



ELMAG 3.01: A three-dimensional Monte Carlo simulation of electromagnetic cascades on the extragalactic background light and in magnetic fields^{☆,☆☆}

M. Blytt^a, M. Kachelrieß^{a,*}, S. Ostapchenko^{b,c}

^a Institutt for fysikk, NTNU, Trondheim, Norway

^b Frankfurt Institute for Advanced Studies, Frankfurt, Germany

^c D. V. Skobel'syn Institute of Nuclear Physics, Moscow State University, Russia

ARTICLE INFO

Article history:

Received 30 September 2019

Received in revised form 13 December 2019

Accepted 31 December 2019

Available online 20 January 2020

Keywords:

Electromagnetic cascades

Extragalactic background light

Extragalactic magnetic fields

ABSTRACT

The version 3.01 of ELMAG, a Monte Carlo program for the simulation of electromagnetic cascades initiated by high-energy photons and electrons interacting with extragalactic background light (EBL), is presented. Pair production and inverse Compton scattering on EBL photons as well as synchrotron losses are implemented using weighted sampling of the cascade development. New features include, among others, the implementation of turbulent extragalactic magnetic fields and the calculation of three-dimensional electron and positron trajectories, solving the Lorentz force equation. As final result of the three-dimensional simulations, the program provides two-dimensional source images as function of the energy and the time delay of secondary cascade particles.

Program summary

Program Title: ELMAG 3.01

Program Files doi: <http://dx.doi.org/10.17632/gsr7dfnyx6.1>

Licensing provisions: CC by NC 3.0.

Programming language: Fortran 90

Supplementary material: See <http://elmag.sourceforge.net/>

Nature of problem: Calculation of secondaries produced by electromagnetic cascades on the extragalactic background light (EBL), including deflections in magnetic fields.

Solution method: Monte Carlo simulation of pair production and inverse Compton scattering on EBL photons; weighted sampling of the cascading secondaries; recording of energy, observation angle and time delay of secondary particles at the present epoch in a 1.5-dimensional approximation or of sky-maps in a three-dimensional approach.

Restrictions: Using the three-dimensional approach, the energies of cascade photons should be above 1 GeV for $B_{\text{rms}} \lesssim 10^{-10}$ G.

© 2020 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

In Ref. [1], the Monte Carlo program ELMAG for the simulation of electromagnetic cascades initiated by high-energy photons and electrons interacting with extragalactic background light (EBL) was introduced. Main purpose of that program has been the calculation of contributions to the diffuse extragalactic gamma-ray background and the study of TeV blazars and the influence

of the extragalactic magnetic field (EGMF) on their spectra. The initial version 1.01 presented in Ref. [1] contained only two EBL models. The spatial structure of turbulent magnetic fields was approximated by patches of a randomly oriented homogeneous field; the size of these patches was chosen to correspond to the desired coherence length of the magnetic field. Deflections and time-delays of charged particles were then calculated in the small-angle approximation, using a random walk picture to account for the varying field orientations in different field patches. As a result of these simplifications, the in general asymmetric two-dimensional source images were approximated by radially symmetric ones. In subsequent versions, mainly minor improvements in the code as well as additional EBL backgrounds were added. However, for potential applications of the program to studies of gamma-ray sources, using the high-quality data

[☆] This paper and its associated computer program are available via the Computer Physics Communication homepage on ScienceDirect (<http://www.sciencedirect.com/science/journal/00104655>).

^{☆☆} The review of this paper was arranged by Prof. Z. Was.

* Corresponding author.

E-mail address: michael.kachelrieß@ntnu.no (M. Kachelrieß).

of future instruments as, e.g., the Cherenkov Telescope Array, a description of the electromagnetic cascades in the intergalactic space as realistic as possible is highly desirable.

The version 3.01 of ELMAG presented here includes as an option to go beyond the limitations of the 1.5-dimensional simulation, and to employ a more realistic description of the turbulent magnetic field. In this approach, the actual trajectories of electrons¹ are calculated solving the Lorentz force equation using a numerical solver with adaptive step-size. The turbulent magnetic field is modeled as an isotropic, divergence-free Gaussian random field with a prescribed power-law spectrum. Alternatively, the user can read-in his own magnetic field defined on a chosen grid. With the change to the version 3.01, also the input of user data has been updated. We describe the new format, as well as the output of the three-dimensional approach, which consists of two-dimensional sky images visualizing the expected brightness profile of the considered source. These sky images are functions of the energy and the time delay of the secondary cascade particles, and depend moreover on the offset angle of the source with respect to the line-of-sight and the angular profile of the jet.

