(article starts on next page)

CRPropa 3.2: a public framework for high-energy astroparticle simulations

S. Aerdker,[1,2] **R. Alves Batista,**[3] **J. Becker Tjus,**[1,2,4] **J. Dörner,**[1,2] **A. Dundovic,**[5,6]
B. Eichmann,[1,2] **A. Frie,**[1,2] **C. Heiter,**[7,8] **M. Hoerbe,**[1,2,9] **K.-H. Kampert,**[10,2]
L. Merten,[1,2,*] **G. Müller,**[7] **P. Reichherzer,**[9] **S. Rossoni,**[11] **A. Saveliev,**[12,13]
L. Schlegel,[1,2] **G. Sigl,**[11] **A. van Vliet**[14] **and T. Winchen**[8]

[1]*Theoretical Physics IV, Faculty for Physics & Astronomy, Ruhr University Bochum, Germany*

[2]*Ruhr Astroparticle and Plasma Physics Center (RAPP Center), Germany*

[3]*Instituto de Física Teórica UAM-CSIC, Madrid, Spain*

[4]*Department of Space, Earth and Environment, Chalmers University of Technology, Gothenburg, Sweden*

[5]*Gran Sasso Science Institute (GSSI), L'Aquila, Italy*

[6]*Institute for Cosmology and Philosophy of Nature (ICPN), Križevci, Croatia*

[7]*RWTH Aachen University, III. Physikalisches Institut A, Aachen, Germany*

[8]*Max Planck Institute for Radio Astronomy, Bonn, Germany*

[9]*Department of Physics, University of Oxford, Oxford, United Kingdom*

[10]*Bergische Universität Wuppertal, Department of Physics, Wuppertal, Germany*

[11]*II. Institute for Theoretical Physics, Universität Hamburg, Hamburg, Germany*

[12]*Immanuel Kant Baltic Federal University, Institute of Phys., Math. and Inf. Techn., Kaliningrad, Russia*

[13]*Lomonosov Moscow State University, Faculty of Comp. Mathematics and Cybernetics, Moscow, Russia*

[14]*Department of Physics, Khalifa University, Abu Dhabi, United Arab Emirates*

E-mail: crpropa@desy.de

CRPropa is a Monte Carlo framework for simulating the propagation of (ultra-) high-energy particles in the Universe, including cosmic rays, gamma rays, electrons, and neutrinos. It covers energies from ZeV down to GeV for gamma rays and electrons, and TeV for cosmic rays and neutrinos, supporting various astrophysical environments such as the surroundings of astrophysical sources, galactic, and extragalactic environments. The newest version, CRPropa 3.2, represents a significant leap forward towards a universal multi-messenger framework, opening up the possibility for many more astrophysical applications. This includes extensions to simulate cosmic-ray acceleration and particle interactions within astrophysical source environments, a full Monte Carlo treatment of electromagnetic cascades, improved ensemble-averaged Galactic propagation, significant performance improvements for cosmic-ray tracking through magnetic fields, and a user-friendly implementation of custom photon fields, among many more enhancements. This contribution will give an overview of the new features and present several applications to cosmic-ray and gamma-ray propagation.

*Speaker

1. Introduction

In the light of recent multi-wavelength and sometimes even multi-messenger observation of astrophysical objects a detailed modelling of transport and interaction of cosmic rays and their secondary particles is vital. Since its last major release [3] the simulation framework CRPropa has made progress to become an all particle (almost) all energies modelling tool. Since version 3.1 [9] CRPropa can not only propagate cosmic-ray `Candidates` by solving their equation of motion but also solve an ensemble-averaged Fokker Planck transport equation based on a stochastic differential equation (SDE) solver.

Ever since, CRPropa's modular structure makes improving and extending the existing software easy. In these proceedings we report about the latest developments of the code that have been implemented since the last major release version 3.2 and the publication of the corresponding paper [6]. This work is structured as follows: In section 2 technical improvements, such as `tagOrigin` and *plugins*, are introduced. New physics use cases and example simulations are discussed in section 3. And outlook into the future of CRPropa is given in section 4.

