para prognóstico de falhas em rolamentos.

As abordagens baseadas em dados históricos requerem pouco ou nenhum conhecimento acerca da física do processo. A previsão da vida útil do equipamento pode ser feita utilizando técnicas Bayesianas, por exemplo (Mosallam et al., 2014). Em (Saidi et al., 2017) utiliza-se a técnica de regressão por vetores suporte para previsão da condição em eixos de alta velocidade em turbinas eólicas. Redes neuro-*fuzzy* foram utilizados em (Razavi-Far et al., 2016) para estimar a vida útil remanescente em baterias de íons de lítio.

Este trabalho apresenta uma abordagem baseada em granulação *fuzzy* associada à máquinas de vetores suporte (SVM) e foi adaptada da proposta apresentada em (Yin et al., 2017), com consideráveis modificações na metodologia para otimização de parâmetros e para geração de grânulos. A vantagem dessa abordagem é a utilização de grânulos de informação, pois, comparados aos dados brutos, os grânulos podem representar informações num nível maior de abstração. A abordagem proposta foi testada em dados de desgaste de uma máquina de comando numérico computadorizado (CNC) e apresentou resultados significativos no contexto de prognóstico de falhas. O restante do

artigo está organizado da seguinte maneira: na Seção 2 a metodologia e os algoritmos utilizados são detalhados. A Seção 3 ilustra a base de dados, delimita o experimento e apresenta os resultados obtidos. Por fim, na Seção 4 são apresentadas as conclusões e sugestões para estudos futuros.

## 2 Metodologia

A metodologia proposta foi inspirada no modelo apresentado em Yin et al. (2017) para previsão de uma série temporal granulada. O esquema básico desta metodologia é utilizar uma técnica de granulação dos dados, treinar algum algoritmo de regressão com os dados granulares, otimizar parâmetros dessa regressão e utilizar o modelo construído para prever novos grânulos da série, como ilustrado no diagrama de blocos da Figura 1.

Neste trabalho, a técnica de granulação escolhida foi a técnica proposta por Pedrycz and Vukovich (2002), o algoritmo de regressão utilizado foi o SVM por mínimos quadrados (LS-SVM) (Suykens et al., 2002) e o algoritmo de otimização dos parâmetros utilizado foi o método simplex de Nelder–Mead (Nelder and Mead, 1965).

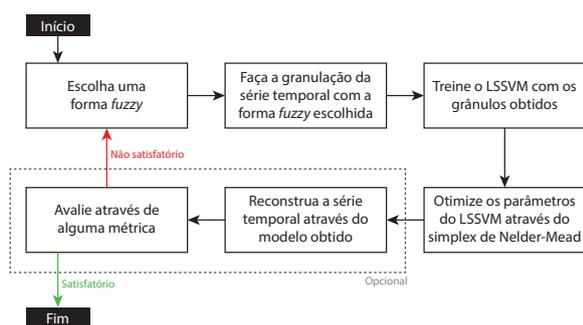


Figura 1: Diagrama de blocos da fase de treinamento da metodologia.

A principal diferença do presente trabalho em relação à metodologia proposta por Yin et al. (2017) é a utilização do simplex de Nelder–Mead para otimização dos parâmetros de regressão, possibilitando a utilização da metodologia num contexto de prognóstico de falhas.

As simulações foram feitas no MATLAB 2015b e, para fins de comparação, outros três algoritmos para otimização dos parâmetros foram testados, são eles: algoritmo genético, enxame de partículas (*Particle Swarm Optimization*: PSO) e recozimento simulado (*Simulated Annealing*). Para cada algoritmo, foram utilizados os parâmetros preestabelecidos pelo MATLAB<sup>1</sup>, ou seja, não houve ajuste que pudesse melhorar sua resposta ao problema.

<sup>1</sup>Disponível em: <https://www.mathworks.com/help/gads>

### 2.1 Granulação Fuzzy

Os grânulos de informação são coleções de entidades que se originam, tipicamente, num nível numérico e são agrupadas de acordo com sua similaridade, adjacência funcional, indistinguibilidade, coerência ou semelhança (Bargiela and Pedrycz, 2003).

