ISAPP school MAD$^{(\gamma)}$ 2021

ISAPP 2021
International School on AstroParticle Physics
MAD$^{(\gamma)}$
Gamma rays to shed light on dark matter
21 - 30 June
ONLINE EVENT

Gamma rays to shed light on dark matter

Hands-on session: *DRAGON2* code

Pedro de la Torre Luque - on behalf of the DRAGON team
pedro.delatorreluque@fysik.su.se

Session outline

Hands-on session
***DRAGON2* code**

(i) The code and its structure

(ii) Running examples: Running a *DRAGON* input

(iii) Reading and handling the *DRAGON* outputs

(iv) Some recent scientific contributions using *DRAGON2*

The *DRAGON2* code

- **Where to find and download it**

  https://github.com/cosmicrays/DRAGON2-Beta_version

- **Previous version of the code**

  https://github.com/cosmicrays/DRAGON

Technical papers

Numerical solver and astrophysical ingredients

https://arxiv.org/abs/1607.07886

Nuclear interactions with the interstellar gas

https://arxiv.org/abs/1711.09616

The *DRAGON2* code

DRAGON2 is written in C++, taking also some routines written in Fortran.

Generally, it is aimed at numerically solving the propagation equation of charged cosmic rays (CRs) in the **Galaxy**. It is able to compute the density of CR particles $\left(\frac{\partial N}{\partial R\, \partial A\, dt\, d\Omega}\right)$ at whatever position in the Galaxy and energy range.

It implements some predefined functions for each of the terms involved in the propagation equation and offers the possibility to you adjust their parameters in the input.

$$\vec{\nabla}\cdot(-D\,\nabla N_i - \vec{v}_\omega N_i) + \frac{\partial}{\partial p}\left[p^2 D_{pp}\frac{\partial}{\partial p}\left(\frac{N_i}{p^2}\right)\right] = Q_i + \frac{\partial}{\partial p}\left[\dot{p}N_i - \frac{p}{3}\left(\vec{\nabla}\cdot\vec{v}_\omega N_i\right)\right]$$
$$-\frac{N_i}{\tau_i^f} + \sum \Gamma^s_{j\to i}(N_j) - \frac{N_i}{\tau_i^r} + \sum \frac{N_j}{\tau^r_{j\to i}}$$

The *DRAGON2* code

- **Where to find and download it**

  https://github.com/cosmicrays/DRAGON2-Beta_version

Technical papers

Numerical solver and astrophysical ingredients

https://arxiv.org/abs/1607.07886

Nuclear interactions with the interstellar gas

https://arxiv.org/abs/1711.09616

✓ **All what you need for the session at**

https://github.com/tospines/ISAPP-school-2021_HandsOn-DRAGON2

Let’s put the hands on it

Download the contents of the Github repository:

git clone https://github.com/tospines/ISAPP-school-2021_HandsOn-DRAGON2.git

Follow the instructions in the “Installation_instructions-DRAGON2” file to download *DRAGON2:*

git clone https://github.com/cosmicrays/DRAGON2-Beta_version.git

Install *DRAGON2:*

start.sh

configure --with-numcpu=2 --with-cfitsio=$CFITSIO_DIR --with-gsl-path=$GSL_bin_path

make

The *DRAGON2* code: Structure

Main modules are the .cc files → They contain the main ingredients of the equation

include directory with the header (.h) files

data folder contains tables of cross sections, interstellar radiation field (ISRF) and the isotope list (list of isotopes and their decay modes)

Some Fortran routines that are compiled with the code (.f files and dmspec.F)

examples directory contains the samples of inputs to be used

The **output** folder saves the output files created by *DRAGON*

plots directory contains plotting routine, resultant plots and experimental datasets

The *DRAGON2* code: Structure

- The basic idea is that primary particles are accelerated in astrophysical sources (namely SNRs) and propagate throughout the Galaxy, occasionally interacting with gas, mainly in the disc of the Galaxy, and there they produce secondary nuclei through spallation.

