Package 'libamtrack'

October 18, 2010

Version 0.2

Date 2010-10-18

Title libamtrack package

Author Felix Klein <f.klein@dkfz.de>.

Maintainer Steffen Greilich <s.greilich@dkfz.de>

Depends R (>= 2.2.0)

Description This package is the R interface to the open-source, ANSI C library libamtrack (http://libamtrack.dkfz.org). libamtrack provides computational routines for the prediction of detector response and radiobiological efficiency in heavy charged particle beams. It is designed for research in proton and ion dosimetry and radiotherapy. libamtrack also provides many auxiliary physics routines for proton and ion beams.

License GPL (version 3 or later)

R topics documented:

`AT.Bethe.Mass.Stopping.Power.MeV.cm2.g` *AT.Bethe.Mass.Stopping.Power.MeV.cm2.g*

Description

Computes the mass stopping power using the Bethe formula for many particles according to ICRU49, p.6, Eq. 2.1 BUT WITHOUT shell or density, Bloch or Barkas correction!

Usage

```
AT.Bethe.Mass.Stopping.Power.MeV.cm2.g(n, E.MeV.u, particle.no, material.no, E.r
```

Arguments

`n`	number of particles
`E.MeV.u`	energies of particle per nucleon (array of size n)
`particle.no`	particle indices (array of size n)
`material.no`	material index (single value)
`E.restricted.keV`	if positive and smaller than maximally transferable energy, the restricted stopping power will be computed (single value)

Value

`Mass.Stopping.Power.MeV.cm2.g`
(array of size n)

`AT.D.RDD.Gy` *AT.D.RDD.Gy*

Description

Returns RDD as a function of distance r_m

Usage

```
AT.D.RDD.Gy(n, r.m, E.MeV.u, particle.no, material.no,
rdd.model, rdd.parameter, er.model)
```

Arguments

`n`	number of particles (length of r_m vector)
`r.m`	distance [m] (array of size n)
`E.MeV.u`	particle (ion) energy per nucleon [MeV/u]
`particle.no`	particle code number
`material.no`	material code number
`rdd.model`	Radial Dose Distribution model code number
`rdd.parameter`	Radial Dose Distribution model parameters vector (array of size 3)
`er.model`	delta electron range model code number

Value

D.RDD.Gy	dose [Gy] (array of size n)
status	status

AT.dose.weighted.E.MeV.u

AT.dose.weighted.E.MeV.u

Description

Computes the dose-weighted average energy of a particle field

Needed by SuccessiveConvolutions

Usage

```
AT.dose.weighted.E.MeV.u(n, E.MeV.u, particle.no, fluence.cm2, material.no)
```

Arguments

n	length of vectors for parameters
E.MeV.u	energy of particles in the mixed particle field (array of size n)
particle.no	particle index (array of size n)
fluence.cm2	fluences of particles in the mixed particle field (array of size n)
material.no	material index

Value

result	result

AT.fluence.weighted.E.MeV.u

AT.fluence.weighted.E.MeV.u

Description

Computes the fluence-weighted average energy of a particle field

Needed by SuccessiveConvolutions

Usage

```
AT.fluence.weighted.E.MeV.u(n, E.MeV.u, fluence.cm2)
```

Arguments

n	length of vectors for parameters
E.MeV.u	energy of particles in the mixed particle field (array of size n)
fluence.cm2	fluences of particles in the mixed particle field (array of size n)

Value

`average.E.MeV.u`
: average.E.MeV.u

`AT.fluence.weighted.LET.MeV.cm2.g` *AT.fluence.weighted.LET.MeV.cm2.g*

Description

Computes the fluence-weighted average LET of a particle field

Needed by SuccessiveConvolutions

Usage

```
AT.fluence.weighted.LET.MeV.cm2.g(n, E.MeV.u, particle.no, fluence.cm2, material
```

