



CALIBRAÇÃO NEURAL DE PARÂMETROS DE UM OSCILADOR AMORTECIDO

Estevão Fuzaro de Almeida¹, Samuel da Silva¹, Hernán Garrido^{2,3}, Martín Domizio^{2,3}, Daniel Ambrosini^{2,3}

¹ *Universidade Estadual Paulista, Faculdade de Engenharia, Departamento de Engenharia Mecânica, Ilha Solteira, SP, Brasil, estevao.fuzaro@unesp.br, samuel.silva13@unesp.br;*

² *Consejo Nacional de Investigaciones Científicas y Técnicas, Argentina;*

³ *Universidad Nacional de Cuyo, Facultad de Ingeniería, Instituto de Mecánica Estructural y Riesgo Sísmico, Mendoza, Argentina.*

RESUMO: Os modelos computacionais tornaram-se uma ferramenta poderosa nas ciências quantitativas para compreender o comportamento de sistemas complexos que evoluem no domínio temporal. Contudo, em muitos casos, eles contêm um número significativo de parâmetros livres cujos valores não podem ser estimados através da teoria, mas precisam ser estimados a partir de dados. Isso é particularmente importante para modelos de epidemiologia computacional, economia, sistemas dinâmicos ou até mesmo caracterização de materiais. No entanto, a maioria dos métodos atuais de estimativa de parâmetros é complexa e demorada para ser executada. Este trabalho apresenta um método computacional simplificado e rápido para obter estimativas precisas para os parâmetros do modelo, utilizando equações diferenciais neurais. Apresentamos assim um *pipeline* composto por um modelo de um oscilador amortecido que atua como gerador de dados de séries temporais sintéticas (com ruído) e também como *solver* direto (sem ruído), além de uma rede neural que extrai parâmetros dos dados gerados pelo modelo. Os dois métodos combinados resultam em uma ferramenta poderosa que permite estimar de forma rápida e precisa os parâmetros do modelo mesmo com dados ruidosos.

Palavras-chave: Equações diferenciais neurais; Calibração de modelos; Estimativa de parâmetros; Matemática Aplicada à Engenharia.

1. INTRODUÇÃO

Vivemos em uma era de complexidade crescente. Com uma grande variedade de recursos computacionais modernos e potentes, combinados com uma vastidão de bancos de dados, os pesquisadores podem, cada vez mais, enxergar os problemas do nosso cotidiano como complexos e dinâmicos. Uma área de grande interesse na engenharia é a calibração de parâmetros de modelos matemáticos utilizando dados, criando, dessa forma, ferramentas preditivas que são computacionalmente eficientes e possuem fundamentação teórica. Os modelos computacionais têm sido usados para compreender como ocorre o contágio de doenças (Cunha Jr et al., 2023), a dinâmica dos pedestres (Helbing e Molnár, 1995), as oscilações sincronizadas (Kuramoto, 1975), a dinâmica das redes sociais (Bak-Coleman et al., 2022), a dinâmica

populacional (Winkelmann et al., 2021) e a dinâmica dos sistemas econômicos (Elam et al., 2018).

Há um grande número de métodos que permitem estimar parâmetros a partir de dados, tais como os estimadores de máxima verossimilhança (Chen, 2021), métodos de Monte Carlo via Cadeia de Markov com um paradigma Bayesiano (Raqueti et al., 2022), o princípio de maximização da entropia (Timme e Casadiego, 2014) e métodos de aprendizado baseados em regressão (Paixão et al., 2020). Recentemente, um método promissor surgiu nas redes neurais artificiais. Embora as redes neurais sejam amplamente utilizadas como ferramentas poderosas para o reconhecimento de padrões e modelos preditivos (Wei et al., 2022), elas apresentam capacidade computacional em serem aplicadas em áreas matemáticas que antes eram dominadas por métodos mais clássicos, como a calibração de parâmetros.

Este trabalho tem como objetivo contribuir para a compreensão mais aprofundada das possibilidades das redes neurais como ferramentas de calibração de modelos matemáticos que representam sistemas mecânicos. Apresentamos e analisamos assim um esquema computacional simplificado para obter uma estimativa dos parâmetros do modelo de um oscilador amortecido a partir de dados de séries temporais sintéticas com adição de ruído.