2. Modeling of three-dimensional cascades

The simulation of pair production and inverse Compton scattering processes on EBL photons, of synchrotron losses as well as the weighted sampling procedure are essentially unchanged compared to version 1.01, for a description of these procedures see Ref. [1]. In the following, we describe only the changes and the additions included for the calculation of three-dimensional cascades.

2.1. Turbulent magnetic field

The calculation of the turbulent magnetic field is based on the algorithm described in Refs. [2,3]. In that approach, the magnetic field $\mathbf{B}(\mathbf{r})$ at the point \mathbf{r} is obtained as a superposition of transverse Fourier modes with left- or right-circular polarization,

$$\mathbf{B}(\mathbf{r}) = \sum_{j=1}^{n_k} B_j [\cos \alpha_j \hat{\mathbf{e}}_{x'} + i h_j \sin \alpha_j \hat{\mathbf{e}}_{y'}] e^{i(k_j z' + \beta_j)}. \quad (1)$$

Here, $\hat{\mathbf{e}}_i$ and $\hat{\mathbf{e}}_{i'}$ denote two sets of Cartesian unit vector, which are connected by a rotation, $\mathbf{r}' = R\mathbf{r}$ with

$$R(\vartheta_j, \phi_j) = \begin{bmatrix} \cos \vartheta_j \cos \phi_j & \cos \vartheta_j \sin \phi_j & -\sin \vartheta_j \\ -\sin \phi_j & \cos \phi_j & 0 \\ \sin \vartheta_j \cos \phi_j & \sin \vartheta_j \sin \phi_j & \cos \vartheta_j \end{bmatrix}. \quad (2)$$

For each mode, two random phases α_j and β_j , two rotation angles ϕ_j and ϑ_j , and the polarization h_j have to be chosen: For an isotropic magnetic field, ϕ_j is uniformly distributed between $[0 : 2\pi]$, while $\cos \vartheta_j$ is uniformly distributed between $[-1 : 1]$. The random phases α_j and β_j are uniformly distributed between $[0 : 2\pi]$. The expectation value $\langle h_j \rangle$ of $h_j = \pm 1$ determines the helicity of the magnetic field, with $|\langle h_j \rangle| = 1$ for a fully left- or right-helical field and $\langle h_j \rangle = 0$ for a field with vanishing helicity. Finally, the amplitude B_j of the mode j , for a power-law spectrum between k_{\min} and k_{\max} with root mean square value B_{rms} and slope γ , is given by

$$B_j = B_{\min} (k_j/k_{\min})^{-\gamma/2}. \quad (3)$$

Here, B_{\min} denotes the field strength of the lowest Fourier mode, which is determined by normalizing the total strength of the turbulent field to B_{rms} .

Note also that the coherence length L_c of a turbulent magnetic field with a power-law like spectrum is connected to its slope and its maximal scale as

$$L_c = \frac{L_{\max}}{2} \frac{\gamma - 1}{\gamma} \frac{1 - (L_{\min}/L_{\max})^\gamma}{1 - (L_{\min}/L_{\max})^{\gamma-1}} \simeq \frac{L_{\max}}{2} \frac{\gamma - 1}{\gamma}, \quad (4)$$

where the approximation is valid for $L_{\max} = 2\pi/k_{\min} \gg L_{\min} = 2\pi/k_{\max}$ and $\gamma > 1$. For a Kolmogorov spectrum with $\gamma = 5/3$, it follows $L_c = L_{\max}/5$, while a steep ($\gamma \gg 1$) or monochromatic ($L_{\min} = L_{\max}$) spectrum leads to $L_c = L_{\max}/2$. Electrons will propagate typically in the ballistic regime for the energies of interest. In this case, $n_k = 200$ field modes distributed over two decades below L_{\max} should be sufficient to ensure an isotropic field with the desired average helicity. Since for deflections of electrons fluctuations with $k \sim 1/R_L$ are most effective, users have to check if the chosen value of k_{\min} is sufficiently low for the photon energies simulated.

2.2. Calculation of trajectories

Fluctuations in the EGMF are typically spread over a wide range of scales, $k_{\min} \ll k_{\max}$. Solving the Lorentz force equation efficiently requires therefore a scheme with an adaptive step-size solver. The solver employed is based on routines from Numerical Recipes [4], using the Runge-Kutta formulas from Fehlberg with the Cash-Karp parameters. The initial step-size h_1 is set to $h_1 = 10^{-3} L_{\text{coh}}$, where L_{coh} denotes the chosen value of the coherence length.

