

USINE, CRDB, and CLUMPY tools

David Maurin^{a,*}

^aLPSC, Université Grenoble Alpes, CNRS/IN2P3, 53 avenue des Martyrs, 38026 Grenoble, France

E-mail: david.maurin@lpsc.in2p3.fr

In the context of indirect dark matter detection with charged cosmic rays and γ -rays, several public tools have been developed at LPSC (Laboratoire de Physique Subatomique et Cosmologie). I present a brief overview of these tools, namely USINE for the calculation of astrophysical nuclear fluxes and associated dark matter signals, the cosmic-ray database CRDB to retrieve cosmic-ray data and associated meta-data, and CLUMPY for the calculation of γ -ray and ν signals from dark matter structures. I stress the improvements made for the various releases of these tools. I then conclude with ongoing developments and a wish list for future releases. The main emphasis is put on the USINE code and the recent results obtained with it, as it was at the core of the presentation made at this workshop.

Tools for High Energy Physics and Cosmology - TOOLS2020
2-6 November, 2020
Institut de Physique des 2 Infinis (IP2I), Lyon, France

*Speaker

1. Introduction

Antiprotons and γ -rays are among the most constraining channels for WIMP dark matter (DM) candidates in the GeV-TeV mass range [1]. The modelling of the expected signals depends on the DM distribution, but unambiguously identifying a signal depends on the confidence we have in the measured and modelled astrophysical backgrounds (which involves many uncertain input ingredients). Last but not least, the potential for a discovery or for setting limits on DM is set by the quality of the experimental data at hand. To fully benefit from the advent of a high-precision era for CR and γ -ray measurements, we must ensure that the modelling precision meets that of the data: this is a key requirement for developing or improving tools for DM indirect searches.

Below, I present the USINE code¹, developed to interpret Galactic Cosmic-Ray (GCR) data for nuclei and anti-nuclei. I then describe the cosmic-ray database CRDB², which gathers all recent and past CR data. I also give a brief description of the CLUMPY code³, which aims at calculating γ -ray and ν signals from DM structures, from local sub-halos to the extragalactic background.

2. The tools

2.1 Galactic cosmic-ray transport with USINE

The transport of charged species in the Galaxy depends on many ingredients: astrophysical sources, geometry and strength of the regular and turbulent magnetic fields, gas distribution, radiation fields pervading the Galaxy, nuclear cross sections for production and destruction of the many CR species, and energy gains and losses in the interstellar medium (ISM). Tackling the full problem from the microscopic scale remains years if not decades ahead of us, owing to the complexity of the multi-scale approach and existing coupling of the CR, gas, and magnetic field components. For all practical purposes, the interpretation of GCR data rely on the phenomenological transport equation, where it is assumed that transport can be described by a macroscopic effective diffusion coefficient—as motivated by results obtained in the quasi-linear theory [2]. With this simplification, the transport equation is a second order differential one both in space and momentum, with several source and sink terms. Several techniques and public codes exist to solve this transport equation.

- The GALPROP [3] and DRAGON [4] codes rely on finite differences [5]. The advantage is that the equations rely on simple algebra that only need to be written once, regardless of any spatial and energy dependence of the transport and other ingredients. The disadvantage is that numerical stability can never be taken for granted, and depending on the problem/geometry solved, the calculation time grows quickly. These codes should be preferred to address difficult questions in the field (e.g. impact of 3D spatial structures, anisotropic diffusion, etc.) or to have a comprehensive multi-messenger prediction (nuclei, leptons, and γ -rays).
- A second way to solve the transport equation is to rely on the formalism of stochastic differential equations. The calculation amounts to drawing random numbers for a Markovian

¹<https://lpsc.in2p3.fr/usine>

²<https://lpsc.in2p3.fr/crdb>

³<https://lpsc.in2p3.fr/clumpy>

process (to determine the CR next position). The benefit is that statistical properties of the CR distribution along the path (forward or backward) can be studied. Owing to the simplicity of this scheme and to the improved efficiency on modern computer architectures, solvers for stochastic differential equations are becoming widespread (see [6] for a public code). However, many particles need to be drawn to reach a very good statistical precision, and as of today, these methods remain very rarely used for CR calculations in practice.

