# Package ‘crone’

October 12, 2022

**Type** Package

**Title** Structural Crystallography in 1d

**Version** 0.1.1

**Date** 2019-08-25

**Description** Functions to carry out the most important crystallographic calculations for crystal structures made of 1d Gaussian-shaped atoms, especially useful for methods development. Main reference: E. Smith, G. Evans, J. Foadi (2017) <doi:10.1088/1361-6404/aa8188>.

**License** GPL-2

**LazyLoad** yes

**LazyData** true

**RoxygenNote** 6.1.1

**Suggests** knitr

**VignetteBuilder** knitr

**Encoding** UTF-8

**Depends** R (>= 3.5.0)

**NeedsCompilation** no

**Author** James Foadi [cre, aut]

**Maintainer** James Foadi <j.foadi@bath.ac.uk>

**Repository** CRAN

**Date/Publication** 2019-08-24 14:40:04 UTC

## R topics documented:

## anomalous_data *Theoretical scattering factors for all atomic species*

### Description

This dataset is a list where each element is an atomic species. Each element of the list is a dataframe with 3 columns.

### Usage

```
anomalous_data
```

### Format

A list whose elements are dataframes corresponding to atomic elements. Each dataframe has the following columns:

**lambda** Specific value of wavelength in angstroms.

**f1** Real scattering component.

**f2** Imaginary scattering component.

## atoms *Atom names and atomic number*

### Description

This is a dataframe including a 2-characters string indicating the atomic element name and an integer indicating the atomic number Z.

### Usage

```
atoms
```

### Format

The dataframe has the following columns:

**anames** 2-character string indicating the atomic name.

**Z** Integer. The atomic number.

## atom_gauss *Gaussian atom*

### Description

Gaussian atom

### Usage

```
atom_gauss(x, a, x0 = 0, Z = 1, B = 0, k = ksigma)
```

### Arguments

| | |
|---|---|
| x | Point in the 1D cell at which this function is calculated. |
| a | A real number. The width of the unit cell in which the gaussian atom is placed. |
| x0 | A real number. The point corresponding to the atom's peak. |
| Z | An integer number. Z is the atomic number of the atom (Z(H)=1, Z(He)=2,Z(Li)=3,Z(B)=4, etc). |
| B | A real number. This is the B factor characterizing the atom's thermal agitation. It is given as B=8*pi^2*U, where U is the variance of the position of the atoms' nucleus around the equilibrium position. |
| k | A real number. It controls the standard deviation of the gaussian function describing the atom and, thus, the shape of the associated peak. The standard deviation sigma is given by: `sigma = k * sqrt(Z)` |

### Value

A vector of length equal to the length of vector x, with values equal to the evaluated gaussian atom.

### Examples

```
# Carbon gaussian atom in the middle of a cell
a <- 10
x0 <- 5
Z <- 6
x <- seq(0,a,length=1000)
rho <- atom_gauss(x,a,x0,Z)
plot(x,rho,type="l",xlab="x",ylab=expression(rho))
```

## choose_a *Suggests unit cell side, a, based on atom content*

### Description

The unit cell side is roughly calculated by adding two times the half-width of the widest gaussian atom to the largest inter-atomic distance. The half-width of the largest gaussian is computed as Ma times the gaussian sigma. If the "P-1" symmetry is present, D is doubled.

### Usage

```
choose_a(Z, D, SG = "P1", k = ksigma, Ma = 5)
```

### Arguments

| | |
|---|---|
| Z | A vector of atom Z numbers. |
| D | A real number. The distance between the two furthest atoms in the cell. |
| SG | 2-letters character string. Symmetry. There are only two symmetries possible when working within 1D crystallography, P1 (no symmetry)and P-1 (inversion through the origin). SG can be either "P1" or "P-1" for this function. |
| k | A real number. It controls the standard deviation of the gaussian function describing the atom and, thus, the shape of the associated peak. The standard deviation sigma is given by: `sigma = k * sqrt(Z)` |
| Ma | A real number. Each gaussian atom has tails truncated at a distance of Ma * sigma from its peak. |

### Value

A real number that suggests a feasible unit cell side containing all atoms.

## Examples

```
# 2 carbon atoms, a sulphur and an oxygen
Z <- c(6,8,16,6)

# Distance between the two carbons is 15 angstroms
D <- 15
a <- choose_a(Z,D)
```

---

## diffraction *Simulation of 1D diffraction pattern*

---

### Description

Analytic Fourier transform of electron density corresponding to an array of `Ncell` unit cells calculated using numerical integration with the trapezoid rule. The diffraction peaks' height is proportional to the number of unit cells ( `Ncell`). The number of diffraction peaks included in the 1D diffraction pattern is related to the maximum resolution `D` provided in the input. The number of grid points for both the simulated electron density and the resulting diffraction pattern can also be provided as input. A further input parameter is the radius of the beamstop disc to stop diffraction close to the incoming beam (as the resulting intensity far outweigh the rest of the diffracted intensities).

### Usage

```
diffraction(sdata, D, Ncell = 10, N = 1000, n = 100, bstop = NULL)
```

### Arguments

`sdata` A named list, normally obtained through the use of functions `read_x` or `standardise_sdata`. The list names correspond to different object types:

- a. Real numeric. The size of the unit cell.
- SG. Character string. Space group symbol; either "P1" or "P-1"
- x0. Vector of real numerics indicating the expanded atomic positions in the unit cell.
- Z. Vector of integers indicating the expanded atomic numbers for all atoms in the unit cell.
- B. Vector of real numerics indicating the expanded B factors for all atoms in the unit cell.
- occ. Vector of real numerics indicating the expanded occupancies for all atoms in the unit cell.

`D` Real numeric. Maximum resolution in angstroms.

`Ncell` Positive integer. It is the number of unit cells in the 1D crystal. The default value is `Ncell=10`.

`N` Positive integer indicating the number of grid points for the electron density. The default value is `N=1000`.

`n` Positive integer determining the reciprocal space grid. The grid is made of `2*n+1` regularly-spaced points from `-1/D` to `1/D`. The value 0 is always at the centre of the grid. The default value is `n=100`.

`bstop` Real numeric. Is the radius of the backstop disc. Intensities at points closer to the origin than `bstop` are reduced to 0. The presence of a backstop improves the contrast for all diffracted intensities because it gets rid of the overwhelming intensity corresponding to the origin of the reciprocal space. The default is not to include any backstop.

