diffsph

Release 1.0.0.

Martin Vollmann, Finn Welzmueller, Lovorka Gajovic

Jan 09, 2024

CONTENTS:

CHAPTER

ONE

DIFFSPH

1.1 diffsph package

1.1.1 Subpackages

diffsph.profiles package

Submodules

diffsph.profiles.analytics module

diffsph.profiles.analytics.**cobrA**(*t*, *rs*, *rh*)

Brightness H-function for the 'constant' top-hat source in the regime-A approximation

Parameters

- **t** – $D\sin(\theta)/r_h$, where θ (theta), r_h (rh) and D (dist) are defined below
- **theta** – angular radius in rad
- **rs** – scale radius
- **rh** – diffusion radius parameter
- **dist** – distance to the source

diffsph.profiles.analytics.**cobrC**(*t*, *rs*, *rh*)

Brightness H-function for the 'constant' source in the regime-C approximation

Parameters

- **t** – $D\sin(\theta)/r_h$, where θ (theta), r_h (rh) and D (dist) are defined below
- **theta** – angular radius in rad
- **rs** – scale radius
- **rh** – diffusion radius parameter

diffsph.profiles.analytics.**cofdA**(*theta*, *rs*, *rh*, *dist*)

Flux-density H-function for the 'constant' top-hat source in the regime-A approximation

Parameters

- **theta** – angular radius in rad
- **rs** – scale radius

- **rh** – diffusion radius parameter
- **dist** – distance to the source

`diffsph.profiles.analytics.`**cofdAmax**(*rs*, *rh*, *dist*)

Maximum value for the flux-density H-function for the 'constant' top-hat source in the regime-A approximation

Parameters

- **rs** – scale radius
- **rh** – diffusion radius parameter
- **dist** – distance to the source

`diffsph.profiles.analytics.`**cofdC**(*theta*, *rs*, *rh*, *dist*)

Flux-density H-function for the 'constant' top-hat source in the regime-C approximation

Parameters

- **theta** – angular radius in rad
- **rs** – scale radius
- **rh** – diffusion radius parameter
- **dist** – distance to the source

`diffsph.profiles.analytics.`**cofdCmax**(*rs*, *rh*, *dist*)

Maximum value for the flux-density H-function for the 'constant' top-hat source in the regime-C approximation

Parameters

- **rs** – scale radius
- **rh** – diffusion radius parameter
- **dist** – distance to the source

`diffsph.profiles.analytics.`**psbrA**(*theta*, *rs*, *rh*, *dist*)

Brigthness H-function for point sources in the regime-A approximation

Parameters

- **theta** – angular radius in rad
- **rs** – scale radius
- **rh** – diffusion radius parameter
- **dist** – distance to the source

`diffsph.profiles.analytics.`**psbrC**(*t*, *rh*)

Brigthness H-function for point sources in the regime-C approximation Variable t is defined as

Parameters

- **t** – $D\sin(\theta)/r_h$, where θ (`theta`), r_h (`rh`) and D (`dist`) are defined below
- **theta** – angular radius in rad
- **rh** – diffusion radius parameter
- **dist** – distance to the source

`diffsph.profiles.analytics.psfdC(theta, rh, dist)`

Flux-density H-function for point sources in the regime-C approximation

Parameters

- **theta** – angular radius in rad
- **rh** – diffusion radius parameter
- **dist** – distance to the source

`diffsph.profiles.analytics.psfdCmax(rh, dist)`

Maximum value for the flux-density H-function for point sources in the regime-C approximation

Parameters

- **rh** – diffusion radius parameter
- **dist** – distance to the source

`diffsph.profiles.analytics.sisbrA(t, sigmav, rh)`

Brightness H-function for the singular isothermal source in the regime-A approximation

Parameters

- **t** – $D\sin(\theta)/r_h$, where θ (theta), r_h (rh) and D (dist) are defined below
- **theta** – angular radius in rad
- **sigmav** – velocity dispersion parameter
- **rh** – diffusion radius parameter
- **dist** – distance to the source

`diffsph.profiles.analytics.sisbrC(t, sigmav, rh)`

Brightness H-function for the singular isothermal source in the regime-C approximation

Parameters

- **t** – $D\sin(\theta)/r_h$, where θ (theta), r_h (rh) and D (dist) are defined below
- **theta** – angular radius in rad
- **sigmav** – velocity dispersion parameter
- **rh** – diffusion radius parameter
- **dist** – distance to the source

diffsph.profiles.hfactors module

`diffsph.profiles.hfactors.D_factor(theta, dist, rad_temp, **kwargs)`

Generic “D” factor

Parameters

- **theta** – angular distance in rad units
- **dist** – distance to earth
- **rad_temp** – radial template

Keyword arguments

Parameters

- **rs** – scale radius
- **rhos** – characteristic density
- **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
- **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
- **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
- **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
- **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
D factor

diffsph.profiles.hfactors.**H_brightness**(*theta*, *dist*, *rh*, *hyp*, *rad_temp*, *regime*, ***kwargs*)

Generic emissivity halo/bulge function in the Regime "A", "B" or "C" approximations

Parameters

- **theta** – angular distance in rad units
- **dist** – distance to earth
- **rh** – diffusion halo/bulge radius
- **hyp** (*str*) – hypothesis: 'wimp' (**default**), 'decay' or 'generic')
- **halo_model** – DM halo model
- **rad_temp** – radial template
- **regime** – regime of the approximation. Must be either upper or lower case a, b, c or I/II/III.

Keyword arguments

Parameters

- **rs** – scale radius
- **rhos** – characteristic density
- **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
- **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
- **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
- **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
- **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
brightness halo/bulge function

diffsph.profiles.hfactors.**H_emissivity**(*r*, *rh*, *hyp*, *rad_temp*, *regime*, ***kwargs*)

Generic emissivity halo/bulge function in the Regime "A", "B" or "C" approximations

Parameters

- **r** – galactocentric distance

- **rh** – diffusion halo/bulge radius
- **hyp** (*str*) – hypothesis: `'wimp'` (**default**), `'decay'` or `'generic'`)
- **rad_temp** – radial template
- **regime** – regime of the approximation (upper/lower case a, b, c or I/II/III).

Keyword arguments

Parameters

- **rs** – scale radius
- **rhos** – characteristic density
- **alpha** – exponent α in the `diffsph.profiles.templates.hdz()` profile
- **beta** – exponent β in the `diffsph.profiles.templates.hdz()` profile
- **gamma** – exponent γ in the `diffsph.profiles.templates.hdz()` profile
- **alphaE** – parameter α_E in the `diffsph.profiles.templates.enst()` profile
- **rc** – core radius parameter r_c in the `diffsph.profiles.templates.cnfw()` profile
- **sigmav** – velocity dispersion parameter σ_v in the `diffsph.profiles.templates.sis()` profile

Returns

emissivity halo/bulge function

`diffsph.profiles.hfactors.`**`H_fluxdens`**(*theta*, *dist*, *rh*, *hyp*, *rad_temp*, *regime*, ***kwargs*)

Generic flux-density halo/bulge function in the Regime "A", "B" or "C" approximations

Parameters

- **theta** – angular distance in rad units
- **dist** – distance to earth
- **rh** – diffusion halo/bulge radius
- **hyp** (*str*) – hypothesis: `'wimp'` (**default**), `'decay'` or `'generic'`)
- **halo_model** – DM halo model
- **rad_temp** – radial template
- **regime** – regime of the approximation. Must be either upper or lower case a, b, c or I/II/III.

Keyword arguments

Parameters

- **rs** – scale radius
- **rhos** – characteristic density
- **alpha** – exponent α in the `diffsph.profiles.templates.hdz()` profile
- **beta** – exponent β in the `diffsph.profiles.templates.hdz()` profile
- **gamma** – exponent γ in the `diffsph.profiles.templates.hdz()` profile
- **alphaE** – parameter α_E in the `diffsph.profiles.templates.enst()` profile
- **rc** – core radius parameter r_c in the `diffsph.profiles.templates.cnfw()` profile

- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
flux density halo/bulge function

`diffsph.profiles.hfactors.`**`H_fluxdens_approx`**(*theta*, *dist*, *rh*, *hyp*, *rad_temp*, *regime*, ***kwargs*)

Generic flux-density halo/bulge function in the Regime "A", "B" or "C" approximations (alternative formula)

Parameters

- **theta** – angular distance in rad units
- **dist** – distance to earth
- **rh** – diffusion halo/bulge radius
- **hyp** (*str*) – hypothesis: 'wimp' (**default**), 'decay' or 'generic')
- **halo_model** – DM halo model
- **rad_temp** – radial template
- **regime** – regime of the approximation. Must be either upper or lower case a, b, c or I/II/III.

Keyword arguments

Parameters

- **rs** – scale radius
- **rhos** – characteristic density
- **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
- **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
- **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
- **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
- **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
flux density halo/bulge function

`diffsph.profiles.hfactors.`**`Hem_A`**(*r*, *rh*, *hyp*, *rad_temp*, ***kwargs*)

Generic emissivity halo/bulge function for Regime A

Parameters

- **r** – galactocentric distance
- **rh** – diffusion halo/bulge radius
- **hyp** (*str*) – hypothesis: 'wimp' (**default**), 'decay' or 'generic')
- **rad_temp** – radial template

Keyword arguments

Parameters

- **rs** – scale radius
- **rhos** – characteristic density

- **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
- **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
- **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
- **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
- **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
emissivity halo/bulge function using the Regime-A approximation

diffsph.profiles.hfactors.**Hem_B**(*r*, *rh*, *hyp*, *rad_temp*, ***kwargs*)

Generic emissivity halo/bulge function for Regime B

Parameters

- **r** – galactocentric distance
- **rh** – diffusion halo/bulge radius
- **hyp** (*str*) – hypothesis: 'wimp' (**default**), 'decay' or 'generic')
- **rad_temp** – radial template

Keyword arguments

Parameters

- **rs** – scale radius
- **rhos** – characteristic density
- **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
- **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
- **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
- **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
- **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
emissivity halo/bulge function using the Regime-B approximation

diffsph.profiles.hfactors.**Hem_C**(*r*, *rh*, *hyp*, *rad_temp*, ***kwargs*)

Generic emissivity halo/bulge function for Regime C

Parameters

- **r** – galactocentric distance
- **rh** – diffusion halo/bulge radius
- **hyp** (*str*) – hypothesis: 'wimp' (**default**), 'decay' or 'generic')
- **rad_temp** – radial template

Keyword arguments

Parameters

- **rs** – scale radius
- **rhos** – characteristic density
- **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
- **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
- **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
- **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
- **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
emissivity halo/bulge function using the Regime-C approximation

`diffsph.profiles.hfactors.`**`J_factor`**`(`*theta*, *dist*, *rad_temp*, ***kwargs*`)`

Generic "J" factor

Parameters

- **theta** – angular distance in rad units
- **dist** – distance to earth
- **rad_temp** – radial template

Keyword arguments

Parameters

- **rs** – scale radius
- **rhos** – characteristic density
- **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
- **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
- **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
- **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
- **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
J factor

`diffsph.profiles.hfactors.`**`halo_factor`**`(`*n*, *rh*, *hyp*, *rad_temp*, ***kwargs*`)`

n-th order halo/bulge factor h_n for a given model (e.g. NFW, Einasto, Plummer, …) Arguments `'n'`, `'rh'`, `'hyp'` and `'rad_temp'` are necessary. Remaining arguments depend on the adopted halo model.