Tomando um exemplo real, podemos definir que os grânulos da cabeça humana são nariz, testa, olhos, boca, ouvido, etc. No entanto, existem elementos cujas fronteiras não são bem definidas, dificultando a utilização da teoria clássica de conjuntos; a fronteira que separa a testa da bochecha, por exemplo, não possui uma definição clara. Neste contexto surge a granulação *fuzzy*, modelando as incertezas que permeiam as fronteiras desses conjuntos (Zadeh, 1997).

Em séries temporais, geralmente, a granulação é feita por adjacência funcional através de duas etapas. A primeira etapa divide a série temporal numa sequência de janelas ( $W_i$ ); uma metodologia bastante utilizada é escolher janelas de tamanho fixo ( $N$ ) sem sobreposições.

A segunda etapa é chamada de granulação e é feita em cada uma das janelas; para cada janela será calculado um grânulo que irá representá-la (Lu et al., 2014). Tomando uma determinada forma *fuzzy* (triangular, trapezoidal, Gaussiana assimétrica, etc), definida por uma função de pertinência  $A$ , os grânulos serão construídos de forma que englobem o máximo de elementos da janela e, ao mesmo tempo, mantenham especificidade. Para isso, é preciso maximizar a pertinência dos dados à forma *fuzzy* escolhida e minimizar o tamanho do suporte dessa forma. Estes dois objetivos são conflitantes e podem ser escritos da seguinte forma:

$$\max \frac{\sum_{i=1}^N A(x_i)}{\text{supp}(A)}, \quad (1)$$

em que  $N$  é o tamanho da janela  $W$ ,  $x_i$  são os valores da série temporal nesta janela,  $A(\cdot)$  é a função de pertinência da forma *fuzzy* escolhida e  $\text{supp}(\cdot)$  é o tamanho do suporte desta função de pertinência. Neste trabalho foi utilizada a forma *fuzzy* triangular, sua função de pertinência é dada pela Equação 2.

$$A(x; a, m, b) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{m-a}, & a < x \leq m \\ \frac{b-x}{b-m}, & m < x \leq b \\ 0, & b < x \end{cases} \quad (2)$$

Os valores dos parâmetros que compõem a forma triangular ( $a, m, b$ ) podem ser encontrados através da solução de três problemas de otimização. O primeiro, dado pela Equação 3, define o valor de  $m$ .

$$m = \operatorname{argmin}_x \sum_{i=1}^N |x - w_i|, \quad (3)$$

em que  $w_i$  é o  $i$ -ésimo escalar da janela  $W$  de tamanho  $N$  e  $x$  o escalar que minimiza o somatório dos módulos.

Os outros dois parâmetros ( $a$ ,  $b$ ) que determinam a forma triangular são encontrados a partir das Equações 4a e 4b.

$$a = \operatorname{argmax}_x \frac{\sum_{i=1}^{n'} A(w_i)}{m - x}, \quad (4a)$$

$$b = \operatorname{argmax}_x \frac{\sum_{i=n'}^N A(w_i)}{x - m}, \quad (4b)$$

em que  $w_i$  é o  $i$ -ésimo escalar da janela  $W$  de tamanho  $N$ ,  $x$  o escalar que maximiza os quocientes, e  $n'$  é o índice do maior elemento da janela  $W$  menor que  $m$ .

## 2.2 LSSVM

Após a granulação, um modelo não-linear autorregressivo (NLAR) é ajustado à série temporal granular através do algoritmo LSSVM. Considere o modelo NLAR a seguir:

$$x_t = F(x_{t-1}, x_{t-2}, \dots, x_{t-L}) + \varepsilon_t, \quad (5)$$

em que  $\varepsilon_t$  é o erro da aproximação,  $L$  é o tamanho da janela autorregressiva e  $F(\cdot)$  uma função não linear. As previsões a partir de um certo instante  $t$  são dadas por:

$$\hat{x}_t = F(x_{t-1}, x_{t-2}, \dots, x_{t-L}) + \varepsilon_t, \quad (6a)$$

$$\hat{x}_{t+1} = F(\hat{x}_t, x_{t-1}, \dots, x_{t-L+1}) + \varepsilon_{t+1}. \quad (6b)$$

em que  $\hat{x}$  denota um valor previsto com base nos  $L$  valores anteriores, sejam eles estimados ou reais.