### Value

A named list with two vectors of real numbers, the values of the reciprocal space grid points (in 1/angstrom units) `xstar` and the intensities `Imod`.

### Examples

```
# Diffraction from just two unit cells of cyanate
sdata <- load_structure("cyanate")

# Max resolution is 1 angstroms; no backstop
ltmp <- diffraction(sdata,D=1,Ncell=1)

# Plot diffraction pattern
plot(ltmp$xstar,ltmp$Imod,type="l",
 xlab=expression(paste("x"^"*")),ylab="Intensity")

# Diffraction from 20 unit cells with backstop of 20 angstroms diameter
ltmp <- diffraction(sdata,D=1,bstop=10)
plot(ltmp$xstar,ltmp$Imod,type="l",
 xlab=expression(paste("x"^"*")),ylab="Intensity")
```

---

## `erf` *Error function for real values*

---

### Description

Error function for real values

### Usage

```
erf(x)
```

### Arguments

`x` A vector of real numbers.

### Value

A real number. The integral of the gaussian, centred on zero and with standard deviation equal to 1, between 0 and x, multiplied by 2/sqrt(pi).

### Examples

```
x <- seq(-1,1,length=1000)
y <- erf(x)
plot(x,y,type=”l”)
```

## expand_to_cell *Expand content of asymmetric unit to whole unit cell*

### Description

Atom positions, types, B factors and occupancies are duplicated if input space group (SG) is P-1; otherwise they are left untouched (space group P1). Value of the occupancy for special positions is barely checked for values outside [0,1] range. Extra-care needed.

### Usage

```
expand_to_cell(sdata, SG = NULL)
```

### Arguments

`sdata` A named list, normally obtained through the use of function `read_x`. The list names correspond to different object types:

- a. Real numeric. The size of the unit cell.
- SG. Character string. Space group symbol; either "P1" or "P-1"
- x0. Vector of real numerics indicating the expanded atomic positions in the unit cell.
- Z. Vector of integers indicating the expanded atomic numbers for all atoms in the unit cell.
- B. Vector of real numerics indicating the expanded B factors for all atoms in the unit cell.
- occ. Vector of real numerics indicating the expanded occupancies for all atoms in the unit cell.

`SG` 2-letters character string. There are only two symmetries possible when working within 1D crystallography, P1 (no symmetry) and P-1 (inversion through the origin). SG can be either "P1" or "P-1" for this function. Default is NULL, in which case the space group is assumed to be equal to the one of the input structure.

## Value

A named list with the following elements:

- a. Real numeric. The size of the unit cell.
- SG. Character string. Name of the space group, either "P1" or "P-1".
- x0. Vector of real numerics indicating the expanded atomic positions in the unit cell.
- Z. Vector of integers indicating the expanded atomic numbers for all atoms in the unit cell.
- B. Vector of real numerics indicating the expanded B factors for all atoms in the unit cell.
- occ. Vector of real numerics indicating the expanded occupancies for all atoms in the unit cell.

## Examples

```
# Asymmetric unit includes 3 atoms between 0 and a/2
a <- 10
SG <- "P-1"
x0 <- c(1,2,4)
Z <- c(6,8,6)
B <- c(1,1.2,1.1)
occ <- c(1,1,0.8)
sdata <- standardise_sdata(a,SG,x0,Z,B,occ)
ltmp <- expand_to_cell(sdata)
print(ltmp)  # Positions, atomic numbers, etc. have doubled

# Nothing changes if imposed SG is "P1" (but you get a warning!)
ltmp <- expand_to_cell(sdata,SG="P1")
print(ltmp)
```

---

# `fluorescent_scan` *Find optimal wavelength for anomalous phasing*

---

## Description

This function mimics the behaviour of the fluorescent scan operated at synchrotron beamlines to find out the optimal wavelength which maximises the anomalous signal used to phase a structure.

## Usage

```
fluorescent_scan(chem_el, lambda_range = NULL)
```

## Arguments

| | |
|---|---|
| `chem_el` | 1- or 2-letters character string. The chemical symbol of interest. |
| `lambda_range` | Real vector of length 2. The two values are the extremes of the wavelength window inside which to carry out the search. Default is for the search to be carried out across the full available range. |

## Value

A named list with two elements: 1) idx is the integer indicating the row in the anomalous_data dataframe related to the specific chemical element, corresponding to the optimal wavelength; 2) the optimal wavelength in angstroms.

## Examples

```
# Optimal wavelength for iron
lFe <- fluorescent_scan("Fe")
print(lFe$lambda)
idx <- lFe$idx

# Load anomalous dataframe for Fe
adFe <- load_anomalous_data("Fe")
print(adFe[idx,])  # Same wavelength as before!

# Optimal wavelength with window restriction
lFe <- fluorescent_scan("Fe",lambda_range=c(6,8))
print(lFe$lambda)
```

# fousynth *From structure factors to density using Fourier synthesis*

## Description

Given a set of structure factors, separately as a vector of amplitudes and a vector of phases in degrees, corresponding to a set of 1D Miller indices, the length of the 1D unit cell, the set of Miller indices and the number of grid points used to calculate the 1D density, this function calculates the 1D density corresponding to the given structure factors.

## Usage

```
fousynth(a, Fmod, Fpha, hidx, N)
```

## Arguments

`a` A real number. The unit cell side length.

`Fmod` A vector of real numbers. The structure factors' amplitudes corresponding to the 1D density to be calculated.

`Fpha` A vector of real numbers. The structure factors' phases (in degrees) corresponding to the 1D density to be calculated.

`hidx` A vector of integer numbers. The set of 1D Miller indices corresponding to the set of structure factors F.

`N` An integer number. The number of grid points used to calculate the 1D density.

### Value

A vector of N real numbers representing the calculated 1D density at each of the regular N grid points.