Parameters

- **n** – order of the halo/bulge factor
- **rh** – diffusion halo/bulge radius
- **hyp** (*str*) – hypothesis: `'wimp'` (**default**), `'decay'` or `'generic'`)
- **rad_temp** – radial template

Keyword arguments

Parameters

- **rs** – scale radius
- **rhos** – characteristic density
- **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
- **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
- **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
- **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
- **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns

halo/bulge factor

diffsph.profiles.massmodels module

`diffsph.profiles.massmodels.D`(*tharcmin*, *galaxy*, *rad_temp*, *manual=False*, ***kwargs*)

Model-specific D factor in GeV/cm ^{2}

Parameters

- **tharcmin** – angular radius in arcmin
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – radial template (`'NFW'`, `'Einasto'`, etc.)
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)

Keyword arguments

- `manual = 'False'`

  Parameters

  ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

  Parameters

  - **rs** – scale radius
  - **rhos** – characteristic density
  - **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
  - **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
  - **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
  - **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
  - **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile

- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
D factor

`diffsph.profiles.massmodels.`**`Hbr`**(*tharcmin*, *galaxy*, *rad_temp*, *hyp*, *ratio*, *regime='B'*, *manual=False*, ***kwargs*)

Model-specific brightness halo/bulge function in the Regime "A", "B" or "C" approximations

Parameters

- **tharcmin** – angular radius in arcmin
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – radial template (`'NFW'`, `'Einasto'`, etc.)
- **hyp** (*str*) – hypothesis: `'wimp'` (**default**), `'decay'` or `'generic'`
- **ratio** – ratio between the diffusion halo/bulge and the half-light radius (default value = 1)
- **regime** – regime of the approximation. Must be either upper or lower case a, b, c or I/II/III.
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)

Keyword arguments

- `manual = 'False'`

  Parameters
  ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

  Parameters

  - **rs** – scale radius
  - **rhos** – characteristic density
  - **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
  - **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
  - **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
  - **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
  - **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
  - **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
brightness halo/bulge function

`diffsph.profiles.massmodels.`**`Hem`**(*r*, *galaxy*, *rad_temp*, *hyp*, *ratio*, *regime='B'*, *manual=False*, ***kwargs*)

Model-specific emissivity halo/bulge function in the Regime "A", "B" or "C" approximations

Parameters

- **r** – galactocentric distance in kpc

- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – radial template (`'NFW'`, `'Einasto'`, etc.)
- **hyp** (*str*) – hypothesis: `'wimp'` (**default**), `'decay'` or `'generic'`
- **ratio** – ratio between the diffusion halo/bulge and the half-light radius (default value = 1)
- **regime** – regime of the approximation. Must be either upper or lower case a, b, c or I/II/III.
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)

Keyword arguments

- `manual = 'False'`

  Parameters
  ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

  Parameters

  - **rs** – scale radius
  - **rhos** – characteristic density
  - **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
  - **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
  - **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
  - **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
  - **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
  - **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

  Returns
  emissivity halo/bulge function

`diffsph.profiles.massmodels.`**`Hfd`**(*tharcmin*, *galaxy*, *rad_temp*, *hyp*, *ratio*, *regime='B'*, *manual=False*, ***kwargs*)

Model-specific flux-density halo/bulge function in the Regime "A", "B" or "C" approximations

Parameters

- **tharcmin** – angular radius in arcmin
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – radial template (`'NFW'`, `'Einasto'`, etc.)
- **hyp** (*str*) – hypothesis: `'wimp'` (**default**), `'decay'` or `'generic'`
- **ratio** – ratio between the diffusion halo/bulge and the half-light radius (default value = 1)
- **regime** – regime of the approximation. Must be either upper or lower case a, b, c or I/II/III.
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)

Keyword arguments

- `manual = 'False'`

Parameters
ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

Parameters

- **rs** – scale radius
- **rhos** – characteristic density
- **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
- **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
- **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
- **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
- **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
brightness halo/bulge function

`diffsph.profiles.massmodels.J`(*tharcmin*, *galaxy*, *rad_temp*, *manual=False*, ***kwargs*)

Model-specific J factor in Gev ^{2}/cm ^{5}

Parameters

- **tharcmin** – angular radius in arcmin
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – radial template (`'NFW'`, `'Einasto'`, etc.)
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)

Keyword arguments

- `manual = 'False'`

Parameters
ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

Parameters

- **rs** – scale radius
- **rhos** – characteristic density
- **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
- **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
- **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
- **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile

- **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
J factor

`diffsph.profiles.massmodels.`**h**(*n*, *galaxy*, *rad_temp*, *hyp*, *ratio*, *manual=False*, ***kwargs*)

Model-specific n-th halo factor

Parameters

- **n** – order of the halo/bulge factor
- **rh** – diffusion halo/bulge radius
- **rad_temp** – radial template
- **hyp** (*str*) – hypothesis: `'wimp'` (**default**), `'decay'` or `'generic'`
- **ratio** – ratio between the diffusion halo/bulge and the half-light radius
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)

Keyword arguments

- `manual = 'False'`

Parameters
ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

Parameters

- **rs** – scale radius
- **rhos** – characteristic density
- **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
- **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
- **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
- **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
- **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
halo factor

`diffsph.profiles.massmodels.`**rho**(*r*, *rad_temp*, *manual=False*, ***kwargs*)

Dark matter density

Parameters

- **r** – galactocentric distance
- **rad_temp** – template (‘NFW’, ‘Einasto’, etc.)

- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)

Keyword arguments

- `manual = 'False'`

  Parameters
  ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

  Parameters

  - **rs** – scale radius
  - **rhos** – characteristic density
  - **alpha** – exponent α in the `diffsph.profiles.templates.hdz()` profile
  - **beta** – exponent β in the `diffsph.profiles.templates.hdz()` profile
  - **gamma** – exponent γ in the `diffsph.profiles.templates.hdz()` profile
  - **alphaE** – parameter α_E in the `diffsph.profiles.templates.enst()` profile
  - **rc** – core radius parameter r_c in the `diffsph.profiles.templates.cnfw()` profile
  - **sigmav** – velocity dispersion parameter σ_v in the `diffsph.profiles.templates.sis()` profile

  Returns
  dark matter density

diffsph.profiles.templates module

`diffsph.profiles.templates.`**`bkrt`**`(`*r*, *rs*, *rhos*`)`

Burkert dark-matter halo profile.

$$\rho(r) = \frac{\rho_s}{(1 + r/r_s)(1 + r^2/r_s^2)}$$

Parameters

- **r** – main variable (galactocentric distance)
- **rs** – scale radius
- **rhos** – charactieristic density

Returns
density at galactocentric distance r

`diffsph.profiles.templates.`**`cnfw`**`(`*r*, *rs*, *rhos*, *rc*`)`

Cored Navarro/Frenk/White dark-matter halo template.

$$\rho(r) = \frac{\rho_s}{(r/r_s + r_c/r_s)(1 + r/r_s)^2}$$

Parameters

- **r** – main variable (galactocentric distance)

- **rs** – scale radius
- **rhos** – chraracteristic density
- **rc** – core radius

Returns
density at galactocentric distance r

`diffsph.profiles.templates.const(`*r*, *rs*`)`

Constant (top-hat) template

$$\rho(r) = \frac{3}{4\pi r_s^3}\Theta(r_s - r)$$

Parameters
rs – characteristic radius

Returns
constant density

`diffsph.profiles.templates.enst(`*r*, *rs*, *rhos*, *alphaE=0.17*`)`

Einasto dark-matter halo profile.

$$\rho(r) = \rho_s \exp\left[-\frac{2}{\alpha_E}\left(\frac{r^{\alpha_E}}{r_s^{\alpha_E}} - 1\right)\right]$$

Parameters

- **r** – main variable (galactocentric distance)
- **rs** – scale radius
- **rhos** – charactieristic density
- **alphaE** – power-law slope of the Einasto profile, (default value = 0.17)

Returns
density at galactocentric distance r

`diffsph.profiles.templates.hdz(`*r*, *rs*, *rhos*, *alpha*, *beta*, *gamma*`)`

Hernquist/Diemand/Zhao dark-matter halo template.

$$\rho(r) = \frac{\rho_s}{(r/r_s)^{\gamma}(1 + (r/r_s)^{\alpha})^{\frac{\beta-\gamma}{\alpha}}}$$

Using default values alpha = 1, beta = 3 and gamma = 1 results in the default NFW halo profile.

Parameters

- **r** – main variable (galactocentric distance)
- **rs** – scale radius
- **rhos** – chraracteristic density
- **alpha** – inner exponent
- **beta** – large-r exponent
- **gamma** – small-r exponent

Returns
density at galactocentric distance r

`diffsph.profiles.templates.`**`nfw`**`(`*r*, *rs*, *rhos*`)`

Navarro/Frenk/White dark-matter halo template.

$$\rho(r) = \frac{\rho_s}{(r/r_s)(1+r/r_s)^2}$$

Parameters

- **`r`** – main variable (galactocentric distance)
- **`rs`** – scale radius
- **`rhos`** – chraracteristic density

Returns

density at galactocentric distance r

`diffsph.profiles.templates.`**`plmm`**`(`*r*, *rs*`)`

Plummer template

$$\rho(r) = \frac{3}{4\pi r_s^3} \frac{1}{(1+r^2/r_s^2)^{5/2}}$$

Parameters

- **`r`** – main variable (distance to the center)
- **`rs`** – Plummer radius
- **`rhoa`** – central density

Returns

density of the Plummer sphere at distance r

`diffsph.profiles.templates.`**`ps`**`(`*r*, *rs*`)`

Point source template

$$\rho(r) = \frac{1}{4\pi r^2} \delta(r)$$

Parameters

- **`r`** – main variable (galactocentric distance)
- **`rs`** – characteristic radius

Returns

density at galactocentric distance r

`diffsph.profiles.templates.`**`ps_iso`**`(`*r*, *rs*, *rhos*`)`

Pseudo-isothermal sphere dark-matter halo profile.

$$\rho(r) = \frac{\rho_s}{1+r^2/r_s^2}$$

Parameters

- **`r`** – main variable (galactocentric distance)
- **`rs`** – scale radius
- **`rhos`** – characteristic density

Returns

density at galactocentric distance r

`diffsph.profiles.templates.sis(r, sigmav)`

Singular isothermal sphere

$$\rho(r) = \frac{\sigma_v^2}{2\pi G r^2}$$

Parameters

- **r** – main variable (galactocentric distance)
- **sigmav** – velocity dispersion

Returns

density at galactocentric distance r

Module contents

diffsph.spectra package

Submodules

diffsph.spectra.analytics module

`diffsph.spectra.analytics.Fav(x)`

Synchrotron-power function for randomly-oriented magnetic fields[1].

$$F(x) = x^2 \left(K_{4/3}(x) K_{1/3}(x) - \frac{3}{5} x [K_{4/3}^2(x) - K_{1/3}^2(x)] \right)$$

Returns

Pitch-angle averaged synchrotron function as a function of x

`diffsph.spectra.analytics.M_C(xi, eta, delta)`

Master function in the Regime-C limit

$$\mathcal{M}_C(\xi, \eta, \delta) = \frac{\xi^\delta}{(1-\delta)\eta} F(\xi^2)$$

`diffsph.spectra.analytics.M_i(xi, eta, delta)`

Master function in the large η limit

$$\mathcal{M}_i(\xi, \eta, \delta) = \frac{\Gamma^2(1/3)\eta^{-\frac{5}{3(1-\delta)}}}{5\sqrt[3]{2}(1-\delta)} \Gamma\left(\frac{5}{3(1-\delta)}, \eta\, \xi^{1-\delta}\right) \exp\left(\eta\, \xi^{1-\delta}\right)$$