- The last approach to solve the transport equation is to rely on analytical or semi-analytical methods. By simplifying the geometry and spatial distribution of the ingredients, standard analysis methods (Fourier transform, Bessel expansion, Green functions) allow to solve analytically all (or part) of the spatial and momentum derivatives. The main benefit is that the calculation is very fast and numerical instabilities are reduced. The price to pay is a limitation of the complexity of the ingredients that can be put in the model. Worse, each specific spatial dependence in the ingredients or geometry requires a new derivation of the solution: in most cases, a tractable enough solution does not even exist. The only public code following a semi-analytical approach is the USINE code [7], in which the spatial derivatives are solved analytically and the momentum coordinates numerically (with finite differences). The models currently available in USINE are the Leaky-Box [8] and two-zone diffusion models (thin disc and thick diffusive halo, in 1D [9] or 2D [10]).

USINE in a nutshell USINE is a C++ code, with many classes for the input ingredients, transport parameters, and models. It is interfaced with ROOT/CERN libraries⁴ for displays and minimisations. It also relies on the Function-Parser library⁵ to interpret formulae provided by users in ASCII files as formulae in the code. A dedicated class in USINE gathers all user-declared parameters, providing a mechanism to turn any user-defined parameter into a fitable one—this encompasses geometry, transport, source, solar modulation, and nuclear physics parameters.

USINE relies on a single ASCII initialisation file (with a USINE-specific syntax), and is based on command-line runs as follows:

```
> ./bin/usine [option] [parameters]
```

with [option] taking the possible values:

```
[-l]: simple calculation/display
[-e]: useful comparisons and extra plots
[-i]: displays the inputs (data, nuclear cross sections...)
[-t]: test of all classes (ensure code stability)
[-m]: minimisation (interfaced with Minuit)
```

Typical parameters in the command line are the initialisation file name, the directory in which output files are saved, etc. To get the full list and details on the expected parameters for each option, just type the command line without parameters.

To ensure the code stability and reliability, USINE relies on unit and integration tests—with reference test files to check that the installation is successful and ready to run. The code is on

