Reinforcement learning with function approximation for 3-spheres swimmer

Luca Berti, Zakarya El Khiyati, Youssef Essousy, Christophe Prud'Homme, Laetitia Giraldi

Reinforcement learning with function approximation for 3-spheres swimmer

Luca Berti * **Zakarya El Khiyati** ** **Youssef Essousy** ***
Christophe Prud'homme * **Laetitia Giraldi** **

* *Université de Strasbourg, CNRS, IRMA UMR 7501, Cemosis, F-67000 Strasbourg, France (e-mail: berti@math.unistra.fr, prudhomm@math.unistra.fr).*
** *Université Côte d'Azur - Inria Sophia-Antipolis, France (e-mail: zakarya.el-khiyati@inria.fr, laetitia.giraldi@inria.fr).*
*** *Laboratory MSDA, Mohammed VI Polytechnic University, Green City, Morocco (email: youssef.essousy@um6p.ma).*

Abstract: We study the swimming strategies that maximize the speed of the three-sphere swimmer using reinforcement learning methods. First of all, we ensure that for a simple model with few actions, the Q-learning method converges. However, this latter method does not fit a more complex framework (for instance the presence of boundary) where states or actions have to be continuous to obtain all directions in the swimmer's reachable set. To overcome this issue, we investigate another method from reinforcement learning which uses function approximation, and benchmark its results in absence of walls.

Keywords: Reinforcement learning control, Micro-swimming, Three-sphere swimmer, Function approximation

1. INTRODUCTION

The optimization of micro-swimmers' displacement is a subject of growing interest in recent literature, since it could boost the use of such robots as medical tools (Sitti (2009); Servant et al. (2015)). The solution of such optimization problems requires to take into account the swimmer's hydrodynamics as a constraint. However, solving this coupled problem is a challenging task due to the numerical complexity of the equation of motion of the swimmer. Indeed, the swimmer's displacements are governed by partial differential equations describing the behaviour of the surrounding fluid.

The problem of finding the best strategy of locomotion (Lauga (2020); Tam and Hosoi (2007)) is associated with the optimization of a certain cost function which could be described either as the speed (El Alaoui-Faris et al. (2020)) or as the energy of the system (Lohéac et al. (2013); Nasouri et al. (2019)). One solution to tackle this type of problems is to solve a simpler problem deriving from the first order condition of the optimization problem (see Alouges et al. (2019)). An equivalent strategy is to consider an optimal control approach and to describe the solution using Pontryagin maximum principle (Martín et al. (2016); Giraldi et al. (2015)).

More recently, reinforcement learning (Alageshan et al. (2020); Tsang et al. (2020); Liu et al. (2021)) and deep reinforcement learning (Garnier et al. (2021)) have been applied to fluid mechanics in order to control the fluid flow, optimize the shape or the swimming strategy of swimmers (Esparza López et al. (2019)). Alternative strategies as genetic algorithms have also been investigated (Ishimoto (2016)).

The use of reinforcement learning tabular methods (Q-learning in the first place) is well adapted to study problems where the state and the action belong to small finite dimensional spaces. This aspect was exploited in Tsang et al. (2020); Liu et al. (2021) to study the optimal swimming strategy for sphere-swimmers at low Reynolds number, finding the travelling wave to be the optimal arm activation strategy. However, when the state space has an continuous component (orientation, distance from a wall), tabular methods are not suitable anymore and a modification of reinforcement learning is needed. In a more complex framework (for instance the presence of boundary), states or actions have to be considered in a large finite dimensional space to obtain all directions in the swimmer's reachable set.

This paper investigates another method from reinforcement learning which uses function approximation to overcome this issue. More precisely, we apply the reinforcement learning method called differential semi-gradient SARSA (Sutton and Barto (2018)) to the well-known 3-spheres swimmer (Najafi and Golestanian (2004)). We study the optimal gait of the swimmer when the distances between the spheres can take more than two discrete values. As expected, we find that the optimal strategy remains a square stroke. The paper is organised as follows: in section 2, the 3-spheres swimmer model, its dynamics and the optimization problem are presented; in section 3, the reinforcement learning framework is discussed, justifying its applicability to the problem at hand; in section 4 numerical

⋆ The authors thank the Academy "Complex Systems" for the financial support during their research stay.

results on the optimization of the swimming gait of the 3-spheres swimmer are illustrated.