`diffsph.spectra.analytics.M_raw(xi, eta, delta)`

"Raw" master function

$$\mathcal{M}(\xi, \eta, \delta) = \int_\xi^\infty dx F(x^2) \exp\left(-\eta \left[x^{1-\delta} - \xi^{1-\delta}\right]\right)$$

Returns

above integral

[1] Formula extracted from Ghisellini et al, 1988

`diffsph.spectra.analytics.anltc_Mst(xi, eta, delta)`

Master function

$$\mathcal{M}(\xi,\eta,\delta) = \int_{\xi}^{\infty} dx F(x^2) \exp\left(-\eta\left[x^{1-\delta} - \xi^{1-\delta}\right]\right)$$

Note: Function evaluates the above integral only for those values where no numerical errors are present. Otherwise, it uses the approximate formulas *diffsph.spectra.analytics.M_C()* or *diffsph.spectra.analytics.M_i()*

`diffsph.spectra.analytics.btot(E, B)`

Total energy loss function in GeV/s

Parameters

- **E** – cosmic-ray energy in GeV
- **B** – magnitude of the magnetic field's smooth component in μG

Returns

energy-loss rate in GeV/s

`diffsph.spectra.analytics.lam(E, B, D0, delta=0.3333333333333333)`

Syrovatskii variable in kpc^2

Parameters

- **E** – cosmic-ray energy in GeV
- **B** – magnitude of the magnetic field's smooth component in μG
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm^2/s
- **delta** – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3)

Returns

Syrovatskii variable in kpc^2

diffsph.spectra.synchrotron module

`diffsph.spectra.synchrotron.Enu(B, nu)`

Typical particle energy in GeV for synchrotron radiation at the frequency nu in GHz and for a magnetic field B in μG

Parameters

- **B** – magnitude of the magnetic field's smooth component in μG
- **nu** – frequency in GHz

Returns

Particle energy in GeV.

`diffsph.spectra.synchrotron.Mst(xi, eta, delta)`

Interpolation function for the kernel function $\hat{\mathcal{M}}(\xi,\eta,\delta)$

Parameters

- **xi** – ξ

- **eta** – η
- **delta** – δ

Returns
Spectral-function kernel (as an interpolation function)

`diffsph.spectra.synchrotron.`**`Mst_DM`**(*xi*, *eta*, *m*, *delta*, *channel*)

Master function for dark-matter hypotheses

Parameters

- **xi** – ξ
- **eta** – η
- **delta** – δ
- **m** – WIMP mass in GeV
- **channel** – annihilation/decay channel

Returns
Master function (as an interpolation function) for DM hypotheses

`diffsph.spectra.synchrotron.`**`Mst_pw`**(*eta*, *Gamma*, *delta*)

Master function for the generic power-law hypothesis

Parameters

- **eta** – η
- **Gamma** – Γ
- **delta** – δ

Returns
Master function (as an interpolation function) for the generic power-law hypothesis

`diffsph.spectra.synchrotron.`**`X`**(*nu*, *tau*, *delta*, *B*, *hyp*, ***kwargs*)

Spectral function in erg/GHz for all hypotheses built in diffsph

Parameters

- **nu** – frequency in GHz
- **tau** – diffusion time-scale parameter for a 1 GeV CRE in s
- **delta** – power-law exponent of the diffusion coefficient as a function of the CRE's energy
- **B** – magnitude of the magnetic field's smooth component in μG
- **hyp** (*str*) – hypothesis: `'wimp'`, `'decay'` or `'generic'`

Keyword arguments:

- If `hyp = 'wimp'` or `'decay'`

Parameters

- **mchi** – mass of the DM particle in GeV/c^2
- **channel** (*str*) – annihilation/decay channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- If `hyp = 'generic'`

Parameters

Gamma – power-law exponent of the generic CRE source ($1.1 < \Gamma < 3$)

Returns

spectral function in erg/GHz

`diffsph.spectra.synchrotron.`**`X_DM`**`(`*k*, *mchi*, *channel*, *nu*, *tau*, *delta*, *B*`)`

Spectral function in erg/GHz for all DM hypotheses built in diffsph

Parameters

- **k** – hypothesis index (k=1 for decay and k=2 for annihilation)
- **mchi** – mass of the DM particle in GeV/c^2
- **channel** – annihilation/decay channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.
- **nu** – frequency in GHz
- **tau** – diffusion time-scale parameter for a 1 GeV CRE in s
- **delta** – power-law exponent of the diffusion coefficient as a function of the CRE's energy
- **B** – magnitude of the magnetic field's smooth component in μG

Returns

spectral function in erg/GHz

`diffsph.spectra.synchrotron.`**`X_gen`**`(`*Emin*, *Emax*, *S_func*, *nu*, *tau*, *delta*, *B*`)`

Spectral function in erg/GHz for generic CRE sources

$$X_{\text{gen}}(\nu) = \int_{E_m}^{E_M} dE' \hat{X}(\nu, E') S(E')$$

Parameters

- **Emin** – low-E cutoff energy in GeV of the CRE source `'S_func'`
- **Emax** – high-E cutoff energy in GeV of the CRE source `'S_func'`
- **S_func** – CRE source function
- **nu** – frequency in GHz
- **tau** – diffusion time-scale parameter for a 1 GeV CRE in s
- **delta** – power-law exponent of the diffusion coefficient as a function of the CRE's energy
- **B** – magnitude of the magnetic field's smooth component in μG

Returns
spectral function in erg/GHz

`diffsph.spectra.synchrotron.X_pw`(*Gamma*, *nu*, *tau*, *delta*, *B*)

Spectral function in erg/GHz for the generic power-law hypothesis

Parameters

- **Gamma** – power-law exponent of the generic CRE source ($1.1 < \Gamma < 3$)
- **nu** – frequency in GHz
- **tau** – diffusion time-scale parameter for a 1 GeV CRE in s
- **delta** – power-law exponent of the diffusion coefficient as a function of the CRE's energy
- **B** – magnitude of the magnetic field's smooth component in μG

Returns
spectral function in erg/GHz

`diffsph.spectra.synchrotron.eta`(*E*, *B*, *tau*, *delta*)

η variable as a function of the CRE's energy, magnetic field, tau and delta parameters

Parameters

- **E** – CRE energy in GeV
- **B** – magnetic field strength in μG
- **tau** – diffusion time-scale parameter for a 1 GeV CRE in s
- **delta** – power-law exponent of the diffusion coefficient as a function of the CRE's energy

Returns
η variable

`diffsph.spectra.synchrotron.htX`(*E*, *nu*, *tau*, *delta*, *B*, *fast_comp=True*)

Spectral function kernel in erg/GHz $\hat{X}$

Parameters

- **E** – CRE energy in GeV
- **nu** – frequency in GHz
- **tau** – diffusion time-scale parameter for a 1 GeV CRE in s
- **delta** – power-law exponent of the diffusion coefficient as a function of the CRE's energy
- **B** – magnitude of the magnetic field's smooth component in μG
- **fast_comp** (*bool*) – if `'True'`, employs the interpolating method (default value = `'True'`)

Returns
spectral kernel in erg/GHz

`diffsph.spectra.synchrotron.lMst`(*Lxi*, *Leta*, *delta*)

Interpolation function for (kernel) $\log(\hat{\mathcal{M}})$

Parameters

- **Lxi** – $\log(\xi)$
- **Leta** – $\log(\eta)$
- **delta** – δ

Returns
$\log(\hat{\mathcal{M}})$ as a function of $\log(\xi)$, $\log(\eta)$ and δ

`diffsph.spectra.synchrotron.lMst_DM(`*Lxi*, *Leta*, *Lm*, *delta*, *channel*`)`
Interpolation function $\log(\mathcal{M})$ for DM hypotheses

Parameters

- **Lxi** – $\log(\xi)$
- **Leta** – $\log(\eta)$
- **Lm** – $\log(m/\mathrm{GeV})$ (m is the WIMP mass)
- **delta** – δ
- **channel** – annihilation/decay channel

Returns
$\log(\mathcal{M})$ as a function of $\log(\xi)$, $\log(\eta)$, $\log(m)$ and δ

`diffsph.spectra.synchrotron.lMst_pw(`*Leta*, *Gamma*, *delta*`)`
Interpolation function $\log(\mathcal{M})$ for the gereric power-law hypothesis

Parameters

- **Leta** – $\log(\eta)$
- **Gamma** – Γ
- **delta** – δ

Returns
$\log(\mathcal{M}_{\mathrm{gen}})$ as a function of $\log(\eta)$, Γ and δ

Module contents

diffsph.utils package

Submodules

diffsph.utils.consts module

diffsph.utils.dictionaries module

diffsph.utils.tools module

`diffsph.utils.tools.TB(`*brightness*, *theta*, *nu*, **args*, ***kwargs*`)`
Brightness temperature conversion

$$T_B = \frac{c^2}{2\,k\,\nu^2} I_\nu$$

Parameters

- **brightness** – generic brightness function in Jy/sr
- **theta** – angular radius (as the first argument of the generic brighness function)
- **nu** – frequency (as the second argument of the generic brighness function)

Returns
brightness temperature in mK

`diffsph.utils.tools.approxhalo_fd(n, theta, dist, rh)`

Partial (θ-dependent) flux-density halo/bulge factor (approximate formula):

$$\mathcal{H}_n(\theta) = \mathcal{H}_n(r_h, R) - 2\int_{R\sin(\theta)}^{r_h} dr\, r\, \kappa_1(r, R, \theta)\frac{\sin\left(\frac{n\pi r}{r_h}\right)}{r}$$

where R, rh and n are, respectively the distance, halo radius and Fourier index

`diffsph.utils.tools.approxhalo_fd_tot(n, dist, rh)`

Total flux-density halo/bulge factor (approximate formula):

$$\mathcal{H}_n(r_h, R) \simeq 4\pi\int_0^{r_h} dr\, r^2\frac{\sin\left(\frac{n\pi r}{r_h}\right)}{r}\ ,$$

where R, rh and n are, respectively the distance, halo radius and Fourier index

`diffsph.utils.tools.check_cache()`

Function checks whether the /.diffsph_cache/ folder exists. If it does not exists, it creates it

Returns
folder directory name

Return type
str

`diffsph.utils.tools.delta_float(inp)`

Float number for variable `'delta'`

Parameters
inp – variable `'delta'` as *str* (`'kol'`, `'kra'`, etc.) or *float*

Returns
float number associated with `'inp'`

Return type
float

`diffsph.utils.tools.df(func, **kwargs)`

`diffsph.utils.tools.evaluate(f, x, **kwargs)`

Function converts string into a python function's name and evaluates it

Parameters

- **f** – function to be evaluated
- **x** – first argument of f

Returns
$f(x)$

`diffsph.utils.tools.f(n, x)`

Basis function in Fourier-expanded brightness formula

$$f_n(x) = 2\int_x^1 \frac{\sin(n\pi y)dy}{\sqrt{y^2 - x^2}}$$

Returns
f_n as a function of x

`diffsph.utils.tools.`**`fwhm`**(*brightness*, *thmax*, **args*, ***kwargs*)

Full width at half maximum

Parameters

- **`brightness`** – generic brightness function
- **`thmax`** – signal's angular radius

Returns
Full width at half maximum in arcmin

`diffsph.utils.tools.`**`g`**(*n*, *x*)

Basis function in Fourier-expanded flux density formula

$$g_n(x) = 2\int_x^1 \sqrt{y^2 - x^2}\,\sin(n\pi y)dy$$

Returns
g_n as a function of x

`diffsph.utils.tools.`**`halo_fd`**(*n*, *theta*, *dist*, *rh*)

Partial (θ-dependent) flux-density halo/bulge factor:

$$\mathcal{H}_n(\theta) = \mathcal{H}_n(r_h, R) - 2\int_{R\sin(\theta)}^{r_h} dr\, r\, \kappa_1(r, R, \theta)\frac{\sin\left(\frac{n\pi r}{r_h}\right)}{r}\ ,$$

where R, rh and n are, respectively the distance, halo radius and Fourier index

`diffsph.utils.tools.`**`halo_fd_tot`**(*n*, *dist*, *rh*)

Total flux-density halo/bulge factor:

$$\mathcal{H}_n(r_h, R) = 2\int_0^{r_h} dr\, r\, \kappa_0(r, R)\frac{\sin\left(\frac{n\pi r}{r_h}\right)}{r}\ ,$$

where R, rh and n are, respectively the distance, halo radius and Fourier index

Returns
Halo flux-density factor

`diffsph.utils.tools.`**`hfd`**(*fluxdens*, *thmax*, **args*, ***kwargs*)

Half-flux diameter

Parameters

- **`brightness`** – generic brightness function
- **`thmax`** – signal's angular radius

Returns
Half-flux diameter in arcmin

`diffsph.utils.tools.`**`hypothesis_index`**(*hyp*)

Index of the hypothesis (1 for decaying DM or generic scenario, 2 for WIMP self-annihilation).