⁴<http://root.cern.ch>

⁵<http://warp.povusers.org/FunctionParser/fparser.html>

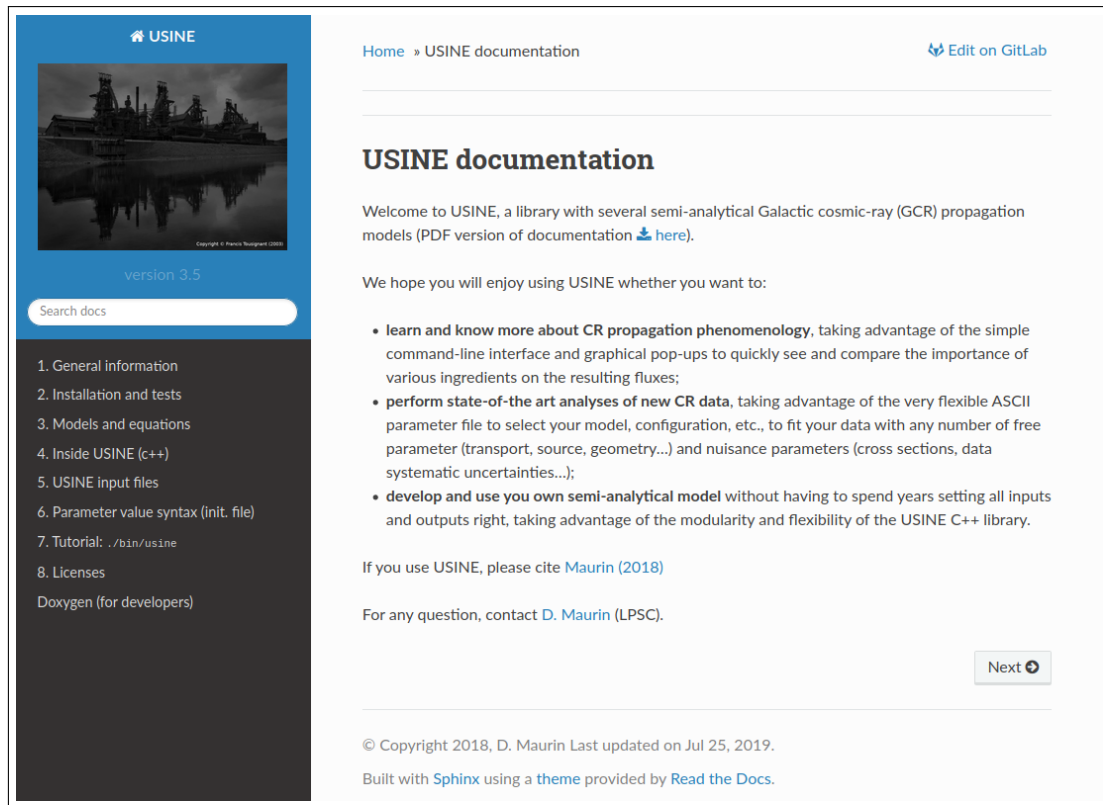


Figure 1: Snapshot of the USINE documentation webpage.

Git-Lab⁶, with a full Sphinx-based⁷ (readthedocs theme⁸) documentation⁹, as illustrated in Fig. 1. We stress that the syntax for the USINE keywords (in the initialisation file) can be highly non-trivial. For this reason, I urge the user to rely on the search box (showed in the blue box on the left of the documentation page, see Fig. 1) to find the full description of the keywords expected syntax.

USINE releases and recent results: why should you use USINE? The development of USINE started in 1999, but USINE remained until very recently a private code. There was mostly three development periods, with long hiatuses between these periods. The first unreleased version of USINE allowed to extract the CR transport parameters [10, 11], in order to calculate the anti-proton astrophysical background [12], and later-on to provide the so-called Min/Med/Max transport parameters [13], still widely used as benchmarks when setting constraints on DM candidates. In a second major update in 2007-2010 (also unreleased), we introduced for the first time in this field the use of a Markov Chain Monte Carlo analysis to determine CR transport coefficients [8]. I also implemented in USINE at that time anti-proton and anti-deuteron signals from DM, as used for instance in [14, 15]. The last development period was triggered by the release of high-precision data by the AMS collaboration¹⁰. This led to the first public release (V3.4 and 3.5) of the USINE

⁶<https://gitlab.com/dmaurin/USINE/>

⁷www.sphinx-doc.org

⁸<https://readthedocs.org>

⁹<https://dmaurin.gitlab.io/USINE>

¹⁰<https://home.cern/science/experiments/ams>

code in 2019 [7].

Given the strong limitations of the currently implemented models, namely (i) ISM and energy losses and gains restricted to be in a thin disc, (ii) astrophysical sources at best following a radial distribution in the same thin disc, and (iii) propagation of nuclei and anti-nuclei only, users may wonder why they should ever consider using USINE. There may be not many reasons left, but here they are. First, as of today, the simple models in USINE provide good fits to the recent data, though the transport parameters extracted are to some extent effective only. Simple models remain valid because CR (anti-)nuclei are averaged over large volumes [16, 17], meaning that the space-time discreteness of the source and the small scale structures of the ISM are not crucial. However, whereas many effects would lead to a departure to this standard approach [18], there is no clear indication yet on what to consider next, so that pushing these standard models to their limit still brings interesting informations. Second, USINE is way much faster than other codes, and it remains as such an excellent test-bed for a few new ideas (see next paragraph). Third, if you find a new set of analytical solutions, implementing them in the USINE library would be the obvious choice to benefit from all the I/O and fit niceties in USINE. Last, semi-analytical models remain the preferred framework for DM indirect detection studies, owing to their relative simplicity, and this will probably remain the case in the coming years.

To illustrate that USINE still enables to address state-of-the-art science questions, here is a brief list of the recent results we obtained: (i) indication of a high-energy break in the diffusion coefficient [19], (ii) list of nuclear cross-sections to improve in order to reach a modelling precision on par with the AMS-02 data 3% precision [20], (iii) improved methodology to analyse and fit without biases high precision data [21], (iv) new benchmark transport parametrisations SLIM/BIG/QUAINT [22, 23], (v) consistency of antiproton data with a pure astrophysical origin [24], constraints on the diffusive halo size [25]. Several more studies are on their way, including an update of the Min/Med/Max benchmark parameters and an update of DM constraints from anti-protons.

