The Journal of Open Source Software

Cocofest: an Open-Source Python Package for Functional Electrical Stimulation Optimization in Optimal Control

Kevin Co [1], **Pierre Puchaud** [1,2], **Florent Moissenet** [1,3,4], **and Mickaël Begon** [1]

1 Laboratoire de Simulation et Modélisation du Mouvement, Université de Montréal, Montréal, Québec, Canada **2** Auctus, Inria, Centre de l'Université de Bordeaux, Talence, France **3** Biomechanics Laboratory, Geneva University Hospitals and University of Geneva, Geneva, Switzerland **4** Kinesiology Laboratory, Geneva University Hospitals and University of Geneva, Geneva, Switzerland

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor:

Submitted: 06 September 2025
Published: unpublished

Summary

Functional electrical stimulation (FES) is a rehabilitation method intended to promote motor recovery notably after neurological impairment. Applying coordinated electrical pulses to muscles elicits functional movements like walking, reaching, and grasping. FES rehabilitation mostly relies on empirical settings, as responses to stimulation vary across populations and muscles. Empirical settings often cause overstimulation and premature fatigue (Ibitoye et al., 2016), shortening rehabilitation sessions and diminishing therapeutic benefit. Consequently, advanced control approaches like optimal control-driven FES are gaining interest in personalizing and improving FES rehabilitation efficiency, meanwhile delaying muscle fatigue (Co et al., 2025). To address this need, we designed `Cocofest` (Custom Optimal COntrol for Functional Electrical STimulation), an open-source Python package for optimal control-driven FES. `Cocofest` provides a framework to generate personalized pulse trains (Fig. 1) based on nonlinear dynamics models for FES (Table. 1), for several musculoskeletal models and motor tasks. The package includes over 10 examples, covering optimization of FES-related pulse train parameters (including frequency, pulse width, pulse intensity), FES model parameters identification from in-vivo measurements, and long duration predictive simulations.

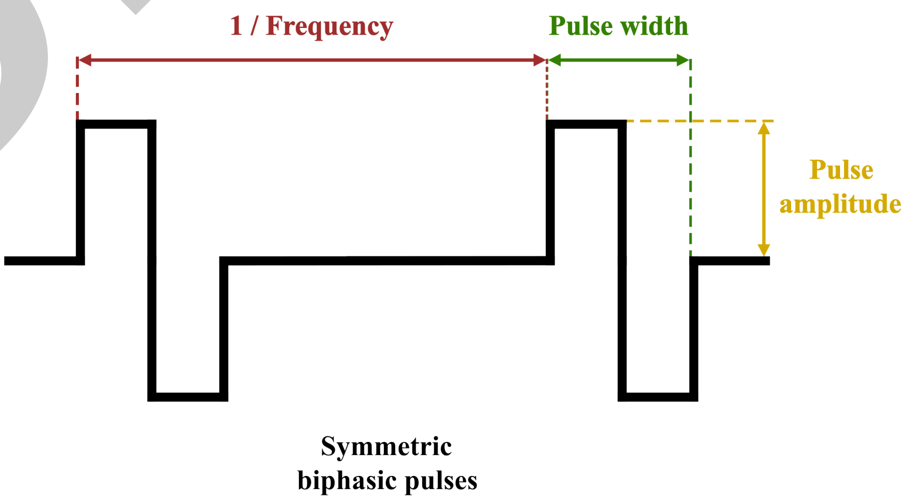

Figure 1: Pulse train parameters that can be optimized in Cocofest

Statement of Need

Since the pioneer study on optimal control-driven FES (Hunt et al., 1997), no code has been shared in the field, limiting objective comparison and replicability across studies. The lack of open-source practice led to an absence of consensus on how to choose nonlinear dynamics for FES, and which cost functions to use for dedicated clinical needs, hindering standardization and cumulative progress (Co et al., 2025). To address these challenges and support collective scientific progress, `Cocofest` fulfills the following four needs:

Firstly, the relationship between the pulse train parameters (e.g., frequency, pulse width and intensity; Fig. 1) and the resulting muscle force, joint torque, and muscle fatigue can be modeled with different nonlinear dynamics (Ding et al., 2003; Veltink et al., 1992). Gathering them within a unified package would facilitate comparison for more informed modelling choices.

