Documentation DMγSpec

Kai Urban

March 3, 2022

1 Introduction

The Python code DMγSpec produces photon spectra from wino and Higgsino DM annihilation $\chi^0\chi^0 \to \gamma + X$ Sudakov-resummed at the endpoint [1, 2, 3], and merged with the parton-shower calculation PPPC4DM [4, 5] to provide full photon spectra for all values of $x = E_\gamma/m_\chi$. In addition, the code includes the Sommerfeld effect using NLO electroweak (EW) potentials for wino [6, 7] and Higgsino DM [8]. For more details on the physics background, see

- M. Beneke, K. Urban and M. Vollmann, *Matching resummed endpoint and continuum γ-ray spectra from dark-matter annihilation*, 22xx.xxxx

and references mentioned below.
DMγSpec is available from

dmyspec.hepforge.org

or alternatively from https://users.ph.tum.de/t31software/dmyspec. The code provides top-level functions that generate differential cross section and cumulative versions from pretabulated data grids that cover the range of DM mass $m_\chi = (0.5 - 100)\,\text{TeV}$. The data grids are generated from a private Mathematica implementation that takes care of the distributional nature of the result at the endpoint and performs the Sudakov resummation. The Python implementation is chosen to make the results fast and easily accessible and to avoid any complications that arise when folding the distribution-valued differential "partonic" cross section, e.g., with a detector resolution function.
In the following, we detail the required Python packages to be installed, and how to access the relevant functions. Furthermore, we discuss the code validation and the precision to be expected compared to the Mathematica reference implementation. If you use DMγSpec in your work, please cite

- M. Beneke, K. Urban and M. Vollmann, *Matching resummed endpoint and continuum γ-ray spectra from dark-matter annihilation*, 22xx.xxxx
- M. Cirelli, G. Corcella, A. Hektor, G. Hutsi, M. Kadastik, P. Panci, M. Raidal, F. Sala and A. Strumia, *PPPC 4 DM ID: A Poor Particle Physicist Cookbook for Dark Matter Indirect Detection*, *JCAP* **03** (2011) 051 [1012.4515]. [Erratum: JCAP 10, E01 (2012)]

If you use the wino spectra with the NLO EW potentials, please cite in addition

- M. Beneke, A. Broggio, C. Hasner and M. Vollmann, *Energetic γ-rays from TeV scale dark matter annihilation resummed*, *Phys. Lett. B* **786** (2018) 347–354 [1805.07367]. [Erratum: Phys.Lett.B 810, 135831 (2020)]
- M. Beneke, A. Broggio, C. Hasner, K. Urban and M. Vollmann, *Resummed photon spectrum from dark matter annihilation for intermediate and narrow energy resolution*, *JHEP* **08** (2019) 103 [1903.08702]. [Erratum: JHEP 07, 145 (2020)]
- M. Beneke, R. Szafron and K. Urban, *Wino potential and Sommerfeld effect at NLO*, *Phys. Lett.* **B800** (2020) 135112 [1909.04584]

and if you use the Higgsino spectra with the NLO EW potentials

- M. Beneke, C. Hasner, K. Urban and M. Vollmann, *Precise yield of high-energy photons from Higgsino dark matter annihilation*, *JHEP* **03** (2020) 030 [1912.02034]
- K. Urban, *NLO electroweak potentials for minimal dark matter and beyond*, *JHEP* **10** (2021) 136 [2108.07285]

instead.

2 Installation

DMγSpec is written in Python 3 and distributed as free software under the GNU GPLv3 license. In addition to the Python installation, DMγSpec requires the packages

- numpy 1.17.4 [10]
- scipy 1.3.3 [11]

with which it has been tested. For using and accessing the example Jupyter notebook file included in the distribution, in addition to a LaTeXinstallation

- matplotlib 3.1.2 [12]
- Jupyter 1.0.0 [13]

are required. Using different package versions, may result in unexpected behaviour or errors. In this case, please update the packages, which are part of all common Python environment and package management distributions, such as conda, the related Anaconda, or pip/pip3. The packages can also be installed to the required version using conda from the main DMγSpec directory in a Unix shell

```
conda config --add channels conda-forge
conda install --file requirements.txt
```

with the provided requirements.txt package-list. The equivalent command for pip reads

```
pip install -r requirements.txt
```

An example use of loading the packages is shown in the example notebook file accessible, e.g., via

```
jupyter notebook ./example.ipynb
```

in a Unix shell. If DMγSpec is to be used outside of the provided example notebook, the four top-level functions can also be loaded via

```
from resummation import function_name
```

in a Python interpreter, where function_name is to be replaced with the name of the functions discussed below.

