USINE propagation code and associated tools

David Maurin[*]

LPSC, Université Grenoble-Alpes, CNRS/IN2P3, 53 avenue des Martyrs, 38026 Grenoble, France

E-mail: dmaurin@lpsc.in2p3.fr

I present the first public release of the USINE code for charged galactic cosmic-ray (GCR) propagation. USINE is a C++ toolbox handling GCR ingredients and several semi-analytic propagation models (1D and 2D). Non-public versions of this code were used in the last 10 years to fit the transport parameters, study radioactive nuclei, antinuclei and possible DM contributions, etc. The complementary of USINE with existing fully numerical models is that: (i) a typical model run is fast, so that the many user interfaces and plots provided in USINE make it a very pedagogical tool to better understand CR propagation; (ii) USINE is interfaced with an MCMC engine that enables fits of CR data in a few hours of laptop CPU; (iii) users who wish to develop their own semi-analytic model benefit from having a lot of tools and ingredients at their disposal; (iv) users who wish to provide new inputs (cross sections, etc.) can quickly see their impact on propagation. USINE, as well as CRDB, and a web interface to get solar modulation values for any time period are all part of a suite of public tools provided to ease GCR studies.

1. Introduction

In the last years, the FERMI and PAMELA measurements in the GeV-TeV energy have confirmed our general understanding of galactic cosmic rays (GCRs), but they also demonstrated the necessity to improve existing models for GCR propagation. The recent results from the AMS-02 detector on the International Space Station (since May 2011) with its high statistics—more than 10 times the whole statistic previously accumulated since the discovery of cosmic rays one century ago—also pose a new challenge for GCR data interpretation.

The transport and confinement of GCRs rely on the interactions with magnetic fields. The microscopic description of the complex interactions of the GCRs with the latter and the gas in the Galaxy requires a full MHD description. However, this approach probably has a very long way to go before becoming of routine use for GCR data interpretation. A successful approach from the early days of GCR studies is to rely on a phenomenological description of the transport, by means of a diffusion equation [13]: to solve it, a full numerical approach can be used, or some assumptions can be made in order to have semi-analytic solutions of the equation. The first approach allows to address any category of transport models (e.g., space-dependent diffusion, complex convection dependencies, etc.), and examples of public codes are GALPROP[1], DRAGON[2] and PICARD [3]. Semi-analytic approaches address only a subclass of all the propagation models. They were developed earlier and are still used nowadays. They have demonstrated their complementarity to full numerical approaches. One of their advantage in the context of high precision data and parameter space scans is that they are much faster to run.

Below, I present a quick description of the USINE propagation code along with some of its outputs. USINE was started and developed fifteen years ago to interpret nuclear data, and to test the astrophysical or DM origin of the GCR anti-protons and anti-deuterons. Some key results obtained with this (or previous versions of this) code are (i) the first automatic scan of the transport parameter space [6], (ii) the use of these parameters to show the compatibility of the models with antiprotons data [1, 5], (iii) providing benchmark propagation models (mid/med/max) for dark matter studies [3], (iv) the first use of a Markov Chain Monte Carlo in the context of GCR physics [9, 10, 11].

2. USINE

The flux of any GCR is determined by a second order differential equation both in momentum and space. The equation has to be solved for nuclei (e.g., about 90 species from H to Fe), anti-nuclei, and leptons. Any dark matter contribution is calculated from the same equation, but with a different source term, as dark matter sources are mainly located in the dark halo of the Galaxy, rather than in the thin disc (for the astrophysical sources). The various ingredients of the transport equation are (i) diffusive and convective spatial transport; (ii) source terms (diffusive shock acceleration in GCR sources, fragmentation of nuclei into light fragments or production of secondary e^+, e^-, γ, etc.); (iii) destruction terms (interactions in the ISM, decay for unstable nuclei); (iv)

