Applying a Differentiable Physics Simulation to Move Objects with Fluid Streams

Oskars Teikmanis, Laura Leja, Kārlis Freivalds
Institute of Electronics and Computer Science (EDI), Riga, Latvia
{oskars.teikmanis, laura.leja, karlis.freivalds}@edi.lv

***Abstract*—We investigate the application of differential physics in two scenarios involving the movement of objects using fluid streams. Through a PhiFlow-based simulation, we demonstrate the feasibility of moving an object horizontally using vertical streams and vice-versa, through iterative optimisation of the properties of the controlled flow sources. This intriguing discovery has the potential to enable the development of devices capable of moving distant objects in various directions using fixed unidirectional blowers.**

***Index Terms*—Differentiable physics, numerical optimisation, computational fluid dynamics, simulation.**

I. Introduction

The study of object displacement with air or water is a captivating subject in the fields of robotics and logistics, as it can provide insights into solving complex interactions between solids and fluids. The systems developed for such tasks must respect the physical laws governing the interactions between an agent and its surroundings. However, even when an accurate physical model is available, some interactions are nearly impossible to calculate analytically. An example of such a problem is the calculation of the exact timing and intensity of a stream of air to move a certain object to the desired location, while simultaneously satisfying additional constraints, such as the desired trajectory.

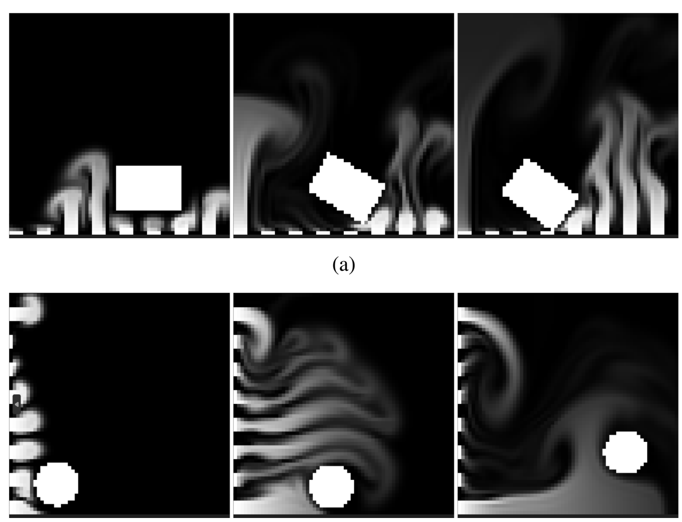
Previous studies have applied physics-based training to scenarios involving object interactions with air or water flows. For instance, reinforcement learning (RL) is used to train controllers that use air or water jets to maintain an object's position under gravity or even direct the ball in a simulated volleyball game [1]. Unfortunately, these controllers rely on fluid flow information from the simulator, which cannot be obtained in the real world.

To combine the benefits of traditional physical modelling and machine learning, one promising approach is to employ differentiable physics (DP), a machine learning framework that employs differentiable states to model the interactions between physical entities. Differentiability of simulated states provides significant potential for using optimisation algorithms to discover control inputs that facilitate a wide range of highly complex simulated interactions [2]. This approach is explored in [3], where a neural network-based controller is trained to move an object to a desired location while overcoming external forces resulting from turbulent air flows. Notably, this controller relies solely on the object's observable properties, such as its orientation, velocity, and related characteristics, to make the appropriate decisions. The authors show that controllers trained using DP perform better than RL or handcrafted PID-controller baselines. Yet, how to train realistic devices that manipulate objects with fluid flows still needs to be explored. In this article, we investigate the efficacy of DP for discovering control signals that move objects perpendicular to the air streams produced by simulated blowers.

This research is funded by the Latvian Council of Science, project "Smart Materials, Photonics, Technologies and Engineering Ecosystem", project No. VPP-EM-FOTONIKA-2022/1-0001.

II. Method

We consider two types of simulated control tasks, depicted in Fig. 1. The first *horizontal* scenario (Fig. 1a) aims to propel a box to the left as far and as fast as possible, by employing an array of eight regulated vertical blowers. The controlled variable is the speed of the fluid exiting each blower. The box can translate and rotate in any direction while being influenced by gravitational forces. The second *vertical* scenario (Fig. 1b) follows a comparable approach involving a circular object and eight controlled horizontal blowers. Here, the goal is to swiftly propel the object upwards. To achieve this, we perform differentiable physics-based optimisation.



(a)

(b)

Fig. 1: Three frames of the resulting simulations of solids interacting with fluid streams (from left to right): (a) rectangular object moved to the left with a vertical flow; (b) the round object lifted from the ground with a horizontal flow.

