

NAME

SPStrGen.pl - Generate structures for Sphingophospholipids (SP)

SYNOPSIS

SPStrGen.pl SPAbbrev|SPAbbrevFileName ...

SPStrGen.pl [-c, --ChainAbbrevMode *MostLikely | Arbitrary*] [-h, --help] [-m, --mode *Abbrev | AbbrevFileName*] [-p, --ProcessMode *WriteSDFFile | CountOnly*] [-o, --overwrite] [-r, --root rootname] [-w, --workingdir dirname] <arguments>...

DESCRIPTION

Generate Sphingophospholipids (SP) structures using compound abbreviations specified on a command line or in a CSV/TSV Text file. All the command line arguments represent either compound abbreviations or file name containing abbreviations. Use mode option to control the type of command line arguments.

A SD file, containing structures for all SP abbreviations along with ontological information, is generated as an output.

SUPPORTED ABBREVIATIONS

The current support for SP structure generation includes the following main classes and sub classes, with additional abbreviations details available under the examples subsection:

o Sphingoid bases

- . Sphing-4-enines (Sphingosines)
- . Sphinganines
- . 4-Hydroxysphinganines (Phytosphingosines)
- . Sphingoid base homologs and variants
- . Sphingoid base 1-phosphates
- . Lysosphingomyelins and lysoglycosphingolipids

o Ceramides

- . N-acylsphingosines (ceramides)
- . N-acylsphinganines (dihydroceramides)
- . N-acyl-4-hydroxysphinganines (phytoceramides)

o Phosphosphingolipids

- . Ceramide phosphocholines (sphingomyelins)
- . Ceramide phosphoethanolamines
- . Ceramide phosphoinositols

o Neutral glycosphingolipids

- . Simple Glc series (GlcCer, LacCer, etc)
- . GalNAcb1-3Gala1-4Galb1-4Glc- (Globo series)
- . GalNAcb1-4Galb1-4Glc- (Ganglio series)
- . Galb1-3GlcNAcb1-3Galb1-4Glc- (Lacto series)
- . GalNAcb1-3Gala1-3Galb1-4Glc- (Isoglobo series)
- . GlcNAcb1-2Manal-3Manb1-4Glc- (Mollu series)
- . GalNAcb1-4GlcNAcb1-3Manb1-4Glc- (Arthro series)
- . Gal- (Gala series)

o Acidic glycosphingolipids

. Gangliosides

OPTIONS

-c, --ChainAbbrevMode *MostLikely|Arbitrary*

Specify what types of acyl chain abbreviations are allowed during processing of complete abbreviations: allow most likely chain abbreviations containing specific double bond geometry specifications; allow any acyl chain abbreviation with valid chain length and double bond geometry specifications. Possible values: *MostLikely* or *Arbitrary*. Default value: *MostLikely*.

Arbitrary value of **-c, --ChainAbbrevMode** option is not allowed during processing of abbreviations containing wild cards.

During *MostLikely* value of **-c, --ChainAbbrevMode** option, only the most likely acyl chain abbreviations specified in ChainAbbrev.pm module are allowed. However, during *Arbitrary* value of **-c, --ChainAbbrevMode** option, any acyl chain abbreviations with valid chain length and double bond geometry can be specified. The current release of lipidmapstools support chain lengths from 2 to 50 as specified in ChainAbbrev.pm module.

In addition to double bond geometry specifications, valid substituents can be specified for in the acyl chain abbreviations.

-h, --help

Print this help message

-m, --mode *Abbrev|AbbrevFileName*

Controls interpretation of command line arguments. Two different methods are provided: specify compound abbreviations or a file name containing compound abbreviations. Possible values: *Abbrev* or *AbbrevFileName*. Default: *Abbrev*

In *AbbrevFileName* mode, a single line in CSV/TSV files can contain multiple compound abbreviations. The file extension determines delimiter used to process data lines: comma for CSV and tab for TSV. For files with TXT extension, only one compound abbreviation per line is allowed.

Wild card character, *, is also supported in compound abbreviations.

Examples:

```
Specific structures: Cer(d18:0/0:0) Cer(d18:1(4E)/0:0)
                  Cer(d19:1(4E)/24:4(5Z,8Z,11Z,14Z))
Specific structures: SM(d18:0/16:0) SM(d19:0/24:1(15Z))
Specific possibilities: Cer(*0:0) Cer(d18:1(4E)/*)
All possibilities: *((*//*) or *((*//*)
```

With wild card character, +/- can also be used for chain lengths to indicate even and odd lengths at sn1/sn2/sn3 positions; additionally > and < qualifiers are also allowed to specify length requirements. Examples:

```
Odd and even number chains at sn1 and sn2: *((-:*/+*:*)
Odd and even number chains at sn1 and sn2 with length longer than
18
and 22: *((->18:*/+>22:*)
```

-p, --ProcessMode *WriteSDFile|CountOnly*

Specify how abbreviations are processed: generate structures for specified abbreviations along with generating a SD file or just count the number of structures corresponding to specified abbreviations without generating any SD file. Possible values: *WriteSDFile* or *CountOnly*. Default: *WriteSDFile*.