---

[1] http://galprop.stanford.edu/

[2] http://www.dragonproject.org/Home.html

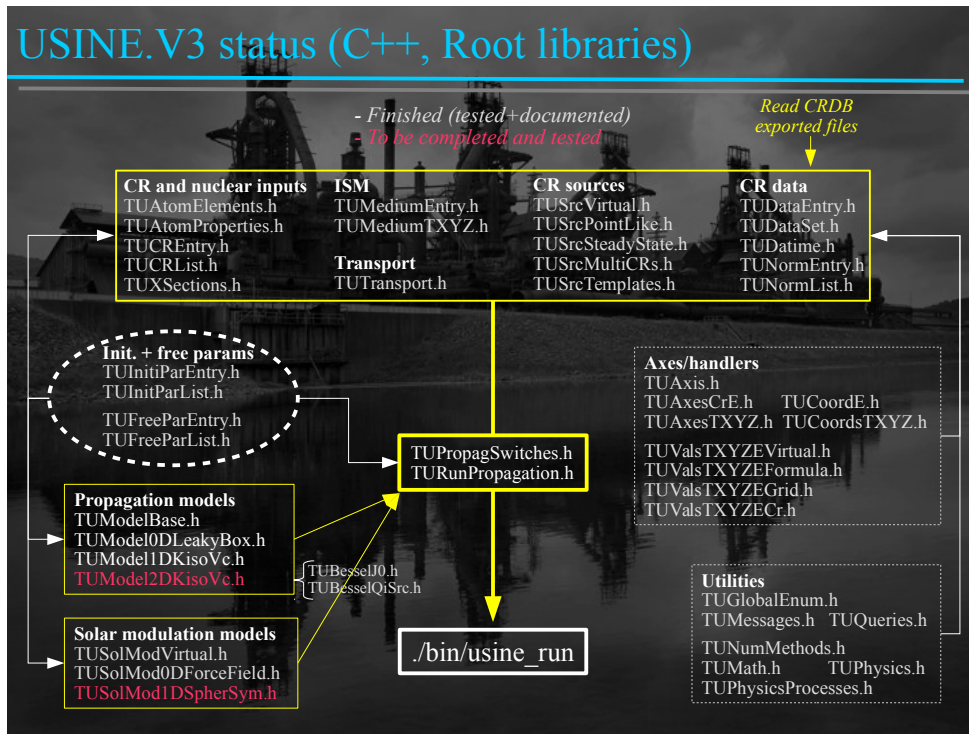[3] http://astro-staff.uibk.ac.at/~kissmrbu/Picard.html

Figure 1: Organisation of the C++ classes in USINE to match the physics ingredients and models for propagation.

energy gains and losses (energy losses in the radiation field of the Galaxy and in the ISM, energy gain by means of diffusion in momentum space).

Regardless of the propagation model selected, many inputs must be specified: list of cosmic rays and their parents, interaction cross-sections, gas distribution, etc. After their transport in the Galaxy, GCRs are also modulated by the Solar activity. Only at this stage can the model predictions be compared to data (taken from balloon, satellite, or ground-based experiments). To handle all these ingredients, the USINE code was developed and first applied to the thin disc/thick halo diffusion/convection model [6]. The first version was written in 1999 in C. The second version was written (between 2003 and 2005) C++ and interfaced with ROOT[4] for the plotting routines and with the MCMC engine GreAT [12]. The current version (V3) is based on a complete rewriting of V2, which I started in 2011, in order to provide the community with a more generic and flexible code.

The code structure is illustrated in figure 1. The C++ classes are sorted according to the ingredients needed for propagation and are organised as follows:

- GCR data read from CRDB[5] exports [8];

- CR description (charts) and interaction cross sections, ISM description, and GCR sources (spatial distribution, spectra);

---

[4] https://root.cern.ch
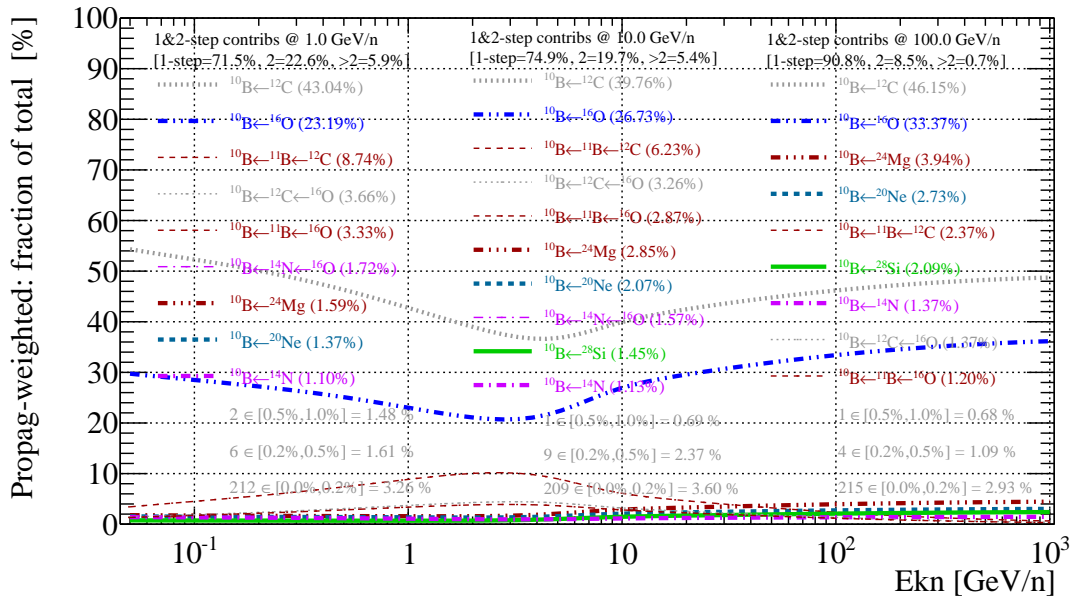[5] Cosmic-Ray DataBase: lpsc.in2p3.fr/crdb