Secondly, no study has compared different optimal control problem (OCP) formulations applied to FES, due to OCP implementation challenges (Co et al., 2025). Easily customizable OCP formulation, involving objective functions, models, and transcriptions is required to provide an adequate research framework. Having the possibility to switch between various OCP transcriptions (e.g., direct collocation or direct multiple shooting) is essential when dealing with stiff differential equations (Puchaud et al., 2023), often embedded in FES models. Muscle fatigue is the primary challenge in FES. Enabling the development and comparison of different OCP formulations could help address research questions, yield novel stimulation patterns and enhance fatigue reduction. Moreover, using receding-horizon estimation for longer simulations reduces the computational complexity associated with time-varying dynamics (e.g., fatigue) (Ding et al., 2003).

Thirdly, predictive simulations of FES-driven or FES-assisted motions (e.g., walking, cycling, reaching, and grasping) require the coupling of FES models with the equations of motion as well as adequate muscle force-length-velocity relationships. Predictive simulations are usually actuated through Hill-type muscle models (Wakeling et al., 2023). A package capable of replacing muscle actuation by FES models in multibody musculoskeletal models will allow us to simulate realistic FES-driven tasks.

Fourthly, personalized rehabilitation strategy is required to facilitate the motor recovery. Therefore, identifying the patient-specific muscle response to FES is a crucial step. Unfortunately, current complex identification methods are a barrier to clinical translation (Le et al., 2010). Providing a robust and customizable framework for the development of more patient-friendly protocols would help to overcome this barrier.

Despite its potential, optimal control–driven FES remains unadopted in clinical practice due to its low technology readiness level (Co et al., 2025). `Cocofest` is a comprehensive package designed to bridge the gaps and foster clinical adoption. It integrates nonlinear muscle dynamics dedicated to FES, manages muscle fatigue, interfaces FES with musculoskeletal models, supports customizable cost functions and parameter identification routines. With the goal of bringing this technology to patient care, we believe this package will contribute to the open-science effort. `Cocofest` is expected to accelerate the increase of technology readiness level by strengthening knowledge foundation.

State of the Field

Several open-source toolkits support optimal control computations for musculoskeletal biomechanics, such as: `OpenSim Moco` (Dembia et al., 2020), a C++ OpenSim extension that enables motion tracking and prediction using efficient direct-collocation formulations coupled to nonlinear programming solvers. `SCONE` (Geijtenbeek, 2019), a C++/C predictive-simulation environment for human and animal motion that optimizes neuromusculoskeletal controllers to achieve task-level objectives (e.g., stable walking at a target speed). `Bioptim` (Michaud et al., 2022), a Python optimal-control framework for biomechanics that supports both direct collocation and multiple shooting, with flexible interfaces to nonlinear programming solvers.

However, these toolkits are not tailored for FES. They control muscle activation as a piecewise linear/constant excitation, whereas FES requires optimizing deliverable stimulation patterns under device and safety constraints. As a result, they lack reusable, validated components for the stimulation-to-force pathway and fatigue/recovery dynamics, limiting reproducible comparison of FES models and slowing translation to practical stimulation design. `Cocofest` addresses this gap by implementing published FES models that can drive musculoskeletal models. This design supports reproducible comparisons of FES modeling assumptions and accelerates prototyping of patient- and task-specific stimulation optimization. `Cocofest` also includes utilities for model identification and receding-horizon optimization to support FES research workflows.

Software Design

`Cocofest` is a Python library that relies on Biorbd, a musculoskeletal physics engine (Michaud & Begon, 2021), and Bioptim, an open-source optimization framework for biomechanical problems (Michaud et al., 2022). Specifically, Bioptim enables easy OCP customization including cost functions, bounds, constraints, transcription methods (e.g., direct collocation), integration methods, and solving methods (e.g., full- and receding-horizon OCPs).

In conventional Hill-type muscle model, muscle force (F_m) is the product of a the muscle activation, F_{max} the maximal isometric muscle force, f_l the force-length, f_v the force-velocity and f_{pas} the passive force-length relationship: $F_m(t) = a(t)\, F_{\max}\, f_l(\tilde{l}_m)\, f_v(\tilde{v}_m) + f_{pas}(\tilde{l}_m)$. `Cocofest` replaces $a(t) \times F_{max}$ by the force obtained using FES models. This approach allows motions driven-FES simulations, meanwhile benefiting from musculoskeletal model properties (e.g., muscle insertion, inertial parameters).

`Cocofest` was developed to maintain a consistent structure between classes and functions to facilitate the OCP customization and new FES model implementation. This shared interface promotes reproducible work and comparisons of optimal control–driven FES strategies.

