PROCEEDINGS OF SCIENCE

Studying dark matter with MadDM 3.1: a short user guide

Chiara Arina,[a] **Jan Heisig,**[a,*] **Fabio Maltoni,**[a] **Luca Mantani,**[a] **Daniele Massaro,**[a,b] **Olivier Mattelaer**[a] **and Gopolang Mohlabeng**[c,d]

[a] *Centre for Cosmology, Particle Physics and Phenomenology (CP3), Université catholique de Louvain, Chemin du Cyclotron 2, B-1348 Louvain-la-Neuve, Belgium*

[b] *Dipartimento di Fisica e Astronomia, Università di Bologna, Viale Berti Pichat 6/2, 40127 Bologna, Italy*

[c] *The Arthur B. McDonald Canadian Astroparticle Physics Research Institute and Department of Physics, Engineering Physics and Astronomy, Queens University, Kingston, Ontario, K7L 3N6, Canada*

[d] *Perimeter Institute for Theoretical Physics, Waterloo, Ontario, N2L 2Y5, Canada*

E-mail: chiara.arina@uclouvain.be, jan.heisig@uclouvain.be, fabio.maltoni@uclouvain.be, daniele.massaro@uclouvain.be, olivier.mattelaer@uclouvain.be, mohlabeng319@gmail.com, luca.mantani@uclouvain.be

MadDM is an automated numerical tool for the computation of dark matter observables for generic new physics models. We announce version 3.1 and summarize its features. Notably, the code goes beyond the mere cross-section computation for direct and indirect detection. For instance, it allows the user to compute the fully differential nuclear recoil rates as well as the energy spectra of photons, neutrinos and charged cosmic rays for arbitrary $2 \rightarrow n$ annihilation processes. This short user guide equips researchers with all the relevant information required to readily perform comprehensive phenomenological studies of particle dark matter models.

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

*Speaker

https://pos.sissa.it/

PoS(TOOLS2020)009

Contents

1. Introduction

The observation of the phenomenon of dark matter on various length scales in our Universe remains one of the major puzzles of modern physics (see *e.g.* Ref. [1] for a review). The hypothesis of a new particle – and possibly an entire new sector of particles beyond the standard model (BSM) – is a widely considered explanation. In particular, the existence of a stable weakly interacting massive particle naturally explains the observed value for the relic density via the thermal freeze-out mechanism. While initially being promoted through the popularity of supersymmetry, by now the idea of frozen-out dark matter has entered the standard recipe for successful dark matter model building well beyond the supersymmetric paradigm. An additional appeal of such a candidate emerges from the promising prospects to detect it. Its production from, scattering off and annihilation into standard-model particles – probed at colliders, direct and indirect detection experiments, respectively – provides three complementary search strategies accessible with current experimental sensitivities. Exploring the interplay of such observables has become a major direction of phenomenological research and brought forth the need for their efficient numerical computation.

This has stimulated the development of automated numerical tools such as microOMEGAs [2], DarkSUSY [3], SuperIso Relic [4] and – in particular, as considered here – MadDM [5].

For the computation of cross sections and widths, MadDM utilizes the automatized matrix element generator MadGraph5_aMC@NLO [6, 7] (MG5_aMC in the following). It is embedded as a plug-in of the MG5_aMC platform. As such, MadDM supports all particle physics models that can be cast into the Universal FeynRules Output (UFO) format [8], generated by *e.g.* FeynRules [9], SARAH [10] or LanHEP [11]. MadDM 1.0 [12], released in 2013, introduced the relic density calculator, while versions 2.0 [13] and 3.0 [5] extended the functionality by a comprehensive set of direct and indirect detection observables, respectively. For direct detection, the code not only computes the elastic spin-independent and spin-dependent dark matter-nucleon cross sections. It also allows for the computation of the double differential event rates as a function of time, scattering angle and energy supporting a variety of target materials. For indirect detection, MadDM allows the user to compute the velocity averaged annihilation cross sections today and the corresponding energy spectra of prompt photons, cosmic rays and neutrinos. The generation of annihilation spectra can either be done by combining pre-computed spectra for individual annihilation channels from PPPC4DMID [14] (fast mode) or by simulating events employing Pythia 8 [15] for showering and hadronization (precise mode). The latter enables full flexibility, in particular allowing the user to consider arbitrary $2 \to n$ processes. The program also includes experimental constraints from the direct detection experiments LUX [16], Xenon1T [17] and Pico-60 [18] as well as indirect detection constraints from gamma-ray observations of dwarf spheroidal galaxies by Fermi-LAT [19] that allow for the computation of a likelihood and exclusion limit.

With this article, we release version 3.1 that introduces various minor improvements, such as a revised display command, an extended output of the relic density computation as well as updated constraints from Xenon1T [17]. We provide a short user guide of MadDM that equips researchers with all relevant information required to readily perform comprehensive phenomenological studies of particle dark matter models. In particular, in section 2 we supply information on the installation and the general functionalities of the code. In section 3 we detail the observable-specific commands and settings. We summarize and give a brief outlook on upcoming developments in section 4.

2. Getting started

In this section, we provide the basic information on how to install MadDM[1] and describe the main commands via a quick tutorial. We depict its folder structure, the relevant output files and give a few tips on how to run it efficiently.