Arguments

`n`	length of vectors for parameters
`E.MeV.u`	energy of particles in the mixed particle field (array of size n)
`particle.no`	particle index (array of size n)
`fluence.cm2`	fluences of particles in the mixed particle field (array of size n)
`material.no`	material index

Value

`result` result

`AT.gamma.response` *AT.gamma.response*

Description

Returns the detector / cell gamma response for a vector of given doses according to the chosen gamma response model

Usage

```
AT.gamma.response(number.of.doses, d.Gy, gamma.model, gamma.parameter, lethal.ev
```

Arguments

number.of.doses	number of doses given in vector d_Gy
d.Gy	doses in Gy (array of size number_of_doses)
gamma.model	gamma response model index
gamma.parameter	vector holding necessary parameters for the chose gamma response model (array of size 4)
lethal.event.mode	TODO

Value

S	gamma responses (array of size number_of_doses)

AT.max.electron.ranges.m
AT.max.electron.ranges.m

Description

Returns the maximum electron range (track diameter) in m for vector of energies

Usage

```
AT.max.electron.ranges.m(number.of.particles, E.MeV.u, material.no, er.model)
```

Arguments

number.of.particles	number of particles in the incident field
E.MeV.u	kinetic energy for particles in the given field (array of size number_of_particles)
material.no	index for detector material
er.model	index for electron-range model chosen

Value

max.electron.range.m	electron range (track diameter) in m (array of size number_of_particles)

AT.RDD.f1.parameters.mixed.field *AT.RDD.f1.parameters.mixed.field*

Description

Pre-calculated many useful parameters characterizing RDD.

Usage

```
AT.RDD.f1.parameters.mixed.field(n, E.MeV.u, particle.no, material.no, rdd.model
rdd.parameter, er.model)
```

Arguments

n	number of particle types in the mixed particle field
E.MeV.u	energy of particles in the mixed particle field (array of size n)
particle.no	type of the particles in the mixed particle field (array of size n)
material.no	material code number
rdd.model	Radial Dose Distribution model code number
rdd.parameter	Radial Dose Distribution model parameters vector (array of size 3)
er.model	delta electron range model code number

Value

f1.parameters
(array of size 8)

AT.run.GSM.method *AT.run.GSM.method*

Description

Computes HCP response and RE/RBE using summation of tracks an a Cartesian grid (the GSM algorithm)

Usage

```
AT.run.GSM.method(n, E.MeV.u, particle.no, fluence.cm2.or.dose.Gy, material.no,
RDD.model, RDD.parameters, ER.model, gamma.model, gamma.parameters,
N.runs, write.output, nX, voxel.size.m, lethal.events.mode)
```

Arguments

n	number of particle types in the mixed particle field
E.MeV.u	energy of particles in the mixed particle field (array of size n)
particle.no	type of the particles in the mixed particle field (array of size n)
fluence.cm2.or.dose.Gy	fluences for the given particles, doses in Gy if negative (array of size n)
material.no	index number for detector material
RDD.model	index number for chosen radial dose distribution
RDD.parameters	parameters for chosen radial dose distribution (array of size 4)
ER.model	index number for chosen electron-range model
gamma.model	index number for chosen gamma response
gamma.parameters	parameters for chosen gamma response (array of size 4)
N.runs	(algorithm specific) number of runs within which track positions will be resampled
write.output	if true, a protocol is written to SuccessiveConvolutions.txt in the working directory
nX	(algorithm specific) number of voxels of the grid in x (and y as the grid is quadratic)
voxel.size.m	side length of a voxel in m
lethal.events.mode	(algorithm specific) if true, allows to do calculations for cell survival

Value

results	to be allocated by the user which will be used to return the results (array of size 10)

AT.run.IGK.method *AT.run.IGK.method*

Description

Computes HCP response and RE/RBE using Katz' Ion-Gamma-Kill approach according to Waligorski, 1988

Usage

```
AT.run.IGK.method(n, E.MeV.u, particle.no, fluence.cm2.or.dose.Gy, material.no,
RDD.model, RDD.parameters, ER.model, gamma.model, gamma.parameters,
saturation.cross.section.factor, write.output)
```

