IMPLEMENTATION OF DEEP REINFORCEMENT LEARNING IN OPENFOAM FOR ACTIVE FLOW CONTROL

SAEED SALEHI[1], HÅKAN NILSSON[2]

[1]*Chalmers University of Technology, saeed.salehi@chalmers.com*

[2]*Chalmers University of Technology, hakan.nilsson@chalmers.com*

Keywords: *Artificial intelligence, Deep reinforcement learning, CFD, OpenFOAM, Active flow control*

Recent advancements in artificial intelligence and machine learning have enabled tackling high-dimensional controlling and decision-making problems. Deep Reinforcement Learning (DRL), as a combination of deep learning and reinforcement learning, can perform immensely complicated cognitive tasks at a superhuman level.

DRL can be utilized in fluid mechanics for different purposes, for instance, training an autonomous glider [1], exploring swimming strategies of multiple fish [2, 3], controlling a fluid-directed rigid body [4], proposing shape optimization [5, 6]. DRL can also be utilized for Active Flow Control (AFC) [7], which is of crucial importance for mitigating damaging effects or enhancing favourable consequences of fluid flows. Optimizing the AFC strategy using classical optimization methods is usually a highly non-linear problem and involves designing various parameters, while DRL can learn sophisticated AFC strategies and fully exploit the abilities of the actuator. It is based on the reinforcement learning concept that explores the state-action-reward sequence and offers a powerful tool for conducting closed-loop feedback control.

In the present work, a coupled DRL-CFD framework was developed within OpenFOAM, as opposed to previous attempts in the literature in which the CFD solver was treated as a black box. Here, the DRL agent is implemented as a boundary condition that is able to sense the *environment state*, perform some *action*, and record the corresponding *reward*. Figure 1 displays a simple flowchart of the developed DRL framework in which a deep neural network (DNN) is used as the decision maker (i.e., policy function).

To test and verify the performance of the developed DRL-CFD software, the simple test case of vortex shedding behind a 2D cylinder is investigated. The actuator is a pair of synthetic jets on top and bottom of the cylinder. The reward function is defined as the reduction of drag and the absolute value of lift. Thereby, the DRL agent (which is a deep neural network here) learns to minimize the drag and lift coefficients by applying the optimum jet flow at each time step. The DRL agent was trained through a total of 1000 CFD simulations. Figure. 2 presents the variation of drag and lift coefficients of the cylinder for both cases. The controlling mechanism starts at $t = 40$ s and it can be seen that both forces reduce significantly.

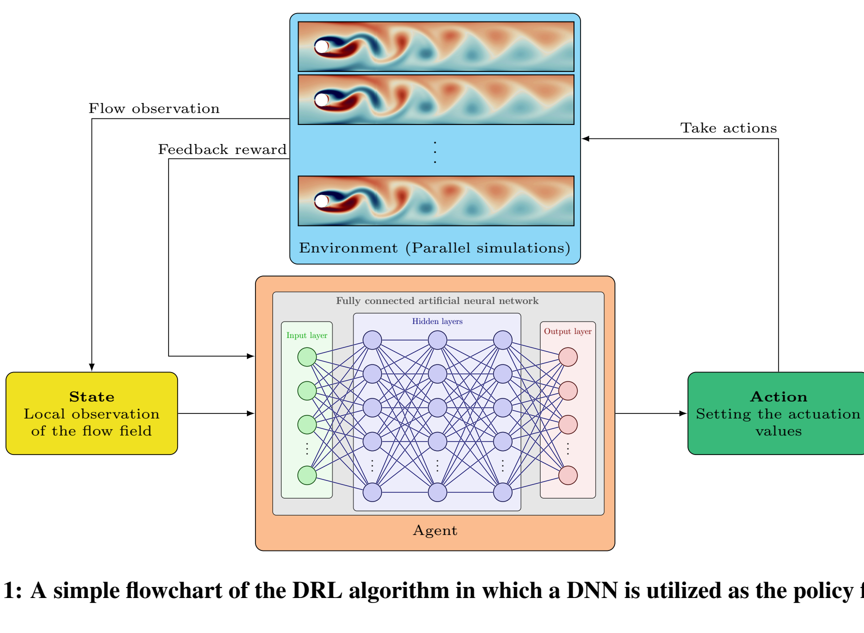
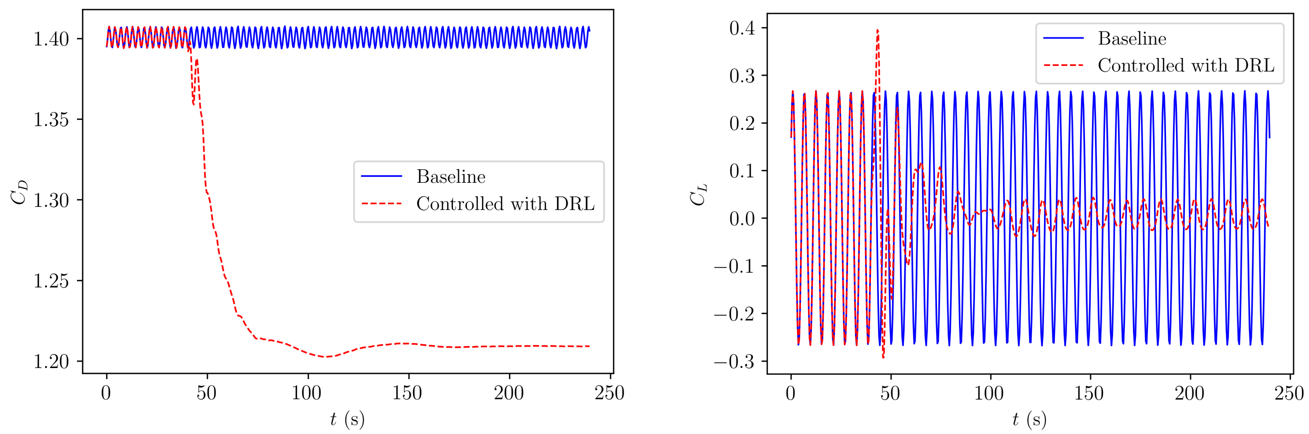