2. MATHEMATICAL MODELLING

2.1 3-spheres swimmer

The 3-spheres swimmer Najafi and Golestanian (2004) is composed of three aligned spheres having the same radius R (see Fig 1). The two outer spheres, B_1 and B_2, are connected to the central one B_3 by thin extensible links. The propulsion of the swimmer is ensured by changing the lengths, d_L and d_R, of the left and right connecting arms in a non-reversible fashion, in order to break the time-reversal symmetry of the Stokes equations (see Scallop theorem in Purcell (1977)).

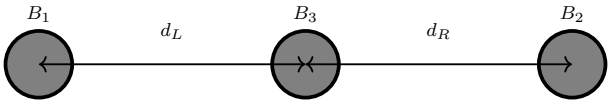


Fig. 1. Three-sphere swimmer and notations.

2.2 Swimmer's dynamics

Let us denote by x^{CM} the center of mass of the swimmer. Since at low Reynolds number inertial effects are absent and the spheres have the same properties, x^{CM} is the average of the centers of the spheres. The motion of the swimmer derives from the solution of the following Stokes problem, with $i = 1, 2, 3$,

$$\begin{cases} -\mu\Delta u + \nabla p = 0, & \text{in } \mathcal{F}_t, \\ \nabla \cdot u = 0, & \text{in } \mathcal{F}_t, \\ u = U + \omega \times (x - x^{CM}(t)) + u_d(t) & \text{on } \partial B_i, \\ m\dot{U} = -F_{fluid} = 0, \\ J\dot{\omega} = -M_{fluid} = 0. \end{cases} \tag{1}$$

where the fluid domain $\mathcal{F}_t$ is equal to $\mathbb{R}^3 \setminus \cup_{i=1,2,3} B_i$, the hydro-dynamical forces F_{fluid} (resp. moments M_{fluid}) are defined by $\sum_{i=1}^{3} \int_{\partial B_i} \sigma(u(s), p(s))\text{ds}$ (resp. $\sum_{i=1}^{3} \int_{\partial B_i} \sigma(u(s), p(s)) \times (x - x^{CM}(t))\text{ds}$), where σ is the Cauchy tensor $\sigma(u, p) := (\nabla u + \nabla u^t) - pId$. The absence of inertia and external forces leads to $F_{fluid} = M_{fluid} = 0$, from which follows that values of U and ω, associated to a change in the elongation speed of the links u_d, encoding the swimming strategy, are instantaneously attained.

System (1) is solved using the Feel++ finite element library (for more details on the numerical schemes see Berti et al. (2021)). Instead of considering the three-dimensional problem, we restrict to its two-dimensional section: this reduction is justified by the qualitatively similar behaviour of the two and three-dimensional 3-spheres swimmer shown in Figure 2, and it also reduces sensibly the computational cost of the numerical simulations.

2.3 Optimization problem

The paper focuses on the optimization problem of finding the best strategy, $t \mapsto (d_L(t); d_R(t))$ with a bounded speed

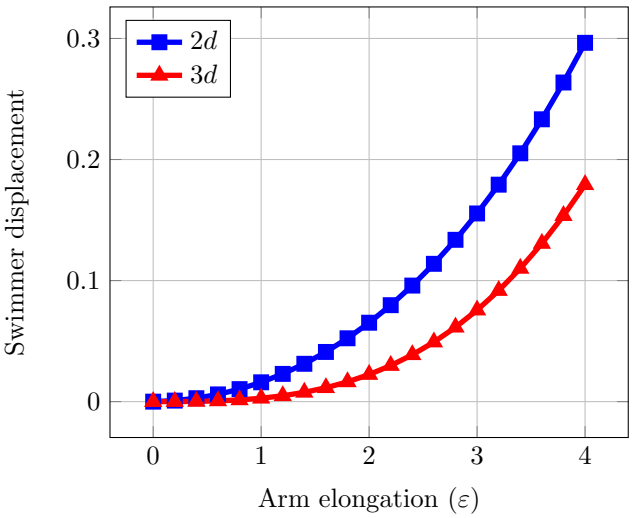


Fig. 2. Comparison of the three-sphere swimmer displacement as a function of its arm elongation. In this case, the spheres have radii $R = 1$, the discrete space of attainable arm lengths is $\mathbb{L} = \{l_0, l_1\}$, where $l_1 = 10$ and $l_0 = l_1 - \varepsilon$. The same qualitative behaviour is seen in the 2D and 3D case.