### Examples

```
# First create the crystal structure (in P1)
a <- 10
SG <- "P1"
x0 <- c(1,4,6.5)
Z <- c(8,26,6)
B <- c(18,20,17)
occ <- c(1,1,1)
sdata <- standardise_sdata(a,SG,x0,Z,B,occ)

# Enough Fourier components (Miller indices)
hidx <- 0:20

# Compute the structure factors
ftmp <- strufac(hidx,sdata)

# Number of grid points
N <- 1000

# Density
rtmp <- fousynth(a,ftmp$Fmod,ftmp$Fpha,hidx,N)

# Density plot in the unit cell
x <- rtmp$x
rho <- rtmp$rr
plot(x,rho,type="l",xlab="x",ylab=expression(rho))
```

## `heaviside` *Heaviside function (step function)*

### Description

Heaviside function (step function)

### Usage

```
heaviside(x, x0 = 0)
```

### Arguments

| | |
|---|---|
| `x` | A vector of real numbers. |
| `x0` | A real number. The x value at which the function step occurs. |

### Value

One of the two numbers 0 or 1.

### Examples

```
x <- seq(-3,5,length=1000)
x0 <- 1
y <- heaviside(x,x0)
plot(x,y,type=”l”)
# Step up and step down
x1 <- seq(-3,5,length=1000)
x10 <- 1
y1 <- heaviside(x1,x10)
x2 <- seq(1,9,length=1000)
x20 <- 5
y2 <- heaviside(x2,x20)
y2 <- 1-y2
plot(x1,y1,type=”l”,xlim=c(-3,9),xlab=”x”,ylab=”y”)
points(x2,y2,type=”l”)
```

## invfousynth *From density to structure factors using inverse Fourier synthesis*

### Description

Given a density as vector calculated in N grid points, the unit cell size and an array of Miller indices hidx, this function calculates amplitudes and phases of the structure factors corresponding to this density, via inverse Fourier transform.

### Usage

```
invfousynth(a, rho, hidx)
```

### Arguments

`a` A real number. The unit cell side length.

`rho` A vector of N real numbers representing the 1D density at each of the regular N grid point.

`hidx` A vector of integer numbers. The set of 1D Miller indices corresponding to the set of structure factors F, to be calculated.

### Value

A vector of N real numbers representing the calculated 1D density at each one of the regular N grid points.

## Examples

```
# First create the crystal structure (in P1)
a <- 10
SG <- "P1"
x0 <- c(1,4,6.5)
Z <- c(8,26,6)
B <- c(18,20,17)
occ <- c(1,1,1)
sdata <- standardise_sdata(a,SG,x0,Z,B,occ)

# 10 Miller indices plus DC component
hidx <- 0:10

# Compute structure factors
ftmp1 <- strufac(hidx,sdata)

# Number of grid points
N <- 1000

# Density
rtmp <- fousynth(a,ftmp1$Fmod,ftmp1$Fpha,hidx,N)

# Using inverse Fourier to obtain structure factors
ftmp2 <- invfousynth(a,rtmp$rr,hidx)

# Comparison
print(abs(ftmp1$Fmod-ftmp2$Fmod))
print(abs(ftmp1$Fpha-ftmp2$Fpha))
```

---

# kgauss *Constant normalizing wrapped gaussian*

---

## Description

Constant normalizing wrapped gaussian

## Usage

```
kgauss(sigma, a)
```

## Arguments

| | |
|---|---|
| sigma | A real number. Is the standard deviation of the wrapped gaussian. |
| a | A real number. The unit cell side length. |

## Value

A real number, the multiplicative constant normalizing the wrapped gaussian atom so that the area under the curve is equal to 1.

### Examples

```
Z <- 16  # Sulphur atom
sigma <- 0.05*sqrt(Z)
a <- 15  # Unit cell size
kk <- kgauss(sigma,a)
print(kk)
```

## `load_anomalous_data` *Load anomalous data for a specific chemical element*

### Description

Returns a dataframe with f' and f" at various wavelengths for the specific chemical element.

### Usage

```
load_anomalous_data(chem_el)
```

### Arguments

| | |
|---|---|
| `chem_el` | 1- or 2-letters character string. The chemical symbol of interest. |

### Value

A dataframe with 3 columns, the specific wavelength in angstroms (lambda), f' (f1) and f" (f2).

### Examples

```
# Load anomalous data for Fe
ano_Fe <- load_anomalous_data("Fe")
print(ano_Fe[1:10,])
```

## `load_data` *Load observed structure factors from 1D structure data in workspace.*

### Description

Function to load structure factors corresponding to one of the many 1D structures available within the *crone* package. The structure factors amplitudes have been generated from calculated data with some simulated error, so that they mimic observed data. Phases are calculated from the correct structure.

### Usage

```
load_data(sname = NULL)
```

### Arguments

`sname` A character string. Name of the structure whose data are to be loaded in the workspace. It can be one of:

- beryllium_fluoride
- carbon_dioxide
- cyanate
- nitronium
- thiocyanate
- xenon_difluoride
- pinkerton2015

Default is NULL, in which case the function returns a list of all structures available.

### Value

A named fdata-type list with the following elements:

- a Real numeric. Unit cell length in angstroms. Always included.
- SG. Spacegroup 2-letters character string. There are only two symmetries possible when working within 1D crystallography, P1 (no symmetry) and P-1 (inversion through the origin). SG can be either "P1" or "P-1". Always included.
- hidx. Real numeric array. 1D unique (positive in the 1D context) Miller indices. Always included.
- Fobs. Real numeric array. Amplitudes of observed structure factors. Not always included.
- sigFobs. Real numeric array. Errors associated with Fobs. Not always included.
- Phicalc. Real numeric array. Phases (in degrees) of structure factors calculated from the correct 1D structure. They are normally used to check correctness of Phiobs. Not always included.

### Examples

```
# Load thiocyanate data
fdata <- load_data("thiocyanate")
print(fdata)

# Default returns all names of structures included
load_data()
```

## `load_structure` *Load 1D structure data in workspace.*

### Description

Function to load data from one of the many 1D structures available within the *crone* package.

### Usage

```
load_structure(sname = NULL)
```

### Arguments

`sname` A character string. Name of the structure whose data are to be loaded in the workspace. It can be one of:

- beryllium_fluoride
- carbon_dioxide
- cyanate
- nitronium
- thiocyanate
- xenon_difluoride
- pinkerton2015

Default is NULL, in which case the function returns a list of all structures available.

