| | |
|---|---|
| Title | Multi-Objective Optimization based 3D Walking of a Neuromuscular Driven Salamander Model in Simulation |
| Author(s) | Arreguit, Jonathan; Ramalingasetty, Tata Shravan; Gupta, Astha et al. |
| Citation | |
| Version Type | VoR |
| URL | https://doi.org/10.18910/84869 |
| rights | |
| Note | |

# Multi-Objective Optimization based 3D Walking of a Neuromuscular Driven Salamander Model in Simulation

Jonathan Arreguit[a*], Shravan Tata Ramalingasetty[a], Astha Gupta[a], Auke Ijspeert[a]
[a]Biorobotics Laboratory, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
* *jonathan.arreguitoneill@epfl.ch*

## 1 Introduction

Robotics experiments and physics simulations are important scientific tools to study animal locomotion. They can complement animal studies by offering a platform for studying the neural architectures and replicating the locomotory behaviours [1]. While robotic experiments can serve as methods for testing neural models in real world physics, neuromechanical simulations are beneficial when testing models difficult to setup in hardware. Moreover, simulations can be coupled with evolutionary algorithms to tune the parameters of neural and muscular models which can be difficult to measure from the animals directly. This approach has been employed in a large number of works including studies of lampreys [2], salamanders [3, 4], ants [5] or humans [6]. More precisely, [2–4] use Genetic Algorithms, [5] uses Covariance Matrix Adaptation Evolution Strategy and [6] uses Particle Swarm Optimization, all of which optimize a single cost function. In this work, we present how multi-objective optimization can lead to a set of diverse results given a salamander model with a fixed neural architecture and a simplified muscle model when optimized against conflicting objectives.

## 2 Methods

### 2.1 Salamander model

The simulated salamander model is 20 [cm] long and weighs 20 [g]. It contains 11 body joints allowing the body to bend laterally and 4 Degrees of Freedom (DoF) for each limb, including 3 DoF at the interface with the body, and one DoF at the elbow/knee. Each of these DoF is modelled as a revolute joint controlled by a flexion-extension muscle pair governed by the following Eq. (1):

$$\tau_i = \underbrace{\alpha_i(M_{F,i} - M_{E,i}) + \beta_i(M_{F,i} + M_{E,i})\Delta\varphi_i}_{\tau_{active,i}} + \underbrace{\beta_i\gamma_i\Delta\varphi_i + \delta_i\dot{\varphi}_i}_{\tau_{passive,i}}, \quad (1)$$

where, similarly to [2–4], $\alpha_i$ represents the active gain, $\beta_i$ the stiffness gain, $\gamma_i$ the intrinsic stiffness, $\delta_i$ the damping coefficient, $M_{F,i}$ and $M_{E,i}$ the neural activities for the flexor and extensor, $\Delta\varphi_i$ the position and $\dot{\varphi}_i$ the velocity for each joint $i$. In particular, we differentiate between the active torque $\tau_{active,i}$ and passive torque $\tau_{passive,i}$. We use the same network equations originally implemented in [7], which is based on a network of weakly-coupled oscillators. The major difference is that our model includes more DoF for each limb, which we adapt by using a pair of oscillators for each joint and connecting them accordingly.

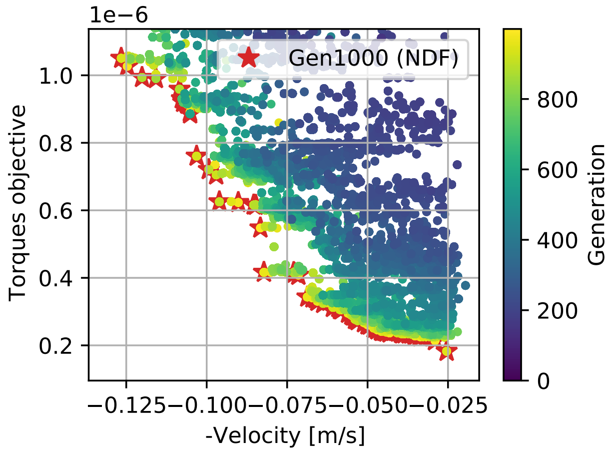


**Figure 1:** Convergence of the evolution along the two objective functions over the generations. Each dot represents a feasible solution. The solutions characterised by the red stars show the non-dominated front obtained in the final $1000th$ generation.

