PRIMAT Tutorial

PRImordial MATter

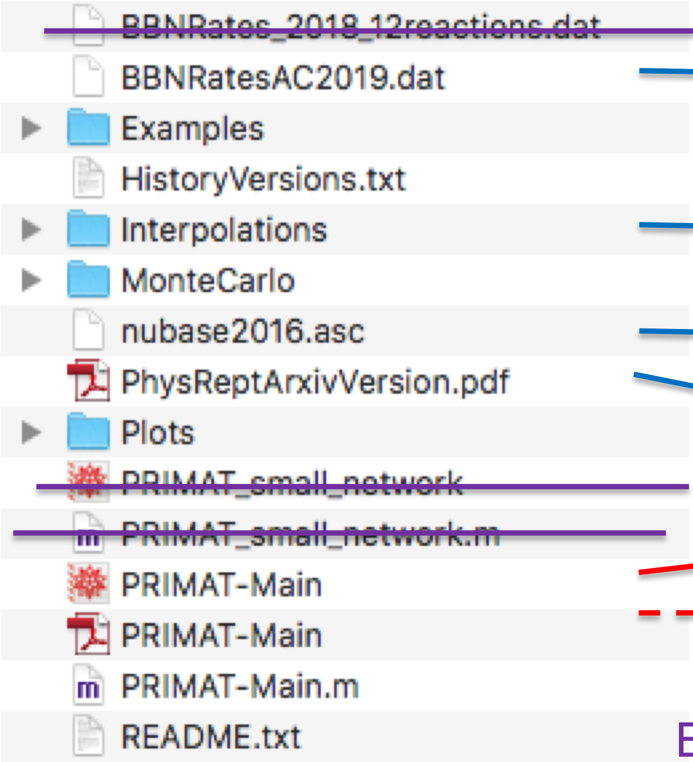
Cyril Pitrou

Download and install

Download the zip file by filling the form on :

http://www2.iap.fr/users/pitrou/primat.htm

Unzip. You should see



~~BBNRates_2018_12reactions.dat~~
BBNRatesAC2019.dat
Examples
HistoryVersions.txt
Interpolations
MonteCarlo
nubase2016.asc
PhysReptArxivVersion.pdf
Plots
~~PRIMAT_small_network~~
~~PRIMAT_small_network.m~~
PRIMAT-Main
PRIMAT-Main
PRIMAT-Main.m
README.txt

Tabulated nuclear rates

Stored interpolations of weak rates

Masses and properties of nuclides

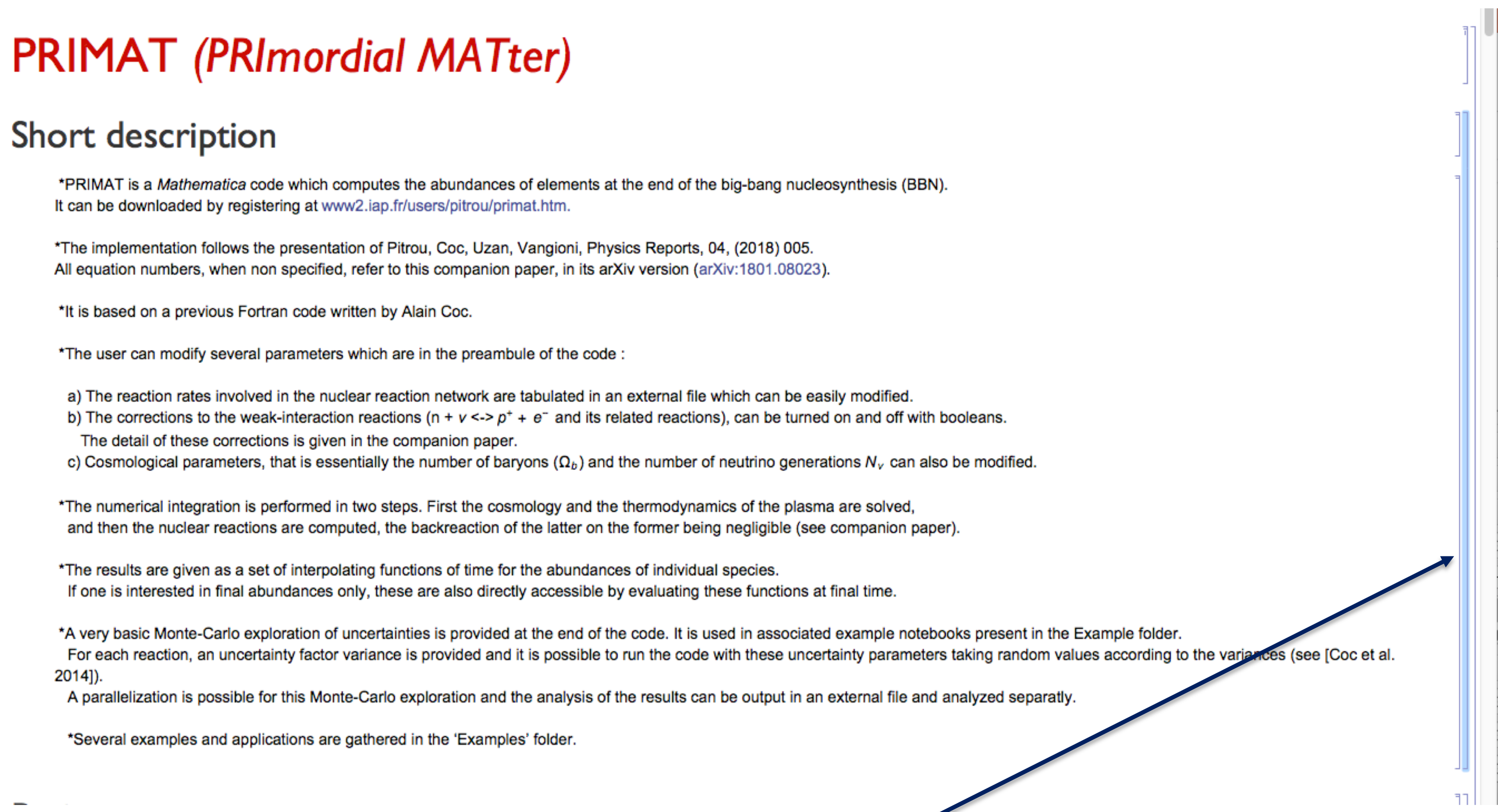
Reference publication

Main notebook

PDF dump to see the code without *Mathematica*

Barred purple files are somehow obsolete
(to be removed in next release)

PRIMAT-Main.nb (need *Mathematica* to open it)



PRIMAT *(PRImordial MATter)*

Short description

*PRIMAT is a *Mathematica* code which computes the abundances of elements at the end of the big-bang nucleosynthesis (BBN).
It can be downloaded by registering at www2.iap.fr/users/pitrou/primat.htm.