### Value

A named list with the following elements:

- a Real numeric. Unit cell length in angstroms.
- SG 2-letters character string. There are only two symmetries possible when working within 1D crystallography, P1 (no symmetry) and P-1 (inversion through the origin). SG can be either "P1" or "P-1".
- x0 Vector of real numerics indicating the expanded atomic positions in the unit cell.
- Z Vector of integers indicating the expanded atomic numbers for all atoms in the unit cell.
- B Vector of real numerics indicating the expanded B factors for all atoms in the unit cell.
- occ Vector of real numerics indicating the expanded occupancies for all atoms in the unit cell.

### Examples

```
# Load thiocyanate data
sdata <- load_structure("thiocyanate")
print(sdata)

# Default returns all names of structures included
load_structure()
```

## local_maxima *Find local maxima in a vector of real values.*

### Description

Find local maxima in a vector of real values.

### Usage

```
local_maxima(x)
```

### Arguments

| | |
|---|---|
| x | A vector of real numbers |

### Value

A vector of integers corresponding to local maxima positions in vector x

### Examples

```
t <- seq(-10,10,length=1000)
x <- dnorm(t,mean=3)+dnorm(t,mean=7)
yM <- local_maxima(x)
```

## plot_absorption_curves *Plot of absorption curves*

### Description

Plot f’ and f” absorption curves for the specified chemical element. Curves can be plotted in specified wavelength regions using parameter "zoom".

### Usage

```
plot_absorption_curves(chem_el, zoom = NULL)
```

### Arguments

| | |
|---|---|
| chem_el | 1- or 2-letters character string. The chemical symbol of interest. |
| zoom | Real vector of length 2. The two values are the extremes of the wavelength window inside which to plot the two curves. Default is for both curves to be plotted across the full available range. |

### Value

Nothing, but causes a 2D plot to be displayed in a graphical window.

### Examples

```
# No zoom
plot_absorption_curves("Fe")

# Zoom
plot_absorption_curves("Fe",c(1,3))
```

## read_h *Read data from a reflections file*

### Description

Read data from a *_h.dat-type file containing cell size, spacegroup symbol and amplitudes and/or phases of observed and/or calculated structure factors. This function loads the file data into a standardised named list for structure factors data.

### Usage

```
read_h(filename)
```

### Arguments

| | |
|---|---|
| `filename` | A character string. Existing file that includes structure factors information. The file name in general has the form "[prefix]_h.dat". |

### Value

A named list with the following elements:

- a Real numeric. Unit cell length in angstroms. Always included.
- SG. Spacegroup 2-letters character string. There are only two symmetries possible when working within 1D crystallography, P1 (no symmetry) and P-1 (inversion through the origin). SG can be either "P1" or "P-1". Always included.
- hidx. Real numeric array. 1D unique (positive in the 1D context) Miller indices. Always included.
- Fobs. Real numeric array. Amplitudes of observed structure factors. Not always included.
- sigFobs. Real numeric array. Errors associated with Fobs. Not always included.
- Fp. Real numeric vector. Amplitudes of the positive component of Friedel (or Bijvoet) pairs (F+). Not always included.
- sigFp. Real numeric vector. Errors associated with Fp. Not always included.
- Fm. Real numeric vector. Amplitudes of the negative component of Friedel (or Bijvoet) pairs (F-). Not always included.

- sigFm. Real numeric vector. Errors associated with Fm. Not always included.
- Phiobs. Real numeric array. Phases (in degrees) of structure factors obtained with one of the methods used for structure solution. Not always included.
- Phicalc. Real numeric array. Phases (in degrees) of structure factors calculated from the correct 1D structure. They are normally used to check correctness of Phiobs. Not always included.

### Examples

```
# Observed structure factors amplitudes and calculated phases
# from thiocyanate structure
datadir <- system.file("extdata",package="crone")
filename <- file.path(datadir,"thiocyanate_h.dat")
fdata <- read_x(filename)
print(names(fdata))
print(fdata$Fobs)
print(fdata$sigFobs)
```

## read_x *Read unit cell content (atom and coordinates).*

### Description

Read unit cell length, space group, atom coordinates and all other parameters from a coordinate file.

### Usage

```
read_x(filename)
```

### Arguments

| | |
|---|---|
| filename | A character string. Existing file that includes all structural information. The file name in general has the form "[prefix]_x.dat". |

### Value

A named list with the following elements:

- a. Real numeric. Unit cell length in angstroms.
- SG. SG 2-letters character string. There are only two symmetries possible when working within 1D crystallography, P1 (no symmetry) and P-1 (inversion through the origin). SG can be either "P1" or "P-1".
- x0. Vector of real numerics indicating the expanded atomic positions in the unit cell.
- Z. Vector of integers indicating the expanded atomic numbers for all atoms in the unit cell.
- B. Vector of real numerics indicating the expanded B factors for all atoms in the unit cell.
- occ. Vector of real numerics indicating the expanded occupancies for all atoms in the unit cell.

## Examples

```
datadir <- system.file("extdata",package="crone")
filename <- file.path(datadir,"carbon_dioxide_x.dat")
sdata <- read_x(filename)
print(names(sdata))
print(sdata)
```

---

## reduce_to_asu *Reduce content of unit cell to asymmetric unit.*

---

### Description

Atom types, B factors and occupancies are restricted to those corresponding to positions in the asymmetric unit, if the input space group (SG) is P-1. Otherwise they are only sorted according to increasing atom positions. If the starting positions do not correspond to a properly symmetry-expanded set, then a warning is issued. If no symmetry is used (P1) the starting set is forced to have values inside the unit cell and it is sorted according to increasing atom positions.

### Usage

```
reduce_to_asu(adata, SG = NULL)
```

### Arguments

adata A named list, normally obtained through the use of function read_x. The list names correspond to different object types:

- a. Real numeric. The size of the unit cell.
- SG. Character string. Space group symbol; either "P1" or "P-1"
- x0. Vector of real numerics indicating the expanded atomic positions in the unit cell.
- Z. Vector of integers indicating the expanded atomic numbers for all atoms in the unit cell.
- B. Vector of real numerics indicating the expanded B factors for all atoms in the unit cell.
- occ. Vector of real numerics indicating the expanded occupancies for all atoms in the unit cell.

SG 2-letters character string. There are only two symmetries possible when working within 1D crystallography, P1 (no symmetry) and P-1 (inversion through the origin). SG can be either "P1" or "P-1" for this function. Default is NULL, in which case the space group is assumed to be equal to the one of the input structure.