2. Technical Improvements

2.1 Improved Propagation Algorithms

CRPropa provides three different propagation algorithms to solve the equation of motion `SimplePropagation`, `PropagationBP`, and `PropagationCK`. For comparisons of the latter two, the reader is referred to [6, 10] and the documentation[1]. Both solvers are implemented adaptively by default. This means the module optimises the propagation step Δs within a user-defined range $\Delta s \in [s_{\min}, s_{\max}]$ to match a given precision. Usually this gives an overall performance improvement compared to non-adaptive algorithms because the integration step will be increased, e.g., in regions of low magnetic field strength. When transport properties such as the diffusion coefficients should be studied it is, however, often advantageous to use a fixed propagation step in the simulation, making the usual error estimation and step optimisation obsolete. Now the corresponding computations are skipped when `PropagationBP` and `PropagationCK` are initialised with the values for the minimal and maximal propagation step $s_{\min} = s_{\max}$, leading to performance improvements of approximately a factor 2-3.[2] The computation time of the adaptive versions remains the same, as well as all simulation results.

2.2 Tracking of Secondary Production

Although CRPropa tracks individual particles it was up to now not possible to differentiate particles due to their production mechanism. With the introduction of a new `Candidate` property called `tagOrigin` the producing interaction is now recorded for all particles. This allows to separately analyse observables, such as secondary spectra, by their origin. This will help to better understand the dominating production mechanism. Furthermore, this feature can be used to identify

---

[1]https://crpropa.github.io/CRPropa3/pages/example_notebooks/propagation_comparison/Propagation_Comparison_CK_BP.html

[2]See (https://github.com/CRPropa/CRPropa3/pull/424) for details on the performed tests.

potentially irrelevant interactions. When these are removed from the model, significant simulation speed-ups are expected.

By default `tagOrigin`s are based on the kind of interaction, e.g. photo-pion production and inverse Compton scattering, but do not differentiate between different photon target fields. This behaviour can be changed by providing individual tags for each module instance before adding it to the `ModuleList`. Furthermore, the new feature can be used to differentiate between different source classes in post-processing as it can be modified by `Sources`, too.

Recording the `tagOrigin` leads to a small performance overhead of approximately three percent for simulations that are dominated by interactions.[3] Note, that old post-processing scripts might break since the `tagOrigin` is now included in some of the predefined output formats.

2.3 Extended Density Grids

In version 3.2 the new property of `Density` fields has been introduced. It was implemented as a first step towards nuclei-nuclei interactions, where they will serve as the description of the target fields. Here, we discuss two upgrades to this feature. Firstly, density grids are now restricted to their original volume defined by the origin and the number of and distance between grid points. Outside of this volume the fields are vanishing. The behaviour can be changed by the user to use reflective or periodic continuation as often done for turbulent magnetic fields. Note, that in general these new features allows to restrict other fields to their original volume, too.

Secondly, a new source class `SourceMassDistribution` was implemented. It allows to sample source positions from any given `Density` class, e.g., sampling cosmic-ray sources that follow one of the predefined gas distributions of the Galaxy.

2.4 Customisation with Plugins

CRPropa plugins are a way to provide extensions to CRPropa that are not planned to become part of the core CRPropa software package. Similar to the main CRPropa software plugins are written in C++ and wrapped for python access with SWIG. After compilation they can be imported as separate python modules and be freely combined with the CRPropa software. The documentation includes now an updated example, which can be used as a starting point for user developments. Although plug-ins are not maintained by the CRPropa developers we encourage users to share their developments with the community.