Future developments and wish list There are not many development directions left with USINE. An important one, in the context of this workshop, is to plug back the module to calculate DM anti-nuclei. Indeed, this module was not upgraded between the unreleased (2008) and public (2019) versions, but it will be part of the next USINE release: contact me if you are interested in using this module now, as it already exists in a development branch (it makes use of PPC4DMID final state spectra [26]). Other improvements for the next release are (i) PyUSINE, a python library to use USINE in notebooks, (ii) the extension of nuclear ingredients to run USINE for CR species $Z > 30$, and (iii) a few other surprises! My initial wish list included electrons and positrons, implemented from the pinching method proposed by Mathieu Boudaud in [27], but sadly, our dear friend and colleague Mathieu passed away in January 2020; it is not clear yet whether I will be able to implement this calculation in the next release. A next step for USINE would be to implement a numerical model as in DRAGON and GALPROP, but unless someone is very excited about it (then please contact me!), this will probably never happen. The next USINE release is expected in 2021, but that aside, no other major upgrade is planned. However, maintenance of the code will be ensured for several more years.

Cosmic-Ray DataBase (CRDB)

Main developers: D. Maurin¹ & F. Melot¹ (logo by H. Dembinski¹)
 Contributors: H. Dembinski, J. Gonzalez, I. Mariş, R. Taillet
 Partners: [KCDC project](#)
 Publications (please cite if you use CRDB):
 • v2.1: Maurin, Melot, and Taillet, *A&A* 569, A32 (2014)
 • v4.0: Maurin, Dembinski, Gonzalez, Mariş, and Melot (arXiv:2005.14663)
 [Acknowledgements / Contact us / Funding support]

DB status
 Current version: v4.0 (May 2020)
 → v4.1 in dev. (include KCDC data)
 Code last changes: 30/07/2020
 Data: 116 expts, 601 sub-expts, 389 publis
 [ChangeLog / Latest data / View traffic]

Data and user interfaces
 This database is a compilation of experimental cosmic-ray data. If you spot some errors or want to contribute please [contact us](#).

- CRDB contains cosmic-ray data from 10^6 eV (Voyager) to 10^{21} eV (Pierre Auger Observatory) for the various quantities:
 - *Leptons*: e^- , e^+ , e^+e^- , $e^-/(e^+e^-)$, and e^+/e^-
 - *Nuclei*: isotopes, elements (from H to U), and groups of elements
 - *Anti-nuclei*: anti-protons, limits on anti-deuterons and anti-nuclei
- Data and meta-data (data about data) can be accessed in contextualised user interfaces or directly retrieved.
 - User interfaces**
 - **Data extraction**: plot, save, and export user-selected CR quantities
 - **Experiments/Data**: sorted lists of experiments, publications, and their data
 - **REST access**: retrieve CRDB data from script or python notebook (REST interface)
 - **Solar modulation**: Force-Field modulation level time series (and REST access)
 - **Submit data**: submit data and their associated meta-data
 - **Export all native data and references in one go** (this may take some time)
 - [USINE](#), [GALPROP](#), or [csv](#) format
 - [ADS bibtext](#) entries and [Latex cite](#) (sorted by sub-experiment)

Behind the scene

- Architecture: [LAMP](#) solution (Linux OS, Apache HTTP server, MySQL database, PHP Hypertext PreProcessor) hosted at LPSC on a virtual server
- Web pages: [PHP](#) language, [AJAX](#) (Asynchronous JavaScript and XML), and third-party libraries for sorting and displays ([jquery](#), [jquery-ui](#), [jquery.cluetip](#), [table-sorter](#))
- Scripts and codes: [c++](#) and [ROOT CERN library](#) for plots, cron job scheduler for meta-data and modulation data updates
- [Rest](#) interfaces enabled
- Data extraction: a lot of sweat, patience, and the help of [DataThief!](#)

Figure 2: Snapshot of the CRDB webpage

2.2 CRDB: the cosmic-ray database

CR data are the backbones of CR analyses, but in the past, scientists have all too often wasted time collecting again and again the same data for their analyses. To remedy that, gathering data in a single database is the obvious way to go. The first public distribution of these data started from the GALPROP¹¹ team as a simple and single ASCII file [28]. As part of the development of the USINE library, I started to push for the development of a database (from data I had gathered independently). CRDB came online in 2013 [29], including a contextualised presentation of the data and experiments, ASCII file exports, and online plot generation. The Italian Space Agency (ASI) soon after developed its own CR database, originally to host ASI-supported experimental data [30], and the initial design and look of CRDB-ASI¹² was strongly influenced by ours.