To check the reliability of the numerical scheme, the routine was tested against the formula for a charged particle being deflected in a turbulent magnetic field. An electron with energy E propagating the distance D in a turbulent magnetic field with the coherence length L_c is deflected by the angle [5]

$$\Theta \simeq 0.025^\circ \sqrt{\frac{D}{L_c}} \frac{L_c}{10 \text{ Mpc}} \frac{B_{\text{rms}}}{10^{-11} \text{ G}} \frac{E}{10^{20} \text{ eV}} \simeq 27.3^\circ \frac{\sqrt{DL_c}}{R_L}, \quad (5)$$

where we introduced the Larmor radius $R_L = E/(eB)$ in the last step.

Fig. 1 shows² the deflection angle obtained with ELMAG 3.01 as function of the propagation distance D/L_c for two values of the ratio R_L/L_c . Additionally, the error control parameter eps has been varied between 10^{-5} and 10^{-8} ; the results for different choices of eps were spread out on the x -axis for better visibility. For each case, 100 realizations of the turbulent magnetic field were used to calculate the mean and the variance of the deflection angle. The results are in good agreement with Eq. (5) up to distances $D \sim 100 L_c$. Since the interaction length of electrons is of the order of tens of kpc, i.e. smaller than the typical coherence lengths assumed for the EGMF, we conclude that the numerical precision is sufficient for our purposes.

As an additional test, we have compared our results from the three-dimensional simulation to those from Elyiv et al. [7]. For the comparison, we use a source at the distance 120 Mpc and a primary photon energy distribution

$$E_\gamma^2 dN_\gamma/dE_\gamma \propto \exp(-E_\gamma/E_{\text{max}}), \quad (6)$$

with an exponential cutoff at $E_{\text{max}} = 300 \text{ TeV}$ as an example. The main difference between the two simulations is the definition of the magnetic field. Elyiv et al. divided the magnetic field into cubic cells of length 1 Mpc, where each cell contains a uniform field with random orientation. In contrast, we use a turbulent magnetic field with a Kolmogorov power-spectrum³ and a coherence length of 1 Mpc.

² For a description of additional tests see also Ref. [6].

³ Note that deflections in the magnetic field do not depend on the slope γ , as long as $\gamma > 0$ [8].

¹ We call from now on electrons and positrons collectively electrons.

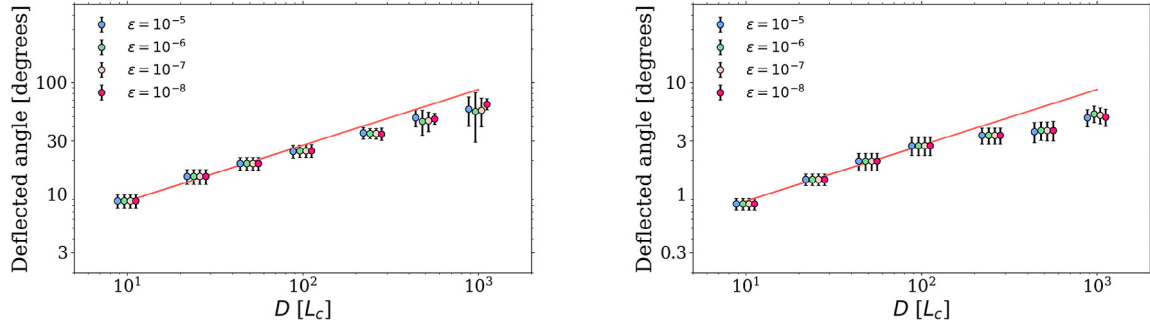


Fig. 1. Deflection angle of electrons as function of the propagation distance D in units of L_c for $R_L = 10L_c$ (left panel) and $R_L = 100L_c$ (right panel) for various values of ϵ . The red line corresponds to the prediction of Eq. (5). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

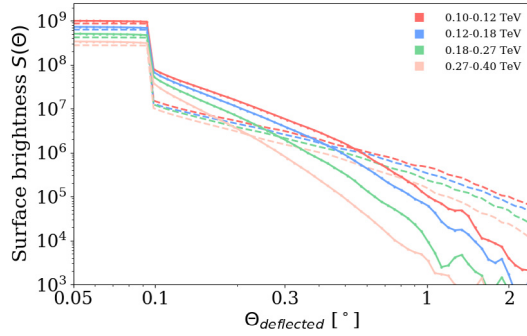


Fig. 2. The surface brightness $S(\theta)$ as function of the deflection angle θ for four different energy ranges; dashed lines for a turbulent magnetic field with $B_{rms} = 10^{-14}$ G, while the solid lines are for $B_{rms} = 10^{-15}$ G.