Parameters
hyp (*`str`*) – hypothesis: `'wimp'`, `'decay'` or `'generic'`)

Returns
hypothesis index

Return type
int

`diffsph.utils.tools.`**`ker_0`**`(`*r*, *dist*`)`

$$\kappa_0(r, R) = \frac{1}{R} \log \sqrt{\frac{R+r}{R-r}}$$

`diffsph.utils.tools.`**`ker_1`**`(`*r*, *theta*, *dist*`)`

$$\kappa_1(\theta, r, R) = \frac{1}{R} \log \frac{R\cos\theta + \sqrt{r^2 - R^2 \sin^2\theta}}{\sqrt{R^2 - r^2}}$$

`diffsph.utils.tools.`**`load_data`**`(`*folder*`)`

Function loads data from folder

Returns
data organized in form of a python dictionary

Return type
dict

`diffsph.utils.tools.`**`sort_kwargs`**`(`***kwargs*`)`

Function sorts keyword arguments alphabetically

Returns
sorted keywords with corresponding entries

Return type
dict

`diffsph.utils.tools.`**`var_to_str`**`(`*inp*`)`

Dictionary for variables `'delta'`, `'hyp'`, `'galaxy'`, `'ref'` and `'rad_temp'`

Parameters
`inp` – input string or number

Returns
default variable name

Return type
str

Module contents

1.1.2 Submodules

1.1.3 diffsph.limits module

`diffsph.limits.`**`decay_rate_gausslim`**`(`*nu*, *a_fit*, *sigma_fit*, *beam_size*, *galaxy*, *rad_temp*, *D0=3e+28*, *delta='kol'*, *B=2*, *mchi=50*, *channel='mumu'*, *manual=False*, ***kwargs*`)`

Maximum dark matter decay rate allowed by the exclusion of a Gaussian-shaped signal

$$a_{\rm fit}\exp\left(-\frac{\theta^2}{2\sigma_{\rm fit}^2}\right)$$

Parameters

- **nu** – frequency in GHz
- **a_fit** – fitted gaussian amplitude in μ Jy / beam
- **sigma_fit** – width parameter of the Gaussian template in arcmin
- **beam_size** – beam size in arcseconds
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – dark matter halo model (`'NFW'`, `'Einasto'`, etc.)
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)
- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or `'kol'`)
- **B** – magnitude of the magnetic field's smooth component in μ G (default value $= 2\mu$ G)
- **mchi** – mass of the DM particle in GeV/c ^{2}
- **channel** (*str*) – decay channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)

Keyword arguments

- `manual = 'False'`

  Parameters

  ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

  Parameters

  - **rs** – scale radius in kpc
  - **rhos** – characteristic density in GeV/cm ^{3}
  - **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
  - **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
  - **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
  - **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
  - **sigmav** – velocity dispersion in km/s for the isothermal sphere *diffsph.profiles.templates.sis()*

Returns

upper limit on the DM decay rate in 1/s

Return type

float

`diffsph.limits.decay_rate_limest`(*nu*, *rms_noise*, *beam_size*, *galaxy*, *rad_temp*, *ratio=1*, *D0=3e+28*, *delta='kol'*, *B=2*, *mchi=50*, *channel='mumu'*, *manual=False*, *high_res=False*, *accuracy=1*, ***kwargs*)

(Estimated) maximum dark matter decay rate given the rms noise level of an observation

Parameters

- **nu** – frequency in GHz
- **rms_noise** – RMS noise level of the observation in μ Jy / beam
- **beam_size** – beam size in arcseconds
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – dark matter halo model (`'NFW'`, `'Einasto'`, etc.)
- **ratio** – ratio between the diffusion halo and half-light radii
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)
- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or `'kol'`)
- **B** – magnitude of the magnetic field's smooth component in μ G (default value $= 2\mu$ G)
- **mchi** – mass of the DM particle in GeV/c ^{2}
- **channel** (*str*) – decay channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)
- **high_res** (*bool*) – spatial resolution. If `'True'`, `synch_emissivity()` computes as many terms as needed in order to converge at $r = 0$. (default value = `'False'`)
- **accuracy** – theoretical accuracy in % (default value = 1%)

Keyword arguments

- `manual = 'False'`

  Parameters
  ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

  Parameters

  - **rs** – scale radius in kpc
  - **rhos** – characteristic density in GeV/cm ^{3}
  - **alpha** – exponent α in the *`diffsph.profiles.templates.hdz()`* profile
  - **beta** – exponent β in the *`diffsph.profiles.templates.hdz()`* profile
  - **gamma** – exponent γ in the *`diffsph.profiles.templates.hdz()`* profile
  - **alphaE** – parameter α_E in the *`diffsph.profiles.templates.enst()`* profile
  - **sigmav** – velocity dispersion in km/s for the isothermal sphere *`diffsph.profiles.templates.sis()`*

Returns
Estimated upper limit on the DM decay rate in 1/s

Return type
float

`diffsph.limits.generic_rate_gausslim(nu, a_fit, sigma_fit, beam_size, galaxy, rad_temp, D0=3e+28, delta='kol', B=2, Gamma=2, **kwargs)`

Maximum CRE production rate (generic power-law hypothesis) allowed by the exclusion of a Gaussian-shaped signal

$$a_{\rm fit} \exp\left(-\frac{\theta^2}{2\sigma_{\rm fit}^2}\right)$$

Parameters

- **nu** – frequency in GHz
- **a_fit** – fitted gaussian amplitude in μ Jy / beam
- **sigma_fit** – width parameter of the Gaussian template in arcmin
- **beam_size** – beam size in arcseconds
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – dark matter halo model (`'NFW'`, `'Einasto'`, etc.)
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)
- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or `'kol'`)
- **B** – magnitude of the magnetic field's smooth component in μ G (default value $= 2\mu$ G)
- **Gamma** – power-law exponent of the generic CRE source ($1.1 < \Gamma < 3$, default value = 2)
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)

Keyword arguments

- `manual = 'False'`

  Parameters
  ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

  Parameters
  - **rs** – scale radius in kpc
  - **sigmav** – velocity dispersion in km/s for the isothermal sphere *diffsph.profiles.templates.sis()*

Returns
upper limit on the generic CRE production rate in 1/s

Return type
float

`diffsph.limits.`**`generic_rate_limest`**(*nu*, *rms_noise*, *beam_size*, *galaxy*, *rad_temp*, *ratio=1*, *D0=3e+28*, *delta='kol'*, *B=2*, *Gamma=2*, *high_res=False*, *accuracy=1*, ***kwargs*)

(Estimated) maximum CRE production rate (generic power-law hypothesis) given the rms noise level of an observation

Parameters

- **nu** – frequency in GHz
- **rms_noise** – RMS noise level of the observation in μ Jy / beam
- **beam_size** – beam size in arcseconds
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – dark matter halo model (`'NFW'`, `'Einasto'`, etc.)
- **ratio** – ratio between the diffusion halo and half-light radii
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)
- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or `'kol'`)
- **B** – magnitude of the magnetic field's smooth component in μ G (default value $= 2\mu$ G)
- **Gamma** – power-law exponent of the generic CRE source ($1.1 < \Gamma < 3$, default value = 2)
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)
- **high_res** (*bool*) – spatial resolution. If `'True'`, `synch_emissivity()` computes as many terms as needed in order to converge at $r = 0$. (default value = `'False'`)
- **accuracy** – theoretical accuracy in % (default value = 1%)

Keyword arguments

- `manual = 'False'`

  Parameters
  ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

  Parameters

  - **rs** – scale radius in kpc
  - **sigmav** – velocity dispersion in km/s for the isothermal sphere *`diffsph.profiles.templates.sis()`*

Returns
Estimated upper limit on the generic CRE production rate in 1/s

Return type
float

`diffsph.limits.`**`sigmav_gausslim`**(*nu*, *a_fit*, *sigma_fit*, *beam_size*, *galaxy*, *rad_temp*, *D0=3e+28*, *delta='kol'*, *B=2*, *mchi=50*, *channel='mumu'*, *self_conjugate=True*, *manual=False*, ***kwargs*)

Maximum WIMP self-annihilation cross-section allowed by the exclusion of a Gaussian-shaped signal

$$a_{\rm fit} \exp\left(-\frac{\theta^2}{2\sigma_{\rm fit}^2}\right)$$

Parameters

- **nu** – frequency in GHz
- **a_fit** – fitted gaussian amplitude in μ Jy / beam
- **sigma_fit** – width parameter of the Gaussian template in arcmin
- **beam_size** – beam size in arcseconds
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – dark matter halo model (`'NFW'`, `'Einasto'`, etc.)
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)
- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or `'kol'`)
- **B** – magnitude of the magnetic field's smooth component in μ G (default value $= 2\mu$ G)
- **mchi** – mass of the DM particle in GeV/c ^{2}
- **channel** (*str*) – annihilation channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.
- **self_conjugate** – if set `'True'` (default value) the DM particle is its own antiparticle
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)

Keyword arguments

- `manual = 'False'`

  Parameters
  ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

  Parameters

  - **rs** – scale radius in kpc
  - **rhos** – characteristic density in GeV/cm ^{3}
  - **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
  - **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
  - **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
  - **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile

Returns
upper limit for the WIMP self-annihilation cross-section in cm ^{3} /s

Return type
float

`diffsph.limits.`**`sigmav_limest`**(*nu*, *rms_noise*, *beam_size*, *galaxy*, *rad_temp*, *ratio=1*, *D0=3e+28*, *delta='kol'*, *B=2*, *mchi=50*, *channel='mumu'*, *self_conjugate=True*, *manual=False*, *high_res=False*, *accuracy=1*, ***kwargs*)

(Estimated) maximum WIMP self-annihilation cross-section given the rms noise level of an observation