3 Documentation functions

The package contains four main functions. All functions accept dark-matter mass values in the range $m_\chi = 500\,\text{GeV}$ to $100\,\text{TeV}$. The functions are generated from grids tabulating the annihilation matrix elements Γ_{IJ} in mass range $0.5 - 100\,\text{TeV}$, and $1-x$ range $2\cdot 10^{-5}$ to 1. The grids are linearly interpolated to avoid difficulties at the sharp features (e.g., γZ-threshold or γW^+W^--threshold). In addition, a zero-bin of common width from endpoint to $1 - x = 2 \cdot 10^{-5}$ (default value) or $1 - x = 0.01$ is implemented. The Sommerfeld factors S_{IJ} are realized as one-dimensional interpolating functions. A quick overview of arguments and outputs is also available via

```
help(function_name)
```

in a Python interpreter.

3.1 diffxsection

Provides the differential cross-section $d(\sigma v)/dx$ in x, for all values, except for $1 - x \leq 2 \cdot 10^{-5}$, where the zero-bin at the absolute endpoint (see below) has to be used. The output is given in units of $[10^{-26}\text{cm}^3/\text{s}]$

```
diffxsection(x,mchi,model,SF)
```

where the function parameters refer to the variable `x` $= E_\gamma/m_\chi$, and `mchi` the dark matter mass in units of TeV. The parameter `model` specifies whether the Higgsino or wino model shall be investigated, possible values are `'wino'` or `'higgsino'` (supplied as a Python string, i.e., including the quotation marks). Finally there is the parameter `SF` that specifies the Sommerfeld table to be used (complete list of possibilities, see below – supplied as a Python string, meaning including quotation marks).
The average evaluation time is about $10^{-4}\,\text{s}$ tested on a Desktop machine with an Intel i3-9100 CPU (3.60GHz) and 8GB of RAM. The evaluation time refers to the average time of 10000 function calls for random values of x, m_χ chosen uniformly in the allowed parameter ranges (as for all functions below).

3.2 cumulxsection

The function cumulates the cross section from the endpoint $x = 1$ to a given $x < 1$, i.e.,

$$\int_{1-x}^{1} dx' \, \frac{d(\sigma v)}{dx'} \, . \tag{3.1}$$

Note that if $1 - x$ falls within the zero-bin, the integration is extended to the zero-bin size (see below). The output is given in units of $[10^{-26}\mathrm{cm}^3/\mathrm{s}]$

```
cumulxsection(x,mchi,model,SF,ZBsize='default',rel=-3)
```

with function arguments as for `diffxsection` above. In addition, there is the optional function value `ZBsize` that if omitted is set to `'default'`. The possible values for this option are either `'default'` corresponding to a zero-bin of $1 - x = 2 \cdot 10^{-5}$ or `'1 %'` for a zero-bin of $1 - x = 0.01$.
Finally, as the function performs numerical integration on the differential cross section over sharp features and several orders of magnitude in $1 - x$, the numerical integration can produce outliers for isolated parameter combinations, if the user chooses mass values off grid (see below). To allow more control, and catch most of these errors the relative error can be specified with the optional parameter `rel`. The parameter corresponds to a relative error requirement 10^n in the underlying `scipy.integrate.quad` routine (see also the `scipy.integrate` documentation, parameter `epsrel`). The default is $n = -3$ if no argument is provided. The average evaluation time (specifications as for `diffxsection` above) is about 0.05 s for the small and 0.02 s for the large zero-bin.

3.3 binnedxsection

Similar function to `cumulxsection` above, however, for a chosen energy bin from E_1 to E_2 with $E_1 < E_2$, i.e.,

$$\int_{E_1/m_\chi}^{E_2/m_\chi} dx \frac{d(\sigma v)}{dx} \, . \tag{3.2}$$

The output of

```
binnedxsection(mchi,E1,E2,model,SF,ZBsize='default',rel=-3)
```

has units of $[10^{-26}\mathrm{cm}^3/\mathrm{s}]$ and the function parameters are analogue to `cumulxsection` above, with the addition of $E_1 < E_2 \leq m_\chi$ both given in units of TeV in exchange for `x`. The average evaluation time (specifications as for `diffxsection` above) is about 0.02 s for both zero-bin prescriptions.