2.1 Installation

To install the MadDM plug-in, the user has to first download and untar the latest stable version of MG5_aMC from https://launchpad.net/mg5amcnlo. At the time of writing, this corresponds to version 2.8.2, which we assume for definiteness in the examples in the following. While MG5_aMC, is now compatible with Python 3, this is not yet the case for MadDM, which works only with Python 2.7. Additionally, the user should make sure that there is a complete installation

[1]See also https://launchpad.net/maddm.

of the SciPy and NumPy modules.[2] Once the MG5_aMC package has been untared, the user has to enter the corresponding directory, start MG5_aMC, and install MadDM via the MG5_aMC command line:

```
mydir$ tar -xzf MG5_aMC_v2.8.2.tar.gz
mydir$ cd MG5_aMC_v2_8_2/
MG5_aMC_v2_8_2$ python2.7 bin/mg5_aMC
MG5_aMC> install maddm
MG5_aMC> quit
MG5_aMC_v2_8_2$
```

The latest version of MadDM will automatically be downloaded and installed as a MG5_aMC plug-in. The corresponding source code in located in `MG5_aMC_v2_8_2/PLUGIN/maddm` while the executable python file `maddm.py` is stored in `MG5_aMC_v2_8_2/bin/`. See section 2.5 and figure 2 for details on the folder structure.

Note that MadDM is automatically interfaced with a few tools that support the computation of indirect detection observables. These are Pythia 8 [15], the PPPC4DMID libraries for annihilation spectra [14], DRAGON [20] and the GALPROP libraries [21] (needed for DRAGON). When performing indirect detection computations with MadDM for the first time, the user is asked whether these packages should be installed automatically. Note that *(i)* the installation can take some time and that *(ii)* Pythia 8 and PPPC4DMID are needed for the computation of annihilation spectra, while DRAGON is needed for cosmic-ray propagation only. The user can also perform the installation at any time via the MadDM command-line interface:

```
MG5_aMC_v2_8_2$ python2.7 bin/maddm.py
MadDM> install pythia8
MadDM> install PPPC4DMID
MadDM> install dragon
MadDM> install dragon_data_from_galprop
MadDM> quit
MG5_aMC_v2_8_2$
```

For further information about indirect detection and the usage of these packages see section 3.3.

2.2 Command-line interface and tutorial mode

Once the user has entered in MadDM by executing the `maddm.py` file, the first steps for any computation are to load a model and define the dark matter candidate. This is achieved by typing:

```
MG5_aMC_v2_8_2$ python2.7 bin/maddm.py
MadDM> import model DMsimp_s_spin0
MadDM> define darkmatter xd
```

where here we have considered a Dirac dark matter candidate denoted by the particle named `xd` within the simplified model called `DMsimp_s_spin0`. More details about models are provided in section 2.7. The relic abundance, direct and indirect detection observables for `xd` are computed via

[2]Note that MG5_aMC 2.8.X furthermore requires the Python module `six`.

PoS(TOOLS2020)009

```
MadDM> generate relic_density
MadDM> add direct_detection
MadDM> add indirect_detection
MadDM> output my_process_dir
```

The commands `generate` and `add` have the same functionalities they have in MG5_aMC. In particular, `generate` is used as the first command, while `add` is used to retain the previously generated processes and add new ones. Note that a subsequent call of the `generate` command will erase all previous processes. The last command above creates a folder `my_process_dir` which contains all the code necessary to launch the required computations. Performing these computations for a given parameter point is done via the `launch` command.

```
MadDM> launch my_process_dir
```

This opens the *launch interface* that allows the user to change settings and model parameters, as shown in figure 1. There are two ways of making changes. First, (repeatedly) entering a number 1–4 allows to alternate between the options displayed, while entering 5 or 6 opens the files `param_card.dat` or `maddm_card.dat`, respectively, with a command-line editor (`vim` by default).[3] These files contain all model parameters and most of the MadDM settings, respectively. A second option is to directly type `set <parameter> <value>` in the launch interface, for instance

```
> set mxd 500
```

for setting the dark matter mass to 500 GeV. Auto-completion is available (via pressing tab) to easily find the name of parameters. The observable-specific settings (the first four entries of figure 1) will

```
The following switches determine which programs are run:
/============ Description ============|====== values ======|======= other options =======\
| 1. Compute the Relic Density        |     relic = ON     |  OFF                        |
| 2. Compute direct(ional) detection  |    direct = direct |  OFF|directional            |
| 3. Compute indirect detection/flux  |  indirect = sigmav |  flux_source|flux_earth|OFF |
| 4. Run Multinest scan               |  nestscan = OFF    |  ON                         |
\=========================================================================================/
 You can also edit the various input card:
 * Enter the name/number to open the editor
 * Enter a path to a file to replace the card
 * Enter set NAME value to change any parameter to the requested value
 /=============================================================================\
 |  5. Edit the model parameters    [param]                                    |
 |  6. Edit the MadDM options       [maddm]                                    |
 \=============================================================================/
 [60s to answer]
 >
```

Figure 1: Example of the launch interface after performing the `launch` command in MadDM. In the specific example, the relic density, direct and indirect detection calculations are turned on, while the performance of a scan with MultiNest is switched off.

[3]Note that the files may, of course, be changed by any other instance instead.

be described in detail in section 3. Once the user is done with all settings the launch interface is finally exited by pressing enter.

Note that the `launch` command can be executed either in the same session or after quitting and restarting MadDM. In the former case, the specification of the directory where the process has been created is not necessary, as MadDM will launch the process of the last output in the session.

A convenient way of being guided through the basic commands is the tutorial model. It is entered by typing `tutorial` in the MadDM command-line interface:

```
MG5_aMC_v2_8_2$ python2.7 bin/maddm.py
MadDM> tutorial
```

The screen output explains the basic commands and options that the user may follow. It can be exited by:

```
MadDM> tutorial stop
```

2.3 The `display` command

When computing the observables for dark matter models, it is possible to use the following commands

```
MadDM> display processes
MadDM> display diagrams
```

to either display a list of the generated processes or the respective Feynman diagrams. From MadDM 3.1 on, the display command allows for the following options:

- `relic`, `direct` or `indirect`: display only processes/diagrams related to relic density, direct detection or indirect detection; a combination of them is supported, see the example below;
- `last`: displays only processes/diagrams generated by the last command called, it overwrites any other option specified;
- `all`: works as the simultaneous presence of `relic`, `direct`, `indirect`: it displays only diagrams relevant for dark matter annihilation and it is the default setting if no options are provided.