Plugins that have been shared are linked on the CRPropa. Some of them are explained exemplary in the following:

Field Line Integration   The `FieldLineIntegrator` module can be used to analyse magnetic field models that are implemented in CRPropa. The module should be used instead of a conventional propagation module, e.g., `PropagationBP`, and will integrate the magnetic field line passing through the source position of the candidate. A simulation set-up to integrate magnetic field lines must not include interaction modules, but can include boundary conditions as the maximum trajectory length. In post-processing candidate positions can be used to, e.g., visualise a magnetic field model, calculate diffusion coefficients in non-homogeneous background fields, or field line random walk coefficients.

---

[3]See (`https://github.com/CRPropa/CRPropa3/pull/407`) for details.

Plasma Instabilities   Electron-positron pairs propagating in the intergalactic medium produce plasma instabilities that can affect their propagation. The exact role played by this effect is not known, but it can be sizeable and influence studies that depend on accurate modelling of electromagnetic cascades, such as constraints on intergalactic magnetic fields [4]. The `GRPlInst` plugin[4] was developed to take this into account. It interfaces with CRPropa and effectively adds a continuous energy-loss term to electrons and positrons, depending on the type of plasma instability and properties of the intergalactic medium [5].

Lorentz Invariance Violation   The recent interest in probing quantum gravity (QG) via multimessenger astronomy [1] was the main motivation behind the creation of the `LIVPropa` plugin [11]. It incorporates the main phenomenological consequence of QG-based Standard Model extensions, namely the modification of the dispersion relations of different particles. Based on that, the propagation parameters are recalculated and can then be used in CRPRopa to evaluate the impact of these modifications on the observations. So far only pair production for photons with LIV is available, but we are planning to extend it to other reactions, such as inverse Compton scattering, soon.

2.5 Customisation with Photon Fields

Although custom photon fields are technically available since version 3.2 a consistent implementation of a user-defined target photon fields was quite complicated. To make the procedure of adding a custom photon field clear, we first elaborate how interactions with photon fields are technically implemented in CRPropa.

Since an on-the-fly derivation of all relevant interaction kinematics would be too time consuming, CRPropa uses pre-calculated tables for parameters, such as the interaction rate, which are already integrated over the target photon fields. To generate these tables a collection of python modules has been developed that can be found in a separate git repository[5]. In addition, some CRPropa modules require direct access to photon field properties such as the number density. This means any new photon field has to be implemented twice: On the one hand, as a class of the CRPropa3-Data repository to allow for the generation of the tabulated interaction rates and on the other hand as CRPropa class to allow for direct access.

However, the actual generation of rates and secondary yields for all available interactions can be done now with only a few lines of python code. This includes routines to copy the generated tables to the correct CRPropa directories, so they can be used in the same way as the default photon fields. An updated example explaining the consistent generation of custom photon target step-by-step is now included in the documentation[6]. Figure 1 shows the spectral energy distributions of two custom photon field examples in comparison with the cosmic microwave background.

---

[4]https://github.com/rafaelab/grplinst

[5]https://github.com/CRPropa/CRPropa3-data

[6]https://crpropa.github.io/CRPropa3/pages/example_notebooks/custom_photonfield/custom-photon-field.html
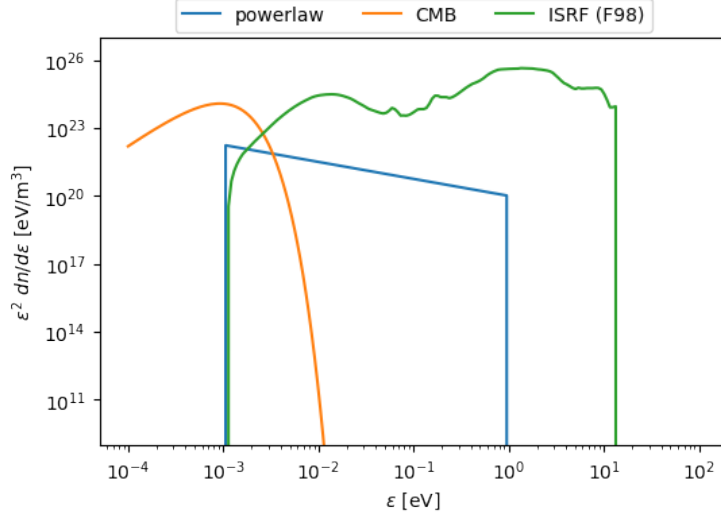
Figure 1: Examples of two custom photon fields in comparison with the cosmic microwave background.