Fig. 2 shows the surface brightness $S(\theta)$ of this source as a function of the deflection angle θ for two magnetic field strengths. The observed energies of 100,000 photons are summed in four bins, and the normalization is fixed to the same values as the one in Fig. 6 from Ref. [7] for better comparison. The plateau visible between 0.05° and 0.1° is caused by “direct” photons, which are smeared out using a point-spread function fitted to data from Fermi-LAT. The results for the surface brightness $S(\theta)$ are in good agreement with those of Ref. [7]: In particular, the drops in surface brightness from the plateau to the following bin coincide well. Our results show, however, a slightly more gradual decline than the one presented by Elyiv et al. which might be connected to the differences in the magnetic field models employed in the two simulations.

2.3. Geometrical setup of source and observer

The TeV emission from active galactic nuclei (AGN) is usually assumed to be relativistically beamed into a narrow cone with an opening angle $\theta_{jet} \simeq 1/(2\Gamma) \simeq 5^\circ[\Gamma/10]$, where Γ is the bulk Lorentz factor of the γ -ray emitting plasma. Blazars are a special type of γ -ray emitting AGN for which the angle θ_{obs} between the line-of-sight and the jet axis satisfies $\theta_{obs} \lesssim \theta_{jet}$. Since most observed TeV blazars should be off-axis with θ_{obs} close to θ_{jet} , the TeV sky-maps of such sources should be asymmetric.

In version 3.01 of ELMAG, additionally to the opening angle θ_{jet} of the blazar jet, its offset θ_{obs} can be chosen. The position of the observer is set at the origin in the xy -plane and at a distance r_{cmb} from the source in z -direction, independent of θ_{obs} . Thus θ_{obs} defines the initial velocity vector of each particle, cf. with Fig. 3. The particle will be traced until it reaches a distance⁴ $> 0.999r_{cmb}$

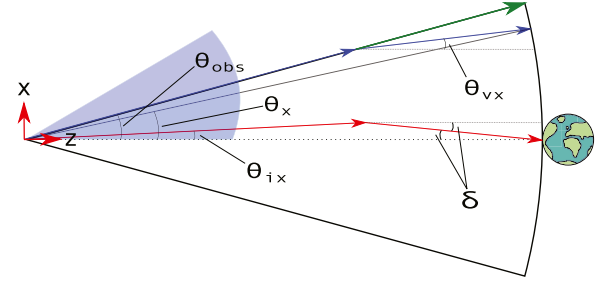


Fig. 3. Sketch of our geometrical setup: The blazar jet shown in blue is offset by $\theta_{obs} = \theta_{jet}$ with respect to the dotted line-of-sight. The green arrow represents the initial starting velocity which is chosen to be in the center of the jet. The blue arrows represent the trajectory of the initial particle, the red arrows the rotated path which hits the Earth. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

from the origin, i.e. until $\sqrt{x^2 + y^2 + z^2} > 0.999r_{cmb}$. After that, it will be projected onto the sphere of radius r_{cmb} . If f denotes the fraction that each final velocity component (v_x, v_y, v_z) has to be multiplied in order to reach the sphere with radius r_{cmb} , then f satisfies

$$r_{cmb}^2 = (x + v_x f)^2 + (y + v_y f)^2 + (z + v_z f)^2. \quad (7)$$

Solving Eq. (7) for f and choosing the smaller absolute value of the two solutions, gives the corrected final position on the sphere, $(x_f, y_f, z_f) = (x + v_x f, y + v_y f, z + v_z f)$. If

$$\theta_{jet} \leq \arctan \left(\frac{\sqrt{x_f^2 + y_f^2}}{z_f} \right) \quad (8)$$

is satisfied, the particle trajectory is rotated to hit the Earth; otherwise the particle is discarded.