For instance, the command

```
MadDM> display processes relic indirect
```

displays all the processes related to relic density and indirect detection.

2.4 Running MadDM from a script

In certain applications, it might not be convenient or even possible to use the command-line interface of MadDM described in section 2.2. An alternative is to control MadDM via a script. To do so the respective commands described in section 2.2 need simply to be written in a plain text file separated by line-breaks. The respective script can be passed as an argument when starting MadDM. The corresponding operations will then be executed. For instance, the user may create the two scripts:[4]

generate.txt

```
import model DMsimp_s_spin0
define darkmatter xd
generate relic_density
add direct_detection
add indirect_detection
output my_process_dir
```

launch.txt

```
launch my_process_dir
indirect = flux_source
direct = direct
set sigmav_method madevent
set nevents 20000
set mxd 500
```

and execute MadDM:

```
MG5_aMC_v2_8_2$ python2.7 bin/maddm.py generate.txt
MG5_aMC_v2_8_2$ python2.7 bin/maddm.py launch.txt
```

Note that the settings `relic`, `direct`, `indirect` and `nestscan` should not be set by entering the numbers 1–4 (*cf.* section 2.2) in scripts as the selected mode can depend on the machine on which the code runs and on the specific MadDM version installed. The observable-specific commands contained in `launch.txt` are detailed in section 3.

2.5 Folder structure

Figure 2 displays the general folder structure of MG5_aMC after installation of the MadDM plug-in. The directory `bin` contains the python code to be executed, while `models` contains all models used, see section 2.7 for more information. Once generated, `my_process_dir` contains all code, input and output for a certain process. Input parameters are stored in various files in the directory `Cards`. For instance, model parameters can be either set via the `set` command in the launch interface (see section 2.2) or by changing the respective parameters in `Cards/param_card.dat`. The output folder contains a sub-directory for each run within the process `run_01`, `run_02`, ... which, in turn, contains all outputs of the computation (stored in `MadDM_results.txt` and `maddm.out`) as well as a copy of the `maddm_card.dat` used. Further output is linked to the directory `Output_Indirect`, see section 3.3 for more details.

2.6 Running scans

There are several ways to run scans over parameter space points within MadDM. First, MadDM may be just called by an external code that performs a scan. In this case, the parameters and settings may just be passed by a script as detailed in section 2.4.

[4]Note that the content of the two scripts could as well be put into one file. The separation of the launch command is, however, often convenient as only this part needs to be rerun when choosing different parameters.

```
MG5_aMC_v2_8_2
├── bin
│   ├── maddm.py
│   ├── mg5_aMC
│   └── ...
├── models
├── my_process_dir
│   ├── bin
│   ├── Cards
│   │   ├── maddm_card.dat
│   │   ├── multinest_card.dat
│   │   ├── param_card.dat
│   │   └── ...
│   ├── Indirect
│   ├── output
│   │   ├── run_01
│   │   ├── run_02
│   │   └── ...
│   └── ...
├── PLUGIN
│   ├── maddm
│   └── ...
└── ...
```

```
run_01
├── maddm_card.dat
├── MadDM_results.txt
├── maddm.out
└── Output_Indirect
    ├── antiprotons_spectrum_pythia8.dat
    ├── gammas_spectrum_pythia8.dat
    ├── neutrinos_e_spectrum_pythia8.dat
    ├── neutrinos_mu_spectrum_pythia8.dat
    ├── neutrinos_tau_spectrum_pythia8.dat
    ├── positrons_spectrum_pythia8.dat
    ├── restx_spectrum_pythia8.dat
    ├── pythia8.log
    ├── run_01_DM_banner.txt
    ├── run_shower.sh
    └── unweighted_events.lhe.gz
```

Figure 2: Schematic structure of the folders and files of MG5_aMC and MadDM. **On the left:** Main directory of MG5_aMC where the python executable file `maddm.py` is located in the `bin` folder, while the source code is in `PLUGIN/maddm/`. The output directory `my_process_dir` contains all relevant setting cards (within `Cards`), and the output files in `output/run_01` for instance. **On the right:** Zoomed view of the `run_01` directory, where the main results are stored, as labeled. The file `MadDM_results.txt` recaps the value of all observable computed by the user. Notice that `Output_Indirect` contains indirect detection files, such as the energy spectra and the lhe event file.

The second option is to employ the sequential grid scan functionality of MadDM, which allows one to scan over an arbitrary number of model parameters with one launch command. To achieve this, instead of setting a given parameter to a fixed value, the respective scan range has to be defined:

```
MadDM> launch my_process_dir
> set mxd scan:range(50,700,25)
```

Note that after the syntax `scan:` any python iterable is accepted, including list comprehension syntax. In the case of multi-dimensional scans, two possibilities are available. Using the syntax `scan:` for two or more parameters will generate a nested loop over the scan ranges, i.e. a complete grid. Another possibility is to use the syntax `scan1:`, which instead creates a parallel scan, namely the values of the iterables are scanned simultaneously. Instead of being specified in the launch interface, this setting can also be done in the `param_card.dat`:

my_process_dir/Cards/param_card.dat

```
...
Block mass
    1 5.040000e-03 # MD
    2 2.550000e-03 # MU
    3 1.010000e-01 # MS
    4 1.270000e+00 # MC
    5 4.700000e+00 # MB
    6 1.720000e+02 # MT
   15 1.777000e+00 # MTA
   23 9.118760e+01 # MZ
   25 1.250000e+02 # MH
   51 1.000000e+01 # MXc
   52 scan:range(50,700,25) # MXd
   54 1.000000e+03 # MY0
   ...
```

The third option is to perform guided scans with the Bayesian inference tool MultiNest (MultiNest [22, 23] is provided together with PyMultiNest [24]) by specifying

```
MadDM> launch my_process_dir
> nestscan = ON
```

and setting multinest parameters in `multinest_card.dat`, which then appears as number 7 in the launch interface:

```
/================================================================\
|  5. Edit the model parameters    [param]                        |
|  6. Edit the MadDM options       [maddm]                        |
|  7. Edit the Multinest options   [multinest]                    |
\================================================================/
[60s to answer]
>
```

For further details see Appendix E.2 of Ref. [5].