O algoritmo LSSVM foi proposto por Suykens et al. (2002) como uma alternativa ao método SVM em que as variáveis de folga são substituídas pelo erro quadrático, resolvendo, assim, um sistema de equações lineares ao invés de um problema de computação quadrática. Por este motivo, o LSSVM reduz a complexidade computacional do SVM tradicional (Ding and Liao, 2011); no entanto, esta redução vem com uma perda de generalização da solução uma vez que os multiplicadores de Lagrange são não nulos em todos os pontos, indicando que todas as amostras são utilizadas na construção do modelo (Yin et al., 2017).

O modelo de regressão do LSSVM pode ser escrito como:

$$y = \mathbf{w} \cdot \phi(\mathbf{x}) + b, \quad (7)$$

em que  $x \in \mathbb{R}^d$ ,  $y \in \{-1, 1\}$ ,  $\mathbf{w}$  é o vetor de pesos,  $b$  é um termo de *bias* e  $\phi(\cdot)$  denota uma transformação não linear. Nesta formulação, a função custo a ser minimizada é dada por:

$$\min_{\mathbf{w}, b} L = \frac{1}{2} \|\mathbf{w}\|^2 + \frac{1}{2} \zeta \sum_{i=1}^n \varepsilon_i^2 \quad (8)$$

$$\text{s.a. } y_i = \mathbf{w} \cdot \phi(\mathbf{x}) + b + \varepsilon_i$$

em que  $\varepsilon_i$  é o erro aleatório e  $\zeta$  denota o parâmetro de compromisso entre suavidade e minimização do erro de treinamento, conhecido por parâmetro de regularização (Zhang et al., 2015).

Este problema pode ser resolvido através do método dos multiplicadores de Lagrange já que a função objetivo está sujeita à restrições de igualdade. Com a introdução de uma função Kernel, o modelo LSSVM pode ser reescrito como:

$$y(\mathbf{x}) = \sum_{i=1}^n \alpha_i K(\mathbf{x}, \mathbf{x}_i) + b, \quad (9)$$

em que  $\alpha_i$  são os multiplicadores de Lagrange. Neste trabalho, o Kernel RBF (*Radial Basis Function*) – também conhecido como kernel Gaussiano – foi escolhido por ser amplamente utilizado e por mapear os dados a um espaço de dimensão infinita (Chang et al., 2010). O Kernel RBF é dado por:

$$K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\delta^2}\right), \quad (10)$$

em que  $\delta$  representa um parâmetro livre.

Analisando (8) e (10) nota-se que a qualidade do modelo gerado por este algoritmo depende da escolha dos parâmetros  $\zeta$  e  $\delta$  que, neste trabalho, são otimizados através do método simplex de Nelder–Mead.

## 2.3 O método simplex de Nelder–Mead

Apesar de ser baseado em Yin et al. (2017), o modelo proposto neste trabalho utiliza o método simplex de Nelder–Mead para otimização dos parâmetros utilizados pelo algoritmo de regressão. Este método foi proposto em 1965 (Nelder and Mead, 1965) para minimização de funções com valores reais tais que  $f(x) \in \mathbb{R}$  e  $x \in \mathbb{R}^n$ . Dois motivos foram cruciais para a escolha deste método: (a) é não-linear, irrestrito e não utiliza o gradiente da função objetivo; (b) é bastante eficiente quando a dimensão do problema é pequeno, no caso, a dimensão no problema a ser tratado é 2. Na seção de resultados será ilustrada a eficiência desta proposta.

O método realiza quatro operações importantes para minimização e a cada uma é associado um parâmetro constante, são elas: reflexão ( $\alpha$ ), expansão ( $\gamma$ ), contração ( $\rho$ ) e encolhimento ( $\sigma$ ) (Lagarias et al., 1998). O algoritmo inicia-se com  $n+1$  vértices  $x \in \mathbb{R}^n$  e itera até que um critério de parada seja satisfeito. Por padrão, o critério escolhido é a análise do desvio padrão dos valores da função objetivo em relação a uma tolerância.