3.4 zerobin

To allow for the inclusion of the absolute endpoint and virtual corrections at the endpoint from $\chi^0\chi^0 \to \gamma\gamma$, a zero-bin has to be provided. The zero-bin ranges are $1 - x = 0$ to $1 - x = 2 \cdot 10^{-5}$ (`'default'`) or $1 - x = 0$ to $1 - x = 0.01$ (`'1 %'`). The function is

identifier	potential	δm_χ	vel. v
`'LO -3'`	LO	164.1 MeV	10^{-3}
`'LO -4'`	LO	164.1 MeV	10^{-4}
`'LO -5'`	LO	164.1 MeV	10^{-5}
`'NLO -3'`	NLO	164.1 MeV	10^{-3}
`'NLO -4'`	NLO	164.1 MeV	10^{-4}
`'NLO -5'`	NLO	164.1 MeV	10^{-5}

Table 1: Available wino Sommerfeld-factor tables.

identifier	potential	δm_χ (χ_1^0 vs. χ^+)	δm_N (χ_1^0 vs. χ_2^0)	vel. v
`'LO -3 dm 355 dmN 20'`	LO	355 MeV	20 MeV	10^{-3}
`'LO -4 dm 355 dmN 20'`	LO	355 MeV	20 MeV	10^{-4}
`'LO -5 dm 355 dmN 20'`	LO	355 MeV	20 MeV	10^{-5}
`'NLO -3 dm 355 dmN 20'`	NLO	355 MeV	20 MeV	10^{-3}
`'NLO -4 dm 355 dmN 20'`	NLO	355 MeV	20 MeV	10^{-4}
`'NLO -5 dm 355 dmN 20'`	NLO	355 MeV	20 MeV	10^{-5}
`'LO -3 dm 355 dmN 015'`	LO	355 MeV	0.15 MeV	10^{-3}
`'LO -4 dm 355 dmN 015'`	LO	355 MeV	0.15 MeV	10^{-4}
`'LO -5 dm 355 dmN 015'`	LO	355 MeV	0.15 MeV	10^{-5}
`'NLO -3 dm 355 dmN 015'`	NLO	355 MeV	0.15 MeV	10^{-3}
`'NLO -4 dm 355 dmN 015'`	NLO	355 MeV	0.15 MeV	10^{-4}
`'NLO -5 dm 355 dmN 015'`	NLO	355 MeV	0.15 MeV	10^{-5}

Table 2: Available Higgsino Sommerfeld-factor tables.

```
zerobin(mchi,model,SF,ZBsize='default')
```

which produces an output in units of $[10^{-26}\mathrm{cm}^3/\mathrm{s}]$. The function arguments are as for `cumulxsection` above. The average evaluation time (specifications as for `diffxsection` above) is about 10^{-4} s.

3.5 Available Sommerfeld-factor tables

The available Sommerfeld-factor tables can be called by the identifiers listed in Table 1 (wino) and Table 2 (Higgsino). The columns specify the EW Sommerfeld potentials, the charged-neutral mass-splitting, the neutral-neutral mass-splitting (Higgsino only), and the single-particle velocity of the lightest DM particle v.

3.6 Cross-section assembly – Sommerfeld and annihilation matrices

The `DM`γ`Spec` code is based on a reference implementation of the results in [1, 2] (wino) and [3] (Higgsino) for the differential spectra in a private `Mathematica` code. In addition, the Sommerfeld factors S_{IJ} for NLO EW potentials [6] (wino) and [8] (Higgsino) are used. In order to provide a fast and stable spectrum generation, the results of the `Mathematica`

reference code are interpolated and function as input for `DMγSpec` .
`DMγSpec` takes tables of the individual Sommerfeld factor elements S_{IJ} and annihilation matrices $\tilde{\Gamma}_{IJ}$ and interpolates them.[1] Many of these elements are redundant and also restricted to be real, e.g. diagonal elements are real, and the off-diagonal ones related via complex conjugation. To avoid redudancy in the interpolation grids, in the wino case, we tabulate only the following elements (all real numbers)

$$\begin{aligned}
&S_{(00)(00)},\ \mathrm{Re}\, S_{(00)(+-)},\ \mathrm{Im}\, S_{(00)(+-)},\ S_{(+-)(+-)} \\
&\tilde{\Gamma}_{(00)(00)},\ \mathrm{Re}\, \tilde{\Gamma}_{(00)(+-)},\ \mathrm{Im}\, \tilde{\Gamma}_{(00)(+-)},\ \tilde{\Gamma}_{(+-)(+-)}\,.
\end{aligned} \tag{3.3}$$