2.7 Importing models

Being a plug-in of MG5_aMC , MadDM can perform computation within any particle physics model that allows for an implementation in the Universal FeynRules Output (UFO) format [8]. The implementation can be achieved with automated tools like FeynRules [9], SARAH [10] or LanHEP [11]. The respective model directory (named as the model to be imported) has to be stored in the directory `models` (*cf.* section 2.5). Note that a database of models can be found at the FeynRules webpage: http://feynrules.irmp.ucl.ac.be/wiki/ModelDatabaseMainPage. For models stored in that database (but not in the local `models` folder) MadDM automatically downloads the model importing it via the MadDM command-line interface (*cf.* section 2.2). This model list can be viewed by the user by typing

```
MadDM> display modellist
```

The entire list of models is then displayed. A user guide for the implementation of a particle physics model into FeynRules can be found in Ref. [9].

3. Dark matter observables

In the following, we detail the capabilities of MadDM to perform computations of the relic density, direct and indirect detection observables, respectively.

3.1 Relic density

MadDM allows for the computation of the relic density in the framework of thermal freeze-out of dark matter. It automatically computes the rates for all relevant $2 \to 2$ annihilation processes including coannihilation processes. Coannihilation is taken into account if the user specifies the coannihilating partner(s), such as

```
MadDM> define coannihilator xco1 xco2 xco3
```

prior to the command `generate relic_density`, see section 2.2. Here `xco1`, `xco2`, `xco3` are exemplary names of coannihilating partners in the model. The code solves the corresponding Boltzmann equation for the dark matter abundance (or dark-sector[5] abundance for the case of coannihilation [25]) numerically, see Ref. [12] for further details. It assumes kinetic equilibrium between all involved particles (as well as chemical equilibrium within the dark sector) during dark matter freeze out.

The output on screen is, for instance:

```
***** Relic Density
OMEGA IS  0.000325869586293
INFO: Relic Density = 3.26e-04 UNDERABUNDANT
INFO: x_f           = 2.80e+01
INFO: sigmav(xf)    = 3.27e-24 cm^3/s
INFO: xsi           = 2.72e-03
```

The relic density is given in terms of Ωh^2, where Ω is the dark matter energy density in units of the critical density and h is the Hubble constant in units of $100\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$. The freeze-out point (approximately the point of chemical decoupling) is given by x_f, where $x = m_\mathrm{DM}/T$. The thermally averaged annihilation cross section at the freeze-out point, $\langle\sigma v\rangle(x_\mathrm{f})$, is given in units of $\mathrm{cm^3\,s^{-1}}$. Since version 3.1, MadDM additionally displays the contributions of the different channels to the relic density. The resulting screen output reads:

```
INFO: Channels contributions:
INFO: xdxdx_hh            : 0.05 %
INFO: xdxdx_zz            : 0.13 %
INFO: xdxdx_ttx           : 98.21 %
INFO: xdxdx_aa            : 0.95 %
```

[5]Here we consider the *dark sector* to comprise the dark matter candidate and all potential coannihilators.

```
INFO: xdxdx_wpwm          : 0.10 %
INFO: xdxdx_az            : 0.57 %
INFO: No contribution from processes: y0y0
```

Models whose relic density undershoots (overshoots) the value measured by Planck, $\Omega h^2 = 0.120 \pm 0.001$ [26], by more than 2σ are flagged as `UNDERABUNDANT` (`OVERABUNDANT`), while values within that range are flagged as `WITHIN_EXP_ERRORS`. However, note that the theoretical uncertainty in the model prediction – typically significantly larger – is not estimated by MadDM. It is up to the user to estimate the error on Ωh^2 to be taken into account *e.g.* in a global fit.

In the case of thermally underabundant dark matter, `xsi` denotes the fraction of the model's thermal abundance of total measured dark matter abundance,

$$\xi = \frac{(\Omega h^2)_{\text{model}}}{(\Omega h^2)_{\text{Planck}}}. \tag{1}$$

In this case constraints from direct and indirect detection, subject to the following sections, are interpreted in two ways:

- One assumes that the model's candidate, indeed only makes up a fraction ξ of the total amount of dark matter, implying the existence of a further unspecified contribution, *e.g.* axions or primordial black holes. This interpretation entails the rescaling of the yields for direct and indirect detection by a factor of ξ and ξ^2, respectively. It is denoted by 'Thermal'.
- Regardless of the underabundant contribution from thermal freeze-out, the model's candidate is assumed to constitute 100% of the measured dark matter abundance. This interpretation applies in the presence of an additional (non-thermal) contribution to dark matter production, *e.g.* through a late decay of a heavier species. It is denoted by 'All DM'.

Note that to enable the interpretation of direct and indirect detection observables in the 'Thermal' scenario the relic density computation has to be performed.

3.2 Direct detection

Direct detection experiments search for dark matter particles scattering off atomic nuclei in low-background environments, *e.g.* deep underground. If the recoil momentum of the nucleus is above the detection threshold, then electrons, photons and/or phonons induced by the nuclear recoil may be detected. The number of dark matter scattering events in a given experiment is then set by a confluence of factors in the dark matter theory parameter space. This includes the mass, scattering cross section and the astrophysical velocity distribution of the dark matter in our local Galactic neighborhood. Since the solar system is moving in a particular direction with respect to the Galactic dark matter halo, the astrophysical distribution can provide both velocity and angular information. These are used to calculate the nuclear recoil energy spectrum and the angular recoil spectrum, by both direct detection and directional detection experiments respectively.