Parameters

- **nu** – frequency in GHz
- **rms_noise** – RMS noise level of the observation in μ Jy / beam
- **beam_size** – beam size in arcseconds
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – dark matter halo model (`'NFW'`, `'Einasto'`, etc.)
- **ratio** – ratio between the diffusion halo and half-light radii
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)
- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or `'kol'`)
- **B** – magnitude of the magnetic field's smooth component in μ G (default value $= 2\mu$ G)
- **mchi** – mass of the DM particle in GeV/c ^{2}
- **channel** (*str*) – annihilation channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.
- **self_conjugate** – if set `'True'` (default value) the DM particle is its own antiparticle
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)
- **high_res** (*bool*) – spatial resolution. If `'True'`, `synch_emissivity()` computes as many terms as needed in order to converge at $r = 0$. (default value = `'False'`)
- **accuracy** – theoretical accuracy in % (default value = 1%)

Keyword arguments

- `manual = 'False'`

  Parameters
  ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

  Parameters

  - **rs** – scale radius in kpc
  - **rhos** – characteristic density in GeV/cm ^{3}
  - **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
  - **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
  - **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
  - **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile

  Returns
  Estimated upper limit on WIMP self-annihilation cross-section in cm ^{3} /s

Return type
float

1.1.4 diffsph.pyflux module

`diffsph.pyflux.Dec_rad(galaxy)`

`diffsph.pyflux.RA_rad(galaxy)`

`diffsph.pyflux.coeff(n, nu, galaxy, rad_temp, hyp, ratio, D0, delta, B, manual=False, **kwargs)`

n-th coefficient participating in the Fourier-expanded Green's function solution of the CRE transport equation

$$s_n = h_n \times X_n$$

Parameters

- **n** – order of the halo/bulge factor
- **theta** – angular radius in arcmin
- **nu** – frequency in GHz
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – radial template (`'NFW'`, `'Einasto'`, etc.)
- **hyp** (*str*) – hypothesis: `'wimp'` (**default**), `'decay'` or `'generic'`
- **ratio** – ratio between the diffusion halo/bulge and half-light radii (default value = 1)
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)
- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or `'kol'`)
- **B** – magnitude of the magnetic field's smooth component in μ G (default value $= 2\mu$ G)
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)

Keyword arguments

- `hyp = 'wimp'` (default)

  Parameters

  - **sv** – annihilation rate (annihilation cross section times relative velocity) σv in cm ^{3}/s (default value = 3×10^{-26} cm ^{3} /s)
  - **self_conjugate** – if set `'True'` (default value) the DM particle is its own antiparticle
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – annihilation channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'decay'`

  Parameters

  - **width** – decay width of the DM particle in 1/s
  - **mchi** – mass of the DM particle in GeV/c ^{2}

   - **channel** (*str*) – decay channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'generic'`

  Parameters

   - **Gamma** – power-law exponent of the generic CRE source ($1.1 < \Gamma < 3$)
   - **rate** – CRE production rate in 1/s

- `manual = 'False'`

  Parameters
  ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

  Parameters

   - **rs** – scale radius in kpc
   - **rhos** – characteristic density in GeV/cm ^{3}
   - **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
   - **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
   - **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
   - **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
   - **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
   - **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

  Returns
  n-th coefficient in the *which_N* function

`diffsph.pyflux.`**synch_TB**(*theta*, *nu*, *galaxy*, *rad_temp*, *hyp='wimp'*, *ratio=1*, *D0=3e+28*, *delta='kol'*, *B=2*, *manual=False*, *high_res=False*, *accuracy=1*, ***kwargs*)

Model-specific brightness temperature from synchrotron radiation

Parameters

- **theta** – angular radius in arcmin
- **nu** – frequency in GHz
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – radial template (`'NFW'`, `'Einasto'`, etc.)
- **hyp** (*str*) – hypothesis: `'wimp'` (**default**), `'decay'` or `'generic'`
- **ratio** – ratio between the diffusion halo/bulge and half-light radii (default value = 1)
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)
- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or `'kol'`)

- **B** – magnitude of the magnetic field's smooth component in μ G (default value $= 2\mu$ G)
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)
- **high_res** (*bool*) – spatial resolution. If `'True'`, *synch_emissivity()* computes as many terms as needed in order to converge at $r = 0$. (default value = `'False'`)
- **accuracy** – theoretical accuracy in % (default value = 1%)

Keyword arguments

- `hyp = 'wimp'` (default)

  Parameters

  - **sv** – annihilation rate (annihilation cross section times relative velocity) σv in cm ^{3}/s (default value = 3×10^{-26} cm ^{3} /s)
  - **self_conjugate** – if set `'True'` (default value) the DM particle is its own antiparticle
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – annihilation channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'decay'`

  Parameters

  - **width** – decay width of the DM particle in 1/s
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – decay channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'generic'`

  Parameters

  - **Gamma** – power-law exponent of the generic CRE source $(1.1 < \Gamma < 3)$
  - **rate** – CRE production rate in 1/s

- `manual = 'False'`

  Parameters

  ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

  Parameters

  - **rs** – scale radius in kpc
  - **rhos** – characteristic density in GeV/cm ^{3}
  - **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
  - **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
  - **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
  - **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile

- **rc** – core radius parameter r_c in the `diffsph.profiles.templates.cnfw()` profile
- **sigmav** – velocity dispersion parameter σ_v in the `diffsph.profiles.templates.sis()` profile

Returns
Brightness temperature in mK

`diffsph.pyflux.synch_TB_approx`(*theta*, *nu*, *galaxy*, *rad_temp*, *hyp='wimp'*, *ratio=1*, *D0=3e+28*, *delta='kol'*, *B=2*, *regime='B'*, *manual=False*, ***kwargs*)

Model-specific brightness temperature in the Regime "A", "B" or "C" approximations

Parameters

- **theta** – angular radius in arcmin
- **nu** – frequency in GHz
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – radial template (`'NFW'`, `'Einasto'`, etc.)
- **hyp** (*str*) – hypothesis: `'wimp'` (**default**), `'decay'` or `'generic'`
- **ratio** – ratio between the diffusion halo/bulge and half-light radii (default value = 1)
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)
- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or `'kol'`)
- **B** – magnitude of the magnetic field's smooth component in μ G (default value $= 2\mu$ G)
- **regime** – regime of the approximation. Must be either upper or lower case a, b, c or I/II/III.
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)

Keyword arguments

- `hyp = 'wimp'` (default)

Parameters

- **sv** – annihilation rate (annihilation cross section times relative velocity) σv in cm ^{3}/s (default value = 3×10^{-26} cm ^{3} /s)
- **self_conjugate** – if set `'True'` (default value) the DM particle is its own antiparticle
- **mchi** – mass of the DM particle in GeV/c ^{2}
- **channel** (*str*) – annihilation channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'decay'`

Parameters

- **width** – decay width of the DM particle in 1/s
- **mchi** – mass of the DM particle in GeV/c ^{2}
- **channel** (*str*) – decay channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'generic'`

Parameters

- **Gamma** – power-law exponent of the generic CRE source ($1.1 < \Gamma < 3$)
- **rate** – CRE production rate in 1/s

- manual = 'False'

Parameters
ref – reference used ('Martinez' or '1309.2641', 'Geringer-Sameth' or '1408.0002', etc.)

- manual = 'True'

Parameters

- **rs** – scale radius in kpc
- **rhos** – characteristic density in GeV/cm ^{3}
- **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
- **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
- **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
- **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
- **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
Brightness temperature in mK

diffsph.pyflux.**synch_brightness**(*theta*, *nu*, *galaxy*, *rad_temp*, *hyp='wimp'*, *ratio=1*, *D0=3e+28*, *delta='kol'*, *B=2*, *manual=False*, *high_res=False*, *accuracy=1*, ***kwargs*)

Model-specific brightness from synchrotron radiation

Parameters

- **theta** – angular radius in arcmin
- **nu** – frequency in GHz
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – radial template ('NFW', 'Einasto', etc.)
- **hyp** (*str*) – hypothesis: 'wimp' (**default**), 'decay' or 'generic'
- **ratio** – ratio between the diffusion halo/bulge and half-light radii (default value = 1)
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)
- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or 'kol')
- **B** – magnitude of the magnetic field's smooth component in μ G (default value $= 2\mu$ G)
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = 'False')

- **high_res** (*bool*) – spatial resolution. If 'True', *synch_emissivity()* computes as many terms as needed in order to converge at $r = 0$. (default value = 'False')
- **accuracy** – theoretical accuracy in % (default value = 1%)

Keyword arguments

- hyp = 'wimp' (default)

  Parameters

  - **sv** – annihilation rate (annihilation cross section times relative velocity) σv in cm ^{3}/s (default value = 3×10^{-26} cm ^{3} /s)
  - **self_conjugate** – if set 'True' (default value) the DM particle is its own antiparticle
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – annihilation channel: $b\bar{b}$ ('bb'), $\mu^+\mu^-$ ('mumu'), W^+W^- ('WW'), etc.

- hyp = 'decay'

  Parameters

  - **width** – decay width of the DM particle in 1/s
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – decay channel: $b\bar{b}$ ('bb'), $\mu^+\mu^-$ ('mumu'), W^+W^- ('WW'), etc.

- hyp = 'generic'

  Parameters

  - **Gamma** – power-law exponent of the generic CRE source ($1.1 < \Gamma < 3$)
  - **rate** – CRE production rate in 1/s

- manual = 'False'

  Parameters
  ref – reference used ('Martinez' or '1309.2641', 'Geringer-Sameth' or '1408.0002', etc.)

- manual = 'True'

  Parameters

  - **rs** – scale radius in kpc
  - **rhos** – characteristic density in GeV/cm ^{3}
  - **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
  - **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
  - **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
  - **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
  - **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
  - **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
Brightness in Jy/sr

Return type
float

`diffsph.pyflux.synch_brightness_approx`(*theta*, *nu*, *galaxy*, *rad_temp*, *hyp='wimp'*, *ratio=1*, *D0=3e+28*, *delta='kol'*, *B=2*, *regime='B'*, *manual=False*, ***kwargs*)

Model-specific brightness from synchrotron radiation in the Regime "A", "B" or "C" approximations

Parameters

- **theta** – angular radius in arcmin
- **nu** – frequency in GHz
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – radial template (`'NFW'`, `'Einasto'`, etc.)
- **hyp** (*str*) – hypothesis: `'wimp'` (**default**), `'decay'` or `'generic'`
- **ratio** – ratio between the diffusion halo/bulge and half-light radii (default value = 1)
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)
- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or `'kol'`)
- **B** – magnitude of the magnetic field's smooth component in μ G (default value $= 2\mu$ G)
- **regime** – regime of the approximation. Must be either upper or lower case a, b, c or I/II/III.
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)

Keyword arguments

- `hyp = 'wimp'` (default)

  Parameters

  - **sv** – annihilation rate (annihilation cross section times relative velocity) σv in cm ^{3}/s (default value = 3×10^{-26} cm ^{3} /s)
  - **self_conjugate** – if set `'True'` (default value) the DM particle is its own antiparticle
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – annihilation channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'decay'`

  Parameters

  - **width** – decay width of the DM particle in 1/s
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – decay channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'generic'`

  Parameters

- **Gamma** – power-law exponent of the generic CRE source ($1.1 < \Gamma < 3$)
- **rate** – CRE production rate in 1/s

- manual = 'False'

  Parameters
  ref – reference used ('Martinez' or '1309.2641', 'Geringer-Sameth' or '1408.0002', etc.)