The cross section is then assembled according to

$$\begin{aligned}
\frac{d(\sigma v)}{dx} = 2\Big\{ & S_{(00)(00)}\tilde{\Gamma}_{(00)(00)} + 2\,\mathrm{Re}\left[S_{(00)(+-)}\right]\mathrm{Re}\left[\tilde{\Gamma}_{(00)(+-)}\right] \\
& -2\,\mathrm{Im}\left[S_{(00)(+-)}\right]\mathrm{Im}\left[\tilde{\Gamma}_{(00)(+-)}\right] + S_{(+-)(+-)}\tilde{\Gamma}_{(+-)(+-)}\Big\}\,.
\end{aligned} \tag{3.4}$$

For the Higgsino, there are more elements, but even more redundancy in the annihilation matrix, as the elements do not know any difference between the $(11) = \chi_1^0\chi_1^0$ and $(22) = \chi_2^0\chi_2^0$ states, and hence $\Gamma_{(11)(11)} = \Gamma_{(22)(22)} = \Gamma_{(11)(22)} = \Gamma_{(22)(11)}$ and $\Gamma_{(11)(+-)} = \Gamma_{(22)(+-)} = \Gamma^*_{(+-)(11)} = \Gamma^*_{(+-)(22)}$. The elements tabulated are

$$\begin{aligned}
&S_{(11)(11)},\ \mathrm{Re}\, S_{(11)(22)},\ \mathrm{Im}\, S_{(11)(22)},\ \mathrm{Re}\, S_{(11)(+-)},\ \mathrm{Im}\, S_{(11)(+-)},\ S_{(22)(22)}, \\
&\mathrm{Re}\, S_{(22)(+-)},\ \mathrm{Im}\, S_{(22)(+-)}, S_{(+-)(+-)} \\
&\tilde{\Gamma}_{(11)(11)},\ \mathrm{Re}\, \tilde{\Gamma}_{(11)(+-)},\ \mathrm{Im}\, \tilde{\Gamma}_{(11)(+-)},\ \tilde{\Gamma}_{(+-)(+-)}\,.
\end{aligned} \tag{3.5}$$

The cross-section is assembled according to

$$\begin{aligned}
\frac{d(\sigma v)}{dx} = 2\Big\{ & \left[S_{(11)(11)} + 2\mathrm{Re}S_{(11)(22)} + S_{(22)(22)}\right]\tilde{\Gamma}_{(11)(11)} \\
& + \left[2\mathrm{Re}S_{(11)(+-)} + 2\mathrm{Re}S_{(22)(+-)}\right]\mathrm{Re}\tilde{\Gamma}_{(11)(+-)} \\
& - \left[2\mathrm{Im}S_{(11)(+-)} + 2\mathrm{Im}S_{(22)(+-)}\right]\mathrm{Im}\tilde{\Gamma}_{(11)(+-)} \\
& + S_{(+-)(+-)}\tilde{\Gamma}_{(+-)(+-)}\Big\}
\end{aligned} \tag{3.6}$$

We save the values of $\tilde{\Gamma}_{IJ}$ to six digit accuracy with respect to the `Mathematica` reference, i.e., far beyond the percent to permille level accuracy assumed for the NLL' calculation. The reason is merely one of convenience, namely to control the grid size to an acceptable level and make evaluation of the interpolation reasonably fast.

3.7 List of exact mass values

The interpolation grids are produced from the private `Mathematica` reference code that generates the differential spectra. For the mass values of the grid, the interpolation in

[1] S_{IJ} and Γ_{IJ} are defined in the references above. Note that $\tilde{\Gamma}_{IJ} = m_\chi \Gamma_{IJ}$ is tabulated to be differential in the endpoint variable $x = E_\gamma/m_\chi$.