MadDM allows the user to choose among two modes called `direct` and `directional`. The setting can be changed in the launch interface either by (repeatedly) entering the number 2 until the requested option is displayed on screen, *cf.* figure 1 and section 2.2, or by directly entering one of the following commands:

```
> direct = direct
> direct = directional
```

The mode `direct` provides the basic computations of spin-independent and spin-dependent dark matter-nucleon cross section as well as their respective limits from LUX [16], XENON1T [17], and PICO-60 [18]. An exemplary screen output from the direct detection module (for the same parameters used above) reads:

```
***** Direct detection [cm^2]:
INFO: SigmaN_SI_p: Thermal = 2.01e-50 ALLOWED, All DM = 7.40e-48 ALLOWED  Xenon1ton ul = 4.16e-46
INFO: SigmaN_SI_n: Thermal = 1.98e-50 ALLOWED, All DM = 7.27e-48 ALLOWED  Xenon1ton ul = 4.16e-46
INFO: SigmaN_SD_p: Thermal = 0.00e+00 ALLOWED, All DM = 0.00e+00 ALLOWED  Pico60 ul     = 2.03e-40
INFO: SigmaN_SD_n: Thermal = 0.00e+00 ALLOWED, All DM = 0.00e+00 ALLOWED  Lux2017 ul    = 1.22e-40
```

As indicated, all direct-detection cross sections are given in cm^2. In the 'Thermal' scenario the cross-section prediction is rescaled by ξ, see section 3.1 for details. This particular model does not have spin-dependent interactions due to the type of mediator interacting with the dark matter.

The mode `directional` additionally provides the fully differential nuclear recoil rates as a function of energy, angle and time. It, therefore, allows the user to explore the directional information of dark matter scattering. For the computation of nuclear recoil rates MadDM allows the user to choose among a large set of detector materials and take into account detector smearing effects. Furthermore, the recoil energy, the detector size, the most probable dark matter velocity and escape velocity as well as the local dark matter density can be specified. Moreover, the nuclear form factor can be customized. All these settings can be adjusted by changing the corresponding entries in the `maddm_card.dat` file. For more information, see [13].

3.3 Indirect detection

Indirect detection probes the (self-)annihilation of dark matter in locally over-dense regions, like the center of the Galaxy. Stable particles that are the final products of these annihilation processes can propagate to us and act as the messengers of the dark matter signal. Photons (gamma rays), neutrinos and stable charged particles (cosmic rays), in particular, antiprotons and positrons, are commonly considered messenger particles. The dark matter annihilation cross section and energy spectra of these messengers need to be computed to confront the signal prediction with data. MadDM provides two different modes and a variety of further settings to supply the user with these observables at an appropriate level of precision and speed.

3.3.1 Running modes and settings

In the `fast` mode, the cross-section computation is performed with a fast phase-space integrator using the Simpson method [27] (also used for the relic density computations). In this mode, no events are generated. Furthermore, it is restricted to $2 \to 2$ processes. It is selected by entering, in the `launch` interface, the command

```
> set fast
```

In `precise` mode, the phase-space integration is performed by MadEvent [28]. Events are generated and arbitrary $2 \to n$ processes can be taken into account. It is selected by

```
> set precise
```

The `precise` mode allows the user to evaluate the cross section either at a fixed dark matter velocity

```
> set sigmav_method = madevent
```

or taking into account a Maxwell-Boltzmann distribution in velocity through a reshuffling and reweighting of events [29, 30]:

```
> set sigmav_method = reshuffling
```

Notice that this is the default setting if nothing is specified. The average velocity (in units $c = 1$) can be set as follows, *e.g.* to 10^{-5}:

```
> set vave_indirect 1e-5
```

which is the default value if nothing is specified. Note that the automatic derivation of constraints from dwarf spheroidal galaxies (see below) requires the velocity to lie between 1.4×10^{-4} and 3×10^{-6}, while typical velocities for the Galactic center is around 10^{-3}. The number of generated events can be set as follows, for instance:

```
> set nevents = 50000
```

Note that the generation of smooth spectra (in particular towards the tails) might require a large number of events (up to several million). However, for analyses of binned spectra much fewer events can often be sufficient. For instance, for the computation of Fermi-LAT limits (see below), based on a binned likelihood function (with 24 bins) an event number between 10000 to 50000 is often sufficient to obtain a good estimate of the constraints. However, as the number and energy of messenger particles per annihilation depends strongly on the dark matter model, these numbers are not universally valid. For instance, considering heavy dark matter with masses larger than a few TeV, Fermi-LAT only probes the low-energy tail of the photon spectrum that is sampled by a small fraction of events. In such cases, care has to be taken when estimating the number of required events.

In both modes, `fast` and `precise`, the user can specify whether to compute just the cross section (`sigmav`) or in addition the energy spectra at sources (`flux_source`) or near Earth (`flux_earth`). The latter is relevant for neutrinos, which oscillate, and for cosmic rays that are subject to a non-trivial propagation process between the source and Earth that, in particular, affect the spectra. This setting can be chosen by repeatedly typing the number 3 in the launch interface to alternate between the three options (and `OFF`), *cf.* figure 1 and section 2.2. Alternatively, it can be set by one of the following commands, respectively:

```
> indirect = sigmav
> indirect = flux_source
> indirect = flux_earth
```

This provides a total of six different running modes. The respective default settings and further options are summarized in table 1. We will briefly discuss them in the following for completeness, see [5] for further details.

	fast mode	precise mode
indirect = sigmav	Default: sigmav_method = inclusive	Default: sigmav_method = reshuffling Other options: sigmav_method = madevent
indirect = flux_source	Default: sigmav_method = inclusive indirect_flux_source_method = PPPC4DMID_ew Other options: indirect_flux_source_method = PPPC4DMID	Default: sigmav_method = reshuffling indirect_flux_source_method = pythia8 Other options: sigmav_method = madevent indirect_flux_source_method = PPPC4DMID_ew indirect_flux_source_method = PPPC4DMID
indirect = flux_earth	Default: sigmav_method = inclusive indirect_flux_earth_method = PPPC4DMID_ep	Default: sigmav_method = reshuffling indirect_flux_source_method = pythia8 indirect_flux_earth_method = dragon Other options: sigmav_method = madevent indirect_flux_earth_method = PPPC4DMID_ep

Table 1: Summary of the MadDM indirect detection functionalities upon the execution of the launch command. We display the default settings and further options of all six occurring combinations.