*The implementation follows the presentation of Pitrou, Coc, Uzan, Vangioni, Physics Reports, 04, (2018) 005.
All equation numbers, when non specified, refer to this companion paper, in its arXiv version (arXiv:1801.08023).

*It is based on a previous Fortran code written by Alain Coc.

*The user can modify several parameters which are in the preambule of the code :

a) The reaction rates involved in the nuclear reaction network are tabulated in an external file which can be easily modified.
b) The corrections to the weak-interaction reactions (n + ν <-> p^+ + e^- and its related reactions), can be turned on and off with booleans.
The detail of these corrections is given in the companion paper.
c) Cosmological parameters, that is essentially the number of baryons (Ω_b) and the number of neutrino generations N_ν can also be modified.

*The numerical integration is performed in two steps. First the cosmology and the thermodynamics of the plasma are solved,
and then the nuclear reactions are computed, the backreaction of the latter on the former being negligible (see companion paper).

*The results are given as a set of interpolating functions of time for the abundances of individual species.
If one is interested in final abundances only, these are also directly accessible by evaluating these functions at final time.

*A very basic Monte-Carlo exploration of uncertainties is provided at the end of the code. It is used in associated example notebooks present in the Example folder.
For each reaction, an uncertainty factor variance is provided and it is possible to run the code with these uncertainty parameters taking random values according to the variances (see [Coc et al. 2014]).
A parallelization is possible for this Monte-Carlo exploration and the analysis of the results can be output in an external file and analyzed separatly.

*Several examples and applications are gathered in the 'Examples' folder.

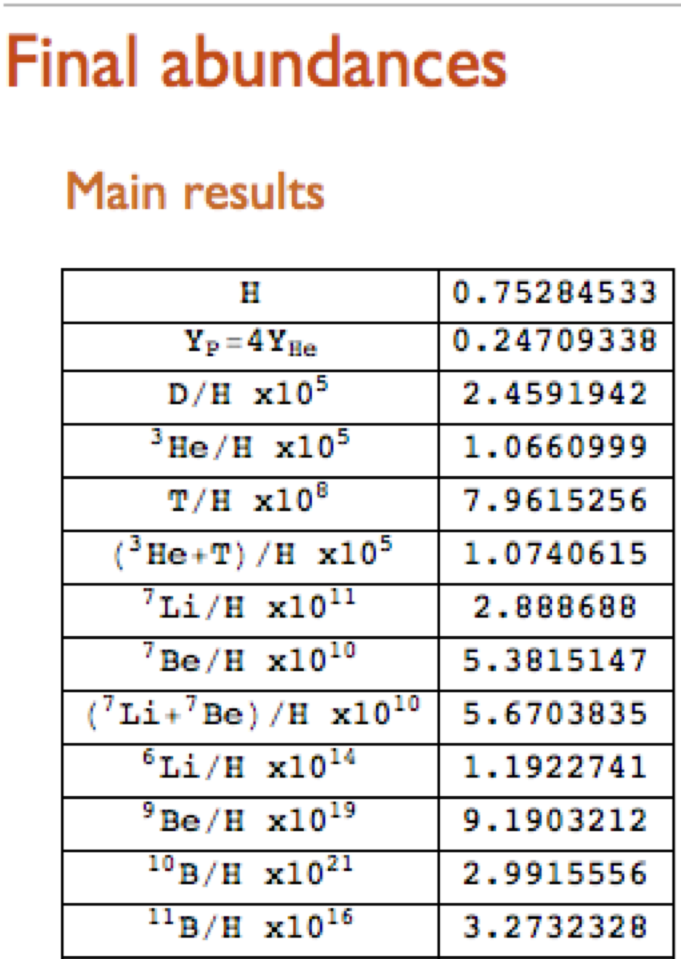
For more readability, you can fold individual parts, sections and subsections by double clicking the cell selection on the right

Run the file

1) Go in menu Evaluation and then select ‘Evaluate Notebook’

2) If *Mathematica* asks to evaluate initialization cells, answer no
(because you want to evaluate everything, not just these cells)

3) To simply read the results, go to the final sections (and unfold them if they are closed)



Final abundances

Main results

H	0.75284533
$Y_P = 4Y_{He}$	0.24709338
D/H x10^5	2.4591942
^{3}He/H x10^5	1.0660999
T/H x10^8	7.9615256
(^{3}He+T)/H x10^5	1.0740615
^{7}Li/H x10^{11}	2.888688
^{7}Be/H x10^{10}	5.3815147
(^{7}Li+^{7}Be)/H x10^{10}	5.6703835
^{6}Li/H x10^{14}	1.1922741
^{9}Be/H x10^{19}	9.1903212
^{10}B/H x10^{21}	2.9915556
^{11}B/H x10^{16}	3.2732328

