

Stopping Power

Generated by Doxygen 1.8.3.1

Fri Mar 29 2013 14:10:31

Contents

1	Hierarchical Index	1
1.1	Class Hierarchy	1
2	Class Index	3
2.1	Class List	3
3	Class Documentation	5
3.1	StopPow Class Reference	5
3.1.1	Detailed Description	6
3.1.2	Constructor & Destructor Documentation	6
3.1.2.1	StopPow	6
3.1.2.2	StopPow	6
3.1.3	Member Function Documentation	6
3.1.3.1	dEdx	6
3.1.3.2	Ein	6
3.1.3.3	Eout	7
3.1.3.4	get_dx	7
3.1.3.5	get_mode	7
3.1.3.6	set_dx	7
3.1.3.7	set_mode	8
3.1.3.8	Thickness	8
3.2	StopPow_BetheBloch Class Reference	8
3.2.1	Detailed Description	9
3.2.2	Constructor & Destructor Documentation	9
3.2.2.1	StopPow_BetheBloch	9
3.2.3	Member Function Documentation	9
3.2.3.1	dEdx_MeV_mgcm2	9
3.2.3.2	dEdx_MeV_um	10
3.3	StopPow_LP Class Reference	10
3.3.1	Detailed Description	10
3.3.2	Constructor & Destructor Documentation	11
3.3.2.1	StopPow_LP	11

3.3.3	Member Function Documentation	11
3.3.3.1	dEdx_MeV_mgcm2	11
3.3.3.2	dEdx_MeV_um	11
3.3.3.3	set_collective	12
3.4	StopPow_SRIM Class Reference	12
3.4.1	Detailed Description	12
3.4.2	Constructor & Destructor Documentation	13
3.4.2.1	StopPow_SRIM	13
3.4.2.2	~StopPow_SRIM	13
3.4.3	Member Function Documentation	13
3.4.3.1	dEdx_MeV_mgcm2	13
3.4.3.2	dEdx_MeV_um	13

Index	14
--------------	-----------

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

StopPow	5
StopPow_BetheBloch	8
StopPow_LP	10
StopPow_SRIM	12

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

StopPow	Generic class for stopping power calculators	5
StopPow_BetheBloch	Calculate Bethe-Bloch stopping power	8
StopPow_LP	Calculate Li-Petrasso stopping power	10
StopPow_SRIM	Cold-matter tabulated stopping	12

Chapter 3

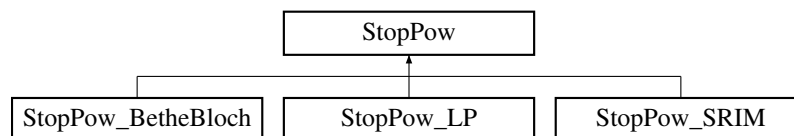
Class Documentation

3.1 StopPow Class Reference

Generic class for stopping power calculators.

```
#include <StopPow.h>
```

Inheritance diagram for StopPow:



Public Member Functions

- [StopPow](#) ()
- [StopPow](#) (int [set_mode](#))
- float [dEdx](#) (float E)
- virtual float [dEdx_MeV_um](#) (float E)=0
- virtual float [dEdx_MeV_mgcm2](#) (float E)=0
- float [Eout](#) (float E, float x)
- float [Ein](#) (float E, float x)
- float [Thickness](#) (float E1, float E2)
- float [get_dx](#) ()
- void [set_dx](#) (float new_dx)
- int [get_mode](#) ()
- void [set_mode](#) (int new_mode)

Static Public Attributes

- static const float **DEFAULT_DX**
- static const float **DEFAULT_DRHOR**
- static const int **MODE_LENGTH**
- static const int **MODE_RHOR**

Protected Attributes

- float **dx**
- int **mode**

3.1.1 Detailed Description

Generic class for stopping power calculators.

In addition to setting the abstract template for stopping power calculators, this also includes several generic methods. The stopping power utilities here can be called as functions of linear distance or areal density. To specify which, the mode must be set correctly.