In fast mode, enabling indirect = flux_source, the energy spectra are taken from the PPPC4DMID database [14] containing pre-computed results for annihilation into pairs of standard model particles only. MadDM combines the spectra of different channels according to their cross sections. The user may choose whether to include electroweak corrections [31] (default) or not, specified by setting indirect_flux_source_method to PPPC4DMID_ew or PPPC4DMID, respectively. Note that in fast mode, propagated cosmic-ray spectra are only available for positrons.

In precise mode, enabling indirect = flux_source, energy spectra are computed using the generated events and Pythia 8 [15] for showering and hadronization, where electroweak corrections are enabled by default.[6] However, the user can also choose to use the pre-computed spectra from PPPC4DMID instead (*cf.* table 1) by using the commands mentioned above. For `indirect = flux_earth` the energy spectra of charged particles are propagated by the numerical code DRAGON [20]. The propagation parameters can be set in the `dragon_card.xml`. Similar to the case of source spectra, for positrons, the propagates spectra can alternatively be taken from the PPPC4DMID database, regardless of the `precise` mode. This is achieved by typing:

```
> indirect_flux_earth_method = PPPC4DMID_ep
```

Note that the propagated neutrino spectra are always computed after oscillations in vacuum employing the very long baseline approximation (see [5] for details).

As explained in section 2.1, both Pythia 8 and DRAGON are automatically installed within the MadDM framework either when first running `indirect_detection` (asked for in the launch interface) or via the command-line interface whenever the user needs them. We ask the user to cite these additional public codes when these are used within MadDM.

3.3.2 Selecting processes

So far we have assumed that the user employs the general command (*cf.* section 2.2)

```
MadDM > generate indirect_detection
```

to specify the considered processes. Note that this command generates all processes where two dark matter particles annihilate into two particles, including all possible standard model particles and BSM particles which are even under the 'dark' group. Alternatively, the user can specify the final state by typing:

```
MadDM > generate indirect_detection u u~
```

forcing MadDM to generate the diagram with a pair of u-quarks in the final state only. In fact, any $2 \to n$ dark matter annihilation process can be computed (requiring running `precise` mode). For instance, one may consider internal bremsstrahlung, where an additional photon `a` is emitted:

```
MadDM > generate indirect_detection u u~ a
```

Note that further MG5_aMC syntax can be employed to specify the process. For instance, to collectively specify a set of particles, multiparticle variables may be defined:

```
MadDM > define q = u d s c b t
MadDM > define qbar =  u~ d~ s~ c~ b~ t~
```

Furthermore, the decay of final state particles may be specified (and hence performed by MadDM). For example, the model used above as reference contains a spin-0 mediator `y0` that can appear in the final state (if it is lighter than dark matter). The mediator can further decay into quarks. Assuming the above multiparticle definition we can hence specify its appearance in the final state and subsequent decay by

[6]This setting can be changed by modifying `pythia_card.dat`.

```
MadDM > generate indirect_detection y0 y0, y0 > q qbar
```

Alternatively, decays can be performed by PYTHIA 8 if the particles' branching ratios are provided. To this end, an automatic computation of branching ratios within MADDM can be performed by setting the corresponding decay width to `AUTO` in the `param_card.dat` [32]. The above process can, hence, also be computed by typing:

```
MadDM > generate indirect_detection y0 y0
```

while the decay width is set in the launch interface:

```
> set wy0 AUTO
```

(or by the corresponding change in the `param_card.dat` by any other instance). With these settings, MADDM automatically computes all branching ratios of the spin-0 mediator while PYTHIA 8 performs the respective decays in the narrow width approximation.

3.3.3 Fermi-LAT constraints

Once the photon energy spectra have been computed with one of the methods described above, MADDM automatically computes the exclusion limit from the Fermi-LAT gamma-ray data from dwarf spheroidal galaxies [19], if the dark matter velocity is set to an allowed value, *i.e.* between 1.4×10^{-4} and 3×10^{-6}. For details on the Fermi-LAT likelihood function implementation, we refer to [5]. The output is given by the excluded annihilation cross section at 95% CL (confidence level) compared to the predicted annihilation cross-section, the Fermi-LAT likelihood and the p-value for the tested model point.

3.3.4 Output

An exemplary screen output from the indirect detection module (for the same parameters used above) reads:

```
****** Indirect detection [cm^3/s]:
INFO: <sigma v> method: madevent
INFO: DM particle halo velocity: 2e-05/c
INFO: xdxdx_zz      Thermal = 1.67e-38  ALLOWED     All DM = 2.26e-33  ALLOWED     Fermi ul = 9.04e-23
INFO: xdxdx_aa      Thermal = 1.24e-37  NO LIMIT    All DM = 1.69e-32  NO LIMIT    Fermi ul = -1.00e+00
INFO: xdxdx_ttx     Thermal = 1.28e-35  ALLOWED     All DM = 1.74e-30  ALLOWED     Fermi ul = 1.11e-25
INFO: xdxdx_hh      Thermal = 6.01e-39  ALLOWED     All DM = 8.15e-34  ALLOWED     Fermi ul = 2.21e-22
INFO: xdxdx_wpwm    Thermal = 1.27e-38  ALLOWED     All DM = 1.73e-33  ALLOWED     Fermi ul = 1.16e-22
INFO: Skipping zero cross section processes for: xrxr, xcxcx, y0y0
INFO: Total limits calculated with Fermi likelihood:
INFO: DM DM > all   Thermal = 1.30e-35  ALLOWED     All DM = 1.76e-30  ALLOWED     Fermi ul = 2.85e-25
INFO:
INFO: *** Fluxes at earth [particle/(cm^2 sr)]:
INFO: gammas Flux          =     1.57e-14
INFO: neutrinos_e Flux     =     8.89e-18
INFO: neutrinos_mu Flux    =     9.71e-18
INFO: neutrinos_tau Flux   =     8.75e-18
```