## Value

A named list with the following elements:

- x0. Vector of real numerics indicating the reduced atomic positions in the asymmetric unit.
- Z. Vector of integers indicating the reduced atomic numbers for all atoms in the asymmetric unit.
- B. Vector of real numerics indicating the reduced B factors for all atoms in the asymmetric unit.
- occ. Vector of real numerics indicating the reduced occupancies for all atoms in the asymmetric unit.

## Examples

```
# Asymmetric unit includes 4 atoms between 0 and a/2, 1 in special position
a <- 10
SG <- "P-1"
x0 <- c(1,2,4,5)
Z <- c(6,8,6,16)
B <- c(1,1.2,1.1,0.8)
occ <- c(1,1,1,0.5)
adata <- standardise_sdata(a,SG,x0,Z,B,occ)
ltmp <- expand_to_cell(adata)
print(ltmp)  # Positions, atomic numbers, etc. have doubled

# Now these expanded values are used as input for reduction
ltmp2 <- reduce_to_asu(ltmp)

# Compare
print(x0)
print(ltmp2$x0)

# SG is "P1"
a <- 16
SG <- "P1"
x0 <- c(1,2,7,9,12,16)
Z <- c(6,6,8,8,7,6)
B <- c(0,0,0,0,0,0)
occ <- c(1,1,1,1,1,1)
adata <- standardise_sdata(a,SG,x0,Z,B,occ)
ltmp3 <- reduce_to_asu(adata)
print(ltmp3) # Now all positions are inside the unit cell
```

# scafac *Scattering factor for 1D gaussian atoms*

## Description

Given unit cell length, atomic number, B factor, occupancy and Miller index h, this function returns the corresponding value of the analytical scattering factor calculated as Fourier transform of the 1D gaussian atom.

### Usage

```
scafac(h, a, Zj, occj, Bj = NULL, k = ksigma)
```

### Arguments

`h` Real numeric. One or more 1D Miller indices. `h` can also have non-integer values in between integer values. This enables plotting of scattering curves.

`a` Real numeric. Length of 1D unit cell.

`Zj` Integer numeric. Atomic number (e.g. Oxygen has Zj <- 8)

`occj` Real numeric. Atomic occupancy, a real number between 0 and 1, where 0 means that the atom is missing in the crystal and 1 means that is present in all unit cells of the crystal.

`Bj` Real numeric. This is the B factor associated with the thermal vibration of the atom. It is measured in squared angstroms and it is equal to 8*pi^2*sigma^2, where sigma is the gaussian atom width.

`k` A real number. It controls the standard deviation of the gaussian function describing the atom and, thus, the shape of the associated peak. The standard deviation sigma is given by: `sigma = k * sqrt(Z)`

### Value

A real numeric. The value of the scattering factor at the sprcified Miller idex or corresponding real value.

### Examples

```
# Values for some Miller indices
h <- 0:10
a <- 20
Zj <- 16
Bj <- 18  # Roughly corresponding to sigma 0.23
occj <- 1
fval <- scafac(h,a,Zj,occj,Bj)
plot(h,fval,pch=16,xlab="Miller index",ylab="Scattering factor",
     ylim=c(0,16))

# Continuous resolution
h <- seq(0,10,length=1000)
fval <- scafac(h,a,Zj,occj,Bj)
points(h,fval,type="l",col=2)

# Scattering curve for a lighter atom
Zj <- 8
fval <- scafac(h,a,Zj,occj,Bj)
points(h,fval,type="l",col=3)

# Scattering curve for the same atom, just with smaller Bj (colder)
Bj <- 10
fval <- scafac(h,a,Zj,occj,Bj)
points(h,fval,type="l",col=4)
```

## sfobs *Generation of structure factors with errors*

### Description

This function generates structure factors calculated starting from the given structure and subject to two types of errors: poissonian counting errors due to the statistical nature of the photons hitting the crystal and normal errors due to the slight random shifting of atoms position in all the unit cells forming the lattice.

### Usage

```
sfobs(hidx, sdata, vx0err = NULL, ntrialP = 100, ntrialG = 100,
  anoflag = FALSE, aK = anoK, lbda = 1, k = ksigma,
  f1f2out = TRUE)
```

### Arguments

`hidx` Real numeric. One or more 1D Miller indices.

`sdata` A named list, normally obtained through the use of function `read_x` or `standardise_sdata`. The list names correspond to different object types:

- a. Real numeric. The size of the unit cell.
- SG. Character string. Space group symbol; either "P1" or "P-1"
- x0. Vector of real numerics indicating the expanded atomic positions in the unit cell.
- Z. Vector of integers indicating the expanded atomic numbers for all atoms in the unit cell.
- B. Vector of real numerics indicating the expanded B factors for all atoms in the unit cell.
- occ. Vector of real numerics indicating the expanded occupancies for all atoms in the unit cell.

`vx0err` A real number. The standard deviation of the random displacement of all atoms composing the structure from their correct position. Default value is NULL, corresponding to the generation of structure factors, with no errors, from the correct structure.

`ntrialP` Integer. The number of simulated Poisson counts for each set of structure factor amplitudes. More counts (high ntrialP) return smaller errors for the structure factor amplitudes. If ntrialP less or equal 0, then poissonian counting errors are not generated.

`ntrialG` Integer. This is the number of randomly generated shifts of each atom of the structure from its true position. The shifts follow a gaussian distribution with mean 0 and standard deviation vx0err.

`anoflag` Logical variable. If TRUE it forces scattering factors to include anomalous contributions. As a consequence, theoretical Friedel's pairs will not be equal.

`aK` Real numeric. This is a fudge factor included to decrease the strength of the anomalous contributions. Without aK the strength is too high for 1D structures, compared to real 3D structures. So aK helps bringing down the anomalous contribution within the 5 met with large-size structures. The default value is aK=0.3 (anoK is included as internal data).

`lbda` Real numeric. This is the wavelength in angstroms. Its value is important in relation to anomalous scattering.

`k` A real number. It controls the standard deviation of the gaussian function describing the atom and, thus, the shape of the associated peak. The standard deviation sigma is given by: `sigma = k * sqrt(Z)` The default value is k=0.05 (ksigma is included as internal data).

`f1f2out` Logical variable. This variable controls output of a small table of f' and f" values for all chemical elements in the structure. Default is for the table to be printed.