Fig. 3 visualizes how the rotation in the two-dimensional xz -plane is performed. The green arrow represents a particle trajectory initialized at $\theta_{obs} = \theta_{jet}$ with no deflection. The blue arrows show a deflected trajectory, with the red arrows representing the deflected track rotated to hit the Earth. The observation angle δ can be obtained from the angles $\theta_{vx} = \arctan(v_x/v_z)$ and $\theta_x = \arctan(x/z)$ by simple geometrical considerations. In order to include a weighted particle distribution inside the jet, the dependence of δ on θ_{ix} has to be known: Mirroring θ_x at the axis defined by $\theta_{obs}/2$ results in $\theta_{ix} = \theta_{obs} - \theta_x$. Observing then that the triangle made by the origin and the two red arrows is the same as the triangle made by the origin and the two blue arrows, one can calculate δ by knowing θ_{obs} , θ_{vx} and θ_{ix} , viz $\delta = \theta_{vx} + (\theta_{ix} - \theta_{obs})$.

⁴ If not otherwise specified, all distances are comoving distances.

2.4. Stacking and weighting

The stack holds all secondary particles that have yet to be calculated in event. In version 3.01, the definition of event was changed to

```
type one_event
  double precision :: y(6), en, z, w, s, dt
  integer :: icq
end type one_event
type (one_event) act, event(n_max)
```

Using the three-dimensional option, the variable $y(6)$ contains the position and velocity vectors of each particle, while s sums the distance traveled. Using the 1.5-dimensional option, the variables x , $the1$, $the2$, xxc , xx are mapped into $y(1:5)$. Calculating the final time delay of each particle can be done by knowing the total traveled distance when the particle reaches the sphere of radius $rcmb$. The time delay is then $dt = (s - rcmb)/cy$, where cy is the speed of light. However, dt must be included in the stack to account for the kinematic time delay when the particle is massive.

The initial weight of each particle is now a product of its weight w_{e0} in the energy spectrum, its weight w_z in the redshift distribution, and its weight w_{jet} in the jet profile.

3. Program structure

The program is distributed among the files `modules301.f90`, `bturb301.f90`, `user301.f90`, `init301.f90`, `elmag301.f90`, and `aux301.f90`. In addition, the two main routines, `main_sp301.f90` and `main_mpi301.f90`, execute a single processor or a MPI simulation, respectively. The physics contained in the routines of the new file `bturb301.f90` for the generation of the turbulent magnetic field has already been described in Section 2.1. Additionally, the file contains the subroutines `init_Bgrid` and `B_interpolation(x, Omega)` which read in an EGMF model on a user-specified Cartesian grid and interpolate these values, respectively.

The changed or added subroutines and functions in the file `elmag301.f90` which constitutes the core of the program are:

- subroutine `cascade(icq, e00, weight0, z_in)`
Follows the evolution of the cascade initiated by a photon ($icq = 0$) or an electron/positron ($icq = \pm 1$) injected at redshift z_{in} with energy $e00$ and weight $weight0$ until all secondary particles have energies below the energy threshold $ethr$ or reached the sphere of radius $rcmb$. Depending on the logical variable `three_dim`, the routine will calculate three-dimensional trajectories or use the 1.5-dimensional approximation. It will also discard particles whose deflection angle exceeds some chosen value.
- subroutine `propagate(y, x, e0)`
Calculates the trajectory of a charged particle in the turbulent magnetic field by calling `odeint(y, nvar, 0, x, eps, h1, hmin, nok, nbad, derivs, rkqs, tf)` located in the file `aux301.f90`. The routine chooses a suitable initial step-size $h1$, depending on the coherence length of the magnetic field, and the acceptable truncation error eps used in the numerical solver for the trajectory. The particle is tracked for the interaction length x , with y containing the particle's position and velocity $[x, y, z, v_x, v_y, v_z]$. $hmin$ is the minimal step-size for the solver to take and $nvar = 6$ is the number of elements in y . The outputs `nok` and `nbad` are the number of good and bad (but retried and fixed) steps taken. The inputs `derivs()` and `rkqs()` are external routines that will be called from inside `odeint()`. The output `tf` is a

logical variable set to `.false.`, if the solver was not able to solve the ODE given the required truncation error threshold eps . If this happens, the particle will be discarded.

- subroutine `normalizer(y)`
Normalizes the velocity components of y so that the absolute value is one.
- subroutine `derivs(x, y, dydx)`
Is the external subroutine called from `odeint()` and `rkck()`, as a part of the numerical solver for the Lorentz force equation. The routine takes as inputs the position and velocity of the particle in the six-vector y , and outputs its derivatives $dydx$.
- subroutine `angle_delay_3d(y, e0, s, thetax, thetay, dt, w2)`
Determines the time-delay dt and the observation angle in the two-dimensional plane from the perspective of a detector on Earth, as described in Section 2.3. The outputs `thetax` and `thetay` are the δ -value from Fig. 3 for the observed angles in respectively the xz -plane and the yz -plane. The output variable `w_jet` is the weighting variable accounting for a non-uniform jet distribution. The latter is calculated by calling the subroutine `jet_distribution(the_s, w_jet)`, where the desired jet distribution profile is specified.