For each annihilation channel, the velocity averaged annihilation cross section today $\langle\sigma v\rangle_0$ is displayed in units of $\mathrm{cm}^3\mathrm{s}^{-1}$. For the 'Thermal' scenario, the cross section predictions are rescaled by ξ^2, see section 3.1 for details. Processes with zero cross section are listed below. Each channel is compared to the limit coming from the Fermi-LAT constraints on prompt photons from dwarf spheroidal galaxies. Subsequently, the total cross section is shown, with the limit computed performing the full Fermi-LAT likelihood analysis. The last lines show the values of the total integrated flux for prompt photons and neutrinos.

In addition, several output files are produced. In the case of single point runs, the relevant output is written into `my_process_dir/output/run_01/MadDM_results.txt` (*cf.* figure 2). The file contains a summary of the computed observables, including the Fermi-LAT likelihood and p-value for the considered parameter point. It is formatted conveniently to enable parsing. The output spectra generated from the PPPC4DMID database can be found in the same directory. The ones generated by using simulated events (in the `precise` mode) are stored in the `my_process_dir/Indirect/Events/run_01/` directory, *cf.* the folder structure shown on the right in figure 2. In case of a scan, `MadDM_results.txt` is not created. Instead, the file `my_process_dir/output/scan_run_01.txt` is written. It contains a list of the computed observables for all parameter points scanned over.

4. Summary and outlook

MadDM is a comprehensive numerical tool for performing computations of dark matter observables. In particular, it supports a detailed interpretation of direct and indirect dark matter searches by providing *e.g.* the fully differential nuclear recoil rates (as a function of energy, angle and time) as well as the photon, neutrino and cosmic-ray spectra for arbitrary $2 \to n$ annihilation processes at source or near Earth. For the latter, MadDM is interfaced to Pythia 8 and Dragon.

Being a plug-in of MG5_aMC, MadDM is conveniently installed and run through a user-friendly command-line interface. It provides an interactive and self-explanatory tutorial mode while the experienced user may prefer running MadDM via scripting. The MG5_aMC framework provides further features inherited by MadDM such as automated width computation or natural support of any particle physics model that can be cast in a UFO format.

MadDM is subject to ongoing developments that further enlarge its capabilities. For the next release, the computations are extended to general loop-induced processes. In particular, we will provide a framework to analyze the gamma-ray line spectrum arising from the annihilation of dark matter into photons, like $\gamma\gamma$, γZ, γh. An automated computation of constraints from the gamma-ray line searches from observations of the Galactic center [33, 34] will also be supplied.

Acknowledgements

C.A. is supported by the Innoviris ATTRACT 2018 104 BECAP 2 agreement. J.H. acknowledges support from the F.R.S.-FNRS, of which he is a postdoctoral researcher. This work has received funding from the European Union's Horizon 2020 research and innovation program as part of the Marie Skłodowska-Curie Innovative Training Network MCnetITN3 (grant agreement no. 722104). Computational resources have been provided by the supercomputing facilities of the Université catholique de Louvain (CISM/UCL) and the Consortium des Équipements de Calcul Intensif en Fédération Wallonie Bruxelles (CÉCI) funded by the Fond de la Recherche Scientifique de Belgique (F.R.S.-FNRS) under convention 2.5020.11 and by the Walloon Region.

References

[1] G. Bertone, D. Hooper and J. Silk, *Particle dark matter: Evidence, candidates and constraints*, *Phys.Rept.* **405** (2005) 279–390, [hep-ph/0404175].

[2] G. Bélanger, F. Boudjema, A. Goudelis, A. Pukhov and B. Zaldivar, *micrOMEGAs5.0 : Freeze-in*, *Comput. Phys. Commun.* **231** (2018) 173–186, [1801.03509].

[3] T. Bringmann, J. Edsjö, P. Gondolo, P. Ullio and L. Bergström, *DarkSUSY 6 : An Advanced Tool to Compute Dark Matter Properties Numerically*, *JCAP* **07** (2018) 033, [1802.03399].

[4] A. Arbey, F. Mahmoudi and G. Robbins, *SuperIso Relic v4: A program for calculating dark matter and flavour physics observables in Supersymmetry*, *Comput. Phys. Commun.* **239** (2019) 238–264, [1806.11489].

[5] F. Ambrogi, C. Arina, M. Backovic, J. Heisig, F. Maltoni, L. Mantani et al., *MadDM v.3.0: a Comprehensive Tool for Dark Matter Studies*, *Phys. Dark Univ.* **24** (2019) 100249, [1804.00044].

[6] J. Alwall, M. Herquet, F. Maltoni, O. Mattelaer and T. Stelzer, *MadGraph 5 : Going Beyond*, *JHEP* **06** (2011) 128, [1106.0522].

[7] J. Alwall, R. Frederix, S. Frixione, V. Hirschi, F. Maltoni, O. Mattelaer et al., *The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations*, *JHEP* **07** (2014) 079, [1405.0301].

[8] C. Degrande, C. Duhr, B. Fuks, D. Grellscheid, O. Mattelaer and T. Reiter, *UFO - The Universal FeynRules Output*, *Comput. Phys. Commun.* **183** (2012) 1201–1214, [1108.2040].

[9] A. Alloul, N. D. Christensen, C. Degrande, C. Duhr and B. Fuks, *FeynRules 2.0 - A complete toolbox for tree-level phenomenology*, *Comput. Phys. Commun.* **185** (2014) 2250–2300, [1310.1921].