Author

Alex Zylstra

Date

2013/03/29

3.1.2 Constructor & Destructor Documentation

3.1.2.1 StopPow::StopPow ()

Simple constructor for the generic class

3.1.2.2 StopPow::StopPow (int *set_mode*)

Construct a new [StopPow](#) object given a starting mode

Parameters

<i>set_mode</i>	the mode you want to use (defined using class constants)
-----------------	--

3.1.3 Member Function Documentation

3.1.3.1 float StopPow::dEdx (float *E*)

Calculate stopping power. Return units depend on mode.

Parameters

<i>E</i>	the particle energy in MeV
----------	----------------------------

Returns

dE/dx in MeV/um [MeV/(mg/cm2)]

Exceptions

<i>invalid_argument</i>	
-------------------------	--

3.1.3.2 float StopPow::Ein (float *E*, float *x*)

Get incident energy for a particle.

Parameters

E	the particle energy in MeV
x	thickness of material in um [mg/cm2]

Returns

initial particle energy in MeV

Exceptions

<i>invalid_argument</i>	
-------------------------	--

3.1.3.3 float StopPow::Eout (float E , float x)

Get energy downshift for a particle.

Parameters

E	the particle energy in MeV
x	thickness of material in um [mg/cm2]

Returns

final particle energy in MeV

Exceptions

<i>invalid_argument</i>	
-------------------------	--

3.1.3.4 float StopPow::get_dx ()

Get the current step sized being used for calculations.

Returns

dx the step size in um [mg/cm2]

3.1.3.5 int StopPow::get_mode ()

Get the current mode being used for calculations.

Returns

mode Either StopPow.MODE_LENGTH or StopPow.MODE_RHOR

3.1.3.6 void StopPow::set_dx (float *new_dx*)

Set the step size for calculations

Parameters

<i>new_dx</i>	the new step size to use, in um [mg/cm2]
---------------	--

Exceptions

<i>invalid_argument</i>	
-------------------------	--

3.1.3.7 void StopPow::set_mode (int new_mode)

Set the mode for calculations

Parameters

<i>new_mode</i>	Either StopPow.MODE_LENGTH or StopPow.MODE_RHOR
-----------------	---

Exceptions

<i>invalid_argument</i>	
-------------------------	--

3.1.3.8 float StopPow::Thickness (float E1, float E2)

Get thickness of material traversed.

Parameters

<i>E1</i>	the initial particle energy in MeV
<i>E2</i>	the final particle energy in MeV

Returns

material thickness in um [mg/cm2]

Exceptions

<i>invalid_argument</i>	
-------------------------	--

The documentation for this class was generated from the following file:

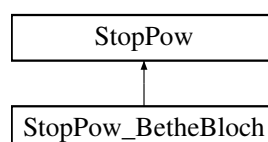
- /Users/alex/Dropbox/Research/workspace/StopPow/src/StopPow.h

3.2 StopPow_BetheBloch Class Reference

Calculate Bethe-Bloch stopping power.

```
#include <StopPow_BetheBloch.h>
```

Inheritance diagram for StopPow_BetheBloch:



Public Member Functions

- [StopPow_BetheBloch](#) (float mt, float Zt, vector< float > mf, vector< float > Zf, vector< float > nf)

- float [dEdx_MeV_um](#) (float E)
- float [dEdx_MeV_mgcm2](#) (float E)

Additional Inherited Members

3.2.1 Detailed Description

Calculate Bethe-Bloch stopping power.

Implement a stopping-power calculator for arbitrary cold matter, using the simple Bethe-Bloch theory.

Author

Alex Zylstra

Date

2013/03/29

3.2.2 Constructor & Destructor Documentation

3.2.2.1 `StopPow_BetheBloch::StopPow_BetheBloch (float mt, float Zt, vector< float > mf, vector< float > Zf, vector< float > nf)`

Initialize the Bethe-Bloch calculator.