Figure 2: Reactions whose cross section matter most for the production of ^{10}B in GCRs. The curves show the energy dependence, whereas the text gives the detail of the reactions for three energies.

- Propagation models and Solar modulation models;

- Initialisation parameters (ASCII files) to configure all the above quantities for a run;

- Handler for all free parameters (collected for all classes) and used for minimisation studies;

- A simple text user-interface to run and analyse models.

Thanks to the many properties of C++ (abstraction, inheritance, overloading,...), the above classes are designed to enable 1D to 3D (with or without time) dependencies and generic formulae using pre-defined energy keywords (R, Etot, beta, etc.) and free parameters (using the Function Parser library[6]). USINE can be viewed as a toolbox in which many inputs can be easily modified (cross-sections and data files are ASCII files), and in which the burdensome of displays is solved by the ROOT interface (ROOT but also ASCII outputs are provided). The details of the USINE code and its capabilities will be given elsewhere. Below, we provide a few examples of plots related to GCR nuclei.

3. Illustration with a few USINE plots

As shown in [7], the uncertainties on the production cross sections are already a limiting factor for the determination of the transport parameters. For any secondary nucleus (not present in the source), it is important to know which channels matter the most. Figure 2 shows as a function of the kinetic energy the most important reactions (they correspond to a balance between the source
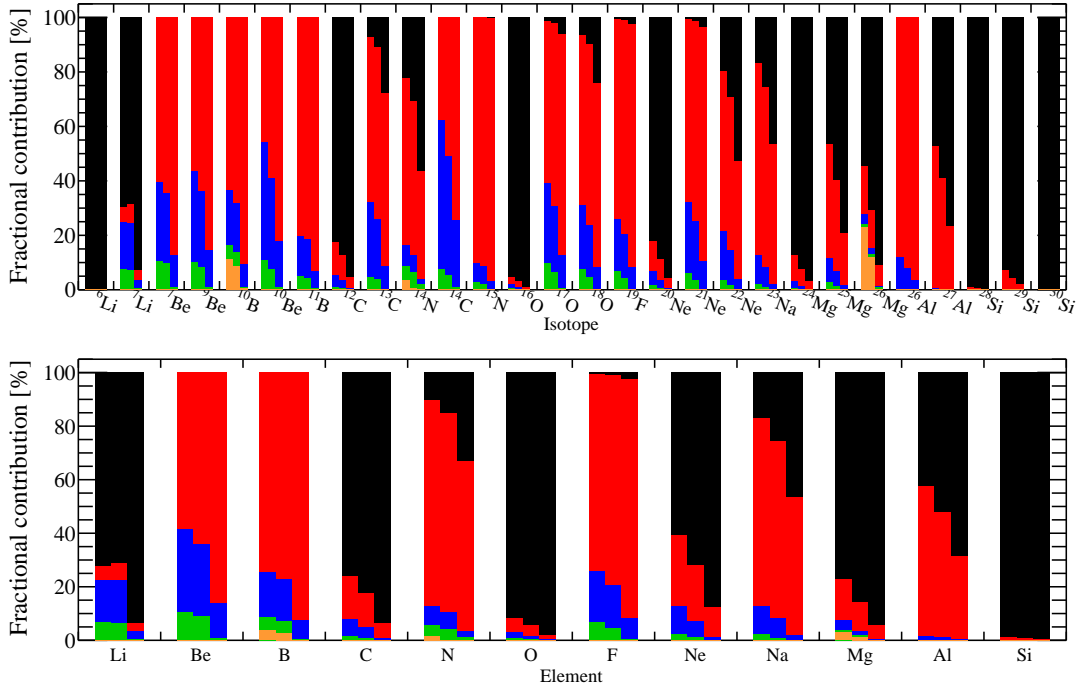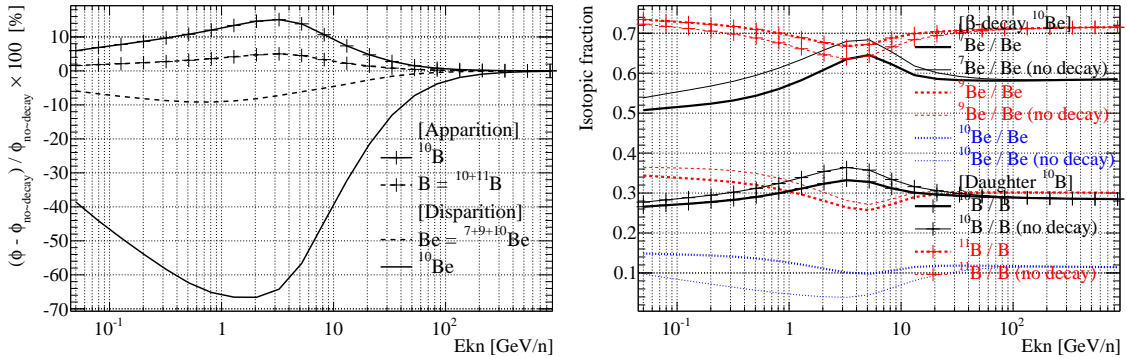
---