of deformation, for swimming as fast as possible. Below, the maximal speed of deformation is a positive real number given by M. For a given large T, the optimization problem reads as

$$\begin{aligned} \max_{t \mapsto (d_L(t); d_R(t))} \quad & x^{CM}(T) - x^{CM}(0). \\ (x^{CM}, d_L, d_R) \quad & \text{solution of (1)} \\ & ||u_d|| \leq M \end{aligned} \tag{2}$$

3. REINFORCEMENT LEARNING METHODS

3.1 Methods

We focus on solutions of (2) where d_L and d_R are stepwise constant functions with values in a finite set $\mathbb{L} \doteq \{l_0, ..., l_N\}$ (see Figure 3). More precisely, the arm lengths will vary only in a predefined set of values and only one arm at a time.

To this end, we will be using reinforcement learning (RL) tools.

In the reinforcement learning framework, the 3-spheres

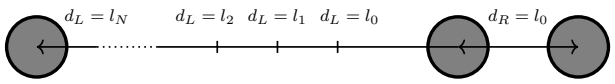


Fig. 3. Representation of the set $\mathbb{L}$ of preset lengths.

swimmer will be learning through interaction with its environment how to map *states* to *actions* in such way as to maximize some perceived rewards.

The formal framework under which RL algorithms are studied is that of Markov Decision Processes (MDPs).

Given a probability space, an MDP is defined as a quadruplet $(\mathcal{S}, \mathcal{A}, P, R)$, where $\mathcal{S}$ is a finite states space, $\mathcal{A}$ is a finite actions space, $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ defines the state transition probability kernel and $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the immediate reward function.

We will denote by S_t, A_t and R_t the state, the action and the reward at time t respectively.

The state of the 3-spheres swimmer at time t is completely determined by the tuple $(d_L(t), d_R(t)) = S_t \in \mathbb{L}^2$, so our states space $\mathcal{S}$ is $\mathbb{L}^2$. The actions space will consist of four actions: either extending, or retracting, the left or the right arm. The reward R_t will consist of the horizontal displacement of the center of mass of the swimmer between times $t-1$ and t, i.e., $R_t \doteq x^{CM}(t) - x^{CM}(t-1), \forall t \geq 1$.

A key property here is the Markov property - that is, the distribution of (S_{t+1}, R_{t+1}) is independent of the previous states and actions $S_{0:t-1}, A_{0:t-1}$ conditionally on the current action S_t. This property is verified by our system since it is deterministic and (A_t, S_t) completely determines (S_{t+1}, R_{t+1}).

Mapping states to actions will be done according to a *policy* π which can be defined as a function $\pi(.|.) : \mathcal{A} \times \mathcal{S} \to [0, 1]$ such that $\pi(a|s)$ is the probability to select action $\boldsymbol{a}$ given that the current state is $\boldsymbol{s}$.

Our goal will be to maximize a long term return. Two different forms of the return will be used in this work depending on the method. First, the discounted return G_t is defined, for a discount factor $\gamma \in [0, 1]$, as

$$G_t \doteq \sum_{k=t}^{\infty} \gamma^{k-t} R_{k+1} \forall t = 0, 1, 2, ...$$

The second form of the return G_t is introduced in section 3.3.

We further define the value function of a state s under a policy π as

$$v_\pi(s) \doteq \mathbb{E}_\pi(G_0|S_0 = s),$$

and the value of state-action pair (s, a) under a policy π as

$$q_\pi(s, a) \doteq \mathbb{E}_\pi(G_0|S_0 = s, A_0 = a),$$

where $\mathbb{E}_\pi$ is the expectation conditioned on the agent following the policy π.

The value function v_π indicates how good a given state is when following a policy π. Similarly, the action-value function q_π assesses the attractiveness of a state-action pair (s, a) by giving the expected return of taking the action a starting from the state s and following the policy π after that.

A policy π is said to be better than another policy π' if for all states $s \in \mathcal{S}$, $v_\pi(s) \geq v_{\pi'}(s)$.