### 2.2 Optimization

In nature, the ability to locomote serves many purposes. Salamanders need to move in their environment for different tasks including hunting prey, escaping predators or finding mates. Essentially, locomotion can be roughly reduced to moving fairly quickly from one point to another while minimising energy consumption, which are two conflicting objectives. In the case of this work, we optimize a salamander model to walk along a straight path, formulated in Eqs. (2)-(6):

$$\min_{\mathbf{p}} \quad -\|\mathbf{x_h}^f - \mathbf{x_h}^i\| \qquad \text{(Distance objective)} \quad (2)$$

$$\min_{\mathbf{p}} \quad \textstyle\sum_{i=0}^{N} \tau_{i,active}^2 \qquad \text{(Torques objective)} \quad (3)$$

$$\text{s.t.} \quad \mathbf{x_{min}} \leq \mathbf{x_h} \leq \mathbf{x_{max}} \qquad \text{(Position boundaries)} \quad (4)$$

$$\min_{j=0,1,2,3}(\|\mathbf{s_j}\|) > 0 \qquad \text{(Contacts handling)} \quad (5)$$

$$\|\dot{\mathbf{x}}_\mathbf{h}\| < \mathrm{V}_{\mathrm{max}} \qquad \text{(Maximum velocity)} \quad (6)$$

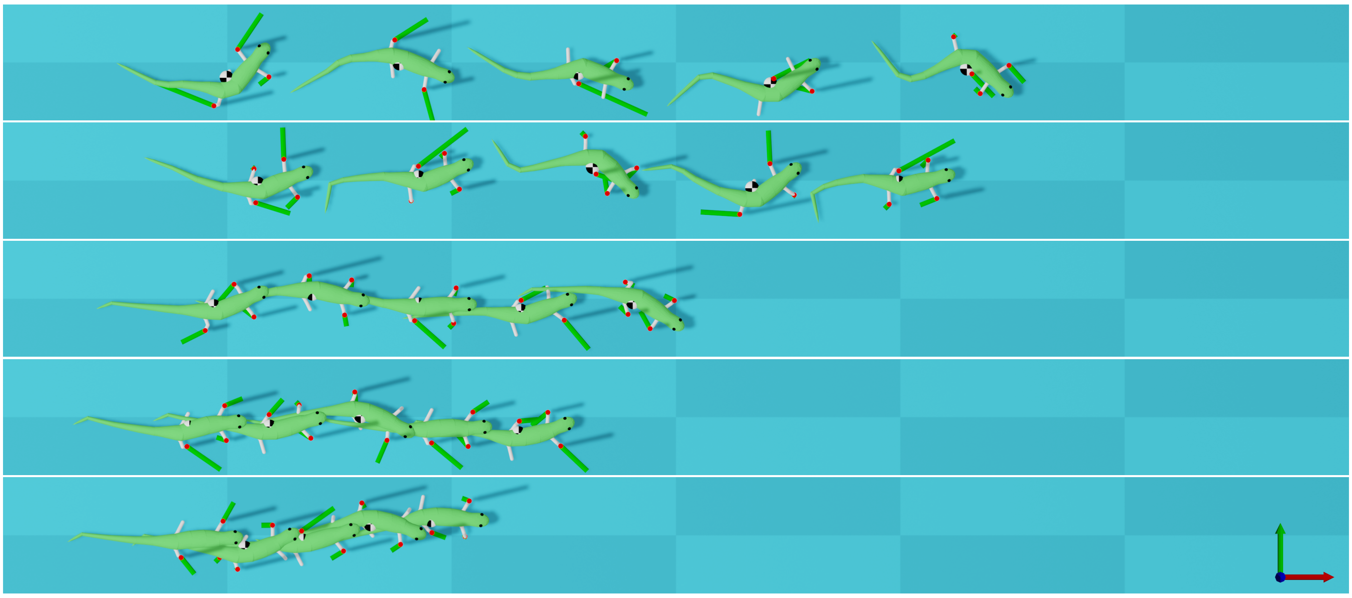

**Figure 2:** Illustration of different walking solutions obtained from the non-dominated front of the final generation. The snapshots shown for each solution correspond to 2 [s], 4 [s], 6 [s], 8 [s] and 10 [s] of the simulation, The images represent the solutions starting from the fastest at the top to the least energy consuming at the bottom. The floor is tiled with squares of 0.25 [m] side lengths.