DMγSpec is essentially equivalent to the Mathematica reference code and produces minimal interpolation errors. The list of these mass values is (all in TeV – 500 data points)

$$0.5, 0.505, 0.51, 0.515, 0.52, 0.525, 0.53, 0.535, 0.54, 0.545, 0.55, 0.555, 0.56, 0.565, 0.57, 0.575, 0.58,$$
$$0.585, 0.59, 0.595, 0.6, 0.605, 0.61, 0.615, 0.62, 0.625, 0.63, 0.635, 0.64, 0.645, 0.65, 0.655, 0.66, 0.665,$$
$$0.67, 0.675, 0.68, 0.685, 0.69, 0.695, 0.7, 0.705, 0.71, 0.715, 0.72, 0.725, 0.73, 0.735, 0.74, 0.745, 0.75,$$
$$0.755, 0.76, 0.765, 0.77, 0.775, 0.78, 0.785, 0.79, 0.795, 0.8, 0.805, 0.81, 0.815, 0.82, 0.825, 0.83, 0.835,$$
$$0.84, 0.845, 0.85, 0.855, 0.86, 0.865, 0.87, 0.875, 0.88, 0.885, 0.89, 0.895, 0.9, 0.905, 0.91, 0.915, 0.92,$$
$$0.925, 0.93, 0.935, 0.94, 0.945, 0.95, 0.955, 0.96, 0.965, 0.97, 0.975, 0.98, 0.985, 0.99, 1., 1.01, 1.02, 1.03,$$
$$1.04, 1.05, 1.06, 1.07, 1.08, 1.09, 1.1, 1.11, 1.12, 1.13, 1.14, 1.15, 1.16, 1.17, 1.18, 1.19, 1.2, 1.21, 1.22,$$
$$1.23, 1.24, 1.25, 1.26, 1.27, 1.28, 1.29, 1.3, 1.31, 1.32, 1.33, 1.34, 1.35, 1.36, 1.37, 1.38, 1.39, 1.4, 1.41,$$
$$1.42, 1.43, 1.44, 1.45, 1.46, 1.47, 1.48, 1.49, 1.5, 1.51, 1.52, 1.53, 1.54, 1.55, 1.56, 1.57, 1.58, 1.59, 1.6,$$
$$1.61, 1.62, 1.63, 1.64, 1.65, 1.66, 1.67, 1.68, 1.69, 1.7, 1.71, 1.72, 1.73, 1.74, 1.75, 1.76, 1.77, 1.78, 1.79,$$
$$1.8, 1.81, 1.82, 1.83, 1.84, 1.85, 1.86, 1.87, 1.88, 1.89, 1.9, 1.91, 1.92, 1.93, 1.94, 1.95, 1.96, 1.97, 1.98,$$
$$1.99, 2., 2.02, 2.04, 2.06, 2.08, 2.1, 2.12, 2.14, 2.16, 2.18, 2.2, 2.22, 2.24, 2.26, 2.28, 2.3, 2.32, 2.34, 2.36,$$
$$2.38, 2.4, 2.42, 2.44, 2.46, 2.48, 2.5, 2.52, 2.54, 2.56, 2.58, 2.6, 2.62, 2.64, 2.66, 2.68, 2.7, 2.72, 2.74, 2.76,$$
$$2.78, 2.8, 2.82, 2.84, 2.86, 2.88, 2.9, 2.92, 2.94, 2.96, 2.98, 3., 3.02, 3.04, 3.06, 3.08, 3.1, 3.12, 3.14, 3.16,$$
$$3.18, 3.2, 3.22, 3.24, 3.26, 3.28, 3.3, 3.32, 3.34, 3.36, 3.38, 3.4, 3.42, 3.44, 3.46, 3.48, 3.5, 3.52, 3.54, 3.56,$$
$$3.58, 3.6, 3.62, 3.64, 3.66, 3.68, 3.7, 3.72, 3.74, 3.76, 3.78, 3.8, 3.82, 3.84, 3.86, 3.88, 3.9, 3.92, 3.94, 3.96,$$
$$3.98, 4., 4.05, 4.1, 4.15, 4.2, 4.25, 4.3, 4.35, 4.4, 4.45, 4.5, 4.55, 4.6, 4.65, 4.7, 4.75, 4.8, 4.85, 4.9, 4.95, 5.,$$
$$5.05, 5.1, 5.15, 5.2, 5.25, 5.3, 5.35, 5.4, 5.45, 5.5, 5.55, 5.6, 5.65, 5.7, 5.75, 5.8, 5.85, 5.9, 5.95, 6., 6.05,$$
$$6.1, 6.15, 6.2, 6.25, 6.3, 6.35, 6.4, 6.45, 6.5, 6.55, 6.6, 6.65, 6.7, 6.75, 6.8, 6.85, 6.9, 6.95, 7., 7.1, 7.3, 7.4,$$
$$7.6, 7.7, 7.8, 8., 8.1, 8.3, 8.5, 8.6, 8.8, 9., 9.1, 9.3, 9.5, 9.7, 9.9, 10., 10.2, 10.4, 10.6, 10.8, 11., 11.3, 11.5,$$
$$11.7, 11.9, 12.1, 12.4, 12.6, 12.9, 13.1, 13.4, 13.6, 13.9, 14.1, 14.4, 14.7, 15., 15.3, 15.5, 15.8, 16.1, 16.5,$$
$$16.8, 17.1, 17.4, 17.8, 18.1, 18.4, 18.8, 19.2, 19.5, 19.9, 20.3, 20.7, 21.1, 21.5, 21.9, 22.3, 22.7, 23.2, 23.6,$$
$$24.1, 24.5, 25., 25.5, 26., 26.5, 27., 27.5, 28., 28.5, 29.1, 29.7, 30.2, 30.8, 31.4, 32., 32.6, 33.2, 33.9, 34.5,$$
$$35.2, 35.9, 36.5, 37.2, 38., 38.7, 39.4, 40.2, 41., 41.7, 42.5, 43.4, 44.2, 45., 45.9, 46.8, 47.7, 48.6, 49.5,$$
$$50.5, 51.4, 52.4, 53.4, 54.5, 55.5, 56.6, 57.6, 58.8, 59.9, 61., 62.2, 63.4, 64.6, 65.8, 67.1, 68.4, 69.7, 71.,$$
$$72.4, 73.8, 75.2, 76.6, 78.1, 79.6, 81.1, 82.7, 84.3, 85.9, 87.5, 89.2, 90.9, 92.7, 94.5, 96.3, 98.1, 100. \quad (3.7)$$