Under the MDP framework, there always exists a policy that is better than all the other policies. All such policies share the same value and action-value functions, they are denoted by π_* and their value function (resp. action-value function) by v_* (resp. q_*).

3.2 Q-learning

Algorithm 1 is guaranteed to converge almost surely to the optimal action-value function $q*$ under the following assumptions (Watkins and Dayan (1992))

(1) Every action is visited an infinite number of times in the limit;
(2) The learning rate sequence (α_t) decreases according to the usual stochastic approximation conditions:

Algorithm 1 Q-learning (Watkins)

```
Initialize the state S
Initialize the action A
for t = 1, 2, ... do
    Run the simulation with action A
    Observe the reward R and next state S'
    Choose the next action A' (ε-greedily)
    q(S, A) ← q(S, A) + α(R + γ max q(S', .) − q(S, A))
    S ← S'
    A ← A'
end for
```

$$\sum_{k \geq 0} \alpha_k(s, a) = \infty, \forall (s, a) \in \mathcal{S} \times \mathcal{A}, \tag{3}$$

$$\sum_{k \geq 0} \alpha_k^2(s, a) < \infty, \forall (s, a) \in \mathcal{S} \times \mathcal{A}. \tag{4}$$

This algorithm was used to successfully find the optimal gait for the 3-spheres swimmer with $N = 1$ in Tsang et al. (2020).

3.3 Function approximation

The Q-learning methods is useful for problems with a small state space, as the amount of time and data needed to obtain reliable estimations of the value functions becomes unrealistic as the state space gets larger. In this section we describe an alternative, approximate solution method named differential semi-gradient SARSA in Sutton and Barto (2018). The problem we ultimately want to tackle using this method is that of 3-spheres swimmer near a wall where few arm lengths is no longer sufficient to describe all the reachable set of the swimmer and in addition its orientation and the distance from the wall are also needed; the continuous nature of these additional state variables prohibits the use of exact solution methods.

Our discrete observations of the new state variables no longer satisfy the Markov property and with that we lose some of the results we previously had. For instance, the existence of an optimal policy in the sense that we defined above is no longer guaranteed (Singh et al. (1994)). A commonly used way to order policies in this context is by resorting to the average reward of policy $r(\pi)$ defined as follows:

$$r(\pi) \doteq \lim_{\infty} \mathbb{E}_\pi(R_t|S_0).$$

The return is then defined as:

$$G_t \doteq R_{t+1} - r(\pi) + R_{t+2} - r(\pi) + R_{t+3} - r(\pi) + ...,$$

and the definitions of the value function and the action-value function remain the same after replacing the return with its new form. Policies are then ordered according to their corresponding average rewards.

The main idea of approximate solution methods is to use an approximate representation $\hat{q}$ of the value functions (and/or the policy, but not in our case). We write

$$q(s, a) \approx \hat{q}(s, a, w),$$

where $w \in \mathbb{R}^d$ are weight parameters that will be modified instead of updating the state-action values directly.

We use a linear representation for $q(s, a) \approx x(s, a)^T w$, where the feature vector x is constructed using tile coding

Table 1. Symmetries of the three-sphere swimmer that are exploited by our algorithm.

Initial state	Action	Reward
(x, l_i)	extend right arm	δ
(x, l_{i+1})	retract right arm	$-\delta$
(l_{i+1}, x)	retract left arm	δ
(l_i, x)	extend left arm	$-\delta$

(see Sutton and Barto (2018)), to which we apply algorithm 2 from Sutton and Barto (2018) for our control problem. The numerical results are shown in the following section. To reduce simulation time during the learning process, we leverage the deterministic nature of our system and use the simulation only when a new state-action pair arises for the first time. We also exploit the symmetries in Table 1 satisfied by the system to further reduce the required computations.