4. Example input and output

The file `user301.f90` is an example file for the input/output subroutines which should be developed by the user for the desired task. We discuss now the file contained in the distribution as an example of how the results of the three-dimensional simulations can be visualized.

4.1. Example input

In previous versions, input parameters were defined as parameters in the module `user_variables` or were specified in routines like `initial_particle(e0, weight)` and `bemf(r)`. In order to facilitate the scanning of a grid of parameters, most input parameters are now read from the files `input_b`, `input_src` and `input_oth`. The files contain comments which should make their use straight-forward. For instance, the file `input_b` starts as

```
2      ! b_model = 1 uniform, 2 turbulent,
          3 read from file
1.d-15 ! B_rms in Gauss
1d0    ! correlation length L_c/Mpc
...
```

If the option `b_model = 3` is chosen, a user-specified model for the EGMF defined on a Cartesian grid with $n_x \times n_y \times n_z$ points is read in from the file `Bgrid`. The z direction of the grid should agree with the direction to the considered source.

Additionally, some options can be chosen in the following routines of the file `user301.f90`:

- subroutine `initial_particle(e0, weight, z)`
Assigns the initial photon/electron energy: fixed $e0$, sampled from a broken power-law or a user-specified function depending on the value of the parameter `en_dist`. Chooses the initial redshift of a particle, either as fixed $z=z_{max}$ or sampled from a distribution $(1+z)^m$ up to z_{max} , depending on the value of the parameter `z_dist`.
- subroutine `jet_distribution(the_s, w_jet)`
Additional weighting according to a jet distribution, by default a radial Gaussian distribution with weight one at the center and variance `th_jet**2`.

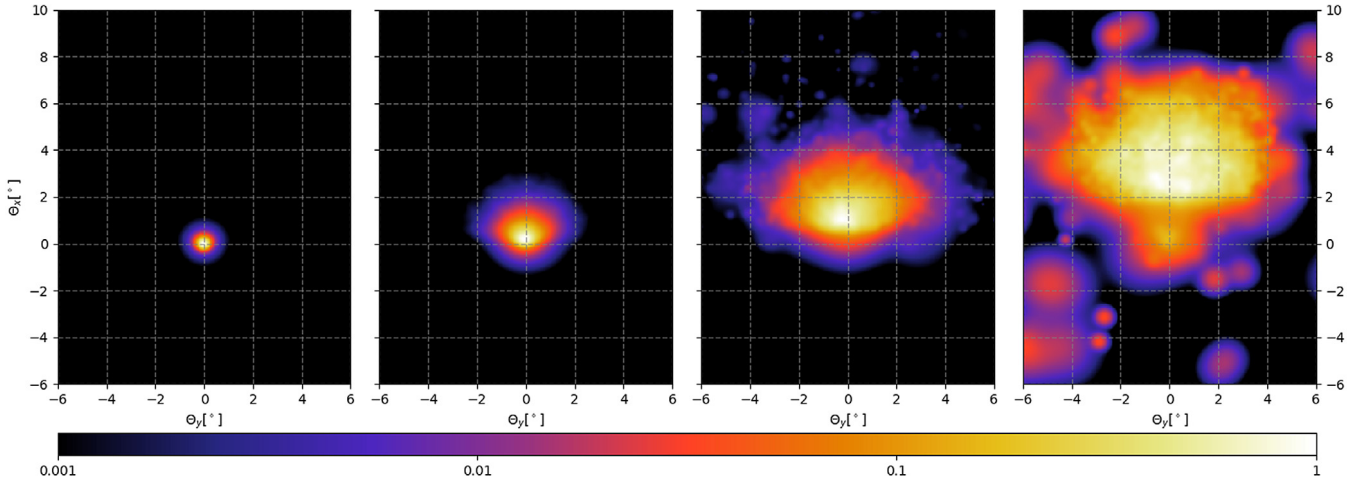


Fig. 4. Surface brightness S for photons with energy $E_\gamma > 1$ GeV in the sky region around a blazar emitting a photon distribution according to Eq. (6); with $\Theta_{\text{obs}} = \Theta_{\text{jet}} = 3^\circ$, and $B_{\text{rms}} = 10^{-14}$ G. The four panels correspond to the time bins 1, 2, 3 and 4. The color code of the surface brightness is shown below. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