Cada iteração do algoritmo passa pelos seguintes passos:

1. **Ordenação.** Os  $n + 1$  vértices são ordenados de acordo com o valor da função objetivo, satisfazendo  $f(x_1) \leq f(x_2) \leq \dots \leq f(x_{n+1})$ . Além disso, calcula-se o centroide dos vértices com exceção do último,  $\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$ .

2. **Reflexão.** Calcula-se o vértice de reflexão dado por:

$$x_r = \bar{x} + \alpha(\bar{x} - x_{n+1})$$

caso  $f(x_1) \leq f(x_r) \leq f(x_n)$ , aceita-se o vértice (leia-se: substituição de  $x_{n+1}$  por  $x_r$ ) e a iteração é encerrada.

3. **Expansão.** Caso  $f(x_r) < f(x_1)$ , é calculado o vértice de expansão, dado por:

$$x_e = \bar{x} + \gamma(x_r - \bar{x})$$

caso  $f(x_e) < f(x_r)$ , o vértice de expansão  $x_e$  é aceito e a iteração é encerrada; caso contrário aceita-se o vértice de reflexão  $x_r$  e a iteração é encerrada.

4. **Contração.** Note que neste caso tem-se  $f(x_r) \geq f(x_n)$ ; dessa forma, o vértice de contração é dado por:

$$x_c = \bar{x} + \rho(x_{n+1} - \bar{x})$$

caso  $f(x_c) < f(x_{n+1})$  aceita-se o vértice de contração e a iteração é encerrada.

5. **Encolhimento.** Por fim, caso nenhum dos casos sejam explorados, faz-se uma contração dos pontos, trazendo-os mais próximos do melhor, da forma:

$$x_i = x_1 + \sigma(x_i - x_1), \quad i \in [2, n + 1]$$

Foram utilizados os valores padrão dos parâmetros de reflexão, expansão, contração e encolhimento  $\alpha = 1$ ,  $\gamma = 2$ ,  $\rho = 0.5$  e  $\sigma = 0.5$ . Esses valores foram escolhidos em consonância com as análises de convergência encontradas em Nelder and Mead (1965). Em cada iteração desse algoritmo, uma função objetivo deve ser avaliada; como o foco é reduzir o erro de previsão da série temporal, optou-se por utilizar a raiz quadrada do erro quadrático médio (RMSE) como função objetivo a ser minimizada, otimizando os parâmetros  $\zeta$  e  $\delta$  do LSSVM:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{n}}, \quad (11)$$

em que  $\hat{y}_i$  representa a  $i$ -ésima previsão.

### 3 Resultados

Os experimentos foram realizados com os dados fornecidos pela *Prognostics and Health Management Society* (PHMS)<sup>2</sup>. Este conjunto é composto por dados históricos de seis máquinas ( $c_1, c_2, \dots, c_6$ ) CNC com três fresas cada, realizados em peças metálicas idênticas. Os dados de força, vibração e emissão acústica dos 315 cortes foram gravados por uma estrutura composta de alguns sensores (Figura 2), um cartão de aquisição de dados e um computador (Li et al., 2009).

Foi verificado que, dos seis experimentos, apenas três ( $c_1, c_4$  e  $c_6$ ) possuíam informação do nível de desgaste, portanto, estes três conjuntos foram utilizados para treinamento, validação e teste, respectivamente. Neste trabalho, o nível de desgaste máximo entre as três fresas de cada máquina é utilizado como variável de interesse.

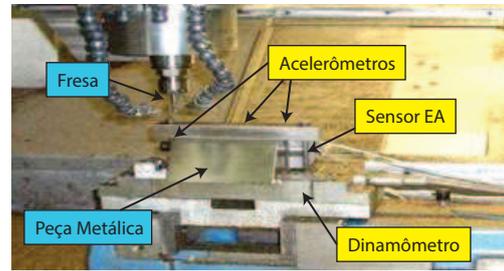


Figura 2: Máquina CNC de alta velocidade Røders Tech RFM760 (Li et al., 2009).