2. METODOLOGIA

O escopo de nossa abordagem abrange modelos formulados como equações diferenciais acopladas do tipo

$$d\varphi = f(\varphi; \mathbf{x}, t, \boldsymbol{\lambda}) dt, \quad (1)$$

onde $\varphi \in \mathbb{R}^N$ é o vetor de estados, $\mathbf{x}$ é uma variável espacial, $\boldsymbol{\lambda} := (\lambda_1, \dots, \lambda_p) \in \mathbb{R}^p$ é um conjunto de parâmetros escalares. A dinâmica de sistemas que são governados por equações como a acima podem depender diretamente da escolha dos parâmetros $\boldsymbol{\lambda}$: em modelos matemáticos mais complexos, as estimativas teóricas para esses parâmetros são, frequentemente, impossibilitadas, pois tais parâmetros são complexos de serem medidos e, portanto, devem ser extraídos dos dados. Em uma ODE neural, os parâmetros escalares são as saídas da rede neural, cujos parâmetros internos devem ser aprendidos através de dados.

2.1. Estimando parâmetros de equações neurais

Apresentamos um método para estimar parâmetros de ODEs através do treinamento de uma rede neural para encontrar um conjunto de parâmetros $\hat{\boldsymbol{\lambda}}$ que, quando inseridos no modelo da Equação 1, reproduzem uma série temporal forne-

cida $\mathbf{T} = (\varphi_1, \dots, \varphi_L)$. Uma rede neural é uma função $u_\theta : \mathbb{R}^{N \times q} \rightarrow \mathbb{R}^p$, onde $q \geq 1$ representa o número de etapas da série temporal que são fornecidas como entrada.

As saídas são os parâmetros estimados $\hat{\lambda}$, que são usados para rodar um *solver* numérico através de B iterações (onde B é o *batch size*) para obter uma série temporal estimada $\hat{\mathbf{T}}(\hat{\lambda}) = (\hat{\varphi}_k, \dots, \hat{\varphi}_{k+B})$. Isso, por sua vez, é usado para treinar os parâmetros internos θ da rede neural (os pesos e os *biases*) através de uma função de perda $J(\hat{\mathbf{T}}, \mathbf{T})$. Dado que $\hat{\lambda} = \hat{\lambda}(\theta)$, pode-se calcular o gradiente $\nabla_\theta J$ para otimizar os parâmetros internos da rede neural através de um método de *backpropagation* via otimizador Adam, por exemplo. O cálculo de $\nabla_\theta J$ requer a diferenciação da série temporal prevista $\hat{\mathbf{T}}$, e também do sistema da Equação 1, em termos de $\hat{\lambda}$; ou seja, a função de perda tem conhecimento da dinâmica do modelo. Finalmente, os dados gerados são inseridos na rede neural para a elaboração de uma nova estimativa do parâmetro $\hat{\lambda}$, e o ciclo é reiniciado.

Uma única passagem por todo o conjunto de dados é denominada *epoch*. O conjunto de dados é processado em lotes, o que significa que a perda é calculada ao longo de B passos da série temporal antes que os pesos sejam atualizados. Se $B = L$, o treinamento é equivalente à *batch gradient descent*; se $B = 1$, o treinamento é equivalente à *stochastic gradient descent*. As complexidades do procedimento de diferenciação são tratadas pelo esquema de autodiferenciação (que trata vetores aleatórios como constantes) e que é bem fundamentado em diversas bibliotecas de *machine learning*. É muito importante notar que a rede neural não está ajustando um conjunto de dados no sentido tradicional, mas sim produzindo um conjunto de parâmetros que, quando inseridos nas equações governantes, geram o conjunto de dados (ou uma estimativa do mesmo). A seguir é apresentado o pseudocódigo desta metodologia descrita:

Algorithm 1 Processo de treinamento da rede neural (uma *epoch*)

Input: $\mathbf{T} = (\varphi_1, \dots, \varphi_L)$ (série temporal de tamanho L) e $B \leq L$ (*batch size*)

- 1: **for** $t \in \{1, \dots, L - B\}$ **do**
- 2: $\hat{\lambda} \leftarrow u_\theta(\varphi_t, \dots, \varphi_{t+q})$ (estimando parâmetros, q elementos da série temporal)
- 3: $\hat{\varphi}_0 \leftarrow \varphi_t$
- 4: **for** $b \in \{1, \dots, B\}$ **do**
- 5: $\hat{\varphi}_b \leftarrow \int f(\hat{\varphi}_{b-1}; \hat{\lambda}) dt$ (série temporal prevista, *solver* numérico)
- 6: **end for**
- 7: Calcule a perda $J(\hat{\varphi}_1, \dots, \hat{\varphi}_B; \varphi_{t+1}, \dots, \varphi_{t+B})$
- 8: Calcule o gradiente $\nabla_\theta J$
- 9: Atualize os parâmetros θ da rede neural utilizando *backpropagation*
- 10: **end for**

3. RESULTADOS E DISCUSSÕES

A Figura 1 apresenta o modelo simples de um oscilador amortecido cujos parâmetros de amortecimento e rigidez serão obtidos via calibração neural. Objetiva-se utilizar os conhecimentos aqui adquiridos para estudar sistemas dinâmicos mais complexos com múltiplos graus de liberdade.