Algorithm 2 Differential semi-gradient sarsa

Initialize the state S
Initialize the action A
for $t = 1, 2, ...$ **do**
    Run the simulation with action A
    Observe the reward R and next state S'
    Choose the next action A' (ϵ-greedily)
    $\delta \leftarrow R - \bar{R} + \hat{q}(S', A', w) - \hat{q}(S, A, w)$
    $\bar{R} \leftarrow \bar{R} + \beta\delta$
    $w \leftarrow w + \alpha\delta\nabla\hat{q}(S, A, w)$
    $S \leftarrow S'$
    $A \leftarrow A'$
end for

4. NUMERICAL RESULTS

The reinforcement learning method we have employed (see Algorithm 2) allowed us to recover the results of Tsang et al. (2020) for the 3-spheres swimmer in infinite domain, where we exploited the qualitative agreement found between the two and three dimensional 3-spheres swimmers, reported in Figure 2. Figure 4 presents these results: the swimming strategy that was recovered at the end of the learning process, as well as the behaviour of the Q-function, are shown. It can be seen that the convergence of the algorithm to the optimal policy takes around 400 iterations, that is four times more than the Q-learning approach by Tsang et al. (2020). This can be justified by the smaller learning rate we considered, which was inversely proportional to the number of tilings that discretized our domain. The results from fluid simulations were fed to the learning algorithm using two approaches, resulting in the same optimal strategy: one one hand, for each agent-environment interaction, a simulation was run; on the other hand, exploiting the time reversibility of the flow and the absence of inertia (see also Table 1), the displacement resulting from each simulation was stored and accessed instead of repeating the action. As it was previously said, the optimal strategy that was found in this case is the travelling wave.

We now enlarge the state space by inserting an intermediate length between the two that were used before. The action space will now be different for each state, containing a minimum of two actions and a maximum of four. The actions correspond to elongation and retraction of one arm

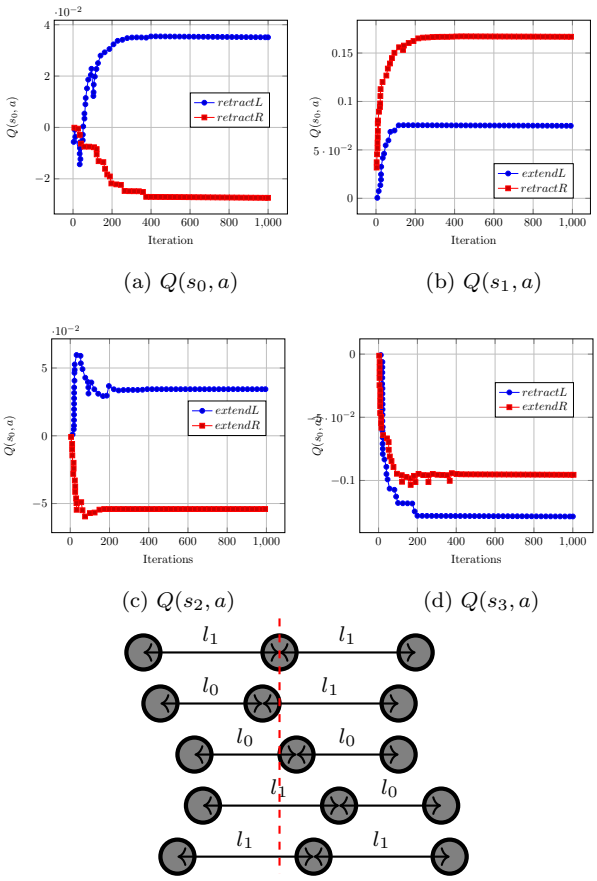
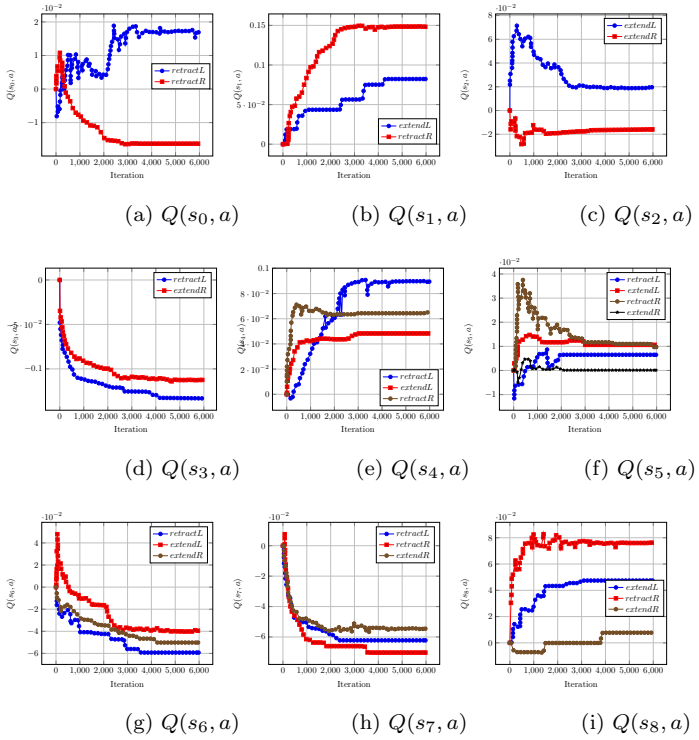