Arguments

n	number of particle types in the mixed particle field
E.MeV.u	energy of particles in the mixed particle field (array of size n)
particle.no	type of the particles in the mixed particle field (array of size n)
fluence.cm2.or.dose.Gy	fluences for the given particles, doses in Gy if negative (array of size n)
material.no	index number for detector material
RDD.model	index number for chosen radial dose distribution
RDD.parameters	parameters for chosen radial dose distribution (array of size 4)
ER.model	index number for chosen electron-range model
gamma.model	index number for chosen gamma response
gamma.parameters	parameters for chosen gamma response (array of size 4)
saturation.cross.section.factor	(algorithm specific) scaling factor for the saturation cross section
write.output	if true, a protocol is written to a file in the working directory

Value

results	to be allocated by the user which will be used to return the results (array of size 10)

AT.run.SPIFF.method

AT.run.SPIFF.method

Description

Computes HCP response and RE/RBE using compound Poison process and successive convolutions (CPP_SC, the SPIFF algorithm)

Usage

```
AT.run.SPIFF.method(n, E.MeV.u, particle.no, fluence.cm2.or.dose.Gy, material.no
rdd.model, rdd.parameters, er.model, gamma.model, gamma.parameters,
N2, fluence.factor, write.output, shrink.tails, shrink.tails.under,
adjust.N2, lethal.events.mode)
```

Arguments

n	number of particle types in the mixed particle field
E.MeV.u	energy of particles in the mixed particle field (array of size n)
particle.no	type of the particles in the mixed particle field (array of size n)
fluence.cm2.or.dose.Gy	fluences for the given particles, doses in Gy if negative (array of size n)
material.no	index number for detector material

rdd.model	index number for chosen radial dose distribution
rdd.parameters	parameters for chosen radial dose distribution (array of size 4)
er.model	index number for chosen electron-range model
gamma.model	index number for chosen gamma response
gamma.parameters	parameters for chosen gamma response (array of size 4)
N2	(algorithm specific) number of bins per factor of two in local dose array
fluence.factor	factor to scale the fluences given as fluence_cm2 with
write.output	if true, a protocol is written to SuccessiveConvolutions.txt in the working directory
shrink.tails	(algorithm specific) if true, tails of the local dose distribution, contributing less than shrink_tails_under are cut
shrink.tails.under	(algorithm specific) limit for tail cutting in local dose distribution
adjust.N2	(algorithm specific) if true, N2 will be increase if necessary at high fluence to ensure sufficient binning resolution
lethal.events.mode	(algorithm specific) if true, allows to do calculations for cell survival

Value

N2	(algorithm specific) number of bins per factor of two in local dose array
results	to be allocated by the user which will be used to return the results (array of size 10) @n

---

`AT.total.D.Gy` *AT.total.D.Gy*

Description

Computes the total dose of a particle field

Needed by SuccessiveConvolutions

Usage

```
AT.total.D.Gy(n, E.MeV.u, particle.no, fluence.cm2, material.no)
```

Arguments

n	length of vectors for parameters
E.MeV.u	energy of particles in the mixed particle field (array of size n)
particle.no	particle index (array of size n)
fluence.cm2	fluences of particles in the mixed particle field (array of size n)
material.no	material index

Value

total.dose.Gy	total.dose.Gy

`AT.total.fluence.cm2` *AT.total.fluence.cm2*

Description

Computes the total fluence of a particle field

Needed by SuccessiveConvolutions

Usage

```
AT.total.fluence.cm2(n, E.MeV.u, particle.no, D.Gy, material.no)
```

Arguments

`n`	length of vectors for parameters
`E.MeV.u`	energy of particles in the mixed particle field (array of size n)
`particle.no`	particle index (array of size n)
`D.Gy`	doses of particles in the mixed particle field (array of size n)
`material.no`	material index

Value

`total.fluence.cm` total.fluence.cm

Index