SF2A 2023
M. N'Diaye, A. Siebert, N. Lagarde, O. Venot, K. Baillié, M. Béthermin, E. Lagadec, J. Malzac, J. Richard (eds)

USING ARTIFICIAL INTELLIGENCE TO IMPROVE THE PERFORMANCE OF RADIATIVE HYDRODYNAMICS SIMULATIONS

G. Radureau[1], C. Michaut[1], A. I. Comport[2] and R. Drouilly[1]

Abstract. Radiation hydrodynamics models the coupling between the dynamics of a hypersonic hot plasma and the radiation it produces or external radiation. Radiation hydrodynamics enables us to understand the flow of matter under the extreme conditions present in many astrophysical objects. Various simplified models exist, but in most cases they are limited or even wrong. It was in this context that the HADES code, which stands for "Hydrodynamics Adapted to the Description of Supersonic Flows", was developed (Nguyen (2011), Michaut et al. (2011), Michaut et al. (2017)). This code solves the general equations of radiative hydrodynamics in the 2D case and uses the M1 model for radiation transfer (Turpault 2003). However, calculating the radiative pressure remains a constraint that slows down the code and also requires search algorithms that do not always converge. We are therefore using "Multi-layer Perceptron" neural network architectures, which are among the simplest and most conventional, because we hope that they will reduce computing time and solve convergence problems. These architectures are composed of "perceptrons", the basic cells of neural networks containing parameters to be set, arranged in layers. The parameters of the perceptrons are adjusted so that the neural network gives the right answers to known examples. At the SF2A conference, I will be presenting my work on the use of these neural networks to calculate the Eddington tensor linking radiative pressure to radiative energy, work inspired by what has been done in the case of the study of the interaction of neutrinos with matter, in the case of supernovae (Harada et al. 2022). Preliminary results show a gain in computation time using these neural networks, but we obtain a loss of precision, which is unacceptable for radiative hydrodynamics simulations. On the other hand, a hybrid method using neural networks to initialise the search algorithms is much more interesting. The neural networks then help these algorithms to converge more quickly and more reliably. This latter solution is the most promising for the HADES code.

Keywords: Radiation hydrodynamics, machine learning

1 Introduction

In this work, we describe the fluid's mechanics with Euler's equations taking into account a source term $\overrightarrow{\mathbf{S}}$ and $c\mathrm{S}^0$, coming from the interaction of the fluid with light. We model the radiative transfer with the M1 multi-group model, which namely in cutting the frequency spectrum into groups of frequencies $\mathcal{G} = \{[\nu_1, \nu_2], [\nu_2, \nu_3], ...\}$, giving us one equation describing the evolution of the radiative energy, flux and pressure for each group :

$$\begin{cases} \partial_t \rho + \overrightarrow{\nabla} \cdot (\rho \overrightarrow{\mathbf{u}}) = 0 \\ \partial_t (\rho \overrightarrow{\mathbf{u}}) + \overrightarrow{\nabla} \cdot (\rho \overrightarrow{\mathbf{u}} \otimes \overrightarrow{\mathbf{u}} + p\mathbb{I}) = \overrightarrow{\mathbf{S}} \\ \partial_t E + \overrightarrow{\nabla} \cdot ((E+p)\overrightarrow{\mathbf{u}}) = c\mathrm{S}^0 \end{cases}$$

Euler equations

$$\begin{cases} \forall g \in \mathcal{G}, \quad \partial_t \mathrm{E_g} + \overrightarrow{\nabla} \cdot \overrightarrow{\mathbf{F}_g} = -c\mathrm{S}^0_\mathrm{g} \\ \forall g \in \mathcal{G}, \quad \partial_t (c^{-2}\overrightarrow{\mathbf{F}_g}) + \overrightarrow{\nabla} \cdot \mathbb{P}_g = -\overrightarrow{\mathbf{S}_g} \\ \mathrm{S}^0 = \sum_{g\in\mathcal{G}} \mathrm{S}^0_\mathrm{g} \ ; \ \overrightarrow{\mathbf{S}} = \sum_{g\in\mathcal{G}} \overrightarrow{\mathbf{S}_g} \end{cases}$$

M1 multi-group radiative transfer equations

Our model also relies on the assumption that light propagates in preferentially in the direction of the radiative flux, which leads to the following relation (Levermore 1984) :

$$\forall g \in \mathcal{G}, \quad \mathbb{P}_g = \mathbb{D}_g \mathrm{E_g} = \begin{bmatrix} \mathbb{D}_{g,xx} & \mathbb{D}_{g,xy} \\ \mathbb{D}_{g,xy} & \mathbb{D}_{g,yy} \end{bmatrix} \mathrm{E_g}$$