3. New Physics Use Cases

In this section, some examples of CRPropa simulations are presented to highlight different capabilities of the software.

3.1 Diffuse Shock Acceleration with Stochastic Differential Equations

Diffusive Shock Acceleration (DSA) can be simulated using the `DiffusionSDE` module based on Stochastic Differential Equations (SDEs). In the ensemble-averaged picture, DSA is described by adiabatic energy change. Using the SDE approach, change in momentum p is given by

$$\mathrm{d}p = -\frac{p}{3}\nabla \cdot \vec{u}\,\mathrm{d}t, \tag{1}$$

with advection field $\vec{u}$. From eq. (1) follows, that ideal shocks cannot be simulated. A planar one-dimensional shock with finite width can be approximated, e.g. by $u(x) \propto \tanh{(x/L_{\mathrm{sh}})}$. In order to simulate an ideal shock, the shock width L_{sh} has to be small compared to the diffusion step length, but pseudo-particles also need to encounter the region of changing advection field to experience sufficient acceleration. For a detailed discussion on the choice of time step and shock width we refer to [2]. Injecting mono-energetic pseudo-particles of energy E_0 at the shock $x = 0$ and considering a stationary flux of particles to arrive at the shock, the time evolution of the spectrum can be approximated using the `TimeEvolutionObserver` (see [2, 9] for details).

Figure 2 shows the time evolution of the energy-spectrum at the downstream side of the shock and the number density in the acceleration region. Energy-dependent diffusion, $\kappa(E) = \kappa_0(E/E_0)^{0.5}$ is considered. The compression ratio of the shock is $q = 4$, leading to a spectral slope $s = -2$ for an ideal shock. To increase statistics at high energies a new `CandidateSplitting` module was used, which will become available in a future release. Candidates are split in two copies once they cross energy-boundaries depending on the expected spectral slope.
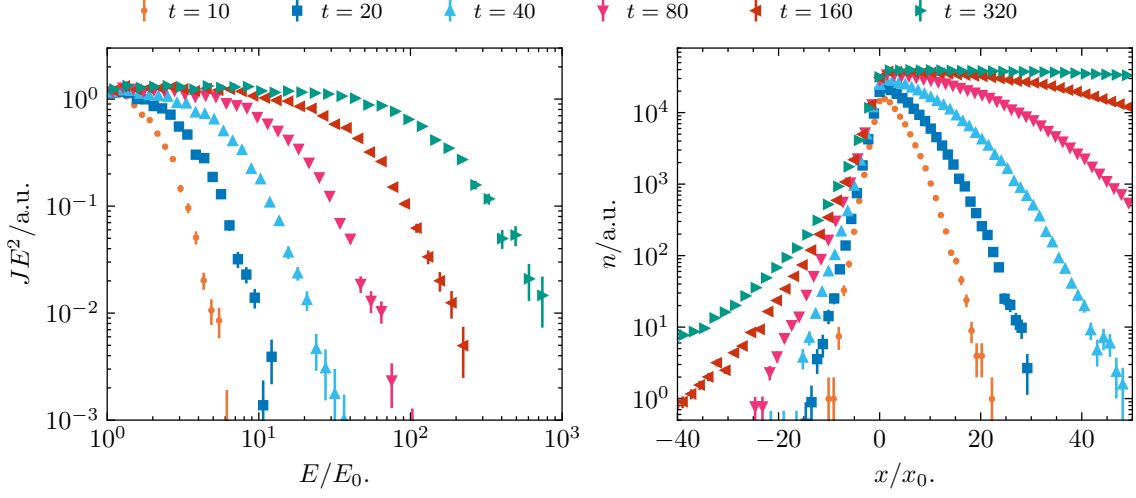
Figure 2: Time evolution of energy spectrum at a planar one-dimensional shock (left) and number density in acceleration region (right). $L_{sh} = 0.02$, upstream speed $u_1 = 1$, compression $q = 4$ all units normalised $x_0 = v_0 t_0 = 1$. Diffusion is energy-dependent, $\kappa(E) = \kappa_0 (E/E_0)^{0.5}$ with $\kappa_0 = 1$.