Fig. 4. The 3-spheres swimmer recovers the travelling wave strategy as the optimal one. On the top, we show the the convergence of the Q-function, and on the bottom, the optimal swimming stroke.

at a time, of a fixed amount: this obliges the agent to pass through the states with intermediate lengths, and to make sure that all actions produce a similar reward. The results for this case are presented in Figure 5, where we see that the optimal strategy is again a travelling wave, and each link passes from one length extremum to the other before the following link is activated. We considered a last case in which two intermediate lengths were available, and we found once more the travelling wave strategy to be the optimal one. We report in Figure 6 the optimal stroke that was repeatedly found, in the 3-spheres swimmer phase space of axes $d_L - d_R$. A travelling wave, composed of $4N$ actions, proved to be the optimal swimming strategy for the 3-spheres swimmer when multiple intermediate link lengths are accessible to the agent. Independently of the number of intermediate arm lengths N, the optimal policy policy π_* is a deterministic policy verifying $\pi_*(a|(l_i, l_j)) = 1$, where

- $a =$ extend right arm when $i = 0$ and $j \in \{0, ..., N-1\}$;
- $a =$ extend left arm when $j = 0$ and $i \in \{0, ..., N-1\}$;
- $a =$ retract right arm when $i = N$ and $j \in \{1, ..., N\}$;
- $a =$ retract left arm when $j = N$ and $i \in \{1, ..., N\}$.

The optimal policy translates to piece-wise functions $(d_L(t), d_R(t))$ prescribing the length of the arms



(a) $Q(s_0, a)$ (b) $Q(s_1, a)$ (c) $Q(s_2, a)$

(d) $Q(s_3, a)$ (e) $Q(s_4, a)$ (f) $Q(s_5, a)$

(g) $Q(s_6, a)$ (h) $Q(s_7, a)$ (i) $Q(s_8, a)$

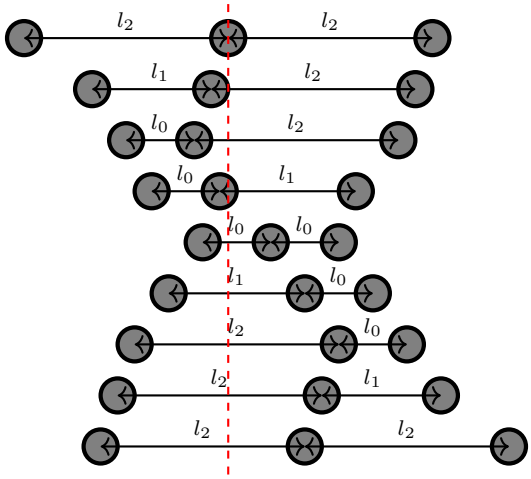


Fig. 5. The results when $\mathbb{L} = \{l_0, l_1, l_2\}$ are presented. On the top, we report the convergence plots of the Q-function. On the bottom, we present the optimal swimming strategy that was found.

$$d_L(t) = \begin{cases} l_0 + \dfrac{l_N - l_0}{N} t, & \text{if } d_R(t) = l_0, 0 \le t \le N, \\ l_N - \dfrac{l_N - l_0}{N} t, & \text{if } d_R(t) = l_N, 0 \le t \le N, \\ l_0, & \text{if } d_R(t) \text{ decreases}, \\ l_N, & \text{if } d_R(t) \text{ increases}, \end{cases} \tag{5}$$

$$d_R(t) = \begin{cases} l_0 + \dfrac{l_N - l_0}{N} t, & \text{if } d_L(t) = l_N, 0 \le t \le N, \\ l_N - \dfrac{l_N - l_0}{N} t, & \text{if } d_L(t) = l_0, 0 \le t \le N, \\ l_0, & \text{if } d_L(t) \text{ increases}, \\ l_N, & \text{if } d_L(t) \text{ decreases}. \end{cases} \tag{6}$$

