

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

Identificação por subespaços usando C/C++ aplicado a um processo industrial

Santos D. M. Borjas¹

Universidade Federal do Rio Grande do Norte, UFRN, Natal, RN

Guilherme A. P. de C. A. Pessoa²

Universidade Federal do Rio Grande do Norte, UFRN, Natal, RN

Grande parte da literatura resolve o problema de identificação de sistemas utilizando o algoritmo de subespaços aplicado ao software MATLAB, como em [3]. Para, usar este software é necessário ter uma licença, que muitas vezes é cara para alguns estudantes e não se tem acesso ao seu código fonte, impossibilitando realizar modificações fundamentais nos algoritmos em estudo. O objetivo deste trabalho é implementar em C/C++ o algoritmo de identificação robusta usando métodos por subespaços, possibilitando embarcar o código em microcontroladores simples como o PIC16F84A. O algoritmo é testado na identificação de três processos industriais: unidade de craqueamento catalítico, processo de fabricação de tubos de vidro e um braço robótico.

Os métodos lineares de identificação por subespaços estão relacionados com modelos da forma:

$$x_{k+1} = Ax_k + Bu_k + Ke_k \quad (1)$$

$$y_k = Cx_k + Du_k + e_k \quad (2)$$

onde A, B, C, D e K são matrizes de dimensões apropriadas [2].

O problema da identificação por subespaços é: dada uma sequência de dados de entrada u e saída y determine a ordem n e as matrizes (A, B, C, D, K) do sistema desconhecido. O algoritmo robusto soluciona este problema através da seguinte sequência [4]

$$1) \Theta_i = Y_f / U_f W_p ; Z_i = Y_f / (\frac{W_p}{U_f}) ; Z_{i+1} = Y_f^- / (\frac{W_p^+}{U_f^-})$$

$$2) \text{Calcular SVD de } \theta_i \Pi_{U_f^-} = USV^T$$

3) Determine n a partir de S

$$4) \text{Compute } \Gamma_i \approx U_1 S_1^{1/2} \text{ e } \Gamma_{i-1} = \underline{\Gamma}_i$$

$$5) \text{As matrizes } A \text{ e } C \text{ são solução da equação } (\frac{\Gamma_{i-1}^+ \cdot Z_{i+1}}{Y_{ii}}) = (\frac{A}{C}) \cdot \Gamma_i^+ \cdot Z_i + G \cdot U_f + (\frac{\rho_w}{\rho_v}).$$

Para computar as matrizes B, D e K ver [4].

Os algoritmos implementados em MATLAB e C/C++ estão disponíveis em:

¹santos.borjas@ufersa.edu.br

²guilhermepillon@hotmail.com

<https://goo.gl/dnLyaY>.

Foram utilizados três processos industriais, com medições reais para validação e comparação do algoritmo robusto implementado em C/C++. Unidade de craqueamento catalítico com 7 entradas, 6 saídas com 8639 amostras. Processo de fabricação de tubos de vidro, 2 entradas, 2 saídas com 1300 amostras. Braço robótico com 1020 amostras. Em todos os casos 75% das amostras foram utilizadas para identificação e 25% foram usadas na validação cruzada. As amostras destes processos encontram-se em [4] e [1].

Tabela 1: Resultados numéricos do desempenho dos algoritmos.

	Planta FCC	Tubo de Vidro	Braço Robótico
	Matlab — C/C++	Matlab — C/C++	Matlab — C/C++
Tempo (s)	0,65 - 0,57	0,198 - 0,099	0,157 - 0,87
FIT (%)	95,43 - 95,43	89,3 - 89,3	98,3 - 98,3

Analisando os valores da Tabela 1, todos os algoritmos tiveram um bom desempenho em termos de validação cruzada. Verifica-se que o tempo de processamento para obtenção do modelo é menor para o algoritmo implementado em C/C++.

Agradecimentos

Agradecemos ao programa institucional de bolsas iniciação científica, PIBITI UFRN.

Referências

- [1] S.D.M. Borjas, C. Garcia, Modelagem de FCC usando métodos de identificação por predição de erro e por subespaços, *IEEE América Latina, Revista virtual No. 2* 108-113, 2004.
- [2] S.D.M. Borjas, Estudo da identificação por subespaços em malha aberta e fechada e proposta de novos algoritmos, Tese de Doutorado, POLI USP, 2009.
- [3] I. W. Jamaludin, N. A. Wahab, N. S. Khalid, S. Sahlan, Z. Ibrahim and M. F. Rahmat, N4SID and MOESP subspace identification methods, *IEEE 9th International Colloquium on Signal Processing and its Applications*, Kuala Lumpur, 2013.
- [4] P.V. Overschee, B.L.R. De Moor. N4SID : Subspace Identification of a Glass Tube, 1998.