#### 3.1 Treinamento e validação

Os primeiros parâmetros a serem definidos para inicialização do treinamento, são o tamanho da janela de granulação ( $N$ ), mostrado em (1), e o tamanho da janela no modelo autorregressivo ( $L$ ), descrito em (5).

Para isso, uma busca em grade foi utilizada com os dados da máquina 1 ( $c_1$ ) no treinamento e os dados da máquina 4 ( $c_4$ ) na validação. Os valores de RMSE para  $N$  variando de 3 a 10 e  $L$  variando de 1 a 8 são mostrados na Tabela 1. O melhor resultado, avaliado pela métrica RMSE, foi alcançado quando foram utilizadas janelas de tamanho  $N = 4$  na fase de granulação e janelas de tamanho  $L = 4$  no modelo autorregressiva.

#### 3.2 Testes

Após selecionar os tamanhos das janelas de granulação e autorregressão, a metodologia proposta foi testada e comparada com a utilização de outros algoritmos para otimização dos parâmetros do LS-SVM. A comparação é feita através de duas métricas: a primeira é a raiz quadrada do erro quadrático médio, dado em (11) e a segunda corresponde

<sup>2</sup>Disponível em: <https://www.phmsociety.org/competition/phm/10>

Tabela 1: Valores de RMSE para validação dos parâmetros de granulação e autorregressão do modelo.

	$N = 3$	$N = 4$	$N = 5$	$N = 6$	$N = 7$	$N = 8$	$N = 9$	$N = 10$
$L = 1$	13.102	12.816	13.252	11.817	11.795	11.969	11.546	11.771
$L = 2$	12.987	16.093	10.237	11.392	10.891	11.169	10.549	11.027
$L = 3$	12.722	10.128	10.192	10.835	10.519	11.138	11.508	14.592
$L = 4$	10.147	<b>10.067</b>	12.221	13.083	12.724	18.104	19.405	12.948
$L = 5$	10.100	12.956	13.426	30.622	44.187	15.031	14.513	12.294
$L = 6$	12.293	13.778	21.438	45.010	13.342	12.302	12.236	14.010
$L = 7$	12.695	17.625	1163.552	13.704	11.474	12.536	14.456	39.079
$L = 8$	13.080	12.677	13.452	11.495	11.851	216.846	281.969	44.316

à média absoluta do erro percentual (MAPE), dada por:

$$\text{MAPE} = \frac{100}{n} \sum_{t=1}^n \left| \frac{y_t - \hat{y}_t}{y_t} \right|, \quad (12)$$

em que  $y_t$  é o valor real da série e  $\hat{y}_t$  é o valor previsto. Os resultados comparativos são mostrados na Tabela 2. Os comportamentos das previsões utilizando os quatro algoritmos para otimização dos parâmetros do LSSVM podem ser vistos nas Figuras 3a, 3b, 3c e 3d.

Os resultados da Tabela 2 levam em consideração um cenário em que os dados da série  $c_1$  são usados para treinamento e os dados da série  $c_6$  para teste. Para confirmar a robustez do método, outras combinações de cenários foram testadas e os valores de RMSE e MAPE para essas combinações são mostrados na Tabela 3.

A previsão é feita em três séries temporais distintas, uma para cada parâmetro da forma triangular utilizada:  $a$ ,  $m$  e  $b$ ; com,  $a_i \geq m_i \geq b_i$ . Cada grânulo representa uma janela da série temporal real, modelando o máximo de informações daquela janela; isso significa que os valores reais de uma janela estão, em sua maioria, dentro dos intervalos granulares. Portanto, mostrar o intervalo de predição (IP) para cada uma das três séries temporais não apresenta informações mais relevantes que mostrar somente o maior e o menor intervalos. Os IP foram calculados em 95% e mostrados nas Figuras 3a, 3b, 3c e 3d.

Pode-se notar que o resultado da previsão de uma das séries temporais ( $b$ ) utilizando o algoritmo genético foi bastante divergente da previsão das outras duas séries ( $a$  e  $m$ ). Uma hipótese que pode explicar este fenômeno é a possibilidade de o algoritmo de otimização dos parâmetros do LSSVM ficar preso em ótimos locais. Esta hipótese também explica alguns valores muito altos de RMSE mostrados na Tabela 1.