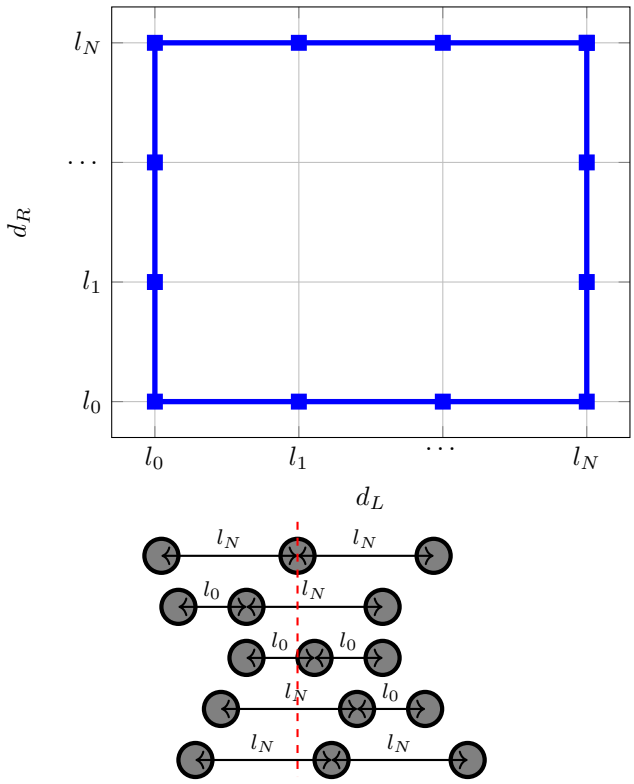


Fig. 6. In the top figure, trajectory in the 3-spheres swimmer phase space of the optimal strokes for the 3-spheres swimmer, with intermediate lengths. In the bottom figure, the optimal stroke when intermediate lengths are available.

We present in Figure 7 the cumulative displacement, in log-log scale, for the cases when $N = 1, 2, 3$. The three curves show the same asymptotic behaviour, which means that the same optimal propulsion speed is reached in the three cases. It can be seen that the optimal strategy is found at different iteration numbers T_N, that are larger as the number of intermediate lengths grows, which also give an approximate idea of the learning time for different values of N.

5. CONCLUSION

We have shown that, using a differential semi-gradient SARSA method, we are able to recover the optimal square stroke of the 3-spheres swimmer. As expected, the introduction of intermediate arm lengths does not vary the optimal swimming strategy. Thus, this paper validates the coupling of a reinforcement learning algorithm with a finite element approach to study the swimmer's dynamics in a toy case. The direct perspective is to study SARSA methods in more complex environments requiring large number of actions or states, for instance the presence of a plane wall.

REFERENCES

Alageshan, J.K., Verma, A.K., Bec, J., and Pandit, R. (2020). Machine learning strategies for path-planning microswimmers in turbulent flows. *Phys. Rev. E*, 101, 043110.

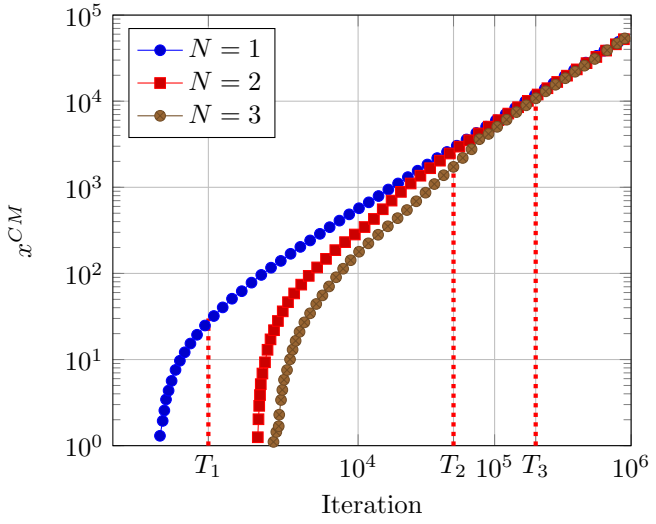


Fig. 7. The cumulative rewards of the cases $N = 1, 2, 3$ are plotted together. They show the same asymptotic behaviour, corresponding to the same optimal swimming speed, reached after the convergence of the algorithm. Different learning times T_N, increasing with N, are empirically identified, and correspond to the iteration where the optimal policy is found.