## Value

A named list with two elements:

- F Array of mean structure factor amplitudes, among all structure factor arrays simulated with specific errors.
- sF Array of structure factors errors. These coincide with the standard deviations of all structure factors arrays simulated with specific errors.

## Examples

```
# Load thiocyanate data
sdata <- load_structure("thiocyanate")

# Miller indices used
hidx <- 1:10

# Correct amplitudes and phases
ftmp <- strufac(hidx,sdata)
Ftrue <- ftmp$Fmod
phitrue <- ftmp$Fpha

# Only poissonian errors
ltmp <- sfobs(hidx,sdata,ntrialP=2)
print(names(ltmp))
Fpois <- ltmp$F

# True density
rtmptrue <- fousynth(sdata$a,Fmod=Ftrue,Fpha=phitrue,hidx,1000)
plot(rtmptrue$x,rtmptrue$rr,type="l",xlab="x",ylab=expression(rho),
 lwd=3)

# Density with poissonian errors
rtmppois <- fousynth(sdata$a,Fmod=Fpois,Fpha=phitrue,hidx,1000)
```

```
points(rtmppois$x,rtmppois$rr,type="l",
 lty=2,col=2,lwd=2) # Very small differences

# Only random atomic errors with standard deviation 0.3 angstroms
ltmp <- sfobs(hidx,sdata,ntrialP=0,vx0err=0.3)
Fcoords <- ltmp$F

# Density with gaussian errors on atom coordinates
rtmpcoords <- fousynth(sdata$a,Fmod=Fcoords,Fpha=phitrue,hidx,1000)
points(rtmpcoords$x,rtmpcoords$rr,type="l",
 lty=3,col=3,lwd=2) # Larger differences
```

## `standardise_fdata` *Standardise reflections data*

### Description

This function output a list with fields needed by most of the functions dealing with structure factors. It is the equivalent of the function *standardise_sdata*, used to prepare atomic structures data.

### Usage

```
standardise_fdata(a, SG, hidx, Fobs = NULL, sigFobs = NULL,
  Fp = NULL, sigFp = NULL, Fm = NULL, sigFm = NULL,
  Phiobs = NULL, Phicalc = NULL)
```

### Arguments

`a` Real numeric. Unit cell length in angstroms.

`SG` SG 2-letters character string. There are only two symmetries possible when working within 1D crystallography, P1 (no symmetry) and P-1 (inversion through the origin). SG can be either "P1" or "P-1".

`hidx` Real numeric array. 1D unique (positive in the 1D context) Miller indices.

`Fobs` Real numeric array. Amplitudes of structure factors. If Fp and Fm are not NULL and Fobs is NULL, then Fobs are calculated as averages of Fp and Fm. If both Fp, Fm and Fobs are included, input Fobs are used, instead of Fp and Fm averages.

`sigFobs` Real numeric array. Errors associated with Fobs. If sigFobs = NULL, errors are estimated from Fp and Fm. Default is NULL.

`Fp` Real numeric vector. Amplitudes of the positive component of Friedel (or Bijvoet) pairs (F+). Default is NULL, i.e. no Fp included.

`sigFp` Real numeric vector. Errors associated with Fp. Default is NULL, i.e. no sigFp included.

`Fm` Real numeric vector. Amplitudes of the negative component of Friedel (or Bijvoet) pairs (F-). Default is NULL, i.e. no Fm included.

`sigFm` Real numeric vector. Errors associated with Fm. Default is NULL, i.e. no sigFm included.

`Phiobs` Real numeric array. Phases (in degrees) of structure factors obtained with one of the methods used for structure solution. Default is NULL.

`Phicalc` Real numeric array. Phases (in degrees) of structure factors calculated from the correct 1D structure. They are normally used to check correctness of Phiobs. Default is NULL.

## Value

A named list with a variable number of elements. Some of them are always included; others are not:

- a Real numeric. Unit cell length in angstroms. Always included.
- SG. Spacegroup 2-letters character string. There are only two symmetries possible when working within 1D crystallography, P1 (no symmetry) and P-1 (inversion through the origin). SG can be either "P1" or "P-1". Always included.
- hidx. Real numeric array. 1D unique (positive in the 1D context) Miller indices. Always included.
- Fobs. Real numeric array. Amplitudes of observed structure factors. Not always included.
- sigFobs. Real numeric array. Errors associated with Fobs. Not always included.
- Fp. Real numeric vector. Amplitudes of the positive component of Friedel (or Bijvoet) pairs (F+). Not always included.
- sigFp. Real numeric vector. Errors associated with Fp. Not always included.
- Fm. Real numeric vector. Amplitudes of the negative component of Friedel (or Bijvoet) pairs (F-). Not always included.
- sigFm. Real numeric vector. Errors associated with Fm. Not always included.
- Phiobs. Real numeric array. Phases (in degrees) of structure factors obtained with one of the methods used for structure solution. Not always included.
- Phicalc. Real numeric array. Phases (in degrees) of structure factors calculated from the correct 1D structure. They are normally used to check correctness of Phiobs. Not always included.

## Examples

```
# Create an arbitrary structure with a heavy atom (Fe)
a <- 20
SG <- "P1"
x0 <- c(1,2,6,16)
Z <- c(6,8,26,7)
B <- c(8,7,5,8)
occ <- c(1,1,1,1)
sdata <- standardise_sdata(a,SG,x0,Z,B,occ)

# Miller indices, from -5 to 5 (to include negatives for anomalous)
hidx <- -5:5
```

```
# Experimental structure factors with anomalous contribution
# (lambda = 1.74) for creating Fm and Fp. Errors only due to
# photons fluctuations.
set.seed(9195)   # For demo purposes.
ltmp <- sfobs(hidx,sdata,ntrialP=10,anoflag=TRUE,lbda=1.74)

# Fp and sigFp (Miller indices from 1 to 5)
isel <- 1:5
idx <- match(isel,hidx)
Fp <- ltmp$F[idx]
sigFp <- ltmp$sF[idx]

# Fm and sigFm
isel <- (-1):(-5)
idx <- match(isel,hidx)
Fm <- ltmp$F[idx]
sigFm <- ltmp$sF[idx]

# Now only positive Miller indices
hidx <- 1:5

# Create standardised data for reciprocal space
fdata <- standardise_fdata(a,SG,hidx,Fp=Fp,sigFp=sigFp,
         Fm=Fm,sigFm=sigFm)

# Fp and Fm
print(fdata$Fp)
print(fdata$sigFp)
print(fdata$Fm)
print(fdata$sigFm)

# Fobs and sigFobs automatically created
print(fdata$Fobs)
print(fdata$sigFobs)

# Structure factors without anomalous data for the same structure
hidx <- 1:5
set.seed(9195)   # For demo purposes.
ltmp <- sfobs(hidx,sdata,ntrialP=10)
Fobs <- ltmp$F
sigFobs <- ltmp$sF
fdata <- standardise_fdata(a,SG,hidx,Fobs=Fobs,sigFobs=sigFobs)
print(fdata)
```