Utilizar o erro quadrático médio como função objetivo fez com que as fases de treinamento para cada algoritmo durassem um tempo considerável. No entanto, o tempo gasto pela abordagem proposta, utilizando o simplex de Nelder–Mead, foi consideravelmente menor. Isso se dá pelo fato de que a quantidade de novas soluções que precisam

ser avaliadas pela função objetivo em cada iteração do algoritmo é menor.

A semelhança entre os valores de erro para as séries granulares, previstas a partir dos outros algoritmos (Tabela 2), pode estar relacionada com o valor ótimo dos parâmetros  $\zeta$  e  $\delta$  encontrados por estes métodos. Como eles foram treinados utilizando o desgaste da máquina  $c_1$  de forma que o erro quadrático médio de uma previsão da própria  $c_1$  fosse minimizado, é possível que os parâmetros encontrados tenham gerado *overfitting*, ou seja, ao utilizar estes parâmetros para prever o comportamento da série  $c_6$ , os algoritmos não mostraram robustez. Este problema parece não ter ocorrido na utilização do simplex de Nelder–Mead pois, como mostrado na Figura 3d, o comportamento da série de desgaste foi generalizado o suficiente para que o modelo pudesse ser aplicado numa outra série desconhecida  $c_6$ .

A metodologia proposta se mostrou competitiva, inclusive, em relação a (Tobon-Mejia et al., 2012), onde redes Bayesianas dinâmicas foram utilizadas para prognóstico de falha no mesmo conjunto de dados utilizado neste trabalho. Neste trabalho, a acurácia é calculada utilizando a equação:

$$A = \frac{1}{n} \sum_{t=1}^n \exp\left(-\frac{|y_t - \hat{y}_t|}{y_t}\right), \quad (13)$$

no qual  $n$  é a quantidade de cortes,  $y_t$  é o valor de desgaste real e  $\hat{y}_t$  o valor de desgaste predito. A abordagem proposta atingiu uma acurácia média de **95.99%**, enquanto a abordagem proposta em Tobon-Mejia et al. (2012) atingiu uma acurácia média de 84.67%, ambas utilizando os dados da máquina  $c_6$  para teste.

## 4 Conclusões

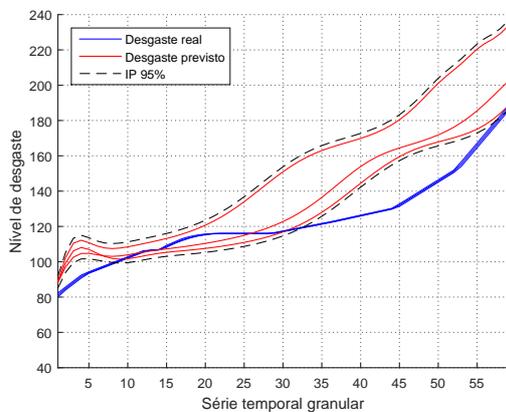
Neste trabalho foi proposto uma metodologia que estende a proposta apresentada em Yin et al. (2017) para o prognóstico de falhas em máquinas CNC e que utiliza o algoritmo simplex de Nelder–Mead para a otimização dos parâmetros da regressão LSSVM. A utilização deste algoritmo se mostrou mais eficiente tanto em acurácia quanto em tempo de processamento quando comparado a proposta em Yin et al. (2017).

Tabela 2: Valores de RMSE e MAPE para as três séries granulares  $a$ ,  $m$  e  $b$  e tempo total de execução para os testes no conjunto  $c_6$ .