It can take substantial amount of time for generating all the structures and writing out a SD file for abbreviations containing wild cards. *CountOnly* value of **--ProcessMode** option can be

used to get a quick count of number of structures to be generated without writing out any SD file.

-o, --overwrite

Overwrite existing files

-r, --root *rootname*

New file name is generated using the root: <Root>.sdf. Default for new file names: SPAbbrev.sdf, <AbbrevFileName>.sdf, or <FirstAbbrevFileName>1To<Count>.sdf.

-w, --workingdir *dirname*

Location of working directory. Default: current directory

EXAMPLES

On some systems, command line scripts may need to be invoked using *perl -s SPStrGen.pl*; however, all the examples assume direct invocation of command line script works.

To generate a specific Sphing-4-enines (Sphingosines) [SP0101] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Cer(d18:1(4E)/0:0)"
```

To enumerate all possible Sphing-4-enines (Sphingosines) [SP0101] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Cer(*:1(*)/0:0)"
```

To generate a specific Sphinganine [SP0102] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Cer(d18:0/0:0)"
```

To enumerate all possible Sphinganine [SP0102] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Cer(*:0/0:0)"
```

To enumerate all possible sphingosine and sphinganine structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Cer(*:*/0:0)"
```

To generate a specific 3-keto sphinganine structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Cer(m18:0/0:0)"
```

To enumerate all possible 3-keto sphingosine and sphinganine structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Cer(m*:*/0:0)"
```

To generate a specific 4-Hydroxysphinganine (Phytosphingosines) [SP0103] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Cer(t18:0/0:0)"
```

To enumerate all possible 4-Hydroxysphinganine (Phytosphingosines) [SP0103] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Cer(t*:0/0:0)"
```

To generate a specific Sphingoid base homologs and variants [SP0104] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "MIPCer(d18:0/0:0)"
% SPStrGen.pl -r SPStructures -o "MIPCer(t18:0/0:0)"
% SPStrGen.pl -r SPStructures -o "MIP2Cer(d18:0/0:0)"
% SPStrGen.pl -r SPStructures -o "MIP2Cer(t18:0/0:0)"
```

To enumerate all possible Sphingoid base homologs and variants [SP0104] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "MIPCer(*:*/0:0)"
% SPStrGen.pl -r SPStructures -o "MIPCer(t*:*/0:0)"
% SPStrGen.pl -r SPStructures -o "MIP2Cer(*:*/0:0)"
% SPStrGen.pl -r SPStructures -o "MIP2Cer(t*:*/0:0)"
```

To generate a specific Sphingoid base 1-phosphates [SP0105] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "CerP(d18:0/0:0)"
```

To enumerate all possible Sphingoid base 1-phosphates [SP0105] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "CerP(*:*/0:0)"
```

To generate a specific Lysosphingomyelins and lysoglycosphingolipids [SP0106] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "SM(d18:0/0:0)"
% SPStrGen.pl -r SPStructures -o "Manb1-4GlcCer(d18:0/0:0)"
```

To enumerate all possible Lysosphingomyelins and lysoglycosphingolipids [SP0106] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "SM(*:*/0:0)"
% SPStrGen.pl -r SPStructures -o "Manb1-4GlcCer(*:*/0:0)"
```

To generate a specific N-acylsphingosines (ceramides) [SP0201] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Cer(d18:1(4E)/16:0)"
```

To enumerate all possible N-acylsphingosines (ceramides) [SP0201] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Cer(*:1(*)/*>0:*)"
```

To generate a specific N-acylsphinganine (dihydroceramides) [SP0202] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Cer(d18:0/16:0)"
```

To enumerate all possible N-acylsphinganine (dihydroceramides) [SP0202] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Cer(*:0/*>0:*)"
```

To generate a specific N-acyl-4-hydroxysphinganine (phytoceramides) [SP0203] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Cer(t18:0/16:0)"
```

To enumerate all possible N-acyl-4-hydroxysphinganine (phytoceramides) [SP0203] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Cer(t*:0/*>0:*)"
```

To generate a specific Ceramide 1-phosphates [SP0205] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "CerP(d18:0/16:0)"
```