Eqs. (2) and (3) denote the distance and torque objectives respectively. The variable $\mathbf{p}$ is the decision vector corresponding to the vector of parameters to be optimized and ${\mathbf{x_h}}^i$ and ${\mathbf{x_h}}^f$ represent the position of the head at the initial and final iteration of the simulation. Eqs. (4) to (6) represent the set of contraints used in this optimization. Eq. (4) forces the model to move within chosen boundaries $\mathbf{x_{min}}$ and $\mathbf{x_{max}}$, while Eqs. (5) and (6) are added to avoid solutions which break the numerical simulation, with $\mathbf{s_j}$ corresponding to the contacts forces for each foot $j \in 0,\ldots,3$ and $\mathrm{V_{max}}$ to the maximum velocity threshold for the model. The simulation was implemented using the Pybullet environment [8]. The optimization problem was solved using the Non-dominated Sorting Genetic Algorithm III [9] implemented in Pymoo [10], with a population size of 120 optimized over the course of 1000 generations.

## 3 Discussion

The progress of the evolution is shown in Figure 1, featuring the formation of a Pareto front as expected given the conflicting objectives. In particular, the non-linear relationship between the two objectives can be observed, where each solution from this front represents the least energy consuming for its given velocity. A sub-sample of solutions from this front are illustrated in Figure 2. We observe that faster solutions tend to make use of larger body and limb amplitudes, while slower solutions tend to repress lateral body bending and primarily use limbs. Due to Eq. 3, the solutions obtained tend to harness the passive dynamics of the muscle model in order to minimise the sum of active torques. In future work, we aim to use additional objectives to optimize over different environments.


## 4 Acknowledgements

Arreguit's work has been supported by the Human Frontier Science Program (HFSP-RGP0027/2017) and Ramalingasetty's work has been supported by the by the Human Brain Project, EU and by the National Institutes of Health (grant no. R01NS090919). The authors declare no conflict of interest.



**References**

[1] Auke J Ijspeert. Biorobotics: Using robots to emulate and investigate agile locomotion. *science*, 346(6206):196–203, 2014.

[2] Auke Jan Ijspeert, John Hallam, and David Willshaw. Evolving swimming controllers for a simulated lamprey with inspiration from neurobiology. *Adaptive behavior*, 7(2):151–172, 1999.

[3] Auke Jan Ijspeert. A connectionist central pattern generator for the aquatic and terrestrial gaits of a simulated salamander. *Biological cybernetics*, 84(5):331–348, 2001.

[4] Nalin Harischandra, Jean-Marie Cabelguen, and Örjan Ekeberg. A 3d musculo-mechanical model of the salamander for the study of different gaits and modes of locomotion. *Frontiers in neurorobotics*, 4:112, 2010.

[5] Shihui Guo, Juncong Lin, Toni Wöhrl, and Minghong Liao. A neuro-musculo-skeletal model for insects with data-driven optimization. *Scientific reports*, 8(1):1–11, 2018.

[6] Florin Dzeladini, Jesse Van Den Kieboom, and Auke Ijspeert. The contribution of a central pattern generator in a reflex-based neuromuscular model. *Frontiers in human neuroscience*, 8:371, 2014.

[7] Auke Jan Ijspeert, Alessandro Crespi, Dimitri Ryczko, and Jean-Marie Cabelguen. From swimming to walking with a salamander robot driven by a spinal cord model. *science*, 315(5817):1416–1420, 2007.

[8] Erwin Coumans and Yunfei Bai. Pybullet, a python module for physics simulation for games, robotics and machine learning. 2016.

[9] Wiem Mkaouer, Marouane Kessentini, Adnan Shaout, Patrice Koligheu, Slim Bechikh, Kalyanmoy Deb, and Ali Ouni. Many-objective software remodularization using nsga-iii. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 24(3):1–45, 2015.

[10] J. Blank and K. Deb. Pymoo: Multi-objective optimization in python. *IEEE Access*, 8:89497–89509, 2020.