$$\frac{\partial N}{\partial R} q(x,y,z)$$

$$\vec{\nabla}\cdot\left(-D\,\nabla N_i + \vec{v}_\omega N_i\right) + \frac{\partial}{\partial p}\left[p^2 D_{pp}\frac{\partial}{\partial p}\left(\frac{N_i}{p^2}\right)\right] = Q_i + \frac{\partial}{\partial p}\left[\dot{p}N_i - \frac{p}{3}\left(\vec{\nabla}\cdot\vec{v}_\omega N_i\right)\right] - \frac{N_i}{\tau_i^f} + \sum \Gamma^s_{j\to i}(N_j) - \frac{N_i}{\tau_i^r} + \sum \frac{N_j}{\tau^r_{j\to i}}$$

$$D_{pp} \propto \frac{p^2\, VA^2}{D}$$

$$\frac{1}{\tau_i^f(T)} = c n_H \beta(T)\left[\sigma_{i\,p}(T) + f_{He}\,\sigma_{i\,He}(T)\right]$$

$$\frac{1}{\tau_i^r(T)} = \frac{1}{\gamma(T)\tau_i^0}$$

$$D = D_0 \beta^\eta \left(\frac{R}{R_0}\right)^\delta F(\vec{r},z)$$

$$\frac{\partial p}{\partial t} = \left(\frac{\partial p}{\partial t}\right)_{Coul,\,Ioniz,\,pi0} + \left(\frac{\partial p}{\partial t}\right)_{IC,\,Bremss,\,Sync}$$

$$\Gamma^s_{j\to i} = \beta_j c n_t \sigma_{j\to i} N_j$$

The *DRAGON2* code: Structure

$$\vec{\nabla}\cdot\left(-D\,\nabla N_i + \vec{v}_\omega N_i\right) + \frac{\partial}{\partial p}\left[p^2 D_{pp}\frac{\partial}{\partial p}\left(\frac{N_i}{p^2}\right)\right] = Q_i + \frac{\partial}{\partial p}\left[\dot{p}N_i - \frac{p}{3}\left(\vec{\nabla}\cdot\vec{v}_\omega N_i\right)\right] - \frac{N_i}{\tau_i^f} + \sum \Gamma^s_{j\to i}(N_j) - \frac{N_i}{\tau_i^r} + \sum \frac{N_j}{\tau^r_{j\to i}}$$

galaxy.cc

particle.cc

galaxy.cc

eloss.cc
bfield.cc

xsec.cc
gas.cc

spectrum.cc
sources.cc

xsec.cc
particle.cc
kamae.cc
gas.cc

diffusion.cc

The *DRAGON2* code: Structure

A grid where the evaluation is performed is built in **grid.cc**

$$\vec{\nabla} \cdot (-D \nabla N_i + \vec{v}_\omega N_i) + \frac{\partial}{\partial p}\left[p^2 D_{pp} \frac{\partial}{\partial p}\left(\frac{N_i}{p^2}\right)\right] = Q_i + \frac{\partial}{\partial p}\left[\dot{p} N_i - \frac{p}{3}\left(\vec{\nabla} \cdot \vec{v}_\omega N_i\right)\right] - \frac{N_i}{\tau_i^f} + \sum \Gamma^s_{j \to i}(N_j) - \frac{N_i}{\tau_i^r} + \sum \frac{N_j}{\tau^r_{j \to i}}$$

Input.cc file reads input parameters

main.cc → **dragon.cc**

- Create the particle to be propagated → **particle.cc**
- Evolve the particle in time (and space) → **crevolutor.cc**
- Normalize the computed spectra to proton observations

Nuclear chain built in the **nucleilist.cc** file

Basic input → .xml file

- **Output information:**

  <Output>

- **A first part characterizing the spatial grid in which we do the simulation**

  <Grid type="2D"> # Also options for 3D

- **Nuclei to be propagated**

  <NuclearChain>

- **Timesteps to perform the computation (make sure it converges!)**

  <Algorithm>

  <OpSplit>

- **Specify ingredients of the Galaxy**

  <Galaxy> ----- <Diffusion>

$$D = D_0 \beta^{\eta} \left(\frac{R}{R_0}\right)^{\delta} F(\vec{r}, z)$$

- **Normalization of protons and electrons and (optionally) their injection parameters**

  <CR>

Basic input → .xml file

- **Ingredients of the Galaxy**

<Galaxy>

- Source distribution
- Gas distribution
- Diffusion coeff.