With CRPropa3.2 it is possible to simulate time-dependent DSA. Also acceleration at shocks with finite width, leading to steeper energy-spectra can be modelled.

3.2 Cosmic Ray Electron Transport in M51

The face-on galaxy M51 was recently used to demonstrate that three dimensional transport modelling helps to differentiate between advective and diffusive propagation [7]. Radio observation hinted on an energy-independent diffusion and a radial dependence of the electron spectral index, which is hard to explain with purely advective or diffusive transport models.

For faster computations the authors implemented an extension of CRPropa for the relevant energy losses (synchrotron radiation and inverse Compton scattering) as continuous loss approximation. These losses use the measured radial dependent magnetic field strength and a homogeneous energy density of the inter stellar radiation field as input parameters. The observed star formation rate surface density has been used to draw the source positions and estimate the wind velocity.

Based on these simulations it was concluded: 1) The diffusion seems to be energy independent, but 2) the scale height of the galaxy changes with radius from $z_0 = 3.2\,\text{kpc}$ to $z_0 = 8.8\,\text{kpc}$. Furthermore, it could be shown that earlier estimates of the wind velocity were approximately a factor five too large according to their model.

4. Summary and Outlook

We summarised new developments and extensions of the CRPropa simulation framework since its last major release version 3.2. Most of the updates are focused on better usability, e.g., by allowing for much simpler creation of custom photon fields, tracking the production of secondaries, or code software updates that improve the simulation in the background, such as the speed up for fixed propagation steps.

These extensions will help to use CRPropa in a more flexible way, allowing to model, e.g., propagation and interactions in and close to the source of cosmic rays, test effects of non-standard redshift evolution of cosmic microwave background (CMB) models[7], on the diffuse gamma-ray flux, or include a consistent description of shock acceleration in ensemble-averaged transport models.

The plugins and examples demonstrate that a flexible simulation framework like CRPropa will be one of the key ingredients in upcoming multi-messenger models and help to improve our understanding of transport and interactions of cosmic rays and their secondaries better.

In the future, CRPropa should be extended to include, e.g., a model of momentum diffusion allowing to describe second-order Fermi acceleration within the SDE approach. Together with ongoing developments of nuclei-nuclei interaction this will open new opportunities to build accurate Galactic transport models.

Information on how to download the code, an online documentation, and examples of applications (including some of those presented in this work) can be found in `https://crpropa.desy.de/`.

Acknowledgements

References

[1] A. Addazi et al. Progress Part. Nucl. Phys. **125** (2022) 103948

[2] S. Aerdker et al. arXiv e-prints (2023) [arXiv.2306.10802]

[3] R. Alves Batista et al. JCAP **05** (2016) 038

[4] R. Alves Batista and A. Saveliev. Universe **7** (2021) 223.

---

[7]This would be expected, e.g, for an SU(2) instead of a U(1) gauge theory (see, e.g., [8]).

[5] R. Alves Batista, A. Saveliev, E. M. de Gouveia Dal Pino. MNRAS **489** (2019) 3836

[6] R. Alves Batista et al. JCAP **09** (2022) 035

[7] J. Dörner, P. Reichherzer, J. Becker Tjus et al. A&A **669** (2023) A111

[8] S. Hahn and R. Hofmann MNRAS **469** (2017) 1

[9] L. Merten et al. JCAP **06** (2017) 046

[10] P. Reichherzer et al. MNRAS **498** (2020) 4

[11] A. Saveliev and R. Alves Batista. PoS **444** (2023) 903