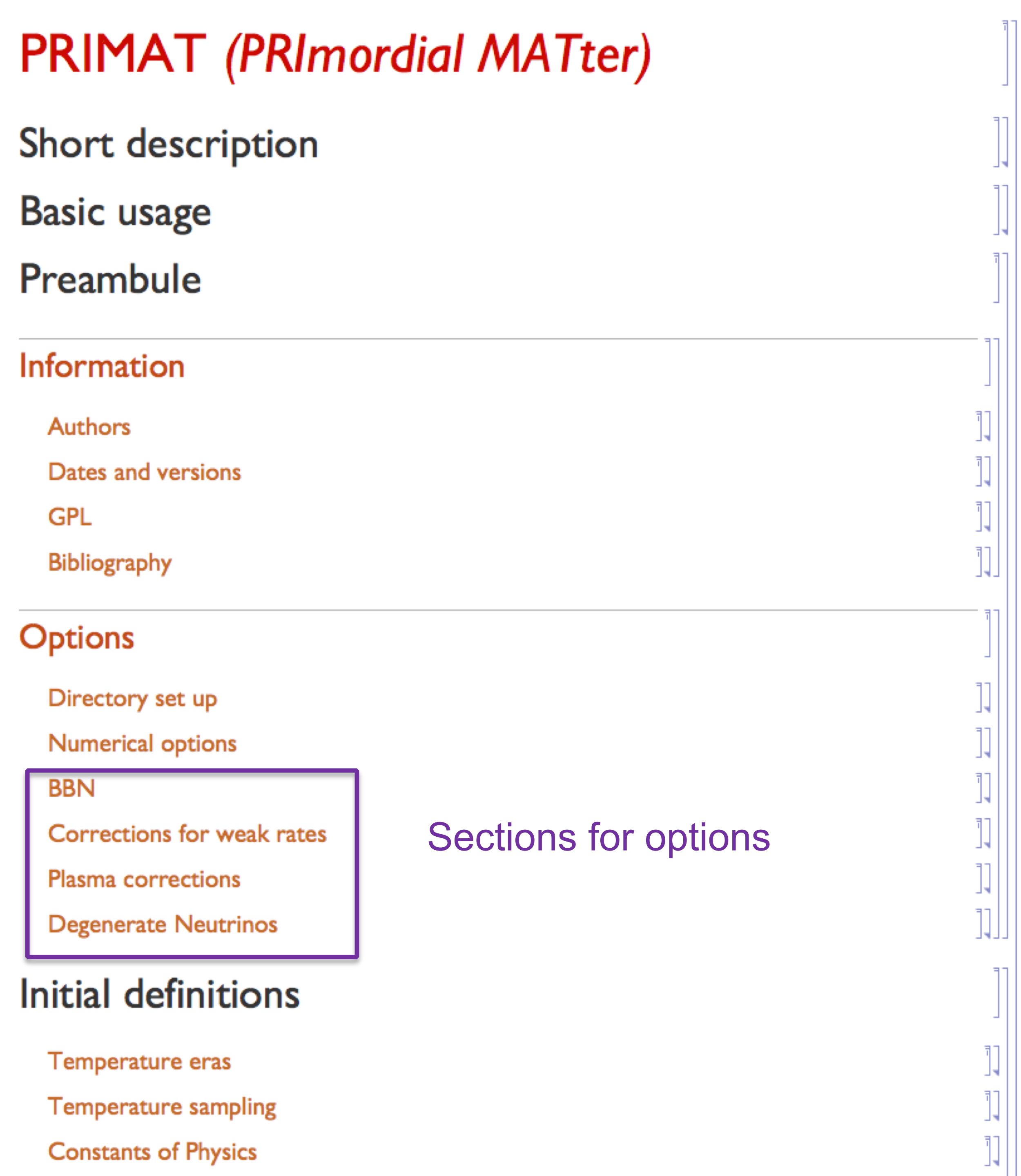
Outline of brief tutorial

1. Modifying options

2. Using results

3. Loading PRIMAT in an external notebook (advanced analysis)

1. Modifying options



PRIMAT *(PRImordial MATter)*

Short description

Basic usage

Preamble

Information

Authors

Dates and versions

GPL

Bibliography

Options

Directory set up

Numerical options

BBN

Corrections for weak rates

Plasma corrections

Degenerate Neutrinos

Sections for options

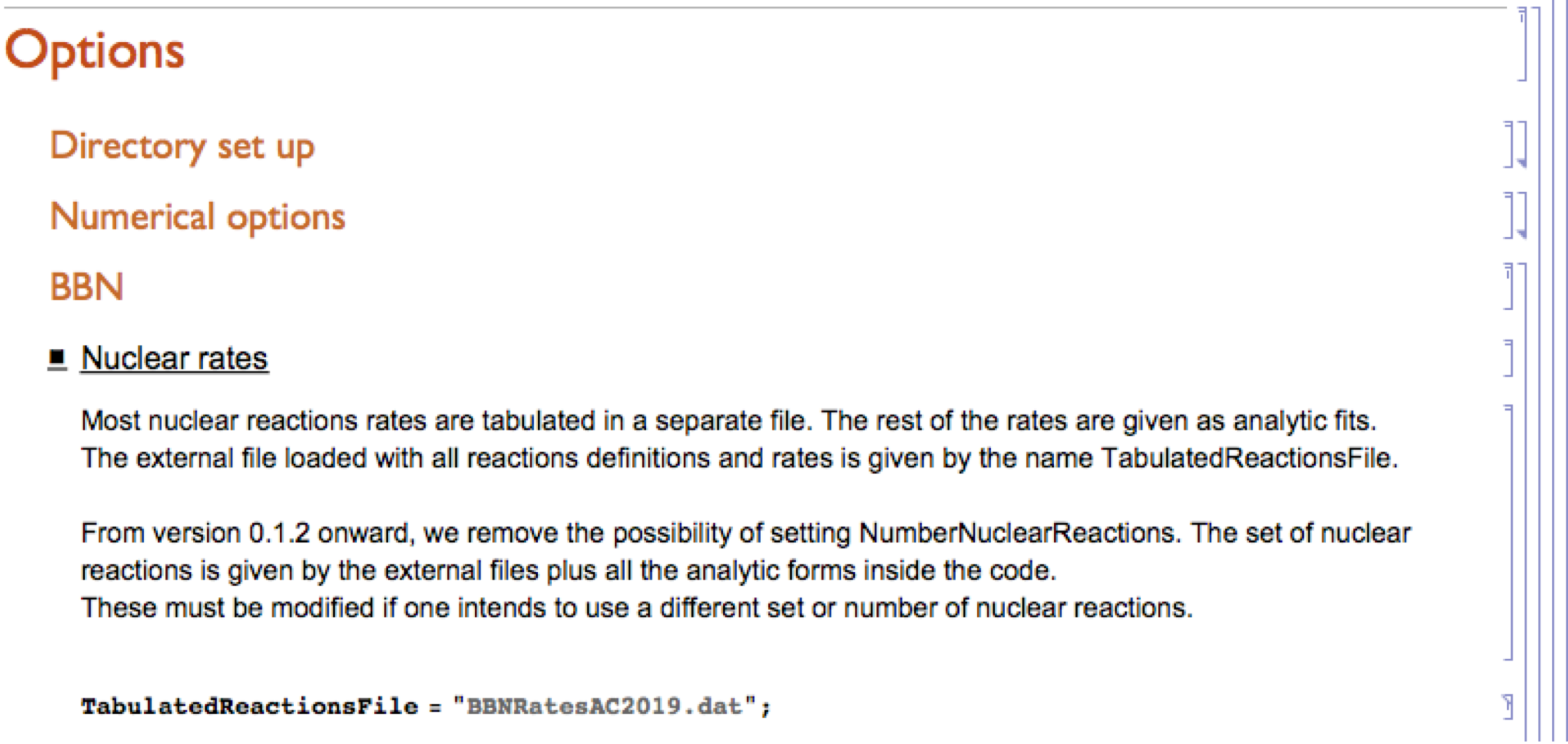
Initial definitions

Temperature eras

Temperature sampling

Constants of Physics

BBN Options



Options

Directory set up

Numerical options

BBN

■ Nuclear rates

Most nuclear reactions rates are tabulated in a separate file. The rest of the rates are given as analytic fits.