<Diffusion>

$$D = D_0 \beta^\eta \left(\frac{R}{R_0}\right)^\delta F(\vec{r}, z)$$

$$D_{pp} = \frac{4}{3\delta(4-\delta^2)(4-\delta)} \frac{v_A^2 p^2}{D(R)}$$

<Reacceleration>

<Convection>

$$v_w(z) = v_{w,0} + \frac{dv_w}{dz} z$$

- Cross sections
- Magnetic field config.

- **Normalization of protons and electrons and (optionally) their injection parameters**

<CR>

- Normalization block

<ProtNormEn_GeV value="53.645" />

<ElNormEn_GeV value="33." />

<ProtNormFlux value="2.57e-1" />

<ElNormFlux value="0.0046" />

- Injection all nuclei
- Injection electrons
- Injection extra-source

Basic input → .source.param file

- **Adjustment of the spectrum of every species individually: Whichever number of breaks**

$Z_1 0 A_1$ $\quad$ Norm_{ZA} $\quad$ γ_1 $\quad$ R_1 $\quad$ γ_2 $\quad$ R_2 $\quad$ γ_3 $\quad$ R_3 $\quad$ … $\quad$ γ_n $\quad$ R_n

$Z_2 0 A_2$ ….

.

.

.

$$Q = \begin{cases} K_1 \times \left(\frac{R}{R_0}\right)^{\gamma_1} & \text{for } R < R_1 \\ K_2 \times \left(\frac{R}{R_0}\right)^{\gamma_2} & \text{for } R_1 < R < R_2 \\ K_3 \times \left(\frac{R}{R_0}\right)^{\gamma_3} & \text{for } R > R_2 \end{cases}$$

$$K_1 = \mathrm{Norm}_{ZA}$$

$$K_i = K_{(i-1)} \left(\frac{R_{(i-1)}}{R_0}\right)^{\gamma_{(i-1)} - \gamma_i} \quad \text{for } i > 1$$

Let's run our inputs and explore the outputs

First, a basic input in which we reproduce the spectra of primary cosmic rays and the B/C,O flux ratios:

./DRAGON Inputs/BaseModel_DRAGONxsec.xml

Go to the *DRAGON* **output folder** and see what has been created.

Visualize our results with the routines given in the **plotting_output-Routines folder.**

Read the **_spectrum.fits.gz** with *Plotting_General_Spectra.ipynb*

Read the **.fits.gz** with *Radial_spectrum.ipynb*

Now, compute the rest of the secondary-to-primary ratios (Be/C, Be/O, Li/C, Li/O) and tell me what you see. What do you think is the reason of that?

Some recent contributions of the code

- Markov chain Monte Carlo analyses of the flux ratios of B, Be and Li with the DRAGON2 code
  https://arxiv.org/abs/2102.13238
- Implications of current nuclear cross sections on secondary cosmic rays with the upcoming DRAGON2 code
  https://arxiv.org/abs/2101.01547
- The theory of cosmic-ray scattering on pre-existing MHD modes meets data
  https://arxiv.org/abs/2011.09197
- Changes in cosmic-ray transport properties connect the high-energy features in the electron and proton data
  https://arxiv.org/abs/2007.15321
- Features in cosmic-ray lepton data unveil the properties of nearby cosmic accelerators
  https://arxiv.org/abs/1907.03696

Let's run our inputs and explore the outputs

Now, we are going to visualize the spectra of CRs at different positions in the Galaxy with the Radial_spectrum.ipynb

Then, a basic input for calculating electron and positron spectra is prepared to run. Read it with Electrons_positrons.ipynb:

./DRAGON Inputs/BaseModel_Electrons.xml

Finally, another example is set to show you how to compute the antiproton spectra. Use this time the DRAGON2_Antiprotons.ipynb