- manual = 'True'

  Parameters

  - **rs** – scale radius in kpc
  - **rhos** – characteristic density in GeV/cm ^{3}
  - **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
  - **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
  - **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
  - **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
  - **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
  - **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

  Returns
  Brightness in Jy/sr

diffsph.pyflux.**synch_emissivity**(*r*, *nu*, *galaxy*, *rad_temp*, *hyp='wimp'*, *ratio=1*, *D0=3e+28*, *delta='kol'*, *B=2*, *manual=False*, *high_res=False*, *accuracy=1*, ***kwargs*)

Model-specific emissivity from synchrotron radiation

Parameters

- **r** – galactocentric distance in kpc
- **nu** – frequency in GHz
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – radial template ('NFW', 'Einasto', etc.)
- **hyp** (*str*) – hypothesis: 'wimp' (**default**), 'decay' or 'generic'
- **ratio** – ratio between the diffusion halo/bulge and half-light radii (default value = 1)
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)
- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or 'kol')
- **B** – magnitude of the magnetic field's smooth component in μ G (default value $= 2\mu$ G)
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = 'False')
- **high_res** (*bool*) – spatial resolution. If 'True', *synch_emissivity()* computes as many terms as needed in order to converge at $r = 0$ (default value = 'False')

- **accuracy** – theoretical accuracy in % (default value = 1%)

Keyword arguments

- `hyp = 'wimp'` (default)

  Parameters

  - **sv** – annihilation rate (annihilation cross section times relative velocity) σv in cm ^{3}/s (default value = 3×10^{-26} cm ^{3} /s)
  - **self_conjugate** – if set `'True'` (default value) the DM particle is its own antiparticle
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – annihilation channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'decay'`

  Parameters

  - **width** – decay width of the DM particle in 1/s
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – decay channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'generic'`

  Parameters

  - **Gamma** – power-law exponent of the generic CRE source ($1.1 < \Gamma < 3$)
  - **rate** – CRE production rate in 1/s

- `manual = 'False'`

  Parameters

  ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

  Parameters

  - **rs** – scale radius in kpc
  - **rhos** – characteristic density in GeV/cm ^{3}
  - **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
  - **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
  - **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
  - **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
  - **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
  - **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

  Returns

  Emissivity in erg/cm ^{3} /Hz/s/sr

Return type
float

`diffsph.pyflux.`**`synch_emissivity_approx`**(*r*, *nu*, *galaxy*, *rad_temp*, *hyp='wimp'*, *ratio=1*, *D0=3e+28*, *delta='kol'*, *B=2*, *regime='B'*, *manual=False*, ***kwargs*)

Model-specific emissivity from synchrotron radiation in the Regime “A”, “B” or “C” approximations

Parameters

- **r** – galactocentric distance in kpc
- **nu** – frequency in GHz
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – radial template (`'NFW'`, `'Einasto'`, etc.)
- **hyp** (*str*) – hypothesis: `'wimp'` (**default**), `'decay'` or `'generic'`
- **ratio** – ratio between the diffusion halo/bulge and half-light radii (default value = 1)
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)
- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE’s energy (default value = 1/3 or `'kol'`)
- **B** – magnitude of the magnetic field’s smooth component in μ G (default value $= 2\mu$ G)
- **regime** – regime of the approximation. Must be either upper or lower case a, b, c or I/II/III.
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)

Keyword arguments

- `hyp = 'wimp'` (default)

  Parameters

  - **sv** – annihilation rate (annihilation cross section times relative velocity) σv in cm ^{3}/s (default value = 3×10^{-26} cm ^{3} /s)
  - **self_conjugate** – if set `'True'` (default value) the DM particle is its own antiparticle
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – annihilation channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'decay'`

  Parameters

  - **width** – decay width of the DM particle in 1/s
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – decay channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'generic'`

  Parameters

  - **Gamma** – power-law exponent of the generic CRE source ($1.1 < \Gamma < 3$)
  - **rate** – CRE production rate in 1/s

- manual = 'False'

  Parameters
  ref – reference used ('Martinez' or '1309.2641', 'Geringer-Sameth' or '1408.0002', etc.)

- manual = 'True'

  Parameters
  - **rs** – scale radius in kpc
  - **rhos** – characteristic density in GeV/cm ^{3}
  - **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
  - **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
  - **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
  - **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
  - **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
  - **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

  Returns
  Emissivity in erg/cm ^{3} /Hz/s/sr

diffsph.pyflux.**synch_flux_density**(*theta*, *nu*, *galaxy*, *rad_temp*, *hyp='wimp'*, *ratio=1*, *D0=3e+28*, *delta='kol'*, *B=2*, *manual=False*, *high_res=False*, *accuracy=1*, ***kwargs*)

Model-specific flux density from synchrotron radiation

Parameters
- **theta** – angular radius in arcmin
- **nu** – frequency in GHz
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – radial template ('NFW', 'Einasto', etc.)
- **hyp** (*str*) – hypothesis: 'wimp' (**default**), 'decay' or 'generic'
- **ratio** – ratio between the diffusion halo/bulge and half-light radii (default value = 1)
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)
- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or 'kol')
- **B** – magnitude of the magnetic field's smooth component in μ G (default value $= 2\mu$ G)
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = 'False')
- **high_res** (*bool*) – spatial resolution. If 'True', *synch_emissivity()* computes as many terms as needed in order to converge at $r = 0$. (default value = 'False')
- **accuracy** – theoretical accuracy in % (default value = 1%)

Keyword arguments

- `hyp = 'wimp'` (default)

  Parameters

  - **sv** – annihilation rate (annihilation cross section times relative velocity) σv in cm ^{3}/s (default value = 3×10^{-26} cm ^{3} /s)
  - **self_conjugate** – if set `'True'` (default value) the DM particle is its own antiparticle
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – annihilation channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'decay'`

  Parameters

  - **width** – decay width of the DM particle in 1/s
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – decay channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'generic'`

  Parameters

  - **Gamma** – power-law exponent of the generic CRE source ($1.1 < \Gamma < 3$)
  - **rate** – CRE production rate in 1/s

- `manual = 'False'`

  Parameters

  ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

  Parameters

  - **rs** – scale radius in kpc
  - **rhos** – characteristic density in GeV/cm ^{3}
  - **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
  - **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
  - **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
  - **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
  - **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
  - **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

  Returns

  Flux density in µJy

`diffsph.pyflux.`**`synch_flux_density_approx`**(*theta*, *nu*, *galaxy*, *rad_temp*, *hyp='wimp'*, *ratio=1*, *D0=3e+28*, *delta='kol'*, *B=2*, *regime='B'*, *manual=False*, ***kwargs*)

Model-specific flux density from synchrotron radiation in the Regime "A", "B" or "C" approximations

Parameters

- **theta** – angular radius in arcmin
- **nu** – frequency in GHz
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – radial template (`'NFW'`, `'Einasto'`, etc.)
- **hyp** (*str*) – hypothesis: `'wimp'` (**default**), `'decay'` or `'generic'`
- **ratio** – ratio between the diffusion halo/bulge and half-light radii (default value = 1)
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)
- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or `'kol'`)
- **B** – magnitude of the magnetic field's smooth component in μ G (default value $= 2\mu$ G)
- **regime** – regime of the approximation. Must be either upper or lower case a, b, c or I/II/III.
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)

Keyword arguments

- `hyp = 'wimp'` (default)

  Parameters

  - **sv** – annihilation rate (annihilation cross section times relative velocity) σv in cm ^{3}/s (default value = 3×10^{-26} cm ^{3} /s)
  - **self_conjugate** – if set `'True'` (default value) the DM particle is its own antiparticle
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – annihilation channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'decay'`

  Parameters

  - **width** – decay width of the DM particle in 1/s
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – decay channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'generic'`

  Parameters

  - **Gamma** – power-law exponent of the generic CRE source ($1.1 < \Gamma < 3$)
  - **rate** – CRE production rate in 1/s

- `manual = 'False'`

Parameters
ref – reference used ('Martinez' or '1309.2641', 'Geringer-Sameth' or '1408.0002', etc.)

- manual = 'True'

Parameters

- **rs** – scale radius in kpc
- **rhos** – characteristic density in GeV/cm ^{3}
- **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
- **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
- **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile
- **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
- **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
Flux density in µJy

class diffsph.pyflux.transport(*rh=None*, *B=None*, *D0=None*, *tau0=None*, *delta=None*)
Bases: object

property D0

Dcoeff(*E*)
Diffusion coefficient in cm ^{2} /s

Parameters

- **E** – cosmic-ray energy in GeV
- **delta** – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or 'kol')

Returns
Diffusion coefficient for CRE with energy E (GeV) in cm ^{2} /s

Elosses(*E*)
Total energy loss function in GeV/s

Parameters

- **E** – cosmic-ray energy in GeV
- **B** – magnitude of the magnetic field's smooth component in μG

Returns
energy-loss rate in GeV/s

Syrovatskii_var(*E*)
Syrovatskii variable in kpc^2

Parameters

- **E** – cosmic-ray energy in GeV

- **B** – magnitude of the magnetic field's smooth component in μG
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm^2/s
- **delta** – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3)

Returns
Syrovatskii variable in kpc^2

eta_var(*E*)

η variable as a function of the CRE's energy, magnetic field, tau and delta parameters

Parameters

- **E** – CRE energy in GeV
- **B** – magnetic field strength in μG
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm^2/s
- **delta** – power-law exponent of the diffusion coefficient as a function of the CRE's energy

Returns
η variable

hatXne(*E*, *E0*)

CRE number-density function kernel in s/GeV $\hat{X}_n$

Parameters

- **E** – CRE energy in GeV
- **E0** – injected CRE's energy in GeV
- **B** – magnetic field strength in μG
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm^2/s
- **delta** – power-law exponent of the diffusion coefficient as a function of the CRE's energy

Returns
Electron number density kernel in s/GeV

property rh

property tau0

`diffsph.pyflux.`**which_N**(*nu*, *galaxy*, *rad_temp*, *hyp*, *ratio*, *D0*, *delta*, *B*, *manual=False*, *high_res=False*, *accuracy=1*, ***kwargs*)

Determines at which order should the Fourier-expanded Green's function solution be truncated and stores the associated $s_n = h_n \times X_n$ coefficients as an array in the `/cache` folder

Parameters

- **nu** – frequency in GHz
- **galaxy** (*str*) – name of the galaxy
- **rad_temp** (*str*) – radial template (`'NFW'`, `'Einasto'`, etc.)
- **hyp** (*str*) – hypothesis: `'wimp'` (**default**), `'decay'` or `'generic'`
- **ratio** – ratio between the diffusion halo/bulge and half-light radii (default value = 1)
- **D0** – magnitude of the diffusion coefficient for a 1 GeV CRE in cm ^{2}/s (default value = 3×10^{28} cm ^{2} /s)

- **delta** (*float, str*) – power-law exponent of the diffusion coefficient as a function of the CRE's energy (default value = 1/3 or `'kol'`)
- **B** – magnitude of the magnetic field's smooth component in μ G (default value $= 2\mu$ G)
- **manual** (*bool*) – manual input of parameter values in rad_temp (default value = `'False'`)
- **high_res** (*bool*) – spatial resolution. If `'True'`, *synch_emissivity()* computes as many terms as needed in order to converge at $r = 0$. (default value = `'False'`)
- **accuracy** – theoretical accuracy in % (default value = 1%)

Keyword arguments

- `hyp = 'wimp'` (default)

  Parameters

  - **sv** – annihilation rate (annihilation cross section times relative velocity) σv in cm ^{3}/s (default value = 3×10^{-26} cm ^{3} /s)
  - **self_conjugate** – if set `'True'` (default value) the DM particle is its own antiparticle
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – annihilation channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'decay'`

  Parameters

  - **width** – decay width of the DM particle in 1/s
  - **mchi** – mass of the DM particle in GeV/c ^{2}
  - **channel** (*str*) – decay channel: $b\bar{b}$ (`'bb'`), $\mu^+\mu^-$ (`'mumu'`), W^+W^- (`'WW'`), etc.