The external file loaded with all reactions definitions and rates is given by the name TabulatedReactionsFile.

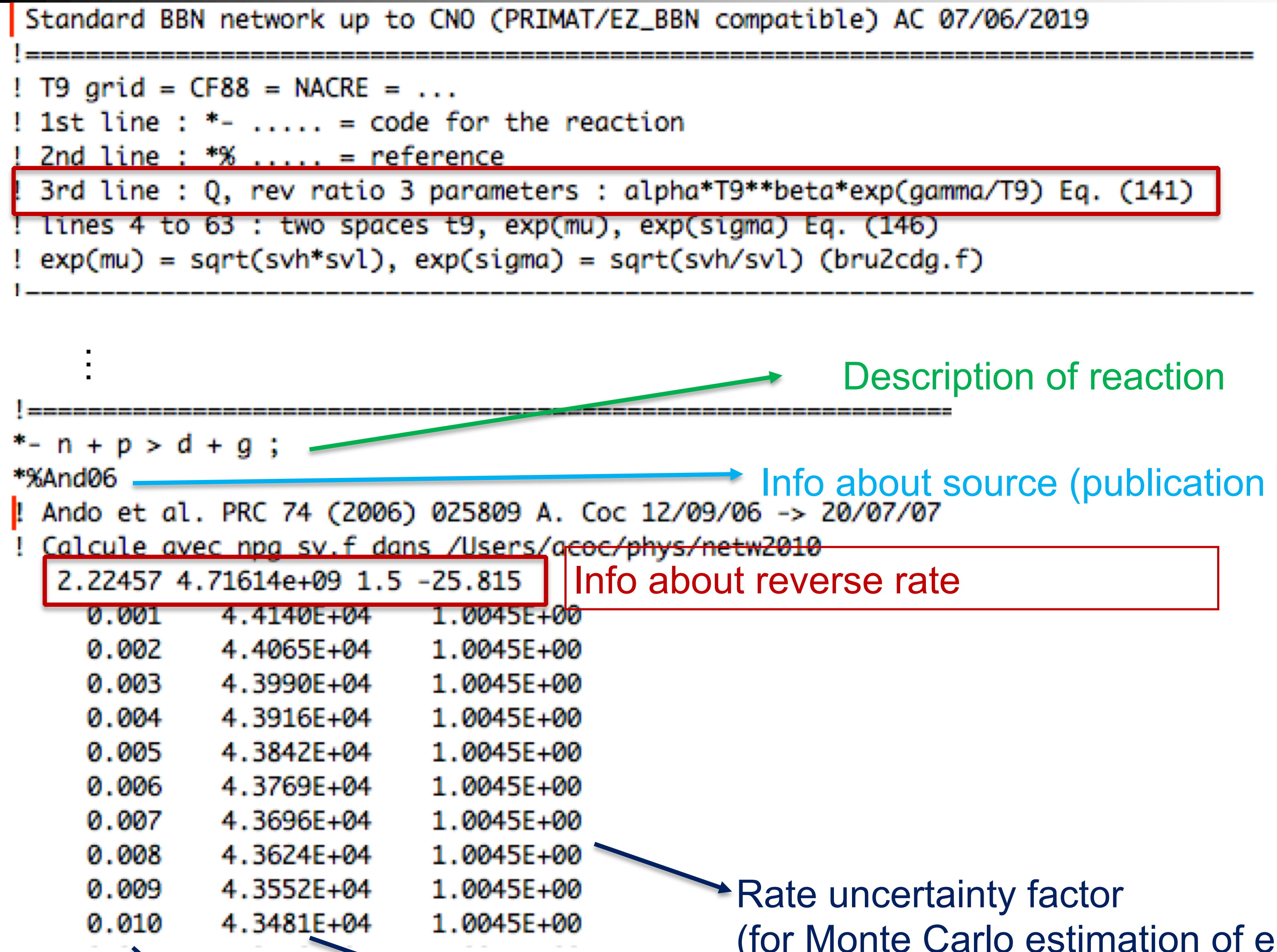
From version 0.1.2 onward, we remove the possibility of setting NumberNuclearReactions. The set of nuclear reactions is given by the external files plus all the analytic forms inside the code.
These must be modified if one intends to use a different set or number of nuclear reactions.

```
TabulatedReactionsFile = "BBNRatesAC2019.dat";
```

This is the external file with reaction rates tabulated.

Note : If you want to change a reaction rate, copy it to a different name,
then change the desired reaction rate, and set TabulatedReactionsFile
to be the name of your new file.