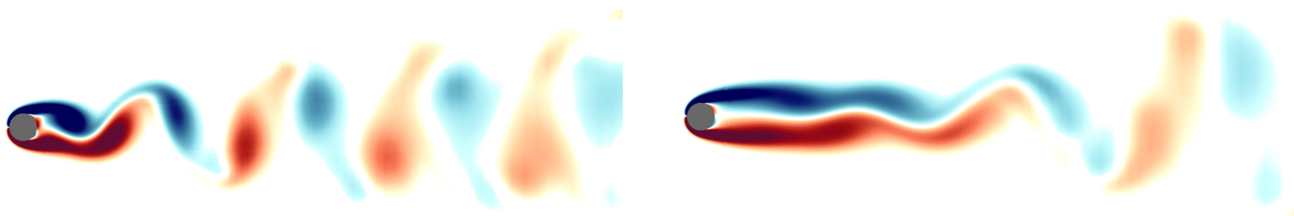
Figure 1: A simple flowchart of the DRL algorithm in which a DNN is utilized as the policy function



(a) Drag coefficient

(b) Lift coefficient

Figure 2: Variation of (a) drag and (b) lift coefficients for the uncontrolled (baseline) and DRL controlled cases. The controlling mechanism starts at $t = 40$ s.



(a) Baseline

(b) Controlled with DRL

Figure 3: Vorticity contours for (a) the uncontrolled (baseline) and (b) DRL controlled cases after reaching quasi stationary condition ($t = 200$ s)

The contours of vorticity behind the cylinder for the uncontrolled (baseline) and controlled cases, after reaching quasi-stationary condition ($t = 200$ s), are presented in Fig. 3. The vortex shedding effect is considerably reduced in the controlled case.

Acknowledgments

The research presented in this work was carried out as a part of "Swedish Centre for Sustainable Hydropower - SVC". SVC has been established by the Swedish Energy Agency, Energiforsk and Svenska kraftnät together with Luleå University of Technology, Uppsala University, KTH Royal Institute of Technology, Chalmers University of Technology, Karlstad University, Umeå University and Lund University, `www.svc.nu`.
The computations were enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS) and the Swedish National Infrastructure for Computing (SNIC) at NSC and C3SE partially funded by the Swedish Research Council through grant agreements no. 2022-06725 and no. 2018-05973.

References

[1] G. Reddy, A. Celani, T. J. Sejnowski, and M. Vergassola, "Learning to soar in turbulent environments," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 113, no. 33, pp. E4877–E4884, 2016.

[2] G. Novati, S. Verma, D. Alexeev, D. Rossinelli, W. M. Van Rees, and P. Koumoutsakos, "Synchronisation through learning for two self-propelled swimmers," *Bioinspiration and Biomimetics*, vol. 12, no. 3, 2017.

[3] S. Verma, G. Novati, and P. Koumoutsakos, "Efficient collective swimming by harnessing vortices through deep reinforcement learning," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 115, no. 23, pp. 5849–5854, 2018.

[4] P. Ma, Y. Tian, Z. Pan, B. Ren, and D. Manocha, "Fluid directed rigid body control using deep reinforcement learning," *ACM Transactions on Graphics*, vol. 37, no. 4, 2018.

[5] X. Y. Lee, A. Balu, D. Stoecklein, B. Ganapathysubramanian, and S. Sarkar, "Flow shape design for microfluidic devices using deep reinforcement learning," *CoRR*, vol. abs/1811.12444, 2018.

[6] J. Viquerat, J. Rabault, A. Kuhnle, H. Ghraieb, A. Larcher, and E. Hachem, "Direct shape optimization through deep reinforcement learning," *Journal of Computational Physics*, vol. 428, 2021.

[7] J. Rabault, M. Kuchta, A. Jensen, U. Réglade, and N. Cerardi, "Artificial neural networks trained through deep reinforcement learning discover control strategies for active flow control," *Journal of Fluid Mechanics*, vol. 865, pp. 281–302, 2019.