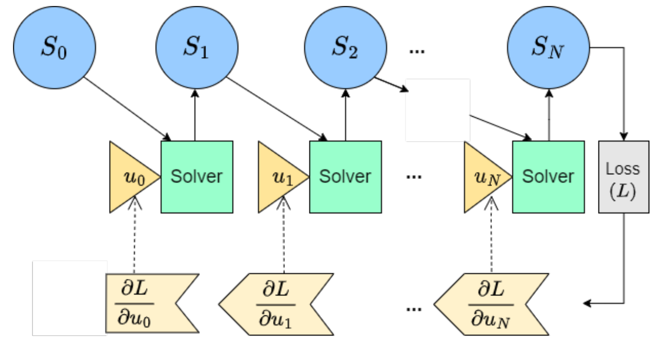


Fig. 2: Overview of the training method. Training is integrated with the physics simulation that is performed in several timesteps. At first, the initial state of the environment S_0, comprising an object and the fluid field, is created. At each timestep, the current state S_i and the values of control variable u_i are passed to the physics solver, which produces the next state of the environment. The loss function is calculated at the final state S_N and its gradient is backpropagated through all time steps to update the control variables.

The mathematics governing this problem are based on incompressible Navier-Stokes equations, which describe the motion of fluids. The two characteristic equations of such issues are the continuity equation (incompressibility condition) and the momentum equation. The momentum equation is discretised and solved in the simulation using finite difference methods. The object's position and orientation are updated by using Newtonian laws of motion.

For hosting and processing the simulated scenarios, we use PhiFlow [4], a differentiable simulation toolkit that enables the integration of physics-based systems for optimisation and machine-learning tasks. It provides gradient-based optimisation techniques for applications primarily focused on computational fluid dynamics (CFD). We adapt PhiFlow's implementation of fluid-solid interaction from [3], which is essential for the simulated scenarios discussed in this section.

A graphical overview of our method is shown in Fig. 2. Its top part depicts the simulation process. At each simulated time-step the PhiFlow-based solver reads in the current control input u_i and the current state of the environment denoted by $S_i = \{x, y, v_o, v_f, \omega_o\}$, where $i \in \{0, 1, 2, \ldots, N\}$ with N denoting the number of time-steps (45 in our case). The state includes the object's coordinates (x and y), velocity (v_o), rotational speed (ω_o), and the velocity field of the flow (v_f). Once this process is completed, the loss function is computed based on the final state S_N. In the *horizontal* scenario, the loss is defined as the sum of x coordinates over all simulation steps. This approach yields lower loss values when the object is successfully moved to the left. In the *vertical* scenario, the loss is defined as the negative sum of y coordinates, which encourages control inputs that result in upward movement.

The bottom part of the diagram depicts training in which the loss gradient is backpropagated through all the training steps. We use the Adam algorithm [5] with a learning rate of 0.05 to update the control variables. We perform 1000 optimisation steps which we found to be sufficient.

Our results are depicted in Fig. 1, which visualises the flow and object movement in both of our simulated scenarios using the control variables that are obtained by the optimisation. Both visualisations show a significant movement of the objects perpendicular to the stream direction, proving that the applied training method can produce adequate values for the fluid stream velocities.

III. Conclusion

The findings of this paper have shown that: (1) objects can be moved with fluids in a direction that significantly deviates from the blower's orientation, (2) it is possible to use gradient-based learning to simulate physical behaviours that would be nearly impossible to program by hand.

In addition, this research forms a basis for more involved, three-dimensional simulations that may enable the realisation of physical devices designed for displacing non-trivial assortments of objects. Applications may range from sorting packages with air flow to lifting underwater objects with concentrated streams.

References

[1] P. Ma, Y. Tian, Z. Pan, B. Ren, and D. Manocha, "Fluid directed rigid body control using deep reinforcement learning," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 1–11, 2018.

[2] Y. Hu, L. Anderson, T.-M. Li, *et al.*, "DiffTaichi: Differentiable Programming for Physical Simulation," in *Proceedings of the 7th International Conference on Learning Representations (ICLR)*, 2019.

[3] B. Ramos, F. Trost, and N. Thuerey, "Control of two-way coupled fluid systems with differentiable solvers," *arXiv preprint arXiv:2206.00342*, 2022.

[4] P. Holl, V. Koltun, and N. Thuerey, "Learning to Control PDEs with Differentiable Physics," in *Proceedings of the 8th International Conference on Learning Representations (ICLR)*, 2020.

[5] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2015.