Since the first release in 2013 [29], about a quarter of a million queries were made. CRDB contains 44 638 data points from more than 400 publications and 100 experiments. It is hosted by the LPSC laboratory website, running on a LAMP open source solution (Linux operating system, Apache HTTP server, MySQL database, and PHP hypertext pre-processor language) and relying on AJAX (Asynchronous JavaScript and XML) web development techniques. Accessing the data can be achieved either via the web pages (via jquery libraries), or via a REST (representational state transfer) interface; the former allows to access the fully contextualised data and meta-data, while the latter allows users to retrieve the data directly from command lines in their own scripts or codes.

Figure 2 shows a snapshot of the CRDB web page, corresponding to CRDB V4.0 released just before this summer [31]. The several tabs correspond to different ways of accessing the (meta-)data

¹¹<http://galprop.stanford.edu>

¹²<https://tools.ssd.cas.ac.cn/CosmicRays>

or CR-related quantities. Among the main improvements made in this release were: multi-species and time-series data selection, extension of the data from MeV to ZeV, improvement of the REST interface (a python notebook example with `curl` or `get` is provided), simpler format for anyone to submit data to CRDB, etc. Of special interest for this workshop are the inclusion of data upper-limits for anti-deuterons and anti-elements, and the extraction of solar modulation levels for any time period, based on neutron monitor data [32].

For the next release, we plan to gather more ultra-high energy data, in collaboration with the KCDC team¹³ [33], and also possibly to include more quantities (neutrino data and dipole anisotropy). And of course, we will continue adding data from the AMS-02, CALET, DAMPE, Nucleon, and other experiments as they are published.

2.3 CLUMPY

We conclude here very briefly with the CLUMPY code. It has been developed to estimate DM-induced fluxes of γ -rays and ν in a large variety of objects (and user-defined configurations) to provide a unified and comprehensive calculation at both the Galactic and extragalactic scales. The first release dates back to 2011 [34] and focused on the optimised calculation of the astrophysical J-factors—integration of the DM density squared (resp. density) along the line of sight for DM annihilation (resp. decay)—for local objects, including sub-halos, dwarf spheroidal galaxies, the Milky Way, and nearby galaxy clusters. The second release in 2016 [35] included the full calculation of γ -ray and ν spectra (computed from PPPC4DMID), and also a Jeans analysis module to compute the DM profile of dwarf spheroidal galaxies from their kinematic data. The third release in 2019 [36] extended the calculation for any object at any cosmological distance, including the mean extragalactic fluxes in a flat Λ CDM universe. So far, beside full numerical simulations from which CLUMPY borrows some end results (to use as inputs), CLUMPY remains the only public code for this task. It has been used in many publications by experimentalists, phenomenologists, and astrophysicists.

CLUMPY is a C/C++ code (but no classes), interfaced with ROOT/CERN, CFITSIO¹⁴, GSL¹⁵, and HEALPIX [37] libraries. It is run by a single command line with options, and self-explanatory messages, with instructions on how to run the various modules if no option is given:

```
> ./bin/clumpy [options]
with [options] as follows:
  -g[index]: galactic calculations
  -h[index]: isolated/list of halos
  -s[index]: run on statistical files
  -o[index]: skymap file manipulation
  -z: gamma and neutrino spectra
  -f: convert J-factor skymap to flux map
  -e[index]: extragalactic module
  -e[index] -D: generates initialisation file
```