Hands-on session: *DRAGON2* code

- Download it here → https://github.com/cosmicrays DRAGON2-Beta_version

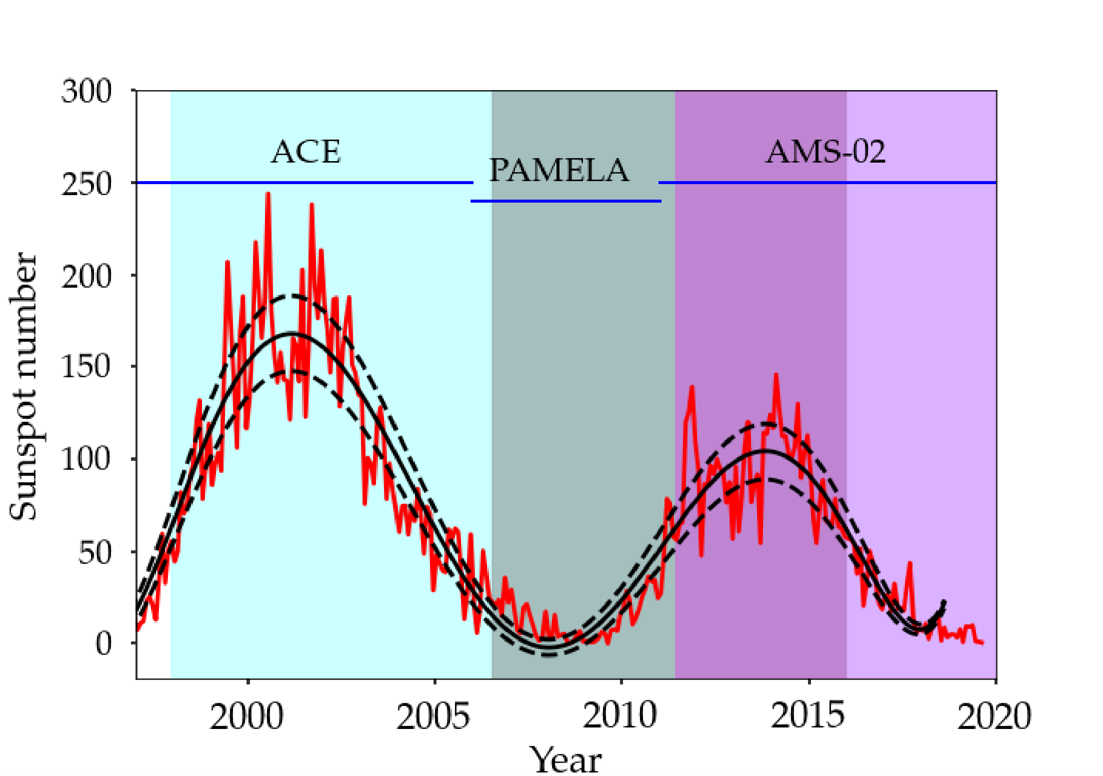
Cosmic-ray propagation with DRAGON2: I. numerical solver and astrophysical ingredients → arXiv:1607.07886