## `standardise_sdata` *Organise atom data in a standard format for later use*

## Description

Function to put any group consisting of cell size a, space group SG, atom coordinates x0, atomic numbers Z, atomic B factors B and atomic occupancies occ, into a named list with 6 fields. This is then used as standard input/output format throughout the *crone* package.

## Usage

```
standardise_sdata(a, SG, x0, Z, B, occ)
```

## Arguments

`a` Real numeric. Unit cell length in angstroms.

`SG` SG 2-letters character string. There are only two symmetries possible when working within 1D crystallography, P1 (no symmetry) and P-1 (inversion through the origin). SG can be either "P1" or "P-1".

`x0` Vector of real numerics indicating the expanded atomic positions in the unit cell.

`Z` Vector of integers indicating the expanded atomic numbers for all atoms in the unit cell.

`B` Vector of real numerics indicating the expanded B factors for all atoms in the unit cell.

`occ` Vector of real numerics indicating the expanded occupancies for all atoms in the unit cell.

## Value

A named list with the following elements:

- a. Real numeric. Unit cell length in angstroms.
- SG. SG 2-letters character string. There are only two symmetries possible when working within 1D crystallography, P1 (no symmetry) and P-1 (inversion through the origin). SG can be either "P1" or "P-1".
- x0. Vector of real numerics indicating the expanded atomic positions in the unit cell.
- Z. Vector of integers indicating the expanded atomic numbers for all atoms in the unit cell.
- B. Vector of real numerics indicating the expanded B factors for all atoms in the unit cell.
- occ. Vector of real numerics indicating the expanded occupancies for all atoms in the unit cell.

## Examples

```
# Create standard format for an arbitrary structure in P1
a <- 20
SG <- "P1"
x0 <- c(2,11,16,19)
Z <- c(6,6,16,8)
B <- c(13,14,5,10)
occ <- c(1,1,1,1)
sdata <- standardise_sdata(a,SG,x0,Z,B,occ)
print(sdata)
```

## structure_gauss *Structure of gaussian atoms*

### Description

Structure formed by all gaussian atoms in the unit cell. Positions, atomic numbers and thermal factors are given by vectors of a same length. Each atom forming the structure is also characterised by a given occupancy (between 0 and 1).

### Usage

```
structure_gauss(sdata, x = NULL, N = NULL, k = ksigma)
```

### Arguments

`sdata` A named list, normally obtained through the use of function `read_x` or `standardise_sdata`. The list names correspond to different object types:

- a. Real numeric. The size of the unit cell.
- SG. Character string. Space group symbol; either "P1" or "P-1"
- x0. Vector of real numerics indicating the expanded atomic positions in the unit cell.
- Z. Vector of integers indicating the expanded atomic numbers for all atoms in the unit cell.
- B. Vector of real numerics indicating the expanded B factors for all atoms in the unit cell.
- occ. Vector of real numerics indicating the expanded occupancies for all atoms in the unit cell.

`x` Point in the 1D cell at which this function is calculated. Default is NULL, in which case a grid is set up internally.

`N` Integer. Number of points in the regular grid, if the grid is not provided directly.

`k` A real number. It controls the standard deviation of the gaussian function describing the atom and, thus, the shape of the associated peak. The standard deviation sigma is given by: `sigma = k * sqrt(Z)`

### Value

A named list of length 2: x is the grid (either input by user or set up internally), rr is a vector of length equal to the length of vector x, with values equal to the evaluated gaussian atoms.

### Examples

```
# Cell, atom types, positions and B factors
a <- 10
SG <- "P1"
x0 <- c(2,5,7)
Z <- c(6,16,8)
```

```
B <- c(0,0,0)

# All occupancies to 1
occ <- c(1,1,1)

# Standard data format
sdata <- standardise_sdata(a,SG,x0,Z,B,occ)

# Grid for unit cell
x <- seq(0,a,length=1000)

# Structure density
rtmp <- structure_gauss(sdata,x)
plot(rtmp$x,rtmp$rr,type="l",xlab="x",ylab=expression(rho))

# Now reduce occupancy of sulphur
occ[2] <- 0.5
sdata <- standardise_sdata(a,SG,x0,Z,B,occ)
rtmp <- structure_gauss(sdata,x)
points(rtmp$x,rtmp$rr,type="l",col=2)

# Increase temperature of oxygen
B[3] <- 10
sdata <- standardise_sdata(a,SG,x0,Z,B,occ)
rtmp <- structure_gauss(sdata,x)
points(rtmp$x,rtmp$rr,type="l",col=3)
```

## strufac *Calculation of structure factors*

### Description

Calculates structure factors corresponding to one or more 1D Miller indices, given the atomic content of one unit cell. Anomalous scattering can be included using logical flag "anoflag" (default is FALSE). Crystal structures are always considered with no symmetry. Thus a 1D structure with the P-1 symmetry will have to be expanded first with function "expand_to_cell".

### Usage

```
strufac(hidx, sdata, anoflag = FALSE, aK = anoK, lbda = 1,
  k = ksigma, f1f2out = TRUE)
```

### Arguments

`hidx` Real numeric. One or more 1D Miller indices.

`sdata` A named list, normally obtained through the use of functions `read_x` or `standardise_sdata`. The list names correspond to different object types:

- a. Real numeric. The size of the unit cell.

- SG. Character string. Space group symbol; either "P1" or "P-1"
- x0. Vector of real numerics indicating the expanded atomic positions in the unit cell.
- Z. Vector of integers indicating the expanded atomic numbers for all atoms in the unit cell.
- B. Vector of real numerics indicating the expanded B factors for all atoms in the unit cell.
- occ. Vector of real numerics indicating the expanded occupancies for all atoms in the unit cell.