The grid in $1-x$ spans 3001 values logarithmically distributed between $1-x = 2 \cdot 10^{-5}$ and $1-x = 1$.

4 Validation of the grids

As mentioned, for the 500 mass points of the grid in DMγSpec (3.7), the interpolation in DMγSpec is essentially exact compared to the private Mathematica reference version. In order to further validate the grid, we go beyond these 500 mass points and generate random mass values in between the exact grid values. We use 249 points distributed with a focus on the low mass region. A complete list of the validated mass points can be found in the package folder, subfolder `validation` in the file `mass_values.csv`. We create the spectra with DMγSpec and with the Mathematica reference, where the latter is taken as the truth. Furthermore, we use randomly selected $1-x$ points for the functions with an energy-resolution dependence. The selected values are distributed in the decades

between zero-bin and $1-x=1$ (50 in total), and can be found in the package subfolder `validation` in the file `oneminx_values.csv`.
The results show very good agreement except for a few outliers, mainly for the small default zero-bin, in the vicinity of the sharp features γZ and γWW threshold, when they move into the zero-bin. For the `'1 % '` zero-bin these thresholds are already part of the zero-bin. In any case, in case of doubt, there are always the 500 mass points given above, for which the results are exact, and there are no discrepancies regardless of the chosen zero-bin between `Mathematica` reference and `DM`γ`Spec` .
For concreteness, we discuss in the following the results of the validation using the NLO EW potential for the interpolation of S_{IJ} at $v=10^{-3}$. We also checked all cases with the LO potential, and the differences in the error are in the fourth to sixth digit relative to the error extracted for the NLO potential, i.e., always on the sub-permille level. This also validates the Sommerfeld tables, as the individual entries of S_{IJ} differ by a large amount between the two potential choices. The bulk of the interpolation error therefore stems from the two-dimensional $\tilde{\Gamma}_{IJ}(m_\chi, x)$ grid as expected. We discuss mean and median deviation to identify if single outliers dominate the average error or if there are dangerous regions where the interpolation does not meet the requirements set. For definiteness, we choose the relative error parameter `rel` $=-5$ for `cumulxsection`, and `binnedxsection` (see above) in the validation.

4.1 Zero-bin validation

We consider the two possible zero-bin sizes of $1-x=2\cdot 10^{-5}$ and $1-x=0.01$. The results for the latter are given in the brackets.
For wino DM, the deviations for the zero-bin are 0.6 (0.1) permille mean deviation and 0.05 (0.04) permille median deviation. The maximal error is found at the mass 10.12 TeV (7.51 TeV) with a deviation of 6.3% (0.06%). For the default zero-bin this cuts into the resummed Z-pole located at $2\cdot 10^{-5}=m_Z^2/(4m_\chi^2)$, i.e. at a mass value of $m_\chi=10.195\,\text{TeV}$ close to the maximum error mass value. Overall only 5 (0) of the 249 probed mass values show a deviation of more than 3 permille, all located in the merging region of Z-pole and zero-bin.
For Higgsino DM (Sommerfeld factor $\delta m_N=20\,\text{MeV}$ and $v=10^{-3}$), the deviations for the zero-bin are 0.15 (0.11) permille mean deviation and 0.05 (0.05) permille median deviation. The maximal error is found at the mass 10.12 TeV (7.51 TeV) with a deviation of 4.7% (0.06%). Overall only 2 (0) of the 249 probed mass values show a deviation of more than 3 permille, again all located in the merging region of γZ-threshold and zero-bin.