BBNRatesAC2019.dat file



```
 Standard BBN network up to CNO (PRIMAT/EZ_BBN compatible) AC 07/06/2019
!==========================================================================
! T9 grid = CF88 = NACRE = ...
! 1st line : *- ..... = code for the reaction
! 2nd line : *% ..... = reference
! 3rd line : Q, rev ratio 3 parameters : alpha*T9**beta*exp(gamma/T9) Eq. (141)
! lines 4 to 63 : two spaces t9, exp(mu), exp(sigma) Eq. (146)
! exp(mu) = sqrt(svh*svl), exp(sigma) = sqrt(svh/svl) (bru2cdg.f)
!--------------------------------------------------------------------------

    ⋮

!=========================================================
*- n + p > d + g ;
*%And06
! Ando et al. PRC 74 (2006) 025809 A. Coc 12/09/06 -> 20/07/07
! Calcule avec npg_sv.f dans /Users/acoc/phys/netw2010
  2.22457 4.71614e+09 1.5 -25.815
    0.001     4.4140E+04     1.0045E+00
    0.002     4.4065E+04     1.0045E+00
    0.003     4.3990E+04     1.0045E+00
    0.004     4.3916E+04     1.0045E+00
    0.005     4.3842E+04     1.0045E+00
    0.006     4.3769E+04     1.0045E+00
    0.007     4.3696E+04     1.0045E+00
    0.008     4.3624E+04     1.0045E+00
    0.009     4.3552E+04     1.0045E+00
    0.010     4.3481E+04     1.0045E+00
  ⋮
```

Description of reaction

Info about source (publication short name)

Info about reverse rate

Rate uncertainty factor
(for Monte Carlo estimation of errors)

Rate

Temperature in T9 (in GigaKelvin)

Changing rates given in analytic form

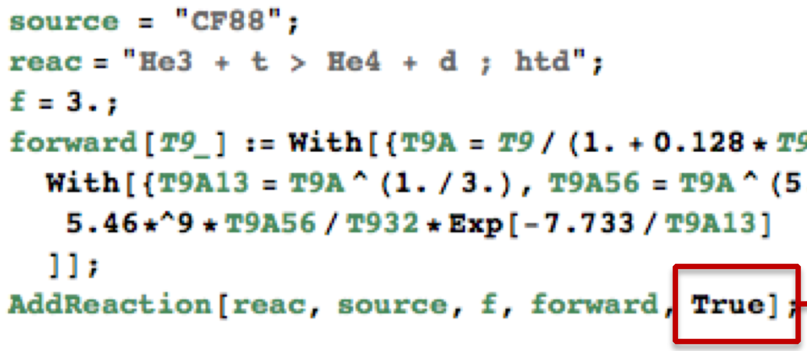
Not all reaction rates are given in the external files with tabulation on a grid.
In the core of the notebook, there is a section :

Lists of analytic reactions (87 reactions)

In this section is defined the function

```
DefineAnalyticRates :=
```

It lists all the analytic expression. A typical reaction is added as :



```
source = "CF88";
reac = "He3 + t > He4 + d ; htd";
f = 3.;
forward[T9_] := With[{T9A = T9 / (1. + 0.128 * T9), T932 = T9 ^ (3 / 2)},
   With[{T9A13 = T9A ^ (1. / 3.), T9A56 = T9A ^ (5. / 6.)},
    5.46*^9 * T9A56 / T932 * Exp[-7.733 / T9A13]
   ]];
AddReaction[reac, source, f, forward, True];
```

Boolean to also add reverse rate

source is the publication short name where the formula can be found

reac string giving initial and final species. After the ; is just a nickname for the reaction (htd here)

f is the uncertainty factor

forward is the analytic expression being defined as a function of T9 (T in GK).

BBN options

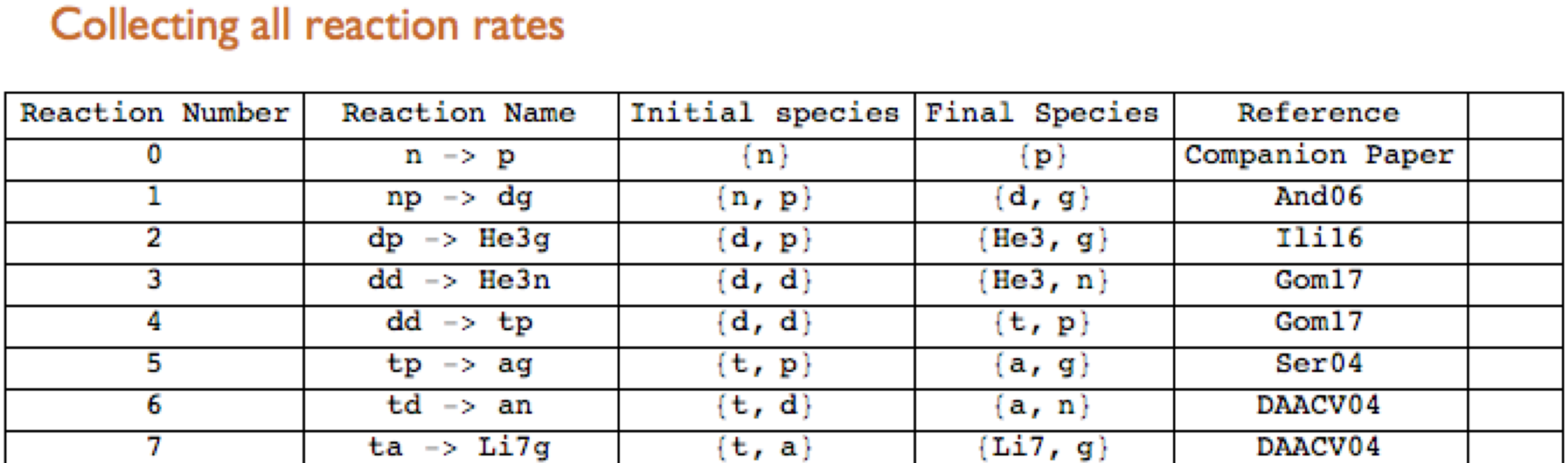
$RandomNuclearRates : set to True only when doing Monte Carlo

$RescaleSomeRates : set to True if you want to rescale rates (to check their influence)

In that case you must specify the rescaling factors :

```
If[$RescaleSomeRates,
 dpTOHe3gFactor = 1;
];
```

The name of each factor being the name of the reaction with ‘Factor’ appended.
A reaction of the type a + b -> c + d has name abTOcd. To know the full list of reactions see section :