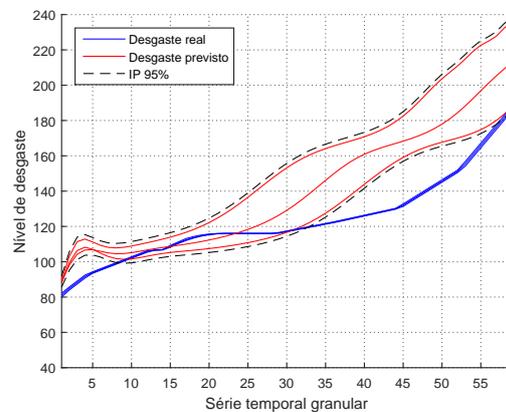
Algoritmo	RMSE			MAPE			Tempo (s)
	$a$	$m$	$b$	$a$	$m$	$b$	
Algoritmo genético	58.371	11.270	<b>9.315</b>	40.140	6.559	5.447	968.522
Enxame de partículas	35.926	11.655	20.875	21.944	6.687	12.005	450.799
Recozimento simulado	34.276	11.802	15.533	20.809	6.753	8.827	527.192
Simplex de Nelder–Mead	<b>9.102</b>	<b>8.898</b>	9.974	<b>4.063</b>	<b>4.035</b>	<b>4.295</b>	<b>47.206</b>

Tabela 3: Valores de RMSE e MAPE para as séries granulares em diferentes cenários.

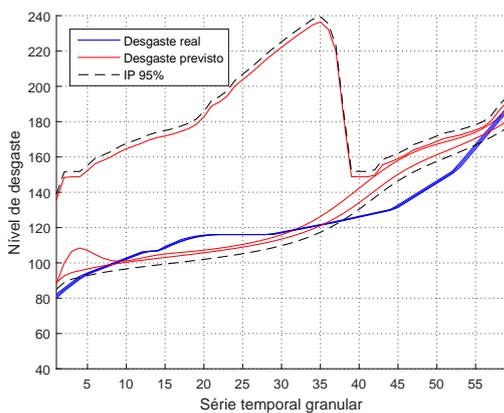
Treinamento	Teste	RMSE			MAPE		
		$a$	$m$	$b$	$a$	$m$	$b$
$c_1$	$c_4$	9.913	10.012	10.067	6.853	6.871	6.792
$c_4$	$c_6$	8.915	9.754	10.200	3.685	3.717	3.892
$c_4$	$c_1$	7.273	8.097	8.521	4.716	5.082	5.295
$c_6$	$c_1$	5.714	5.739	6.371	3.876	3.904	4.191
$c_6$	$c_4$	11.358	12.171	13.407	7.621	8.031	8.746



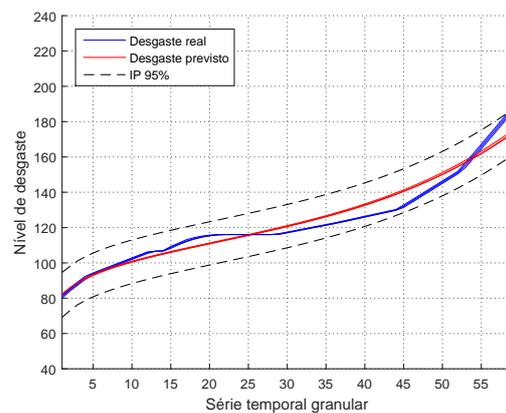
(a) Recozimento Simulado



(b) Enxame de Partículas



(c) Algoritmo Genético



(d) Simplex de Nelder–Mead

Figura 3: Previsão da série  $c_6$  utilizando quatro algoritmos para otimização dos parâmetros do LSSVM.

Uma sugestão para estudos futuros é analisar a eficiência desta metodologia utilizando outras formas *fuzzy* além da triangular e em outras bases de dados. Aliado a isso, é interessante investigar a criação de múltiplos modelos para diferentes estágios de desgaste utilizando uma técnica adequada

para escolha desses modelos em tempo real.

### Agradecimentos

Os autores agradecem o suporte financeiro da CAPES, FAPEMIG e CNPq.