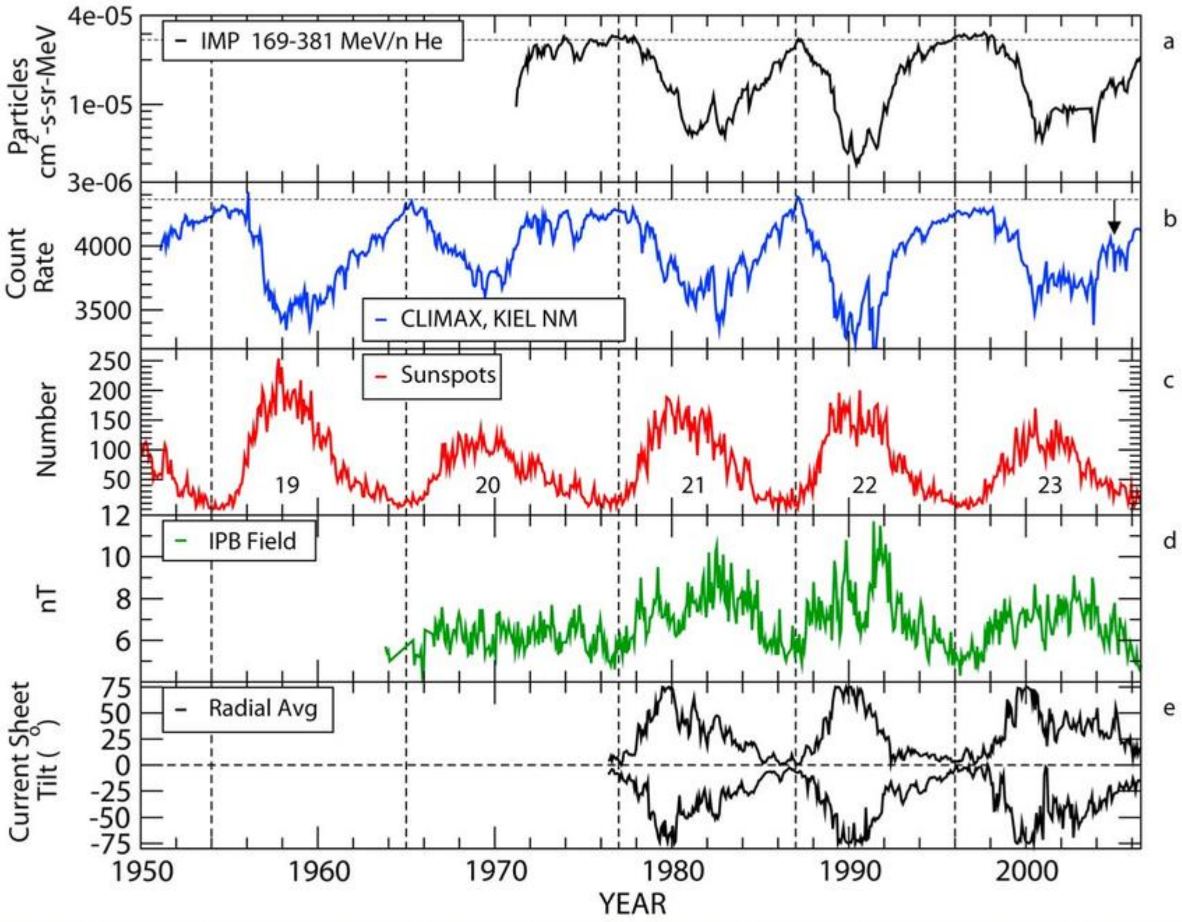
Cosmic-ray propagation with DRAGON2: II. Nuclear interactions with the interstellar gas → arXiv:1711.09616

- Databases for CR data → https://lpsc.in2p3.fr/cosmic-rays-db/
  https://tools.ssdc.asi.it/CosmicRays/