Parameters

<i>mt</i>	the test particle mass in AMU
<i>Zt</i>	the test particle in charge (units of e)
<i>mf</i>	vector containing ordered field particle masses in AMU
<i>Zf</i>	vector containing ordered field particle charges in units of e
<i>nf</i>	vector containing ordered field particle densities in units of 1/cm3

Exceptions

<i>invalid_argument</i>	
-------------------------	--

3.2.3 Member Function Documentation

3.2.3.1 `float StopPow_BetheBloch::dEdx_MeV_mgcm2 (float E) [virtual]`

Calculate the total stopping power

Parameters

<i>E</i>	the test particle energy in MeV
----------	---------------------------------

Returns

stopping power in units of MeV/(mg/cm2)

Exceptions

<i>invalid_argument</i>	
-------------------------	--

Implements [StopPow](#).

3.2.3.2 `float StopPow_BetheBloch::dEdx_MeV_um (float E) [virtual]`

Calculate the total stopping power

Parameters

<i>E</i>	the test particle energy in MeV
----------	---------------------------------

Returns

stopping power in units of MeV/um

Exceptions

<i>invalid_argument</i>	
-------------------------	--

Implements [StopPow](#).

The documentation for this class was generated from the following file:

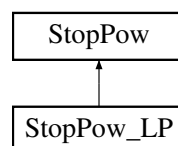
- /Users/alex/Dropbox/Research/workspace/StopPow/src/StopPow_BetheBloch.h

3.3 StopPow_LP Class Reference

Calculate Li-Petrasso stopping power.

```
#include <StopPow_LP.h>
```

Inheritance diagram for StopPow_LP:



Public Member Functions

- [StopPow_LP](#) (float mt, float Zt, vector< float > mf, vector< float > Zf, vector< float > Tf, vector< float > nf)
- float [dEdx_MeV_um](#) (float E)
- float [dEdx_MeV_mgcm2](#) (float E)
- void [set_collective](#) (bool set)

Additional Inherited Members

3.3.1 Detailed Description

Calculate Li-Petrasso stopping power.

Implement a stopping-power calculator for plasma, using the Fokker-Planck theory described in Li and Petrasso, PRL 1993.

Author

Alex Zylstra

Date

2013/03/29

3.3.2 Constructor & Destructor Documentation

3.3.2.1 StopPow_LP::StopPow_LP (float *mt*, float *Zt*, vector< float > *mf*, vector< float > *Zf*, vector< float > *Tf*, vector< float > *nf*)

Initialize the Li-Petrasso stopping power.

Parameters

<i>mt</i>	the test particle mass in AMU
<i>Zt</i>	the test particle in charge (units of e)
<i>mf</i>	vector containing ordered field particle masses in AMU
<i>Zf</i>	vector containing ordered field particle charges in units of e
<i>Tf</i>	vector containing ordered field particle temperatures in units of keV
<i>nf</i>	vector containing ordered field particle densities in units of 1/cm3

Exceptions

<i>invalid_argument</i>	
-------------------------	--

3.3.3 Member Function Documentation

3.3.3.1 float StopPow_LP::dEdx_MeV_mgcm2 (float *E*) [virtual]

Calculate the total stopping power

Parameters

<i>E</i>	the test particle energy in MeV
----------	---------------------------------

Returns

stopping power in units of MeV/(mg/cm2)

Exceptions

<i>invalid_argument</i>	
-------------------------	--

Implements [StopPow](#).

3.3.3.2 float StopPow_LP::dEdx_MeV_um (float *E*) [virtual]

Calculate the total stopping power

Parameters

<i>E</i>	the test particle energy in MeV
----------	---------------------------------

Returns

stopping power in units of MeV/um

Exceptions

<i>invalid_argument</i>	
-------------------------	--

Implements [StopPow](#).

3.3.3.3 void StopPow_LP::set_collective (bool set)

Turn collective effects on or off.