Alouges, F., DeSimone, A., Giraldi, L., Or, Y., and Wiezel, O. (2019). Energy-optimal strokes for multi-link microswimmers: Purcell's loops and Taylor's waves reconciled. *New Journal of Physics*, 21(4), 043050.

Berti, L., Chabannes, V., Giraldi, L., and Prud'homme, C. (2021). Modelling and finite element simulation of multi-sphere swimmers. *Comptes Rendus. Mathématique*, 359(9), 1119–1127.

El Alaoui-Faris, Y., Pomet, J.B., Régnier, S., and Giraldi, L. (2020). Optimal actuation of flagellar magnetic microswimmers. *Physical Review E*, 101(4), 042604.

Esparza López, C., Théry, A., and Lauga, E. (2019). A stochastic model for bacteria-driven micro-swimmers. *Soft Matter*, 15(12), 2605–2616.

Garnier, P., Viquerat, J., Rabault, J., Larcher, A., Kuhnle, A., and Hachem, E. (2021). A review on deep reinforcement learning for fluid mechanics. *Computers & Fluids*, 225, 104973.

Giraldi, L., Martinon, P., and Zoppello, M. (2015). Optimal design of Purcell's three-link swimmer. *Physical Review E*, 91(2), 023012.

Ishimoto, K. (2016). Hydrodynamic evolution of sperm swimming: Optimal flagella by a genetic algorithm. *Journal of Theoretical Biology*, 399, 166–174.

Lauga, E. (2020). Traveling waves are hydrodynamically optimal for long-wavelength flagella. *Physical Review Fluids*, 5(12), 123101.

Liu, Y., Zou, Z., Tsang, A.C.H., Pak, O.S., and Young, Y.N. (2021). Mechanical rotation at low Reynolds number via reinforcement learning. *Physics of Fluids*, 33(6), 062007.

Lohéac, J., Scheid, J.F., and Tucsnak, M. (2013). Controllability and time optimal control for low reynolds numbers swimmers. *Acta Applicandae Mathematicae*, 123(1), 175–200.

Martín, J.S., Takahashi, T., and Tucsnak, M. (2016). An optimal control approach to ciliary locomotion. *Mathematical Control and Related Fields*, 6(2), 293–334.

Najafi, A. and Golestanian, R. (2004). Simple swimmer at low Reynolds number: Three linked spheres. *Physical Review E*, 69(6), 062901.

Nasouri, B., Vilfan, A., and Golestanian, R. (2019). Efficiency limits of the three-sphere swimmer. *Physical Review Fluids*, 4(7), 073101.

Purcell, E.M. (1977). Life at low reynolds number. *American journal of physics*, 45(1), 3–11.

Servant, A., Qiu, F., Mazza, M., Kostarelos, K., and Nelson, B.J. (2015). Controlled In Vivo Swimming of a Swarm of Bacteria-Like Microrobotic Flagella. *Advanced Materials*, 27(19), 2981–2988.

Singh, S.P., Jaakkola, T., and Jordan, M.I. (1994). Learning without state-estimation in partially observable markovian decision processes. In *Machine Learning Proceedings 1994*, 284–292. Elsevier.

Sitti, M. (2009). Voyage of the microrobots. *Nature*, 458(7242), 1121–1122.

Sutton, R. and Barto, A. (2018). *Reinforcement Learning, second edition: An Introduction*. Adaptive Computation and Machine Learning series. MIT Press.

Tam, D. and Hosoi, A.E. (2007). Optimal Stroke Patterns for Purcell's Three-Link Swimmer. *Physical Review Letters*, 98(6), 068105.

Tsang, A.C.H., Tong, P.W., Nallan, S., and Pak, O.S. (2020). Self-learning how to swim at low Reynolds number. *Physical Review Fluids*, 5(7), 074101.

Watkins, C.J. and Dayan, P. (1992). Q-learning. *Machine learning*, 8(3-4), 279–292.