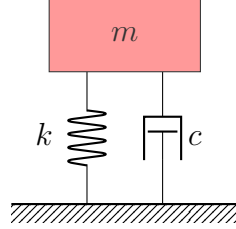


Figura 1. Modelo de oscilador amortecido utilizado para calibração neural de parâmetros.

Explicando a metodologia de forma sucinta: primeiramente foram gerados dados sintéticos de vibração deste sistema utilizando o método de Euler explícito, com adição de ruído gerado por uma distribuição normal do tipo $\mathcal{N}(\mu = 0, \sigma^2 = 1)$. Os parâmetros utilizados para tal foram: $m = 8,0$ kg, $c = 0,3$ Ns/m, $k = 5,0$ N/m e as condições iniciais foram $x_0 = 1,0$ m e $\dot{x}_0 = 0,0$ m/s. A Figura 2 abaixo apresenta respectivamente o deslocamento, velocidade e aceleração do sistema obtidos (as linhas acinzentadas são referentes ao sistema ideal, i.e. sem ruído).

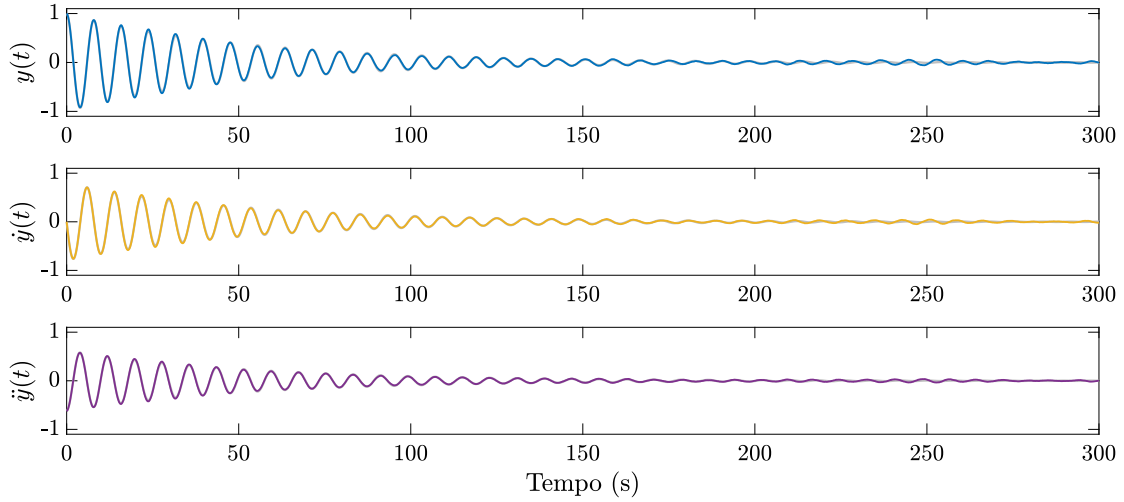


Figura 2. Dados sintéticos de deslocamento, velocidade e aceleração do modelo.

Posteriormente foi criada uma arquitetura de rede neural que recebe três entradas, referentes a cada uma das séries temporais sintéticas de deslocamento, velocidade e aceleração $\mathbf{T} = \{\mathbf{y}(t), \dot{\mathbf{y}}(t), \ddot{\mathbf{y}}(t)\}$. Neste trabalho foram fornecidas as séries temporais completas para o treinamento da rede, i.e. o *batch size* é igual ao tamanho da série temporal ($B = L$); foi fornecida também a massa do sistema ($m = 8,0$ kg), visto que é um parâmetro fácil de ser mensurado em sistemas reais.

A rede neural é então inicializada e treinada e apresenta como saída um vetor de parâmetros estimados $\hat{\lambda} = \{c, k\}$. Estes parâmetros são então utilizados em um outro *solver* numérico ideal (sem ruído) via método de Euler explícito, produzindo assim séries temporais previstas $\hat{T} = \{\hat{y}(t), \hat{\dot{y}}(t), \hat{\ddot{y}}(t)\}$. A rede neural então cria um funcional de perda $J(\hat{T}, T)$ (erro quadrático médio, *Loss MSE*) e inicia um processo de minimização de seus parâmetros internos θ , como já explicado na seção de Metodologia. A Figura 3 apresenta o gráfico referente a perda durante o processo de treinamento e a Figura 4 apresenta o processo de calibração neural dos parâmetros do oscilador amortecido.