## Referências

- Bargiela, A. and Pedrycz, W. (2003). *Granular Computing*, Springer US.
- Chang, Y.-W., Hsieh, C.-J., Chang, K.-W., Ringgaard, M. and Lin, C.-J. (2010). Training and testing low-degree polynomial data mappings via linear svm, *Journal of Machine Learning Research* **11**(Apr): 1471–1490.
- Ding, L. and Liao, S. (2011). Approximate model selection for large scale lssvm, *Asian Conference on Machine Learning*, pp. 165–180.
- Jardine, A. K., Lin, D. and Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance, *Mechanical Systems and Signal Processing* **20**(7): 1483–1510.
- Lagarias, J. C., Reeds, J. A., Wright, M. H. and Wright, P. E. (1998). Convergence properties of the nelder–mead simplex method in low dimensions, *SIAM Journal on Optimization* **9**(1): 112–147.
- Li, X., Lim, B., Zhou, J., Huang, S., Phua, S., Shaw, K. and Er, M. (2009). Fuzzy neural network modelling for tool wear estimation in dry milling operation, *Annual Conference of the Prognostics and Health Management Society*, pp. 1–11.
- Liu, Z., Liu, T., Han, J., Bu, S., Tang, X. and Pecht, M. (2017). Signal model-based fault coding for diagnostics and prognostics of analog electronic circuits, *IEEE Transactions on Industrial Electronics* **64**(1): 605–614.
- Lu, W., Pedrycz, W., Liu, X., Yang, J. and Li, P. (2014). The modeling of time series based on fuzzy information granules, *Expert Systems with Applications* **41**(8): 3799–3808.
- Mosallam, A., Medjaher, K. and Zerhouni, N. (2014). Data-driven prognostic method based on bayesian approaches for direct remaining useful life prediction, *Journal of Intelligent Manufacturing* **27**(5): 1037–1048.
- Nelder, J. A. and Mead, R. (1965). A simplex method for function minimization, *The Computer Journal* **7**(4): 308–313.
- Pedrycz, W. and Vukovich, G. (2002). Feature analysis through information granulation and fuzzy sets, *Pattern Recognition* **35**(4): 825–834.
- Poon, J., Jain, P., Konstantakopoulos, I. C., Spanos, C., Panda, S. K. and Sanders, S. R. (2017). Model-based fault detection and identification for switching power converters, *IEEE Transactions on Power Electronics* **32**(2): 1419–1430.
- Razavi-Far, R., Farajzadeh-Zanjani, M., Chakrabarti, S. and Saif, M. (2016). Data-driven prognostic techniques for estimation of the remaining useful life of lithium-ion batteries, *2016 IEEE International Conference on Prognostics and Health Management (ICPHM)*, IEEE.
- Saidi, L., Ali, J. B., Bechhoefer, E. and Benbouzid, M. (2017). Wind turbine high-speed shaft bearings health prognosis through a spectral kurtosis-derived indices and SVR, *Applied Acoustics* **120**: 1–8.
- Sikorska, J., Hodkiewicz, M. and Ma, L. (2011). Prognostic modelling options for remaining useful life estimation by industry, *Mechanical Systems and Signal Processing* **25**(5): 1803–1836.
- Suykens, J. A. K., Gestel, T. V., Brabanter, J. D., Moor, B. D. and Vandewalle, J. (2002). *Least Squares Support Vector Machines*, WORLD SCIENTIFIC.
- Tobon-Mejia, D., Medjaher, K. and Zerhouni, N. (2012). Cnc machine tool’s wear diagnostic and prognostic by using dynamic bayesian networks, *Mechanical Systems and Signal Processing* **28**: 167–182.
- Venkatasubramanian, V., Rengaswamy, R., Yin, K. and Kavuri, S. N. (2003). A review of process fault detection and diagnosis, *Computers & Chemical Engineering* **27**(3): 293–311.
- Wang, Y., Peng, Y., Zi, Y., Jin, X. and Tsui, K.-L. (2016). A two-stage data-driven-based prognostic approach for bearing degradation problem, *IEEE Transactions on Industrial Informatics* **12**(3): 924–932.
- Yin, S., Jiang, Y., Tian, Y. and Kaynak, O. (2017). A data-driven fuzzy information granulation approach for freight volume forecasting, *IEEE Transactions on Industrial Electronics* **64**(2): 1447–1456.
- Zadeh, L. A. (1997). Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic, *Fuzzy Sets and Systems* **90**(2): 111–127.
- Zhang, N., Kamaha, R. and Behera, P. (2015). Prediction of surface water supply sources for the district of columbia using least squares support vector machines (ls-svm) method, *Advances in Computer Science : an International Journal* **4**(1): 1–9.