- subroutine `psf_spread(e0, thex, they, weight, dt)`
Distributes the detected particle directions according to a PSF, by default a step function with width equal to `theta_reg(e0)`.
- subroutine `user_output(n_max, n_proc)`
Creates (additionally to the output from previous versions 1.1–2.3) in the directory `Output/AngRes` the file `angle_matrix` with the output for the two-dimensional sky-maps. This directory contains also a python routine for the generation of plots.

4.2. Example output and results

After a particle leaves the cascade, it is stored through the subroutine `register(e0, thex, they, weight, dt, icq)`. The three-dimensional array storing the brightness profile from the blazar is `anobs(n_bint, n_binx, n_biny)`, and depends on the parameters `n_bint`, `n_binx`, `n_biny`, `n_bind`, and `shiftx`. These parameters are respectively the number of time bins for the time delay, the number of bins in x - and in y -direction, number of bins per degree, and how many degrees the grid is shifted in x direction. For a brightness profile split into four time bins, with a resolution of 30 bins per degree, ranging in the intervals $x \in [-6^\circ, 12^\circ]$ and $y \in [-6^\circ, 6^\circ]$, one can use the parameters `n_bint=4`, `n_binx=541`, `n_biny=361`, `n_bind=30` and `shiftx=3`.

To take into account the typical characteristics of a detector we use the point spread function (PSF) suggested in Ref. [9], $\vartheta_{95} \simeq 1.68^\circ (E/\text{GeV})^{-0.77} + 0.2^\circ \exp(-10 \text{ GeV}/E)$, where E is the particle energy and ϑ_{95} is the angular containment radius of 95%. This function is an analytical approximation to the PSF from Fermi-LAT, while $\vartheta_{95} = 0.11^\circ$ typical for the PSF of a Cherenkov Telescope is used above 300 GeV. The function `thereg_en(e0)` in the program returns the value of ϑ_{95} . Each particle is mapped in the array `anobs`, through the subroutine `psf_spread(e0, thex, they, weight, dt)`. In this subroutine, the sizes of the bins for the time delay in years can be defined, which are by default: 1: $0 < \tau < 10^5$, 2: $10^5 \leq \tau < 10^6$, 3: $10^6 \leq \tau < 3 \cdot 10^6$, 4: $3 \cdot 10^6 \leq \tau < 10^7$ and 5: $\tau = 0$. This subroutine calls `thereg_en(e0)`, and distributes the brightness according a Gaussian with variance $\sigma \simeq \vartheta_{95}/2$ around the point `(thex, they)`.

All data arrays exist in two versions, e.g. `anobs(n_bint, n_binx, n_biny)` and `anobs_tot(n_bint, n_binx, n_biny)`. Using MPI,⁵ the former arrays contain the result of a single process, which are summed by call `MPI_REDUCE` into `anobs_tot(n_bint, n_binx, n_biny)`,

```
n_array = n_bint*n_binx*n_biny
call MPI_REDUCE(anobs, anobs_tot, n_array,
  MPI_DOUBLE_PRECISION, MPI_SUM, 0, &
  MPI_COMM_WORLD, ierr)
! sum individual arrays spec
```

Finally, the subroutine `user_output(n_max, n_proc)` writes the data arrays with the results to the files contained in the subdirectory `Output`. The files `angle_matrix1` to `angle_matrix4` contain the normalized surface brightness for the time bins 1 to 4. Examples for the sky-maps produced with the Python routine contained in the directory `Output/AngRes` are shown in Figs. 4 and 5. They show the surface brightness around a blazar for two different strengths of the turbulent magnetic field, using one million injected photons.

5. Summary

We have presented the new features in ELMAG 3.01 which include the generation of (helical) turbulent magnetic fields and the three-dimensional tracking of cascade particles. This allows the user to produce two-dimensional sky-images of sources with an arbitrary jet offset angle and jet profile. The format of the new input and output as well as a few tests has been described. In a future release, potential energy losses of electrons caused by plasma instabilities will be included as a new option.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

⁵ For information on MPI see e.g. "Message Passing Interface Forum. MPI: A Message Passing Interface Standard, June 1995" on <http://www.mpi-forum.org>.