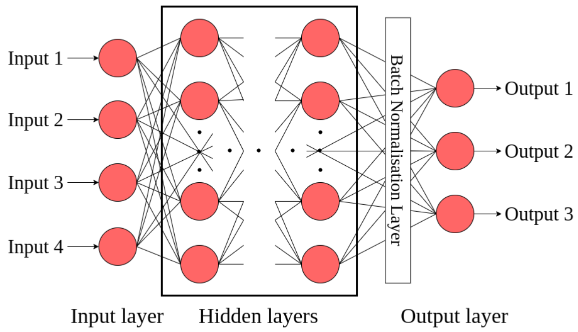
[1] Université Côte d'Azur, CNRS, OCA, Lagrange, France
[2] Laboratoire I3S, Université Côte d'Azur, CNRS, Sophia Antipolis, France

Where $\mathbb{D}_g$ is the Eddington tensors for the group g and $\mathbb{D}_{g,xx}$, $\mathbb{D}_{g,yy}$ and $\mathbb{D}_{g,yy}$ are its components. In radiation hydrodynamics simulations we compute this tensor, but it requires expensive numerical algorithms. We thus have explored the use of MLPs to compute the Eddington tensor in a more efficient way.

2 Our Machine learning approach

2.1 Multi-Layer Perceptron (MLP)

Neural networks are composed of elementary blocks, the *perceptrons*, which are mathematical functions, containing parameters that need to be adjusted to solve a given problem. These elementary structures can be assembled into layers to reproduce more complex functions.



In our case we have used an MLP containing 4 hidden layers, 36 neurons per hidden layer and we have trained the network for 500 epochs. The inputs of our neural network are the radiative energy E_g, the x and y components of the radiative flux $F_{x,g}$ and $F_{y,g}$ and the frequency bounds of the group ν_1 and ν_2. The outputs are the components of the Eddington tensor $\mathbb{D}_{g,xx}$, $\mathbb{D}_{g,yy}$ and $\mathbb{D}_{g,xy}$.

2.2 Use of machine learning algorithms : Eddington tensor

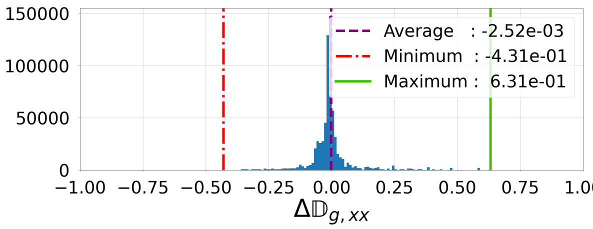


Fig. 1: Prediction error on $\mathbb{D}_{g,xx}$

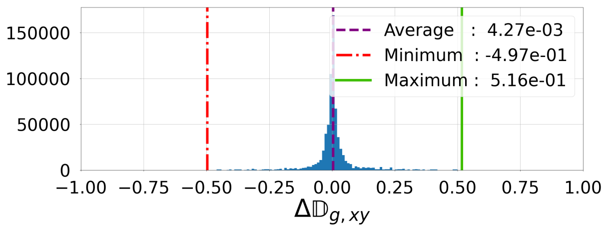


Fig. 2: Prediction error on $\mathbb{D}_{g,yy}$

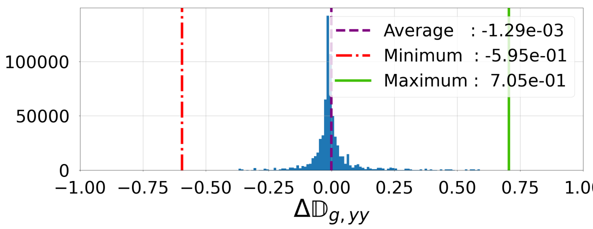


Fig. 3: Prediction error on $\mathbb{D}_{g,xy}$

We have trained an MLP to replace the current algorithm that computes the Eddington tensor in the M1 multi-group case, using random inputs. The error is centered on 0, yet we can get errors that reach 0.7, which is very significant, since $\mathbb{D}_{g,xx}$ and $\mathbb{D}_{g,yy}$ vary in [0, 1] and $\mathbb{D}_{g,xy}$ varies in [-1, 1].

3 Conclusions and perspectives

Our preliminary estimations have shown that using an MLP to compute the Eddington tensor can reduce by a factor 10 the computation time, yet they can introduce large numeric errors. In our future work, we will explore a hybrid method using an MLP and search algorithms to enhance the accuracy of the neural network.

This work was supported by a French government grant managed by the Agence Nationale de la Recherche under the Investissement d'avenir program, reference ANR-19-P3IA-0002.

References

Harada, A., Nishikawa, S., & Yamada, S. 2022, The Astrophysical Journal, 925, 117
Levermore, C. 1984, Journal of Quantitative Spectroscopy and Radiative Transfer, 31, 149
Michaut, C., Di Menza, L., Nguyen, H., Bouquet, S., & Mancini, M. 2017, High Energy Density Physics, 22, 77
Michaut, C., Nguyen, H., & Di Menza, L. 2011, Astrophysics and Space Science, 336, 175
Nguyen, H. C. 2011, PhD thesis, Paris 11
Turpault, R. 2003, Theses, Université Sciences et Technologies - Bordeaux I