Collecting all reaction rates

Reaction Number	Reaction Name	Initial species	Final Species	Reference	
0	n -> p	{n}	{p}	Companion Paper	
1	np -> dg	{n, p}	{d, g}	And06	
2	dp -> He3g	{d, p}	{He3, g}	Ili16	
3	dd -> He3n	{d, d}	{He3, n}	Gom17	
4	dd -> tp	{d, d}	{t, p}	Gom17	
5	tp -> ag	{t, p}	{a, g}	Ser04	
6	td -> an	{t, d}	{a, n}	DAACV04	
7	ta -> Li7g	{t, a}	{Li7, g}	DAACV04	

Name of nuclides

Most nuclides have a Name which is their chemical elements,
on which is added the number of nucleons (A). E.g. Li7, Li6, Be7 etc…

Exceptions are:

d for deuterium (instead of H2)

a for He4

t for tritium (instead of H3)

n for neutron

p for proton (instead of H1)

d for photons

Example : d + n -> t + g means H2 + neutron -> H3 + photon

See section Names and (N,Z) for a table of nuclide names

Corrections for weak rates

Beyond the Born approximation, several corrections are included. The section

Corrections for weak rates

lists all Booleans to select or not specific corrections.

Once a set of options are selected, the rates are computed
when running the notebook, and they are stored in the ‘Interpolation’ folder.
To force the code to revaluate these rates if desired, you must either delete the stored files, or set to True the option :

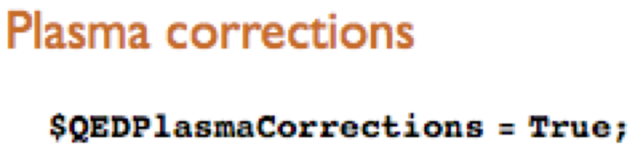
$RecomputeWeakRates

Among the main options which affect weak rates are :

1) $RadiativeCorrections (obviously for radiative corrections)
2) $RadiativeThermal (to also add finite temperature radiative corrections)
3) $FiniteNucleonMass (to account for finite mass of nucleons, e.g. their recoil)

Plasma corrections

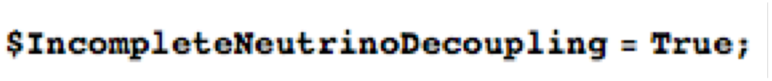
QED corrections need to be included in plasma energy density and pressure. This is controlled in the Plasma corrections section :



Plasma corrections

```
$QEDPlasmaCorrections = True;
```

The effect of neutrino incomplete decoupling (heating function) is controlled via the option (in the same section) :

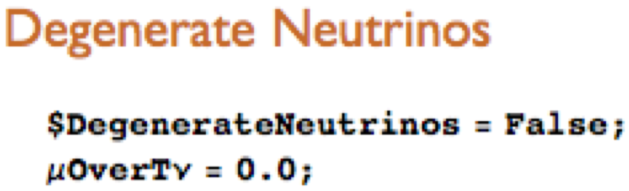


```
$IncompleteNeutrinoDecoupling = True;
```

The QED corrections for plasma energy density and pressure are also stored. To force them to be recomputed, delete the corresponding file in the Interpolation folder or set the option $RecomputePlasmaCorrections to True.

Degenerate neutrinos (neutrino chemical potential)

An option to put a chemical potential (the same for all neutrinos) and setting its value:

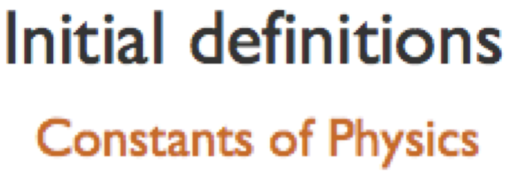


Degenerate Neutrinos

```
$DegenerateNeutrinos = False;
μOverTν = 0.0;
```

Physical constants

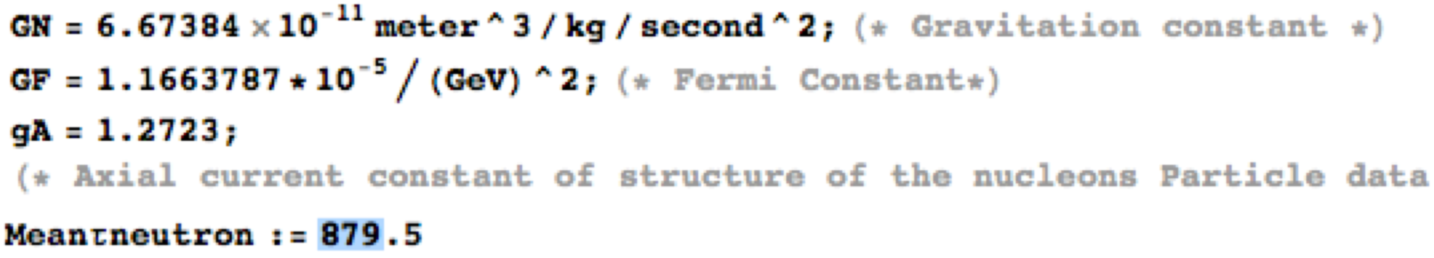
Fundamental constants are all defined in the section



Initial definitions

Constants of Physics

Most relevant to be modified are :

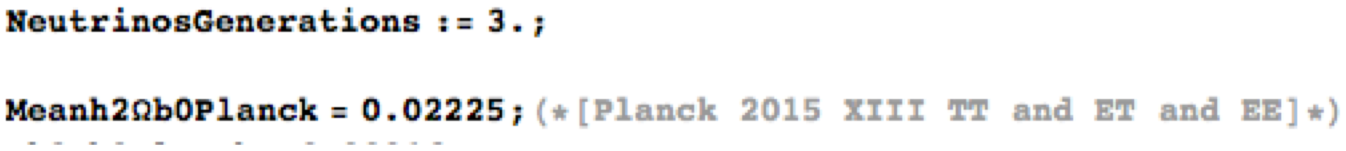


```
GN = 6.67384 × 10^-11 meter ^ 3 / kg / second ^ 2; (* Gravitation constant *)
GF = 1.1663787 * 10^-5 / (GeV) ^ 2; (* Fermi Constant*)
gA = 1.2723;
 (* Axial current constant of structure of the nucleons Particle data
Meanτneutron := 879.5
```

Cosmological constants are in the next section

Cosmological Parameters

Most relevant for BBN are number of neutrinos and baryon density:



```
NeutrinosGenerations := 3.;

Meanh2Ωb0Planck = 0.02225; (* [Planck 2015 XIII TT and ET and EE] *)
```

Structure of code

Once options are chosen and initial definitions are given, the main steps of the code are :

Thermodynamics of the plasma

Section dedicated to QED corrections to the plasma, neutrino decoupling, and the relation a(T) from (approximate) entropy conservation

Weak reactions n+ν ↔ p+e

The weak rates, with all desired corrections are recomputed if not previously stored Or if the option for recomputation is True.