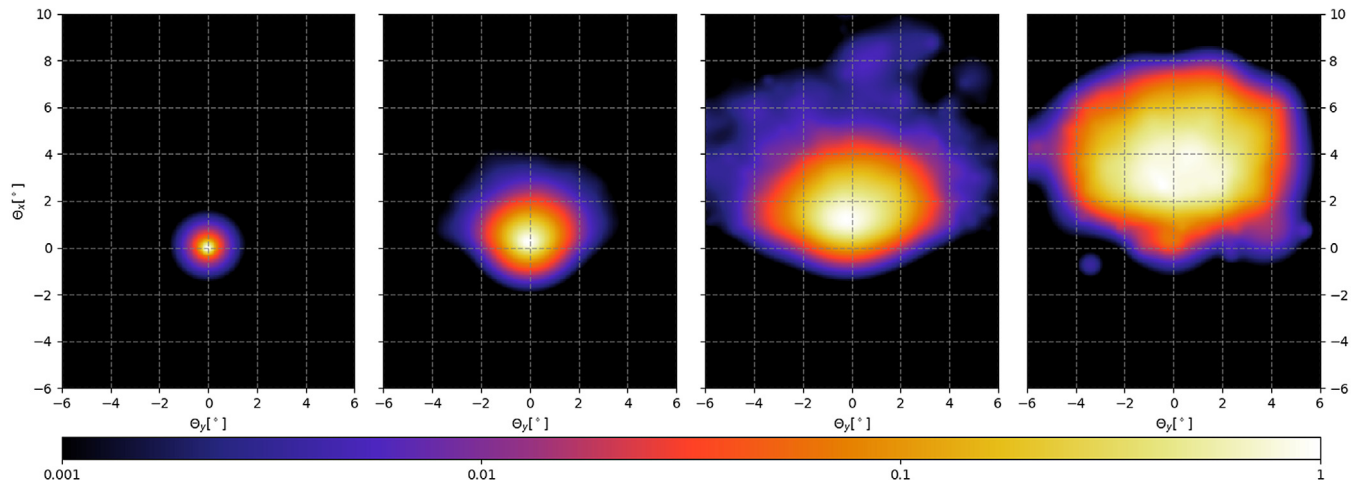


Fig. 5. Surface brightness S around a blazar for the same parameters as in Fig. 4, but with $B_{\text{rms}} = 10^{-15}$ G.

Acknowledgments

We would like to thank Manuel Meyer and Andrey Saveliev for contributing additional EBL backgrounds, and Kumiko Kotera, Manuel Meyer, Marco Muzio and Foteini Oikonomou for useful suggestions which improved ELMAG. S.O. acknowledges support from the project OS 481/2-1 of the Deutsche Forschungsgemeinschaft.

References

- [1] M. Kachelrieß, S. Ostapchenko, R. Tomàs, Comput. Phys. Comm. 183 (2012) 1036–1043, <http://dx.doi.org/10.1016/j.cpc.2011.12.025>, arXiv:1106.5508.
- [2] J. Giacalone, J.R. Jokipii, Astrophys. J. Lett. 430 (1994) L137–L140, <http://dx.doi.org/10.1086/187457>.
- [3] J. Giacalone, J.R. Jokipii, Astrophys. J. 520 (1999) 204–214, <http://dx.doi.org/10.1086/307452>.
- [4] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes in Fortran 77: The Art of Scientific Computing, Vol. 1, second ed., 1996.
- [5] J. Miralda-Escude, E. Waxman, Astrophys. J. 462 (1996) L59–L62, <http://dx.doi.org/10.1086/310042>, arXiv:astro-ph/9601012.
- [6] M. Blytt, Implementation of Three-Dimensional Electromagnetic Cascades into ELMAG (Master's thesis), NTNU Trondheim, 2019, available at <http://hdl.handle.net/11250/2625255>.
- [7] A. Elyiv, A. Neronov, D.V. Semikoz, Phys. Rev. D 80 (2009) 023010, <http://dx.doi.org/10.1103/PhysRevD.80.023010>, arXiv:0903.3649.
- [8] C. Caprini, S. Gabici, Phys. Rev. D 91 (12) (2015) 123514, <http://dx.doi.org/10.1103/PhysRevD.91.123514>, arXiv:1504.00383.
- [9] K. Dolag, M. Kachelrieß, S. Ostapchenko, R. Tomàs, Astrophys. J. 727 (2011) L4, <http://dx.doi.org/10.1088/2041-8205/727/1/L4>, arXiv:1009.1782.