- **Hint 1:** By default, all the simulations start with the phases of the oscillators set to a predetermined value that is assigned in network.py::SalamanderNetwork. The input of such class initialization method includes a SimulationParameters object, the same that you define before running any simulation. You can modify/extend the fields of such class to include the possibility to personalize the oscillators' initialization (to a random or desired value).
- **Hint 2:** When computing the metrics to evaluate the performance of the network, be sure to discard the initial transient.
- **Hint 3:** The hydrodynamic loads acting on the network should have an opposite effect on the left and right sides. Take this into account when assigning the sensory feedback gains.
- **Hint 4:** When studying the feedback-driven behavior, pay attention to the relationship between the sign of the feedback gain and the direction of the emerging locomotion pattern.
- **Hint 5:** When using random quantities, include a mechanism to set the seed of the number generator (e.g.; using the function np.random.seed). This will allow you to replicate the results when running the simulation again.