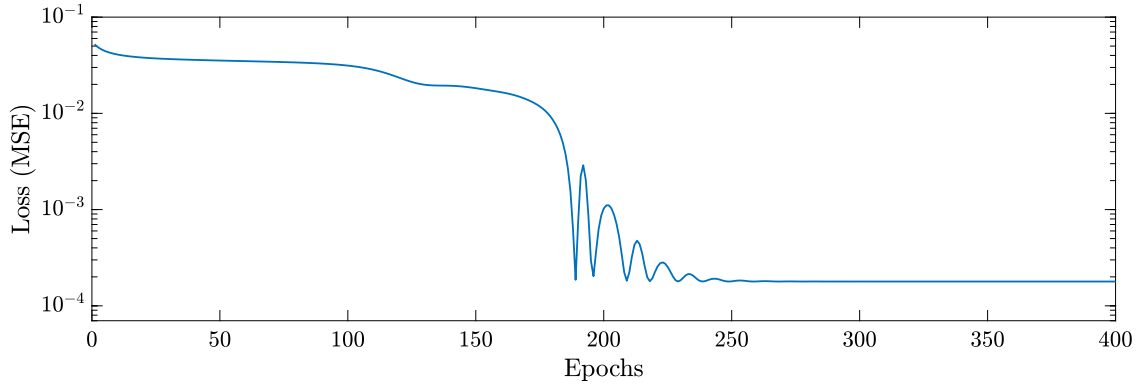


Figura 3. *Loss MSE* em função das *epochs* durante o treinamento do calibrador neural.

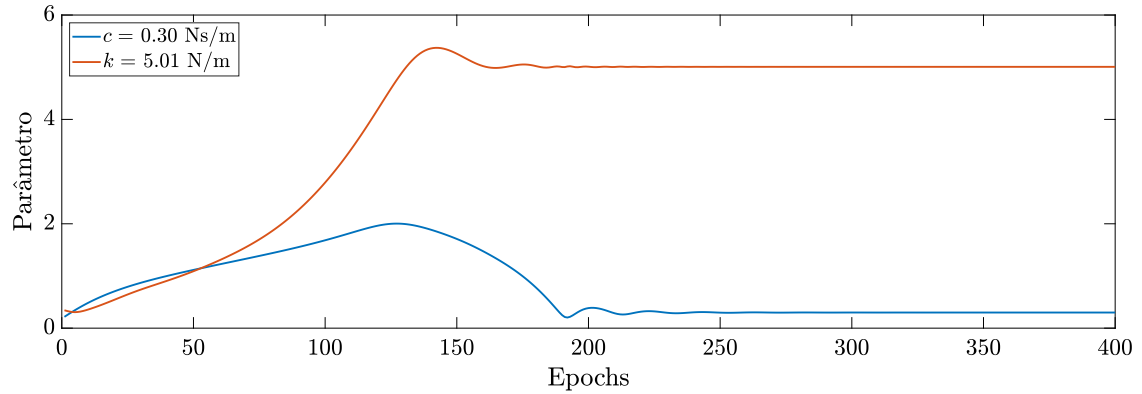


Figura 4. Processo de calibração neural dos parâmetros do oscilador amortecido.

Pode-se observar uma estimativa de parâmetros precisa ($c = 0,30$ Ns/m e $k = 5,01$ N/m), mesmo com a utilização de dados ruidosos. As próximas etapas envolvem a evolução deste calibrador neural para que ao invés de um valor determinístico seja gerada uma densidade marginal dos parâmetros, enriquecendo a calibração neural de forma estocástica. Isto será imprescindível para a calibração de sistemas mais complexos, com mais graus de liberdade.

4. CONCLUSÕES

Em suma, este trabalho apresentou uma abordagem promissora para a calibração neural de parâmetros em modelos de sistemas mecânicos, utilizando um oscilador amortecido como estudo de caso. A metodologia proposta combinou equações diferenciais neurais com dados sintéticos, permitindo a obtenção de estimativas precisas para os parâmetros do modelo, mesmo em presença de ruído nos dados. Os resultados obtidos demonstraram a eficácia do método, destacando sua capacidade de lidar com sistemas complexos e a possibilidade de expansão para problemas com múltiplos graus de liberdade. Futuros desenvolvimentos incluirão a incorporação de abordagens estocásticas para enriquecer a calibração neural, proporcionando uma análise mais completa e robusta dos parâmetros do sistema.