¹³<https://kcdc.ikp.kit.edu/>

¹⁴<http://heasarc.gsfc.nasa.gov/fitsio>

¹⁵<http://www.gnu.org/software/gsl>

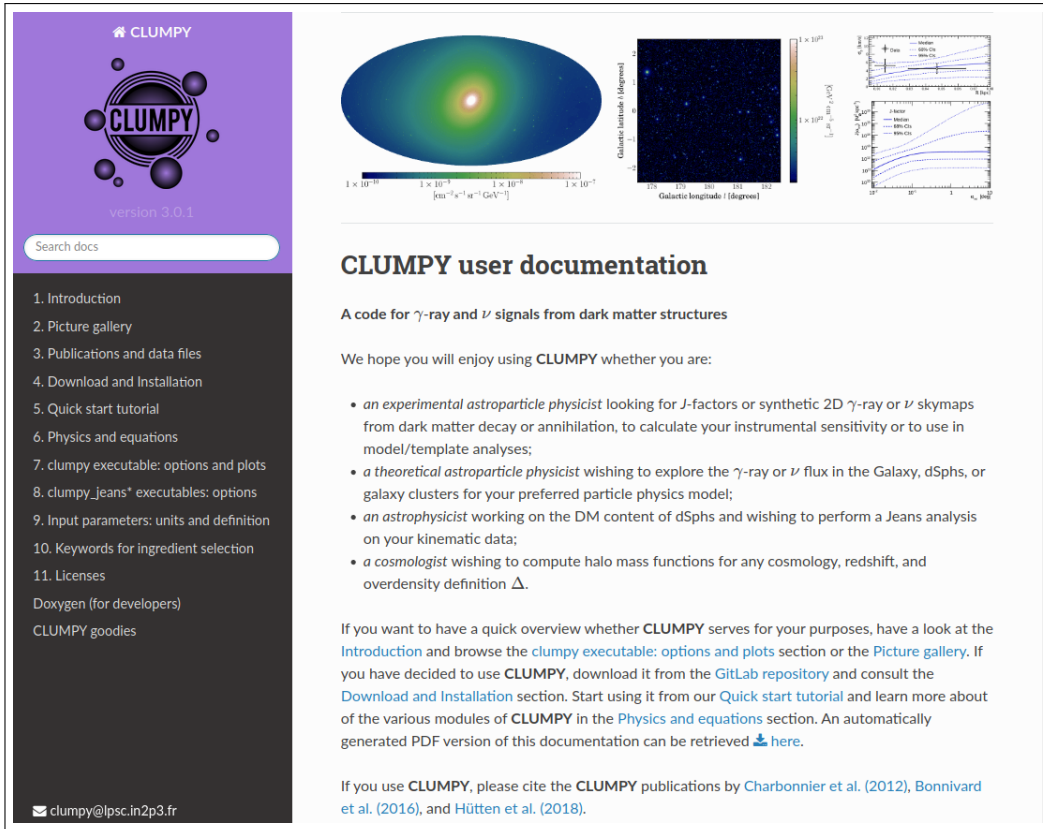


Figure 3: Snapshot of the CLUMPY webpage

As CLUMPY and USINE share one developer, the solutions implemented for testing and documentation are similar (see Fig. 3), with many examples provided.

CLUMPY is not in a development frenzy right now, all the team members being busy with other projects. Nevertheless, a wish list for a future release would be to include a python interface, to enable the drawing of extragalactic clusters to include synthetic skymaps of all DM objects in one go, and to include velocity-dependent particle physics model for all calculations. There are also other funky possibilities in the pipeline if ever we find the time. . .

3. Conclusions

I have presented several tools developed at LPSC in the (sometimes loose) context of indirect DM detection. The two codes and database presented are at a mature stage, benefiting from many years of development, and one or two decades of expertise in the field. The development, maintenance, and support of these tools takes a significant amount of time, and I thank all the users who acknowledge this effort in their publications/presentations; I encourage those who don't, to make the extra step.

To conclude, I stress that some members of these projects were initially mere PhD students or post-docs, having an interest in expanding the tools capabilities and also ready to spend time doing it. Do not hesitate to contact me if this is what you would like to try!