[6]http://warp.povusers.org/FunctionParser

Figure 3: Fractional contribution per process at three energies (1, 10, and 100 GeV/n). Contributions are primary (black), 1-step secondary (red), 2-step secondary (blue), >2-step secondary (green), decay-fed (orange).

abundances and cross-sections). If direct production of ^{10}B is the dominant channel, 2-step reactions can contribute to up to ∼ 23% at low energy. Not only these plots can be obtained for any nucleus, but also for any propagation model implemented in the USINE toolbox.

A similar but slightly different view is provided in figure 3, where the contributing fractions, in the measured flux, of the source (primary), secondary production (in one or several steps), decay-fed (from radioactive parent) are shown for isotopes (top panel) and elements (bottom panel).

As a last example, figure 3 illustrates the impact of a radioactive decay (during propagation) on the parent and daughter nuclei ^{10}Be→^{10}B (left panel) directly, or on the isotopic fractions of these nuclei (right panel).

All these plots are automatically available for any model (and combination of parameters) developed in the USINE framework. Plots for antinuclei are also provided (details on secondary and tertiary production, etc.). Thanks to a text user-interface, any combination (ratio) of nuclei and elements can also be displayed for any modulation level (in the Force-field).

4. Conclusion

USINE is a C++/ROOT code dedicated to the propagation of GCR nuclei and anti-nuclei[7]. Although USINE was developed for two zones (thin disc and thick halo) 1D and 2D models, nothing prevents it to be extended to more complex modelling. A lot of efforts were dedicated towards flexibility (the USINE classes for the propagation ingredients are designed to support 3D and time dependence), simplicity of the inputs (ASCII files for cross-sections, CR data, and initialisation and description of the free parameters of the study), and comparisons to data (import from CRDB, plots, and additional free parameters easily handled in the minimisation). In the context of AMS-02 data, and the many young researchers entering the field, USINE should prove a fast and useful tool to learn or better understand the richness of the physics of GCR nuclei.

Acknowledgements

References

[1] Donato, F., Maurin, F., Brun, P., Delahaye T. and Salati, P., PRL 102, 071301 (2009)

[2] Donato, F., Maurin, D. and Taillet, R., A&A 381, 539 (2002)

[3] Donato, F., Fornengo, N., Maurin, D., Salati, P. and Taillet, R., PRD 69, 063501 (2004)

[4] Donato, F., Fornengo, N. and Maurin, D., PRD 78, 043506 (2008)

[5] Donato, F., Maurin, D., Salati, P., Barrau, A., Boudoul G. and Taillet, R., ApJ 563, 172 (2001)

[6] Maurin, D., Donato, F., Taillet, R. and Salati, P., ApJ 555, 585 (2001)

[7] Maurin, D. , Putze, A. and Derome, L., A&A 516, 67 (2010)

[8] Maurin, D., Melot, F., Taillet, R., A&A 569, 32 (2014)

[9] Putze, A., Derome, L., Maurin, D., Perotto, L. and Taillet, R., A&A 497, 991 (2009)

[10] Putze, A., Derome, L. and Maurin, D., A&A 516, 66 (2010)

[11] Putze, A., Maurin, D. and Donato, F., A&A 526, 101 (2011)

[12] Putze, A., Derome, L., PDU, 5, 29 (2014)

[13] Strong, A. W., Moskalenko, I. V. and Ptuskin, V. S., ARNPS 57, 285 (2007)

---

[7]At the time of the conference, a link will be provided to download the USINE code.