To generate a specific Ceramide phosphocholines (sphingomyelins) [SP0301] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "SM(d18:0/16:0)"
```

To enumerate all possible Ceramide phosphocholines (sphingomyelins) [SP0301] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "SM(*:*/*>0:*)"
```

To generate a specific Ceramide phosphoethanolamines [SP0302] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "PECer(d18:0/16:0)"
```

To enumerate all possible Ceramide phosphoethanolamines [SP0302] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "PECer(*:*/*>0:*)"
```

To generate a specific Ceramide phosphoinositols [SP0303] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "PICer(d18:0/16:0)"
```

```
% SPStrGen.pl -r SPStructures -o "PICer(t18:0/16:0)"
```

To enumerate all possible Ceramide phosphoinositols [SP0303] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "PICer(*:*/*>0:0)"
```

```
% SPStrGen.pl -r SPStructures -o "PICer(t*:*/*>0:0)"
```

To generate a specific Simple Glc series (GlcCer, LacCer, etc) [SP0501] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Manb1-4GlcCer(d18:0/16:0)"
```

```
% SPStrGen.pl -r SPStructures -o "GalCer(d18:0/16:0)"
```

```
% SPStrGen.pl -r SPStructures -o "LacCer(d18:0/16:0)"
```

To enumerate all possible Simple Glc series (GlcCer, LacCer, etc) [SP0501] structures and generate

a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Manb1-4GlcCer(*:*/*>0:*)"
% SPStrGen.pl -r SPStructures -o "LacCer(*:*/*>0:*)"
```

To generate a specific GalNAcb1-3Gala1-4Galb1-4Glc- (Globo series) [SP0502] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "GB3Cer(d18:0/16:0)"
```

To enumerate all possible GalNAcb1-3Gala1-4Galb1-4Glc- (Globo series) [SP0502] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "GB3Cer(*:*/*>0:*)"
```

To generate a specific GalNAcb1-4Galb1-4Glc- (Ganglio series) [SP0503] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "asialo-GM2Cer(d18:0/16:0)"
```

To enumerate all possible GalNAcb1-4Galb1-4Glc- (Ganglio series) [SP0503] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "asialo-GM2Cer(*:*/*>0:0)"
```

To generate a specific Galb1-3GlcNAcb1-3Galb1-4Glc- (Lacto series) [SP0504] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Lc3Cer(d18:0/16:0)"
```

To enumerate all possible Galb1-3GlcNAcb1-3Galb1-4Glc- (Lacto series) [SP0504] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "Lc3Cer(*:*/*>0:0)"
```

To generate a specific GalNAcb1-3Gala1-3Galb1-4Glc- (Isoglobo series) [SP0506] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "iGB3Cer(d18:0/16:0)"
```

To enumerate all possible GalNAcb1-3Gala1-3Galb1-4Glc- (Isoglobo series) [SP0506] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "iGB3Cer(*:*/*>0:0)"
```

To generate a specific GlcNAcb1-2Mana1-3Manb1-4Glc- (Mollu series) [SP0507] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "MolluCer(d18:0/16:0)"
```

To enumerate all possible GlcNAcb1-2Mana1-3Manb1-4Glc- (Mollu series) [SP0507] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "MolluCer(*:*/*>0:*)"
```

To generate a specific GalNAcb1-4GlcNAcb1-3Manb1-4Glc- (Arthro series) [SP0508] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "ArthroCer(d18:0/16:0)"
```

To enumerate all possible GalNAcb1-4GlcNAcb1-3Manb1-4Glc- (Arthro series) [SP0508] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "ArthroCer(*:*/*>0:*)"
```

To generate a specific Gal- (Gala series) [SP0509] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "GalCer(d18:0/16:0)"
```

To enumerate all possible Gal- (Gala series) [SP0509] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "GalCer(*:*/*>0:*)"
```

To generate a specific Gangliosides (Acidic glycosphingolipids) [SP0601] structure and write out a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "GM3Cer(d18:0/16:0)"
```

To enumerate all possible Gangliosides (Acidic glycosphingolipids) [SP0601] structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "GM3Cer(*:*/*>*:*)"
```

To enumerate all possible SP structures and generate a SPStructures.sdf file, type:

```
% SPStrGen.pl -r SPStructures -o "**(*/*)"
```

or

```
% SPStrGen.pl -r SPStructures -o "**(*:*/*:*)"
```

AUTHOR

Manish Sud

CONTRIBUTOR

Eoin Fahy

SEE ALSO

CLStrGen.pl, FAStrGen.pl, GLStrGen.pl, GPStrGen.pl, STStrGen.pl

COPYRIGHT

Copyright (C) 2006-2017. The Regents of the University of California. All Rights Reserved.

LICENSE

Modified BSD License