Nuclear reactions network

After reading reaction in the external file, and those given by analytic forms, it builds the huge system of coupled differential equations (all nuclear reactions plus weak rates)

Time integration of Cosmology and BBN

1) Hubble equation to obtain the relation a(t)
2) Initial conditions for nuclides
3) Integration of the nuclear reactions (in 3 different time ranges).

Results and plots

2 Finding and using results

Final results

Abundance of nuclides as a function of time are given in the functions `YI`

$$\boxed{Y_i = n_i/n_{\rm tot}}$$

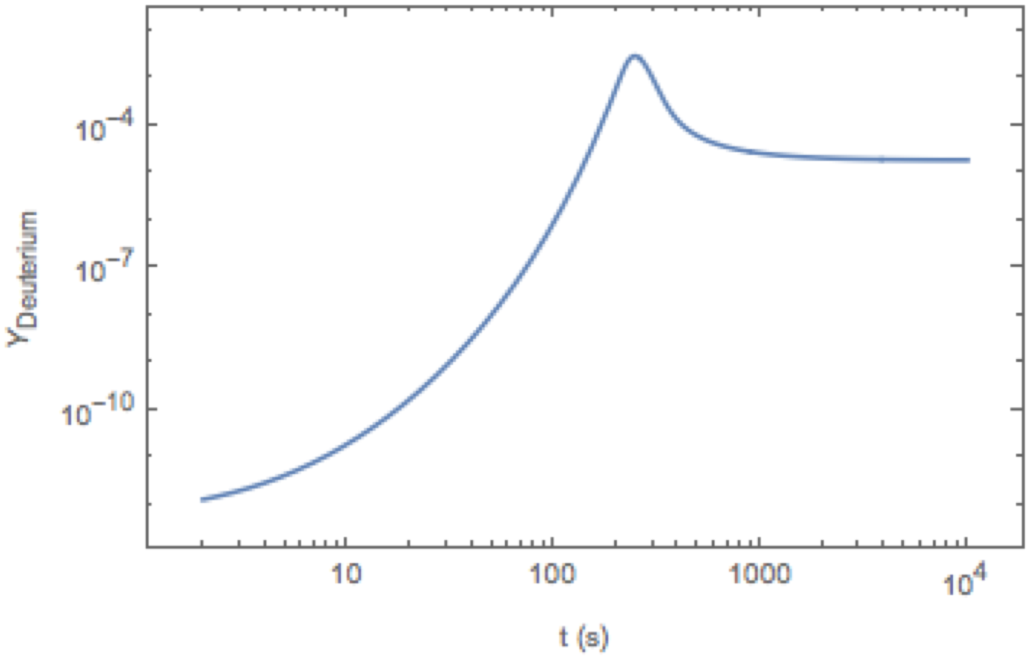
YI[NuclideName][time]

For instance He4 at 1 second is `YI["He4"][1]`

One can use this function to produce plots of abudance as functions of time
For instance evolution of deuterium :

```
In[800]:= LogLogPlot[{YI["d"][tv]}, {tv, 2, 10^4}, Frame → True,
            FrameLabel → {"t  (s)", "YDeuterium"}]
```

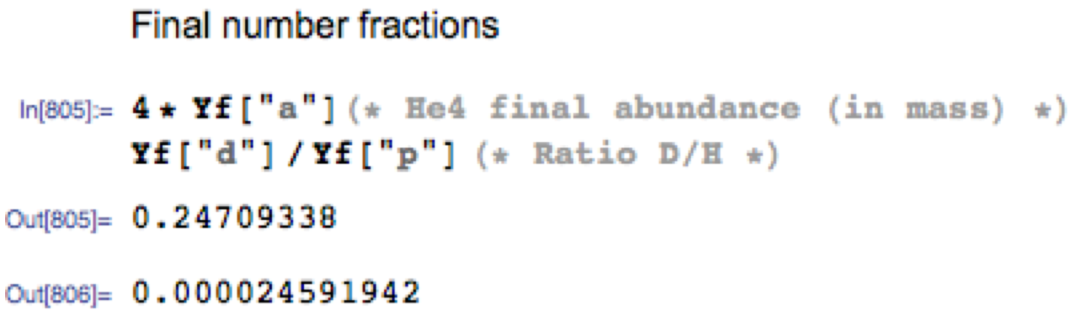
Out[800]=



Final results

The functions XI are related by multiplications by A, that is : $X_i = A_i Y_i$

The functions Yf[name] are the final values of Y for a given nuclide name.



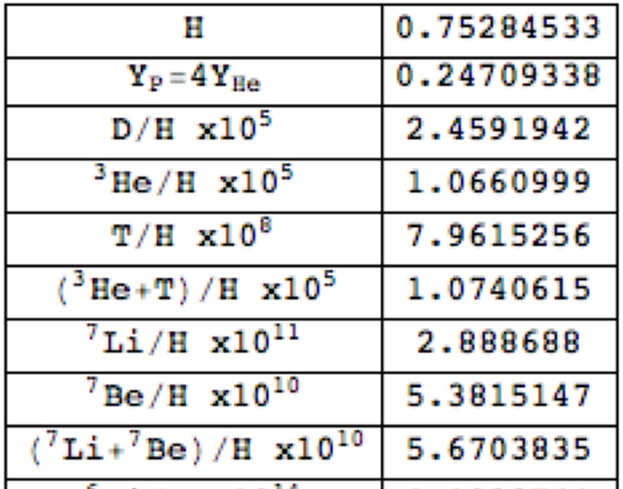
Final number fractions

```
In[805]:= 4 * Yf["a"] (* He4 final abundance (in mass) *)
          Yf["d"] / Yf["p"] (* Ratio D/H *)

Out[805]= 0.24709338

Out[806]= 0.000024591942
```