`anoflag` Logical variable. If TRUE it forces scattering factors to include anomalous contributions. As a consequence, Friedel's pairs will not be equal.

`aK` Real numeric. This is a fudge factor included to decrease the strength of the anomalous contributions. Without aK the strength is too high for 1D structures, compared to real 3D structures. So aK helps bringing down the anomalous contribution within the 5 met with large-size structures. The default value is aK=0.3 (anoK is included as internal data).

`lbda` Real numeric. This is the wavelength in angstroms. Its value is important in relation to anomalous scattering.

`k` A real number. It controls the standard deviation of the gaussian function describing the atom and, thus, the shape of the associated peak. The standard deviation sigma is given by: `sigma = k * sqrt(Z)` The default value is k=0.05 (ksigma is included as internal data).

`f1f2out` Logical variable. This variable controls output of a small table of f' and f" values for all chemical elements in the structure. Default is for the table to be printed.

## Value

A named list with two vectors of real numbers, the structure factors amplitudes, Fmod, and phases, Fpha, corresponding to the Miller indices in input.

## Examples

```
# First create the crystal structure (P1)
a <- 10
SG <- "P1"
x0 <- c(1,4,6.5)
Z <- c(8,26,6)
B <- c(18,20,17)
occ <- c(1,1,1)
sdata <- standardise_sdata(a,SG,x0,Z,B,occ)
lbda <- 1.7   # Close to Fe's absorption

# Miller indices (including DC (h=0) component)
hidx <- 0:10

# Now structure factors without anomalous contribution
ftmp <- strufac(hidx,sdata,lbda=lbda)
```

```
print(length(ftmp$Fmod))  # Includes DC component (h=0)
print(ftmp)          # Amplitudes decrease with increasing
                     # resolution (Miller indices)

# Now try with anomalous contributions
ftmp <- strufac(hidx,sdata,lbda=lbda,anoflag=TRUE)
print(ftmp)  # DC component is not any longer real
```

## write_h *Write structure factors to a reflections file*

### Description

This function writes standardised structure factors data into an ASCII file. The files includes cell size, space group character symbol and Miller indices vector. It can include all of some of observed and/or calculated structure factors amplitudes and phases, either for anomalous or non-anomalous data.

### Usage

```
write_h(filename, fdata)
```

### Arguments

`filename` A character string. Prefix of the structure factors file name. The file name has the form "[prefix]_h.dat".

`fdata` A names list, usually created with functions `standardise_fdata` or `read_h`, and consisting of the following fields:

- a Real numeric. Unit cell length in angstroms. Always included.
- SG. Spacegroup 2-letters character string. There are only two symmetries possible when working within 1D crystallography, P1 (no symmetry) and P-1 (inversion through the origin). SG can be either "P1" or "P-1". Always included.
- hidx. Real numeric array. 1D unique (positive in the 1D context) Miller indices. Always included.
- Fobs. Real numeric array. Amplitudes of observed structure factors. Not always included.
- sigFobs. Real numeric array. Errors associated with Fobs. Not always included.
- Fp. Real numeric vector. Amplitudes of the positive component of Friedel (or Bijvoet) pairs (F+). Not always included.
- sigFp. Real numeric vector. Errors associated with Fp. Not always included.
- Fm. Real numeric vector. Amplitudes of the negative component of Friedel (or Bijvoet) pairs (F-). Not always included.

- sigFm. Real numeric vector. Errors associated with Fm. Not always included.
- Phiobs. Real numeric array. Phases (in degrees) of structure factors obtained with one of the methods used for structure solution. Not always included.
- Phicalc. Real numeric array. Phases (in degrees) of structure factors calculated from the correct 1D structure. They are normally used to check correctness of Phiobs. Not always included.

## Value

This function does not return anything, but will create an ASCII file of name *_h.dat which contains structure factors and other type of information.

## Examples

```
# Data from thiocyanate structure
datadir <- system.file("extdata",package="crone")
filename <- file.path(datadir,"thiocyanate_x.dat")
sdata <- read_x(filename)

# Miller indices
hidx <- 1:10

# Observed structure factors with errors
ltmp <- sfobs(hidx,sdata)
Fobs <- ltmp$F
sigFobs <- ltmp$sF

# Phases from calculated structure factors
ftmp <- strufac(hidx,sdata)
phicalc <- ftmp$Fpha

# Create standardised fdata structure
fdata <- standardise_fdata(sdata$a,sdata$SG,hidx,Fobs=Fobs,
                     sigFobs=sigFobs,Phicalc=phicalc)

# Name of structure factors file (in temporary directory)
wd <- tempdir()
fname <- file.path(wd,"test")

# Write data to file
write_h(fname,fdata)
```

# write_x *Write atomic coordinates to a file.*

## Description

Function to export all information concerning a given structure to a so-called coordinates file of type *_x.dat.

## Usage

```
write_x(filename, sdata)
```

## Arguments

| | |
|---|---|
| `filename` | A character string. Prefix of the output ASCII file to include all structural information. The file name will be "[Prefix]_x.dat". |
| `sdata` | A named list, normally obtained through the use of function `read_x`. The list names correspond to different object types:<br>• a. Real numeric. The size of the unit cell.<br>• SG. Character string. Space group symbol; either "P1" or "P-1"<br>• x0. Vector of real numerics indicating the expanded atomic positions in the unit cell.<br>• Z. Vector of integers indicating the expanded atomic numbers for all atoms in the unit cell.<br>• B. Vector of real numerics indicating the expanded B factors for all atoms in the unit cell.<br>• occ. Vector of real numerics indicating the expanded occupancies for all atoms in the unit cell. |

## Value

This function does not return anything, but will create an ASCII file of name *_x.dat which contains all coordinates of the atoms in the structure and other type of information.

## Examples

```
# Create an arbitrary structure in P1
a <- 23
SG <- "P1"
x0 <- c(2,11,16,19)
Z <- c(6,6,16,8)
B <- c(13,14,5,10)
occ <- c(1,1,1,1)
wd <- tempdir()
prfx <- file.path(wd,"test")
sdata <- standardise_sdata(a,SG,x0,Z,B,occ)
write_x(prfx,sdata)
```

# Index