- `hyp = 'generic'`

  Parameters

  - **Gamma** – power-law exponent of the generic CRE source ($1.1 < \Gamma < 3$)
  - **rate** – CRE production rate in 1/s

- `manual = 'False'`

  Parameters

  ref – reference used (`'Martinez'` or `'1309.2641'`, `'Geringer-Sameth'` or `'1408.0002'`, etc.)

- `manual = 'True'`

  Parameters

  - **rs** – scale radius in kpc
  - **rhos** – characteristic density in GeV/cm ^{3}
  - **alpha** – exponent α in the *diffsph.profiles.templates.hdz()* profile
  - **beta** – exponent β in the *diffsph.profiles.templates.hdz()* profile
  - **gamma** – exponent γ in the *diffsph.profiles.templates.hdz()* profile

- **alphaE** – parameter α_E in the *diffsph.profiles.templates.enst()* profile
- **rc** – core radius parameter r_c in the *diffsph.profiles.templates.cnfw()* profile
- **sigmav** – velocity dispersion parameter σ_v in the *diffsph.profiles.templates.sis()* profile

Returns
series truncation order N

1.1.5 Module contents

CHAPTER

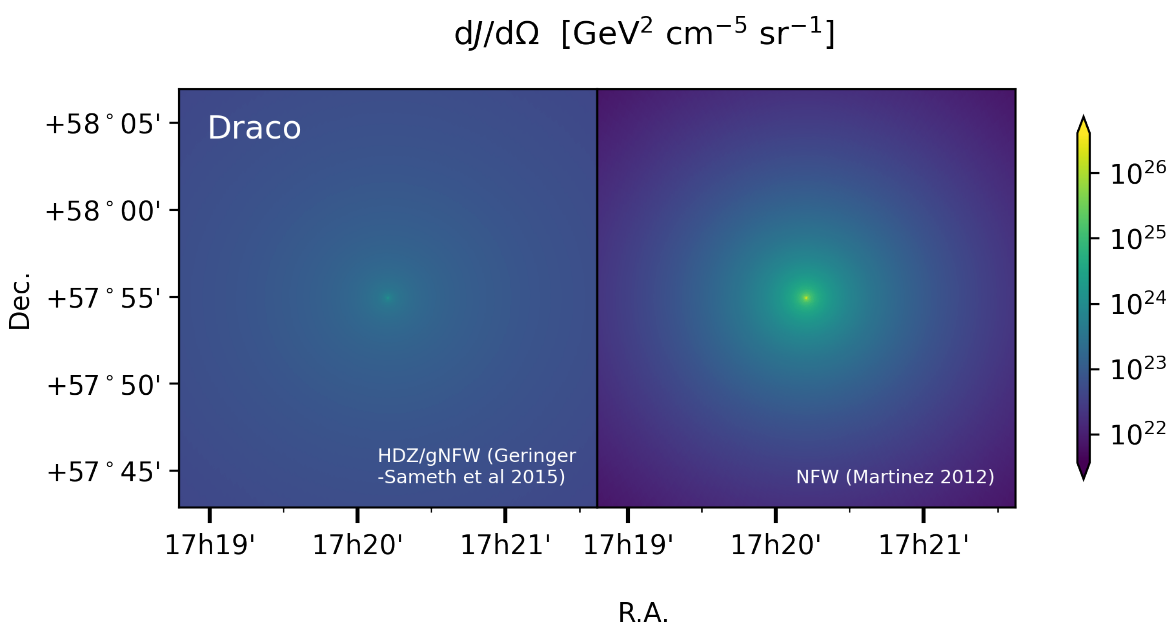
TWO

INDICES AND TABLES

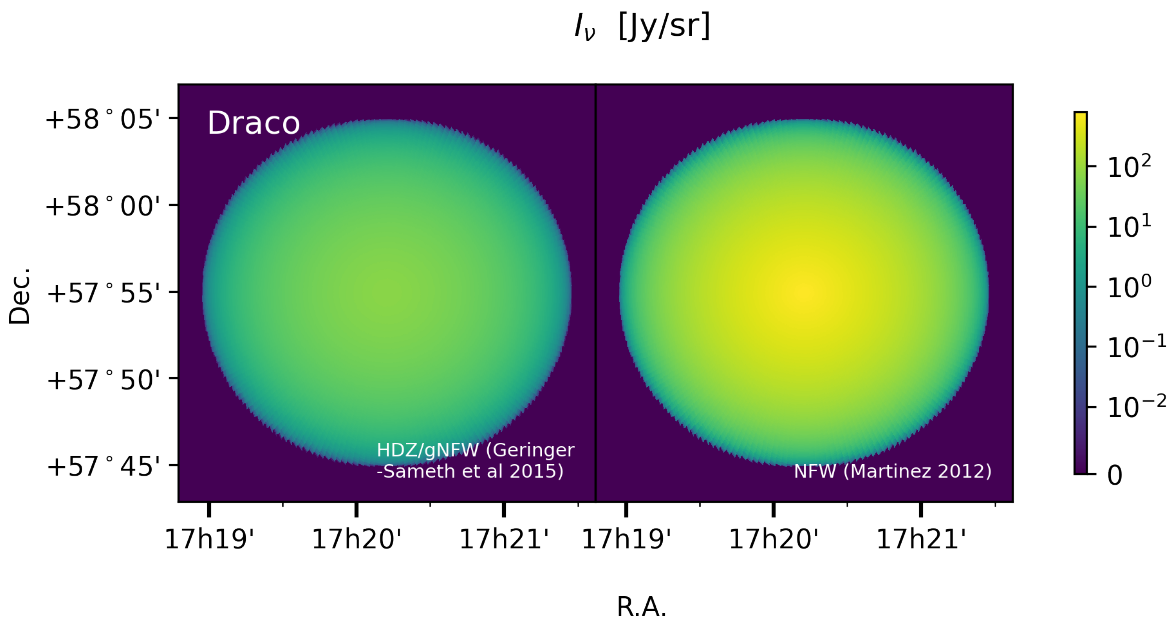
- genindex
- modindex
- search

`diffsph` is a Python package that computes diffuse fluxes from Milky-Way satellite dwarf spheroidal (*dSph*) galaxies. It allows users to obtain

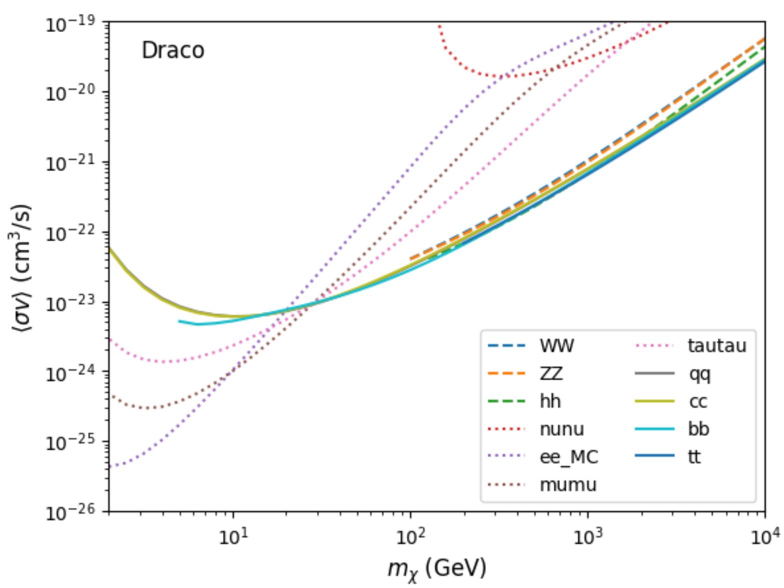
- J factor maps (relevant for gamma-ray astronomy)



- (radio frequency) synchrotron-radiation emission profiles



- bounds on e.~g. Dark Matter annihilation cross sections from radio astronomical observations



CHAPTER THREE

INSTALLATION

Use Git or checkout with SVN using the web URL https://github.com/mertio1/diffsph.git , e. g.:

```
git clone https://github.com/mertio1/diffsph.git
```

or:

```
svn co https://github.com/mertio1/diffsph.git
```

Otherwise download the zip file from the repository https://github.com/mertio1/diffsph

For global installations, while in the `diffsph`'s main folder type:

```
python setup.py bdist_wheel
```

and:

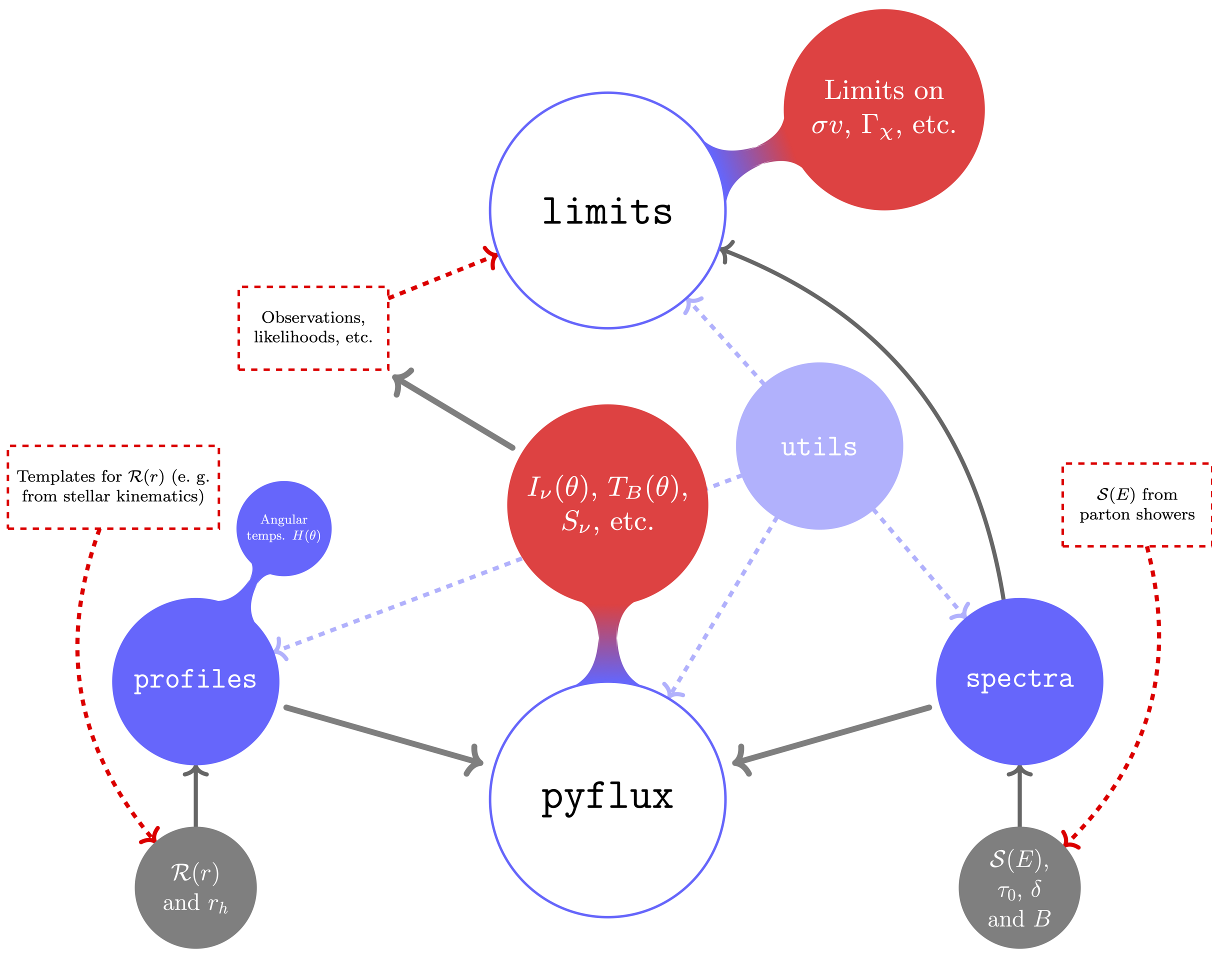
```
pip install .
```

CHAPTER

FOUR

ARCHITECTURE

The main functionalities and algorithmic structure of the code are captured by the diagram below



Users can both compute diffuse emission *fluxes* with the `pyflux` (*diffsph.pyflux*) module and 2σ *limits* on e. g. annihilation cross sections or decay rates of dark matter particles by using the `limits` (*diffsph.limits*) module. Details about the methods used to perform these computations are given in Ref.[1] .