AGRADECIMENTOS

Os autores gostariam de agradecer a Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) pelo apoio ao processo No 2022/16156-9. Os autores também agradecem o apoio financeiro da Asociación de Universidades Grupo Montevideo (AUGM) por possibilitar o intercâmbio e a colaboração científica entre os grupos da Universidade Estadual Paulista (FEIS/UNESP, Brasil) e Universidad Nacional De Cuyo (IMERIS/UNCuyo, Argentina).

REFERÊNCIAS

- Joseph B. Bak-Coleman, Ian Kennedy, Morgan Wack, Andrew Beers, Joseph S. Schafer, Emma S. Spiro, Kate Starbird, e Jevin D. West. Combining interventions to reduce the spread of viral misinformation. *Nature Human Behaviour*, 6(10): 1372–1380, June 2022. ISSN 2397-3374. doi: 10.1038/s41562-022-01388-6. URL <http://dx.doi.org/10.1038/s41562-022-01388-6>.
- Xiaohui Chen. Maximum likelihood estimation of potential energy in interacting particle systems from single-trajectory data. *Electronic Communications in Probability*, 26(none), January 2021. ISSN 1083-589X. doi: 10.1214/21-ecp416. URL <http://dx.doi.org/10.1214/21-ECP416>.
- Americo Cunha Jr, David A. W. Barton, e Thiago G. Ritto. Uncertainty quantification in mechanistic epidemic models via cross-entropy approximate bayesian computation. *Nonlinear Dynamics*, 111(10):9649–9679, February 2023. ISSN 1573-269X. doi: 10.1007/s11071-023-08327-8. URL <http://dx.doi.org/10.1007/s11071-023-08327-8>.
- L. Ellam, M. Girolami, G. A. Pavliotis, e A. Wilson. Stochastic modelling of urban structure. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 474(2213):20170700, May 2018. ISSN 1471-2946. doi: 10.1098/rspa.2017.0700. URL <http://dx.doi.org/10.1098/rspa.2017.0700>.

- Dirk Helbing e Péter Molnár. Social force model for pedestrian dynamics. *Physical Review E*, 51(5):4282–4286, May 1995. ISSN 1095-3787. doi: 10.1103/physreve.51.4282. URL <http://dx.doi.org/10.1103/PhysRevE.51.4282>.
- Yoshiki Kuramoto. *Self-entrainment of a population of coupled non-linear oscillators*, page 420–422. Springer-Verlag, 1975. ISBN 3540071741. doi: 10.1007/bfb0013365. URL <http://dx.doi.org/10.1007/BFb0013365>.
- Jessé Paixão, Samuel da Silva, Eloi Figueiredo, Lucian Radu, e Gyuhae Park. De-lamination area quantification in composite structures using gaussian process regression and auto-regressive models. *Journal of Vibration and Control*, 27(23–24): 2778–2792, October 2020. ISSN 1741-2986. doi: 10.1177/1077546320966183. URL <http://dx.doi.org/10.1177/1077546320966183>.
- Rafael da S Raqueti, Rafael de O Teloli, Samuel da Silva, Philippe Bussetta, e Americo Cunha. On the use of stochastic bouc–wen model for simulating vis-coelastic internal variables from a finite element approximation of steady-rolling tire. *Journal of Vibration and Control*, 29(19–20):4799–4813, September 2022. ISSN 1741-2986. doi: 10.1177/10775463221125038. URL <http://dx.doi.org/10.1177/10775463221125038>.
- Marc Timme e Jose Casadiego. Revealing networks from dynamics: an introduction. *Journal of Physics A: Mathematical and Theoretical*, 47(34):343001, August 2014. ISSN 1751-8121. doi: 10.1088/1751-8113/47/34/343001. URL <http://dx.doi.org/10.1088/1751-8113/47/34/343001>.
- Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, e William Fedus. Emergent abilities of large language models, 2022. URL <https://arxiv.org/abs/2206.07682>.
- Stefanie Winkelmann, Johannes Zonker, Christof Schütte, e Nataša Djurdjevac Conrad. Mathematical modeling of spatio-temporal population dynamics and application to epidemic spreading. *Mathematical Biosciences*, 336:108619, June 2021. ISSN 0025-5564. doi: 10.1016/j.mbs.2021.108619. URL <http://dx.doi.org/10.1016/j.mbs.2021.108619>.