References

- [1] J. Conrad and O. Reimer, *Indirect dark matter searches in gamma and cosmic rays*, *Nature Physics* **13** (2017) 224.
- [2] V.S. Berezhinskii, S.V. Bulanov, V.A. Dogiel and V.S. Ptuskin, *Astrophysics of cosmic rays* (1990).
- [3] A.W. Strong and I.V. Moskalenko, *Propagation of Cosmic-Ray Nucleons in the Galaxy*, *ApJ* **509** (1998) 212.
- [4] C. Evoli, D. Gaggero, D. Grasso and L. Maccione, *Cosmic ray nuclei, antiprotons and gamma rays in the galaxy: a new diffusion model*, *Journal of Cosmology and Astro-Particle Physics* **10** (2008) 18.
- [5] J. Crank and P. Nicolson, *A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type*, *Proceedings of the Cambridge Philosophical Society* **43** (1947) 50.
- [6] A. Kopp, I. Büsching, R.D. Strauss and M.S. Potgieter, *A stochastic differential equation code for multidimensional Fokker-Planck type problems*, *Comput. Phys. Commun.* **183** (2012) 530.
- [7] D. Maurin, *USINE: Semi-analytical models for Galactic cosmic-ray propagation*, *Computer Physics Communications* **247** (2020) 106942.
- [8] A. Putze, L. Derome, D. Maurin, L. Perotto and R. Taillet, *A Markov Chain Monte Carlo technique to sample transport and source parameters of Galactic cosmic rays. I. Method and results for the Leaky-Box model*, *A&A* **497** (2009) 991.
- [9] A. Putze, L. Derome and D. Maurin, *A Markov Chain Monte Carlo technique to sample transport and source parameters of Galactic cosmic rays. II. Results for the diffusion model combining B/C and radioactive nuclei*, *A&A* **516** (2010) A66.
- [10] D. Maurin, F. Donato, R. Taillet and P. Salati, *Cosmic Rays below $Z=30$ in a Diffusion Model: New Constraints on Propagation Parameters*, *ApJ* **555** (2001) 585.
- [11] F. Donato, D. Maurin and R. Taillet, *beta -radioactive cosmic rays in a diffusion model: Test for a local bubble?*, *A&A* **381** (2002) 539.
- [12] F. Donato, D. Maurin, P. Salati, A. Barrau, G. Boudoul and R. Taillet, *Antiprotons from Spallations of Cosmic Rays on Interstellar Matter*, *ApJ* **563** (2001) 172.
- [13] F. Donato, N. Fornengo, D. Maurin, P. Salati and R. Taillet, *Antiprotons in cosmic rays from neutralino annihilation*, *Phys. Rev. D* **69** (2004) 063501.
- [14] F. Donato, N. Fornengo and D. Maurin, *Antideuteron fluxes from dark matter annihilation in diffusion models*, *Phys. Rev. D* **78** (2008) 043506.