[1] M. Vollmann, "Universal profiles for radio searches of dark matter in dwarf galaxies", doi:10.1088/1475-7516/2021/04/068 [arXiv:2011.11947 [astro-ph.HE]].

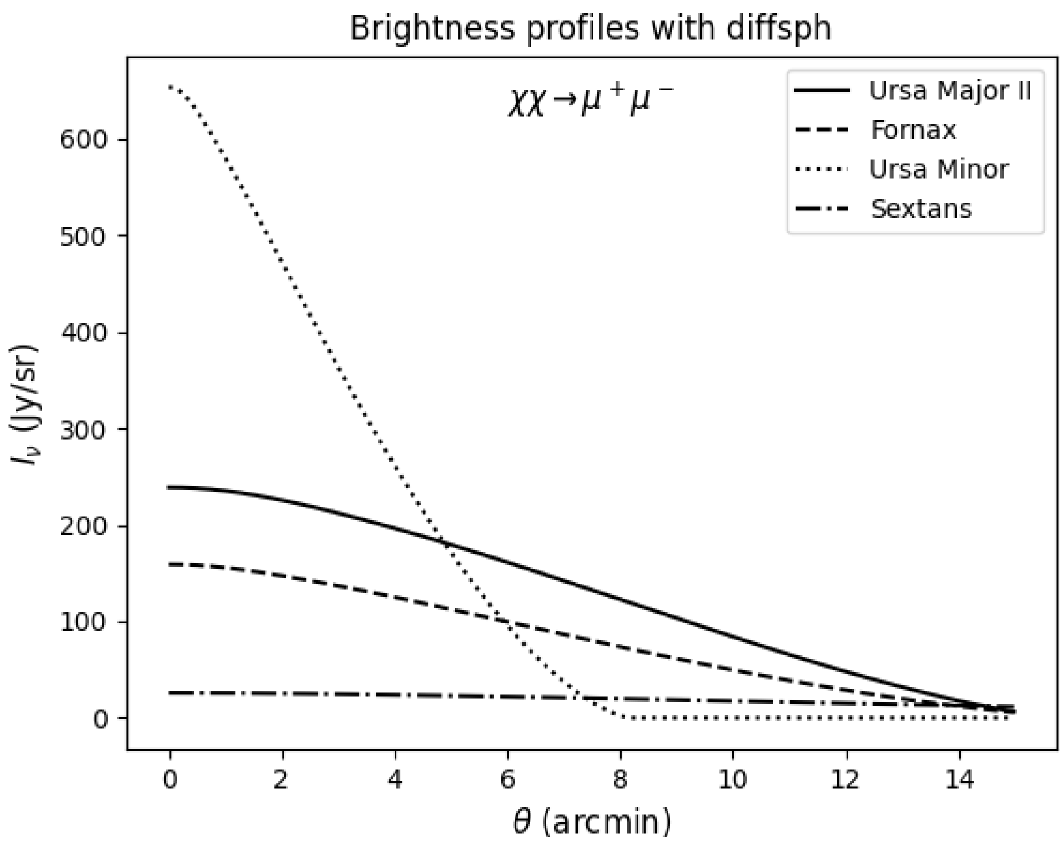
CHAPTER FIVE

EXAMPLES

5.1 pyflux module

In order to get familiar with the code, use the following set of commands to generate the figure below:

```
from diffsph import pyflux as pf
import matplotlib.pyplot as plt
%matplotlib inline

# Angle grid in arcmin

theta_grid = [15 * i / 50 for i in range(0, 50)]

# List of satellite galaxies

dsph_list = ['Ursa Major II', 'Fornax', 'Ursa Minor', 'Sextans']

# diffsph's computations at nu = 150 MHz and for the given model

Inu = [[pf.synch_brightness(th, nu = .150, galaxy = gal, rad_temp = 'HDZ',
                            hyp = 'wimp', ref = '1408.0002', sv = 3e-26,
                            mchi = 10, channel = 'mumu', high_res = True,
                            accuracy = .1)
             for th in theta_grid]
            for gal in dsph_list]

# Plots

plt.plot(theta_grid, Inu[0], "k", label = dsph_list[0])
plt.plot(theta_grid, Inu[1], "--k", label = dsph_list[1])
plt.plot(theta_grid, Inu[2], ":k", label = dsph_list[2])
plt.plot(theta_grid, Inu[3], "-.k", label = dsph_list[3])
plt.legend()
plt.xlabel('$\\theta$ (arcmin)', size = 'large');
plt.ylabel('$I_\\nu$ (Jy/sr)', size = 'large');
plt.title('Brightness profiles with diffsph');
plt.text(7.5, 630, '$\chi\chi\, \\to\, \mu^+\mu^-$',
         horizontalalignment = 'center', size = 'large');
```



5.2 limits module

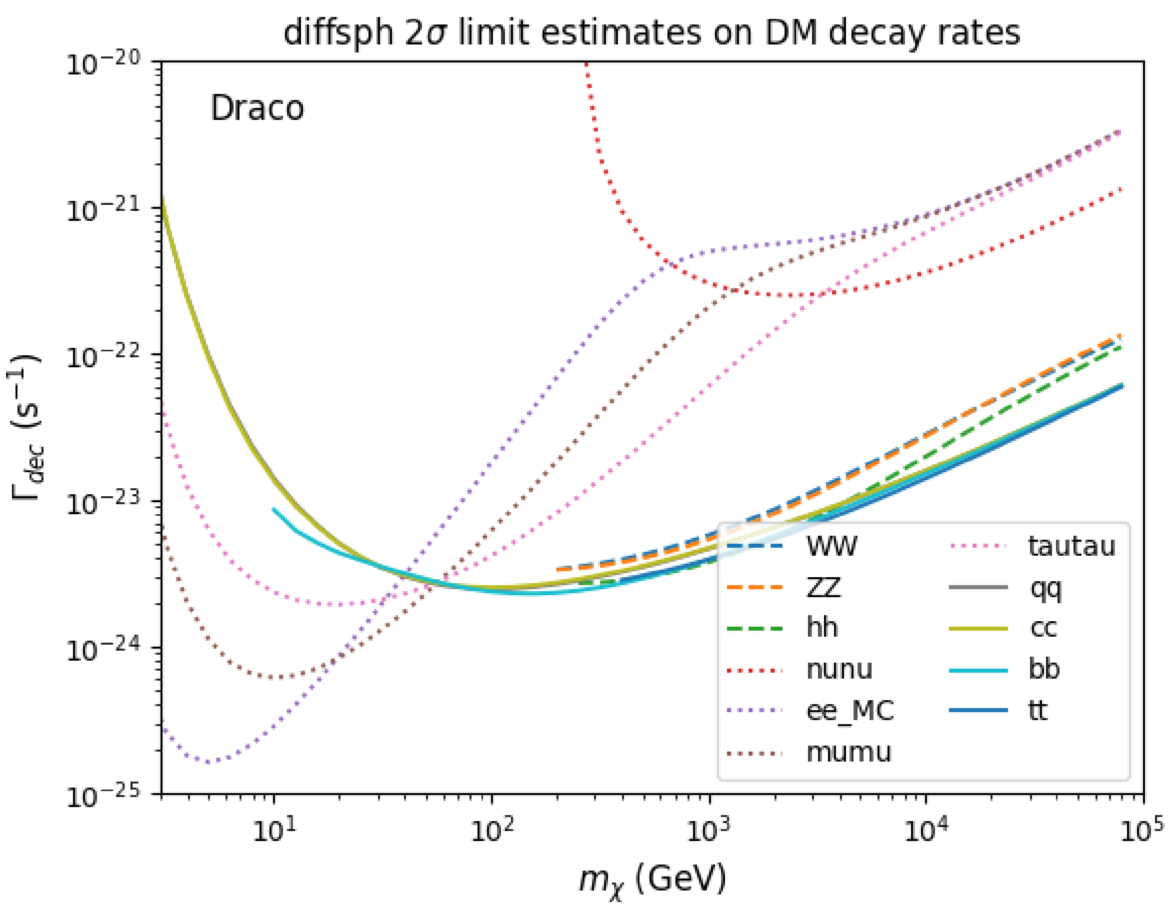
The following example shows how to obtain limits on the decay rate of dark matter particles using non-detection (noise level) in the field of Draco. It can take just a few minutes to compute them all in a modern laptop:

```
from diffsph import limits as lims
import matplotlib.pyplot as plt
%matplotlib inline

# DM mass grid in GeV

mass_grid = [10 ** (5 * i / 50) for i in range(0, 50)]

# List of decay channels

ch_list = ['WW', 'ZZ', 'hh', 'nunu', 'ee_MC', 'mumu', 'tautau', 'qq', 'cc', 'bb', 'tt']

# diffsph's computations at nu = 150 MHz and for the given image

rates = [[lims.decay_rate_limest(nu = .15, rms_noise = 100, beam_size = 20,
                                  galaxy = 'Draco', rad_temp = 'HDZ', mchi = mch,
                                  channel = ch, high_res = True, accuracy = .1,
```

(continues on next page)

(continued from previous page)

```
                                    ref = '1408.0002')
            for mch in mass_grid]
           for ch in ch_list]

# Plots

[plt.loglog(mass_grid, rates[i], label = ch_list[i], ls = '--') for i in range(0, 3)]
[plt.loglog(mass_grid, rates[i], label = ch_list[i], ls = ':') for i in range(3, 7)]
[plt.loglog(mass_grid, rates[i], label = ch_list[i]) for i in range(7, len(ch_list))]
plt.ylim([1e-25, 1e-20]);
plt.xlim([3, 1e5]);
plt.legend(loc = 'lower right', ncols = 2)
plt.xlabel('$m_\chi$ (GeV)', size = 'large');
plt.ylabel('$\Gamma_{dec}$ (s${}^{-1}$)', size = 'large');
plt.title('diffsph 2$\sigma$ limit estimates on DM decay rates');
plt.text(5, 4e-21, 'Draco', size = 'large');
```



References

PYTHON MODULE INDEX

d