BACK UP

SOLAR MODULATION



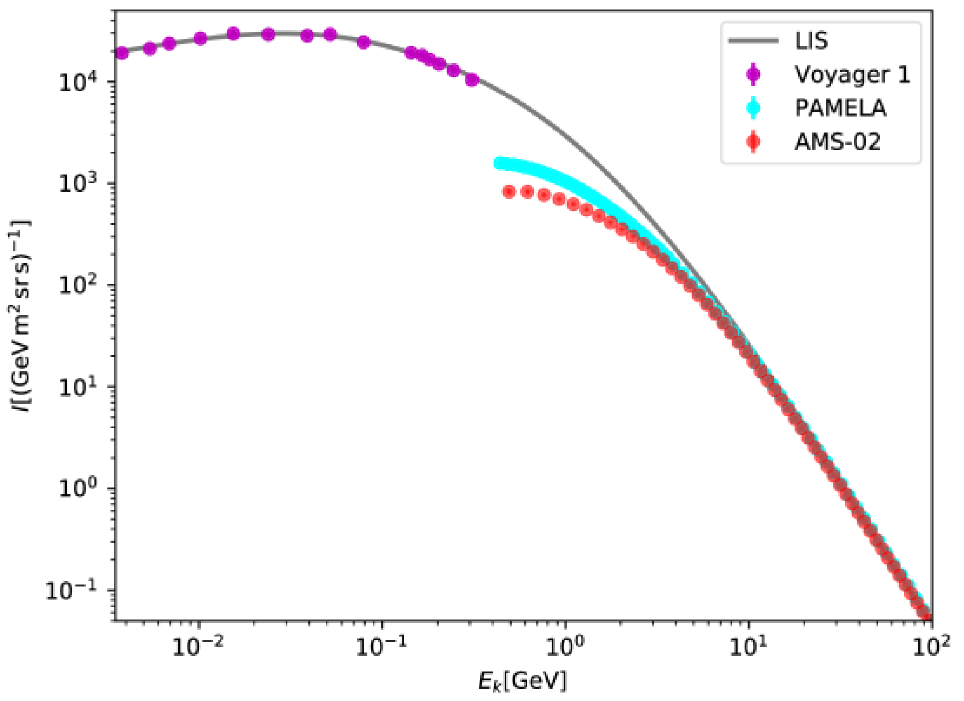


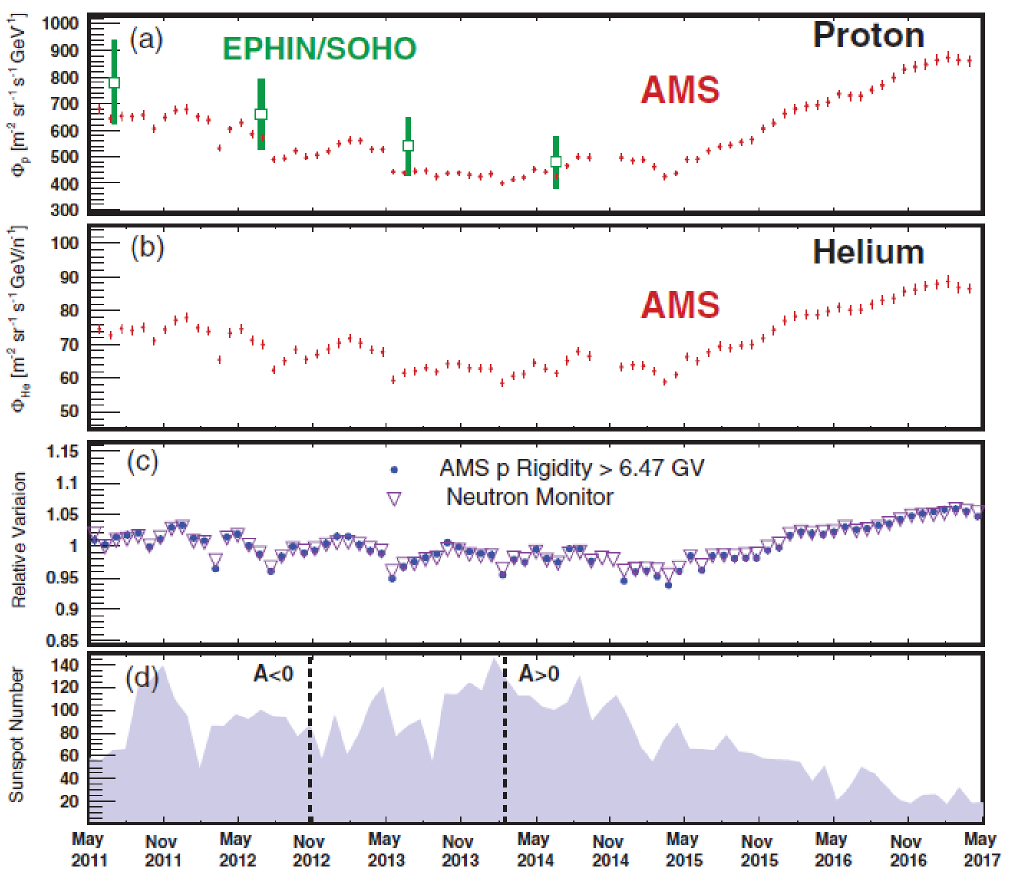
- Force-Field approximation
- Neutron monitor data + Voyager-01 data
- Cholis-Hooper-Linden (arXiv:1511.01507) correction

$$\Phi^{\mathrm{TOA}}(T) = \frac{2mT + T^2}{2m\,(T + \frac{Z}{A}\phi) + (T + \frac{Z}{A}\phi)^2}\,\Phi^{\mathrm{IS}}(T + \tfrac{Z}{A}\phi)$$

$$\phi^{\pm}(t, \mathcal{R}) = \phi_0(t) + \phi_1^{\pm}(t)\;\mathcal{F}\left(\frac{\mathcal{R}}{\mathcal{R}_0}\right)$$