- [15] F. Donato, D. Maurin, P. Brun, T. Delahaye and P. Salati, *Constraints on WIMP Dark Matter from the High Energy PAMELA \bar{p}/p Data*, *Physical Review Letters* **102** (2009) 071301.
- [16] R. Taillet and D. Maurin, *Spatial origin of Galactic cosmic rays in diffusion models. I. Standard sources in the Galactic disk*, *A&A* **402** (2003) 971 [[astro-ph/0212112](#)].
- [17] D. Maurin and R. Taillet, *Spatial origin of Galactic cosmic rays in diffusion models. II. Exotic primary cosmic rays*, *A&A* **404** (2003) 949 [[astro-ph/0212113](#)].
- [18] S. Gabici, C. Evoli, D. Gaggero, P. Lipari, P. Mertsch, E. Orlando et al., *The origin of Galactic cosmic rays: challenges to the standard paradigm*, *arXiv e-prints* (2019) arXiv:1903.11584.
- [19] Y. Génolini, P.D. Serpico, M. Boudaud, S. Caroff, V. Poulin, L. Derome et al., *Indications for a High-Rigidity Break in the Cosmic-Ray Diffusion Coefficient*, *Phys. Rev. Lett.* **119** (2017) 241101.
- [20] Y. Génolini, D. Maurin, I.V. Moskalenko and M. Unger, *Current status and desired precision of the isotopic production cross sections relevant to astrophysics of cosmic rays: Li, Be, B, C, and N*, *Phys. Rev. C* **98** (2018) 034611.
- [21] L. Derome, D. Maurin, P. Salati, M. Boudaud, Y. Génolini and P. Kunzé, *Fitting B/C cosmic-ray data in the AMS-02 era: a cookbook. Model numerical precision, data covariance matrix of errors, cross-section nuisance parameters, and mock data*, *A&A* **627** (2019) A158.
- [22] Y. Génolini, M. Boudaud, P.I. Batista, S. Caroff, L. Derome, J. Lavalley et al., *Cosmic-ray transport from AMS-02 boron to carbon ratio data: Benchmark models and interpretation*, *Phys. Rev. D* **99** (2019) 123028.
- [23] N. Weinrich, Y. Génolini, M. Boudaud, L. Derome and D. Maurin, *Combined analysis of AMS-02 (Li,Be,B)/C, N/O, ^3He , and ^4He data*, *A&A* **639** (2020) A131.
- [24] M. Boudaud, Y. Génolini, L. Derome, J. Lavalley, D. Maurin, P. Salati et al., *AMS-02 antiprotons' consistency with a secondary astrophysical origin*, *Physical Review Research* **2** (2020) 023022.
- [25] N. Weinrich, M. Boudaud, L. Derome, Y. Génolini, J. Lavalley, D. Maurin et al., *Galactic halo size in the light of recent AMS-02 data*, *A&A* **639** (2020) A74.
- [26] M. Cirelli, G. Corcella, A. Hektor, G. Hütsi, M. Kadastik, P. Panci et al., *PPPC 4 DM ID: a poor particle physicist cookbook for dark matter indirect detection*, *J. Cosmology Astropart. Phys.* **3** (2011) 051.
- [27] M. Boudaud, E.F. Bueno, S. Caroff, Y. Génolini, V. Poulin, V. Poireau et al., *The pinching method for Galactic cosmic ray positrons: Implications in the light of precision measurements*, *A&A* **605** (2017) A17.

- [28] A.W. Strong and I.V. Moskalenko, *A Galactic Cosmic-Ray Database*, *ArXiv 0907.0565* (2009) .
- [29] D. Maurin, F. Melot and R. Taillet, *A database of charged cosmic rays*, *A&A* **569** (2014) A32.
- [30] V. Di Felice, C. Pizzolotto, D. D'Urso, S. Dari, D. Navarra, R. Primavera et al., *Looking for cosmic ray data? The ASI Cosmic Ray Database*, in *35th International Cosmic Ray Conference (ICRC2017)*, vol. 301 of *International Cosmic Ray Conference*, p. 1073, jan, 2017.
- [31] D. Maurin, H.P. Dembinski, J. Gonzalez, I.C. Mariş and F. Melot, *Cosmic-Ray Database Update: Ultra-High Energy, Ultra-Heavy, and Antinuclei Cosmic-Ray Data (CRDB v4.0)*, *Universe* **6** (2020) 102.
- [32] A. Ghelfi, D. Maurin, A. Cheminet, L. Derome, G. Hubert and F. Melot, *Neutron monitors and muon detectors for solar modulation studies: 2. phi time series*, *AdSR* **60** (2017) 833.
- [33] A. Haungs, D. Kang, S. Schoo, D. Wochele, J. Wochele, W.D. Apel et al., *The KASCADE Cosmic-ray Data Centre KCDC: granting open access to astroparticle physics research data*, *European Physical Journal C* **78** (2018) 741.
- [34] A. Charbonnier, C. Combet and D. Maurin, *CLUMPY: A code for γ -ray signals from dark matter structures*, *Comput. Phys. Commun.* **183** (2012) 656.
- [35] V. Bonnivard, M. Hütten, E. Nezri, A. Charbonnier, C. Combet and D. Maurin, *CLUMPY: Jeans analysis, γ -ray and ν fluxes from dark matter (sub-)structures*, *Computer Physics Communications* **200** (2016) 336.
- [36] M. Hütten, C. Combet and D. Maurin, *CLUMPY v3: γ -ray and ν signals from dark matter at all scales*, *Computer Physics Communications* **235** (2019) 336.
- [37] K.M. Górski, E. Hivon, A.J. Banday, B.D. Wandelt, F.K. Hansen, M. Reinecke et al., *HEALPix: A Framework for High-Resolution Discretization and Fast Analysis of Data Distributed on the Sphere*, *ApJ* **622** (2005) 759.