Research Impact Statement

`Cocofest` was developed to address several gaps in the literature, including the lack of systematic comparisons of FES models and OCP formulations, accessible tools for FES model identification, and open-source software for reproducible research. It enables researchers to generate personalized stimulation patterns, compare alternative OCP formulations, and simulate realistic FES-driven tasks. By providing a consistent software structure and clear documentation, `Cocofest` aims to streamline research workflows and support translation toward FES rehabilitation applications. Although the project is new and targets a niche domain, it already offers a shared, reproducible environment that can foster discussion, collaboration, and broader adoption of open-source practices within the FES community, which is an important step toward clinical translation of this technique (Co et al., 2025).

AI Usage Disclosure

The authors used ChatGPT only to improve the manuscript clarity and readability. After using this tool/service, the authors reviewed and edited the content as needed and took full responsibility for the content of the publication.

GitHub Copilot and ChatGPT were used to assist in code refactoring and documentation. Authors made all the core design and architectural decisions.

Acknowledgements

The package development was supported by the Fonds de recherche du Québec – Nature et technologies (FRQNT, Grant 341023) and by the FRQ strategic group in Ingénierie de technologies interactives en réadaptation (INTER #160 OptiStim).

References

Co, K., Begon, M., Bailly, F., & Moissenet, F. (2025). Optimal control-driven functional electrical stimulation: A PRISMA-ScR scoping review. *Computers in Biology and Medicine*, *199*, 111311. https://doi.org/https://doi.org/10.1016/j.compbiomed.2025.111311

Dembia, C. L., Bianco, N. A., Falisse, A., Hicks, J. L., & Delp, S. L. (2020). OpenSim moco: Musculoskeletal optimal control. *PLOS Computational Biology*, *16*(12), e1008493. https://doi.org/10.1371/journal.pcbi.1008493

Ding, J., Wexler, A. S., & Binder-Macleod, S. A. (2003). Mathematical models for fatigue minimization during functional electrical stimulation. *Journal of Electromyography and Kinesiology*, *13*(6), 575–588. https://doi.org/10.1016/S1050-6411(03)00102-0

Geijtenbeek, T. (2019). SCONE: Open source software for predictive simulation of biological motion. *Journal of Open Source Software*, *4*(38), 1421. https://doi.org/10.21105/joss.01421

Hunt, K. J., Munih, M., & Donaldson, N. de N. (1997). Feedback control of unsupported standing in paraplegia. I. Optimal control approach. *IEEE Transactions on Rehabilitation Engineering*, *5*(4), 331–340. https://doi.org/10.1109/86.650287

Ibitoye, M. O., Hamzaid, N. A., Hasnan, N., Abdul Wahab, A. K., & Davis, G. M. (2016). Strategies for rapid muscle fatigue reduction during FES exercise in individuals with spinal cord injury: A systematic review. *PloS One*, *11*(2), e0149024. https://doi.org/10.1371/journal.pone.0149024

Le, F., Markovsky, I., Freeman, C. T., & Rogers, E. (2010). Identification of electrically stimulated muscle models of stroke patients. *Control Engineering Practice*, *18*(4), 396–407. https://doi.org/10.1016/j.conengprac.2009.12.007

Michaud, B., Bailly, F., Charbonneau, E., Ceglia, A., Sanchez, L., & Begon, M. (2022). Bioptim, a python framework for musculoskeletal optimal control in biomechanics. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *53*(1), 321–332. https://doi.org/10.1109/TSMC.2022.3183831

Michaud, B., & Begon, M. (2021). Biorbd: A c++, python and matlab library to analyze and simulate the human body biomechanics. *Journal of Open Source Software*, *6*(57), 2562. https://doi.org/10.21105/joss.02562

Puchaud, P., Bailly, F., & Begon, M. (2023). Direct multiple shooting and direct collocation perform similarly in biomechanical predictive simulations. *Computer Methods in Applied Mechanics and Engineering*, *414*, 116162. https://doi.org/10.1016/j.cma.2023.116162

Veltink, P. H., Chizeck, H. J., Crago, P. E., & El-Bialy, A. (1992). Nonlinear joint angle control for artificially stimulated muscle. *IEEE Transactions on Biomedical Engineering*, *39*(4), 368–380. https://doi.org/10.1109/10.126609

Wakeling, J. M., Febrer-Nafría, M., & De Groote, F. (2023). A review of the efforts to develop muscle and musculoskeletal models for biomechanics in the last 50 years. *Journal of Biomechanics*, *155*, 111657. https://doi.org/10.1016/j.jbiomech.2023.111657