[10] F. Staub, *SARAH 3.2: Dirac Gauginos, UFO output, and more*, *Comput. Phys. Commun.* **184** (2013) 1792–1809, [1207.0906].

[11] A. Semenov, *LanHEP — A package for automatic generation of Feynman rules from the Lagrangian. Version 3.2*, *Comput. Phys. Commun.* **201** (2016) 167–170, [1412.5016].

[12] M. Backoviċ, K. Kong and M. McCaskey, *MadDM v.1.0: Computation of Dark Matter Relic Abundance Using MadGraph5*, *Phys. Dark Univ.* **5-6** (2014) 18–28, [1308.4955].

[13] M. Backoviċ, A. Martini, O. Mattelaer, K. Kong and G. Mohlabeng, *Direct Detection of Dark Matter with MadDM v.2.0*, *Phys. Dark Univ.* **9-10** (2015) 37–50, [1505.04190].

[14] M. Cirelli, G. Corcella, A. Hektor, G. Hutsi, M. Kadastik, P. Panci et al., *PPPC 4 DM ID: A Poor Particle Physicist Cookbook for Dark Matter Indirect Detection*, *JCAP* **1103** (2011) 051, [1012.4515].

[15] T. Sjostrand, S. Ask, J. R. Christiansen, R. Corke, N. Desai, P. Ilten et al., *An Introduction to PYTHIA 8.2*, Comput. Phys. Commun. **191** (2015) 159–177, [1410.3012].

[16] LUX collaboration, D. S. Akerib et al., *Limits on spin-dependent WIMP-nucleon cross section obtained from the complete LUX exposure*, Phys. Rev. Lett. **118** (2017) 251302, [1705.03380].

[17] XENON collaboration, E. Aprile et al., *Dark Matter Search Results from a One Ton-Year Exposure of XENON1T*, Phys. Rev. Lett. **121** (2018) 111302, [1805.12562].

[18] PICO collaboration, C. Amole et al., *Dark Matter Search Results from the PICO-60 C_3F_8 Bubble Chamber*, Phys. Rev. Lett. **118** (2017) 251301, [1702.07666].

[19] DES, Fermi-LAT collaboration, A. Albert et al., *Searching for Dark Matter Annihilation in Recently Discovered Milky Way Satellites with Fermi-LAT*, Astrophys. J. **834** (2017) 110, [1611.03184].

[20] C. Evoli, D. Gaggero, D. Grasso and L. Maccione, *Cosmic-Ray Nuclei, Antiprotons and Gamma-rays in the Galaxy: a New Diffusion Model*, JCAP **0810** (2008) 018, [0807.4730].

[21] A. E. Vladimirov, S. W. Digel, G. Johannesson, P. F. Michelson, I. V. Moskalenko, P. L. Nolan et al., *GALPROP WebRun: an internet-based service for calculating galactic cosmic ray propagation and associated photon emissions*, Comput. Phys. Commun. **182** (2011) 1156–1161, [1008.3642].

[22] F. Feroz and M. P. Hobson, *Multimodal nested sampling: an efficient and robust alternative to MCMC methods for astronomical data analysis*, Mon. Not. Roy. Astron. Soc. **384** (2008) 449, [0704.3704].

[23] F. Feroz, M. P. Hobson and M. Bridges, *MultiNest: an efficient and robust Bayesian inference tool for cosmology and particle physics*, Mon. Not. Roy. Astron. Soc. **398** (2009) 1601–1614, [0809.3437].

[24] J. Buchner, A. Georgakakis, K. Nandra, L. Hsu, C. Rangel, M. Brightman et al., *X-ray spectral modelling of the AGN obscuring region in the CDFS: Bayesian model selection and catalogue*, Astron. Astrophys. **564** (2014) A125, [1402.0004].

[25] J. Edsjo and P. Gondolo, *Neutralino relic density including coannihilations*, Phys. Rev. **D56** (1997) 1879–1894, [hep-ph/9704361].

[26] Planck collaboration, N. Aghanim et al., *Planck 2018 results. VI. Cosmological parameters*, Astron. Astrophys. **641** (2020) A6, [1807.06209].

[27] S. Weinzierl, *Introduction to Monte Carlo methods*, hep-ph/0006269.

[28] F. Maltoni and T. Stelzer, *MadEvent: Automatic event generation with MadGraph*, JHEP **02** (2003) 027, [hep-ph/0208156].

[29] R. Kleiss, W. J. Stirling and S. D. Ellis, *A New Monte Carlo Treatment of Multiparticle Phase Space at High-energies*, *Comput. Phys. Commun.* **40** (1986) 359.

[30] O. Mattelaer, *On the maximal use of Monte Carlo samples: re-weighting events at NLO accuracy*, *Eur. Phys. J.* **C76** (2016) 674, [1607.00763].

[31] P. Ciafaloni, D. Comelli, A. Riotto, F. Sala, A. Strumia and A. Urbano, *Weak Corrections are Relevant for Dark Matter Indirect Detection*, *JCAP* **1103** (2011) 019, [1009.0224].

[32] J. Alwall, C. Duhr, B. Fuks, O. Mattelaer, D. G. Öztürk and C.-H. Shen, *Computing decay rates for new physics theories with FeynRules and MadGraph5_aMC@NLO*, *Comput. Phys. Commun.* **197** (2015) 312–323, [1402.1178].

[33] Fermi-LAT collaboration, M. Ackermann et al., *Updated search for spectral lines from Galactic dark matter interactions with pass 8 data from the Fermi Large Area Telescope*, *Phys. Rev.* **D91** (2015) 122002, [1506.00013].

[34] HESS collaboration, H. Abdallah et al., *Search for γ-Ray Line Signals from Dark Matter Annihilations in the Inner Galactic Halo from 10 Years of Observations with H.E.S.S.*, *Phys. Rev. Lett.* **120** (2018) 201101, [1805.05741].