4.2 Differential cross section

Similarly, we check the differential cross-section for all randomly chosen mass and $1-x$ values. For wino DM, we find a mean deviation of 0.6 permille and a median deviation of 0.07 permille. However, there are a few outliers for which the deviation grows into the percent region. The largest is found at $m_\chi\approx 7.83\,\text{TeV}$ at $1-x\approx 3.37\cdot 10^{-5}$, with about 25.8 percent deviation. The value is very close to the Z-pole, which for this mass value is located at $1-x\approx 3.37\cdot 10^{-5}$.

In total, of the checked 12450 points, only 334 show a deviation of larger than 3 permille. Also, the deviations are generically larger around the Z-pole. Away from the Z-threshold, deviations are in most cases below 3 permille, typically even sub-permille. Above $1-x \geq 0.01$, which is the region relevant if the "conservative" 1% zero-bin is used, there are typically no deviations that are larger than 3 permille at any given mass or $1-x$ value. In summary, the by far largest errors are induced by the interpolation of the Z-threshold in $\tilde{\Gamma}_{IJ}$. Further increasing the size of the grid might lower these deviations slightly. However, the limiting factor is the interpolation in $1-x$. For most mass values, the accuracy is more than acceptable, also considering that the spectra are essentially exact at the 500 grid points in mass. If one is unsure about cumulating errors, the large 1% zero-bin can be used that always contains the γZ-threshold inside the zero-bin, avoiding any interpolation issues.
For Higgsino DM, the situation is similar. We find a mean deviation of 0.6 permille and a median deviation of 0.07 permille. However, there are a few outliers where the deviation grows into the percent region. The largest is found at $m_\chi \approx 7.83\,\text{TeV}$ at $1-x \approx 3.37\cdot 10^{-5}$, with about 25.6 percent deviation, similar to the wino above. Of the same checked points, only 305 show the deviation larger than 3 permille, of which, as for the wino above, all located in the region where Z-pole and zero-bin merge. The usage of the "conservative" zero-bin of `'1 %'`, which restricts the needed differential cross-section values to $1-x$ between 0.01 and 1, circumvents any interpolation issues.

4.3 Cumulative cross sections

The cumulative cross section in the wino case, shows the largest deviation from the `Mathematica` results at a mass of 4.03 TeV (6.94 TeV) and a value of $1-x \approx 0.000125$ (0.93), with 2.7% (1.4%) for wino DM. The outliers arise from the integration over sharp features or as in the case of the large zero-bin from the accumulation of errors when integrating to a nearly inclusive spectrum (e.g., at $1-x \approx 0.93$). The mean deviation is 0.7 (0.7) permille for the checked 12450 points with a median deviation of 0.3 (0.3) permille. About 4.3% (3%) of the points show a deviation of larger than 3 permille, and 16 out of 12450 (1 point out of 5976) deviate by more than a percent compared to the `Mathematica` reference code.
In the Higgsino case, we find the largest deviation at a mass of 5.527 TeV (4.605 TeV) and a value of $1-x \approx 0.165\,(0.014)$, with 1.2% (1.3%). The mean deviation is 0.2 (0.2) permille for the checked 12450 points with a median deviation of 0.2 (0.2) permille. Only 6 out of 12450 (1 out of 5976) points show a deviation of larger than 3 permille compared to the `Mathematica` reference code.