SOLAR MODULATION





- Force-Field approximation
- Neutron monitor data + Voyager-01 data
- Cholis-Hooper-Linden (arXiv:1511.01507) correction

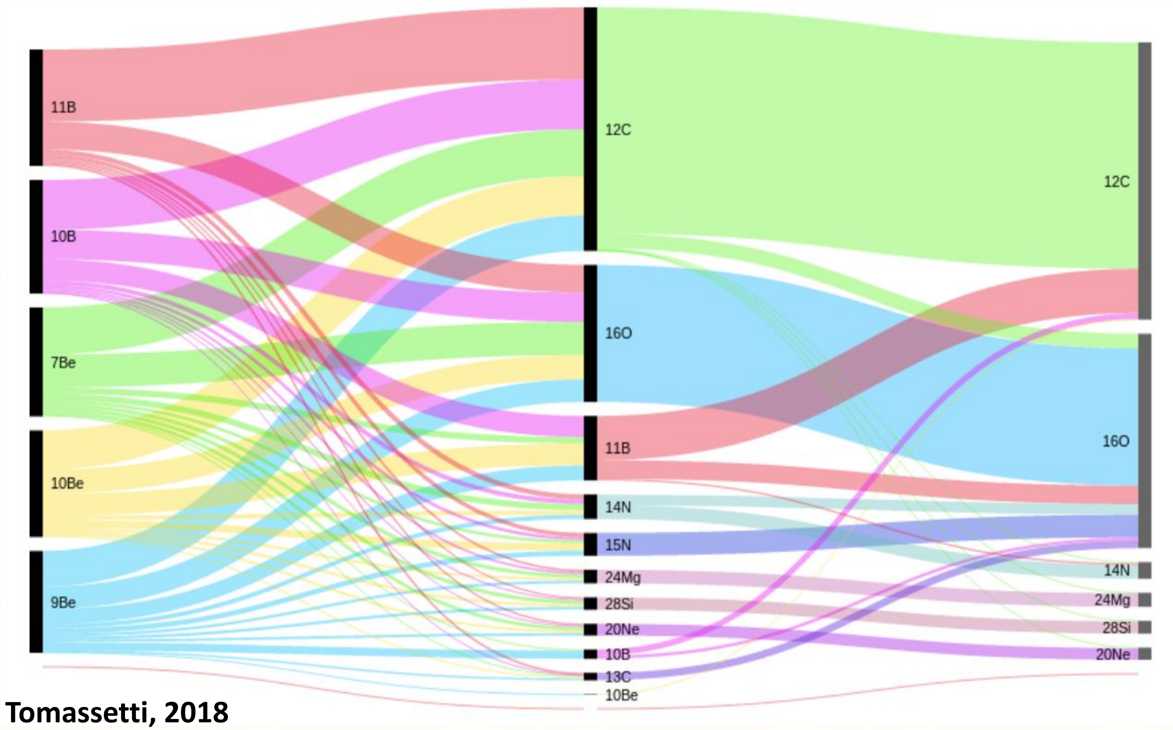
$$\Phi^{\mathrm{TOA}}(T) = \frac{2mT + T^2}{2m\,(T + \frac{Z}{A}\phi) + (T + \frac{Z}{A}\phi)^2}\,\Phi^{\mathrm{IS}}(T + \frac{Z}{A}\phi)$$

$$\phi^{\pm}(t, \mathcal{R}) = \phi_0(t) + \phi_1^{\pm}(t)\;\mathcal{F}\left(\frac{\mathcal{R}}{\mathcal{R}_0}\right)$$

Cross sections → Secondary CRs

$$Q_{sec}(E) \propto \sum^{pr} J_{pr}(E)\, \sigma_{pr \rightarrow sec}(E)$$

Complexity of the CS network



Tomassetti, 2018

Production of secondary CRs

- Main spallation channels: O and C
- Secondary channels (N, Ne, Mg, Si & Fe) are very important for Li and Be (< 50%)
- Tertiary channels also matter:
  e.g. ^{11}B + gas → ^{10}B + X

Genolini et al. 2019 ; arXiv:1803.04686
Tomassetti, 2018 ; arXiv:1707.06917

Diffusion coefficient parametrization