Why Galaxies Care About AGB Stars
Proceedings IAU Symposium No. 343, 2019
F. Kerschbaum, M. Groenewegen, and H. Olofsson, eds.

DOI: 00.0000/X000000000000000X

Magritte: a new multidimensional accelerated general-purpose radiative transfer code

Frederik De Ceuster[1,2]†, **Jeremy Yates**[1], **Peter Boyle**[3], **Leen Decin**[2,4] **and James Hetherington**[5]

[1]Dept. of Physics and Astronomy, University College London, London, WC1E 6BT, UK
[2]Dept. of Physics and Astronomy, KU Leuven, Celestijnenlaan 200D, 3001 Leuven, Belgium
[3]School of Physics and Astronomy, The University of Edinburgh, Edinburgh EH9 3FD, UK
[4] Schoool of Chemistry, University of Leeds, Leeds LS2 9JT, UK
[5] The Alan Turing Institute, Euston Road, London NW1 2DB, UK

Abstract. Magritte is a new deterministic radiative transfer code. It is a ray-tracing code that computes the radiation field by solving the radiative transfer equation along a fixed set of rays for each grid cell. Its ray-tracing algorithm is independent of the type of input grid and thus can handle smoothed-particle hydrodynamics (SPH) particles, structured as well as unstructured grids. The radiative transfer solver is highly parallelized and optimized to have well scaling performance on several computer architectures. Magritte also contains separate dedicated modules for chemistry and thermal balance. These enable it to self-consistently model the interdependence between the radiation field and the local thermal and chemical states. The source code for Magritte will be made publicly available at github.com/Magritte-code.

Keywords. radiative transfer, astrochemistry, methods: numerical

1. Introduction

Radiative transfer plays a key role in the dynamics, the chemistry and the energy balance of various astrophysical objects. Therefore it is essential in astrophysical modelling to properly take into account all radiative processes and their interdependence. The ever growing size and complexity of these models requires fast and scalable methods to compute the radiation field. Magritte is a new general-purpose radiative transfer solver written in modern C++. In contrast to popular (probabilistic) Monte Carlo codes, Magritte is a deterministic ray-tracer which computes the radiation field by solving the transfer equation along a fixed set of rays originating from each grid cell. Being a deterministic code allows for various optimizations and facilitates it to exploit the various layers of parallelism in the calculation.

2. Solving the transfer equation

Magritte's ray-tracing algorithm only uses the locations of the cell centers and the nearest neighbor lists. Hence, it can cope with SPH particle data as well as with structured or unstructured grids. The local emissivities and opacities account for contributions from both lines and continua. Scattering is taken into account iteratively, adding an extra source and opacity. Magritte's solver is modular enough to cope with the most general anisotropic scattering formalisms. Once all emissivities and opacities are computed, the transfer equation is solved using a numerically more stable version of Feautrier's

† email: frederik.deceuster@kuleuven.be

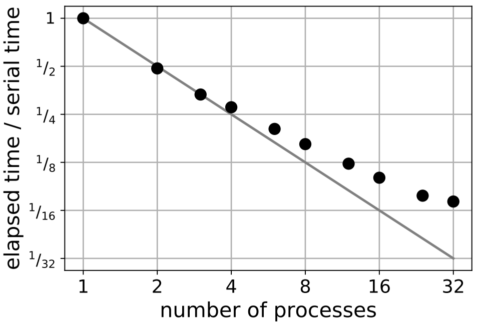

Figure 1. Plot of the (preliminary) strong scaling behaviour of the MPI distributed parallelization over the rays for a test model containing 192 rays, 220 frequency bins and 12,133 grid cells. The total elapsed time when computed on a single core (serial time) was 221 seconds.

second-order formulation (Feautrier 1964; Rybicki & Hummer 1991). Magritte's transfer solver, the chemistry and thermal balance modules are designed to be modular to facilitate coupling it (as a whole or in parts) with other codes.

3. Parallelizing a deterministic ray-tracer

There are three common parallel programming paradigms: message passing, threading, and single instruction multiple data (SIMD) vectorization. Magritte uses a combination of all three to ensure performance on both shared and distributed memory architectures. The computations for different rays in our algorithm are independent within an iteration. Therefore these can easily be distributed over different processes and the results communicated at the end of each iteration. This is done using the standard message passing interface (MPI). Figure 1 shows the (preliminary) scaling of this parallelization layer. Solving the transfer equation along a certain ray requires data from different grid cells and frequencies and thus can better be kept as local as possible in memory. Therefore, within each process, the computations for different cells are threaded using the OpenMP standard. The computations along a certain ray for different frequencies require exactly the same operations but with different values for the emissivities and opacities for each frequency. Hence, these computations are ideally suited for SIMD vectorization. To achieve this in a portable way, Magritte uses the SIMD vector types provided in the Grid library (Boyle et al. 2016). In future versions we will also explore the possibility of offloading the whole radiative transfer solver to graphics processing units (GPUs).

Acknowledgements

FDC is supported by the EPSRC iCASE studentship programme, Intel Corporation and Cray Inc. FDC and LD acknowledge support from the ERC consolidator grant 646758 AEROSOL. This work was performed using the Cambridge Service for Data Driven Discovery (CSD3), part of which is operated by the University of Cambridge Research Computing on behalf of the STFC DiRAC HPC Facility (www.dirac.ac.uk). The DiRAC component of CSD3 was funded by BEIS capital funding via STFC capital grants ST/P002307/1 and ST/R002452/1 and STFC operations grant ST/R00689X/1. DiRAC is part of the National e-Infrastructure.

References

Boyle, P., Yamaguchi, A., Cossu, G., & Portelli, A. 2016, *Proceedings of Science* 251 Lattice2015

Feautrier, P. 1964, *Comptes Rendus Academie des Sciences (serie non specifiee)* 258

Rybicki, G. B. & Hummer, D. G. 1991, *Astron. Astrophys.* 245, 171