Parameters

<i>set</i>	if you want to use collective effects
------------	---------------------------------------

The documentation for this class was generated from the following file:

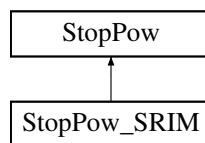
- /Users/alex/Dropbox/Research/workspace/StopPow/src/StopPow_LP.h

3.4 StopPow_SRIM Class Reference

Cold-matter tabulated stopping.

```
#include <StopPow_SRIM.h>
```

Inheritance diagram for StopPow_SRIM:

**Public Member Functions**

- [StopPow_SRIM](#) (string fname)
- [~StopPow_SRIM](#) ()
- float [dEdx_MeV_um](#) (float E)
- float [dEdx_MeV_mgcm2](#) (float E)

Additional Inherited Members**3.4.1 Detailed Description**

Cold-matter tabulated stopping.

A wrapper class for calculating stopping powers using tabulated SRIM data (stored in csv files) Linear interpolation is performed between data points.

Author

Alex Zylstra

Date

2013/03/29

3.4.2 Constructor & Destructor Documentation

3.4.2.1 StopPow_SRIM::StopPow_SRIM (string *fname*)

Constructor for SRIM object. Data file must be in standard SRIM format.

Parameters

<i>fname</i>	file name (or relative path) for the data
--------------	---

Exceptions

<i>ios_base::failure</i>	
--------------------------	--

3.4.2.2 StopPow_SRIM::~StopPow_SRIM ()

Destructor

3.4.3 Member Function Documentation

3.4.3.1 float StopPow_SRIM::dEdx_MeV_mgcm2 (float *E*) [virtual]

Get stopping power from the data.

Parameters

<i>E</i>	the particle energy in MeV
----------	----------------------------

Returns

dE/dx in MeV/(mg/cm2)

Exceptions

<i>invalid_argument</i>	
-------------------------	--

Implements [StopPow](#).

3.4.3.2 float StopPow_SRIM::dEdx_MeV_um (float *E*) [virtual]

Get stopping power from the data.

Parameters

<i>E</i>	the particle energy in MeV
----------	----------------------------

Returns

dE/dx in MeV/um

Exceptions

<i>invalid_argument</i>	
-------------------------	--

Implements [StopPow](#).

The documentation for this class was generated from the following file:

- /Users/alex/Dropbox/Research/workspace/StopPow/src/StopPow_SRIM.h

Index

~StopPow_SRIM
StopPow_SRIM, [13](#)

dEdx
StopPow, [6](#)
dEdx_MeV_mgcm2
StopPow_BetheBloch, [9](#)
StopPow_LP, [11](#)
StopPow_SRIM, [13](#)
dEdx_MeV_um
StopPow_BetheBloch, [10](#)
StopPow_LP, [11](#)
StopPow_SRIM, [13](#)

Ein
StopPow, [6](#)

Eout
StopPow, [7](#)

get_dx
StopPow, [7](#)
get_mode
StopPow, [7](#)

set_collective
StopPow_LP, [12](#)

set_dx
StopPow, [7](#)

set_mode
StopPow, [8](#)

StopPow, [5](#)
dEdx, [6](#)
Ein, [6](#)
Eout, [7](#)
get_dx, [7](#)
get_mode, [7](#)
set_dx, [7](#)
set_mode, [8](#)
StopPow, [6](#)
StopPow, [6](#)
Thickness, [8](#)

StopPow_BetheBloch, [8](#)
dEdx_MeV_mgcm2, [9](#)
dEdx_MeV_um, [10](#)
StopPow_BetheBloch, [9](#)
StopPow_BetheBloch, [9](#)

StopPow_LP, [10](#)
dEdx_MeV_mgcm2, [11](#)
dEdx_MeV_um, [11](#)
set_collective, [12](#)

StopPow_LP, [11](#)
StopPow_LP, [11](#)
StopPow_SRIM, [12](#)
~StopPow_SRIM, [13](#)
dEdx_MeV_mgcm2, [13](#)
dEdx_MeV_um, [13](#)
StopPow_SRIM, [13](#)
StopPow_SRIM, [13](#)

Thickness
StopPow, [8](#)