A table is provided in section

Final abundances

Main results



H	0.75284533
$Y_P = 4Y_{He}$	0.24709338
D/H $\times 10^{5}$	2.4591942
^{3}He/H $\times 10^{5}$	1.0660999
T/H $\times 10^{8}$	7.9615256
(^{3}He+T)/H $\times 10^{5}$	1.0740615
^{7}Li/H $\times 10^{11}$	2.888688
^{7}Be/H $\times 10^{10}$	5.3815147
(^{7}Li+^{7}Be)/H $\times 10^{10}$	5.6703835

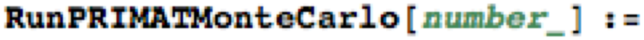
3. Loading PRIMAT in an external notebook (advanced analysis)

Loading PRIMAT in an external notebook

There is a final section in PRIMATMain.nb which is

Tools for Monte-Carlo on nuclear rates

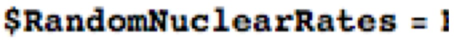
In this section the key function is



```
RunPRIMATMonteCarlo[number_] :=
```

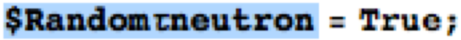
It allows to run ‘number’ times PRIMAT, each time varying randomly the rates according to the uncertainty specified for the reaction (as multiplicative factors).

There are 3 booleans to tell what can be varied.
They need to be set first to the desired value.

1) 

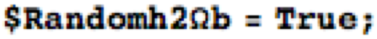
```
$RandomNuclearRates = 1
```

to vary nuclear rates (must be True for sure for Monte Carlo)

2) 

```
$Randomτneutron = True;
```

to vary weak rates (neutron lifetime)

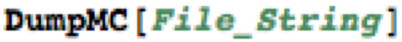
3) 

```
$Randomh2Ωb = True;
```

to vary baryon abundance

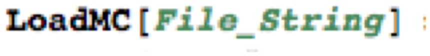
Monte Carlo

After performing a Monte Carlo, results are dumped with



```
DumpMC[File_String]
```

They can be reloaded later with



```
LoadMC[File_String]
```

So that the results can be analyzed independently
(covariance, error on nuclide etc…)

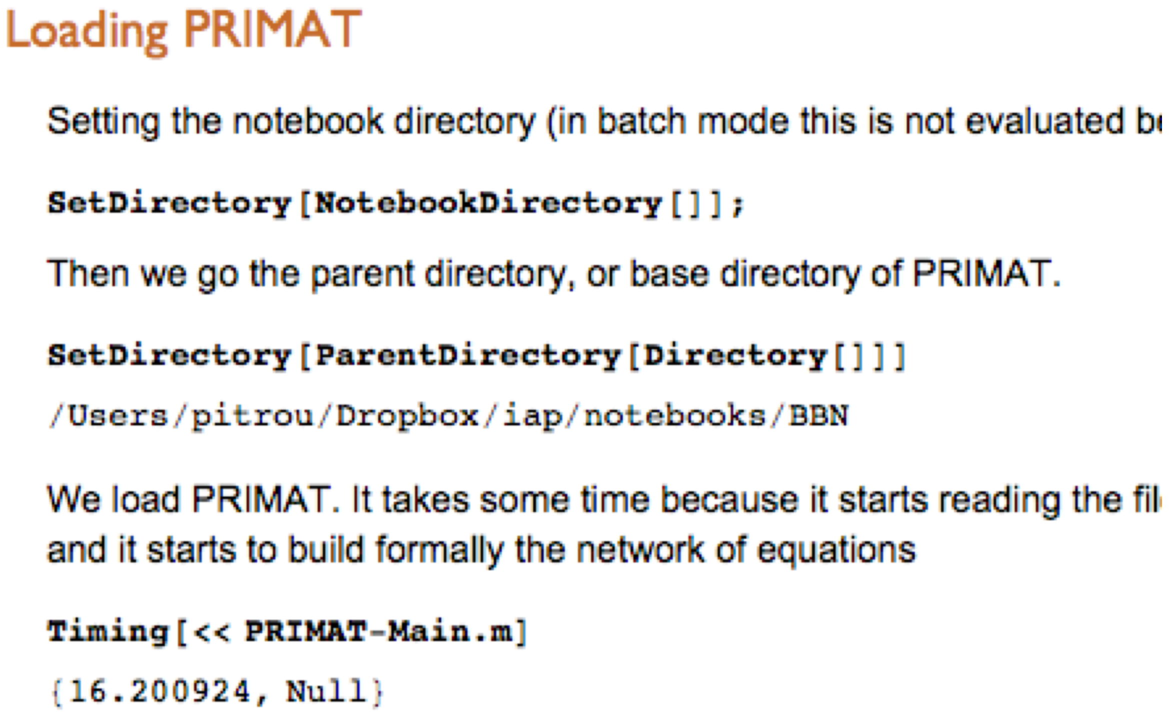
The simplest and self explanatory example is PRIMAT-MonteCarlo-Rates1.nb and is located in the `Example` folder.

Annoying technicalities

When saving PRIMAT-Main.nb, it should generate a PRIMAT-Main.m with all initialization cells (initialization cell is not a good name) in it.

It is this .m file which is loaded by external notebooks. If the .m file is not generated automatically when you save the PRIMAT-Main.nb, then you might force it using File -> Save As (and choose *.m type).

Examples of notebook loading PRIMAT-Main.m are in the Example folder.
They all start by loading the .m file as :



Loading PRIMAT

Setting the notebook directory (in batch mode this is not evaluated b

```
SetDirectory[NotebookDirectory[]];
```

Then we go the parent directory, or base directory of PRIMAT.

```
SetDirectory[ParentDirectory[Directory[]]]
/Users/pitrou/Dropbox/iap/notebooks/BBN
```

We load PRIMAT. It takes some time because it starts reading the fil
and it starts to build formally the network of equations

```
Timing[<< PRIMAT-Main.m]
{16.200924, Null}
```

You can always contact me at pitrou@iap.fr