4.4 Binned cross sections

Finally, we investigate the errors for the binned cross sections. In the wino case, the largest deviation for all bins is about 9.5% at a mass value of $m_\chi = 0.92\,\text{TeV}$ in the bin from $1-x \approx [0.000125, 0.000130]$.[2] The origin of the large relative error in this bin is the

[2]Bins are chosen between the values in `oneminx_values.csv` in the folder `validation`.

very small cross section value, as it is narrow and located below the γZ-threshold. The mean deviation for the binned wino cross section is about 0.4 permille, and the median is even as small as 0.07 permille, so practically negligible, except for the outliers. Of the checked 12201 points, 276 show a deviation of more than 3 permille and only 96 of more than one percent. The points with the largest deviations typically lie below the zero-bin of $1 - x = 0.01$ (i.e., have very small bin size) and hence can be avoided by using the "conservative" 1% zero-bin. For Higgsino DM, we find similar numbers and values for the largest deviations. All interpolation and integration issues are again circumvented by using the large zero-bin.
Note that even though this function also has the option to change the zero-bin, due to the binned format, the validation with the smallest zero-bin automatically covers also the case with a larger zero-bin.

4.5 Summary

In conclusion, the interpolation for wino and Higgsino DM is of similar quality and, on the permille level, accurate for all functions, excluding a few outliers. Using the large zero-bin essentially removes all uncertainties between the `Mathematica` reference code and `DM`γ`Spec` . For both models, most of the errors occur at roughly the same mass values, close to the merging of Z-pole with the zero-bin, further emphasizing that they occur for a very small set of mass m_χ and $1 - x$ values. In addition, we again emphasize that there is also the option to use the 500 mass points (3.7), for which the spectra and all other functions are exact regardless of the chosen zero-bin.

References

[1] M. Beneke, A. Broggio, C. Hasner and M. Vollmann, *Energetic γ-rays from TeV scale dark matter annihilation resummed*, *Phys. Lett. B* **786** (2018) 347–354 [1805.07367]. [Erratum: Phys.Lett.B 810, 135831 (2020)].

[2] M. Beneke, A. Broggio, C. Hasner, K. Urban and M. Vollmann, *Resummed photon spectrum from dark matter annihilation for intermediate and narrow energy resolution*, *JHEP* **08** (2019) 103 [1903.08702]. [Erratum: JHEP 07, 145 (2020)].

[3] M. Beneke, C. Hasner, K. Urban and M. Vollmann, *Precise yield of high-energy photons from Higgsino dark matter annihilation*, *JHEP* **03** (2020) 030 [1912.02034].

[4] P. Ciafaloni, D. Comelli, A. Riotto, F. Sala, A. Strumia and A. Urbano, *Weak Corrections are Relevant for Dark Matter Indirect Detection*, *JCAP* **03** (2011) 019 [1009.0224].

[5] M. Cirelli, G. Corcella, A. Hektor, G. Hutsi, M. Kadastik, P. Panci, M. Raidal, F. Sala and A. Strumia, *PPPC 4 DM ID: A Poor Particle Physicist Cookbook for Dark Matter Indirect Detection*, *JCAP* **03** (2011) 051 [1012.4515]. [Erratum: JCAP 10, E01 (2012)].

[6] M. Beneke, R. Szafron and K. Urban, *Wino potential and Sommerfeld effect at NLO*, *Phys. Lett.* **B800** (2020) 135112 [1909.04584].

[7] M. Beneke, R. Szafron and K. Urban, *Sommerfeld-corrected relic abundance of wino dark matter with NLO electroweak potentials*, *JHEP* **02** (2021) 020 [2009.00640].

[8] K. Urban, *NLO electroweak potentials for minimal dark matter and beyond*, *JHEP* **10** (2021) 136 [2108.07285].

[9] M. Beneke, K. Urban and M. Vollmann, *Matching resummed endpoint and continuum γ-ray spectra from dark-matter annihilation*, 22xx.xxxx.

[10] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del R'ıo, M. Wiebe, P. Peterson, P. G'erard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke and T. E. Oliphant, *Array programming with NumPy*, *Nature* **585** (Sept., 2020) 357–362.

[11] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt and SciPy 1.0 Contributors, *SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python*, *Nature Methods* **17** (2020) 261–272.

[12] J. D. Hunter, *Matplotlib: A 2d graphics environment*, *Computing in Science & Engineering* **9** (2007), no. 3 90–95.

[13] T. Kluyver, B. Ragan-Kelley, F. Pérez, B. Granger, M. Bussonnier, J. Frederic, K. Kelley, J. Hamrick, J. Grout, S. Corlay, P. Ivanov, D. Avila, S. Abdalla and C. Willing, *Jupyter notebooks – a publishing format for reproducible computational workflows*, in *Positioning and Power in Academic Publishing: Players, Agents and Agendas* (F. Loizides and B. Schmidt, eds.), pp. 87 – 90, IOS Press, 2016.