Package 'BiGGR'

April 11, 2018

Type Package

Title Constraint based modeling in R using metabolic reconstruction databases

Version 1.14.0

Date 2014-03-11

Author Anand K. Gavai, Hannes Hettling

Maintainer Anand K. Gavai <anand.gavai@bioinformatics.nl>, Hannes Hettling <hannes.hettling@naturalis.nl>

Depends R (>= 2.14.0), rsbml, hyperdraw, LIM,stringr

Imports hypergraph, limSolve

Description This package provides an interface to simulate metabolic reconstruction from the BiGG database(http://bigg.ucsd.edu/) and other metabolic reconstruction databases. The package facilitates flux balance analysis (FBA) and the sampling of feasible flux distributions. Metabolic networks and estimated fluxes can be visualized with hypergraphs.

License file LICENSE

URL http://www.bioconductor.org/

LazyLoad yes

biocViews Systems Biology,Pathway, Network,GraphAndNetwork,Visualization,Metabolomics

NeedsCompilation no

R topics documented:

`BiGGR-package` *Creates an interface to the BiGG database, provides a framework for simulation and produces flux graphs*

Description

This package provides an interface to simulate metabolic reconstruction from the BiGG database(http://bigg.ucsd.edu/) and other metabolic reconstruction databases. The package aids in performing flux balance analysis (FBA). Metabolic networks and estimated fluxes can be visualized using hypergraphs.

Details

Package:	BiGGR
Type:	Package
Version:	0.99.0
Date:	2013-07-10
Depends:	R (>= 2.14.0), rsbml, hyperdraw, LIM
Imports:	hypergraph
License:	GPL (>=2)
URL:	http://www.bioconductor.org/
Copyright:	see inst/COPYRIGHTS for the license of the BiGG database
biocViews:	NetworkAnalysis, Visualization, Metabolomics
LazyLoad:	yes
Packaged:	2013-08-05 12:44:08 UTC; hettling
Built:	R 3.0.0; ; 2013-08-05 12:44:22 UTC; unix

Index:

E.coli_iAF1260	Ecoli dataset with ORFs and thermodynamic info
E.coli_iJR904	Ecoli genome-scale model
Glycolysis	Metabolic reconstruction of Glycolysis pathway
H.pylori_ilT341	H.pylori in silico genome-scale characterizati
H.sapiens_Recon_1	Reconstruction of human metabolism from the Bi
M.barkeri_iAF692	Metabolic reconstruction of M.barkeri from the
Recon2	Human metabolic reconstruction Recon2
S.aureus_iSB619	Metabolic reconstruction of S.aureus from the
S.cerevisiae_iND750	Metabolic reconstruction of S.cerevisiae from
buildSBMLFromBiGG	Build an SBML model from a given reactions fil
buildSBMLFromGenes	Build an SBML model for specific genes in a gi
buildSBMLFromPathways	Build an SBML model for specific pathway(s) i
buildSBMLFromReactionIDs	Build an SBML model for specific reactions in
createLIMFromBiGG	Create a LIM model object from a BiGG database
createLIMFromSBML	Create a LIM model object from an SBML file
extractGeneAssociations	Extract informations on genes from a given da
extractPathways	Extract all pathways from given database
getPathwaysForSBML	Extract all pathways from a database that are
getRates	Get Optimized Rates
sbml2hyperdraw	Returns a graph representation of an SBML model

Further information is available in the following vignettes:

BiGGR BiGGR (source, pdf)

Author(s)

Anand K. Gavai, Hannes Hettling

Maintainer: Anand K. Gavai <anand.gavai@bioinformatics.nl>, Hannes Hettling <j.hettling@vu.nl>

See Also

rsbml

Examples

```
# library("BiGGR")


library(help="BiGGR")

##load reaction identifiers from package examples
file.name <- system.file("extdata",
 "Glycolysis_TCA_recon1_reactionIDs.txt",
 package="BiGGR")
reaction.ids <- scan(file.name, what=" ")

##load database
data("H.sapiens_Recon_1")

##build SBML model
```

```
sbml.model <- buildSBMLFromReactionIDs(reaction.ids, H.sapiens_Recon_1)

##following term is to be maximized
maximize <- "R_ATPS4m - R_NDPK1m - R_HEX1 - R_PFK - R_PGK + R_PYK"

##specify the external metabolites of the system
externals <- c("M_glc_DASH_D_e", "M_lac_DASH_L_e",
   "M_ala_DASH_L_e", "M_gln_DASH_L_c", "M_h2o_e",
   "M_co2_e", "M_o2_e", "M_h_e", "M_pi_c",
   "M_o2s_m", "M_nh4_m", "M_adp_c",
   "M_atp_c", "M_nadp_c", "M_nadph_c", "M_h_c")
##specify the values of following fluxes:
##R_GLCt1r=0.4, R_O2t=2.4, R_L_LACt2r=R_GLNtm=0

equation.vars <- c("R_GLCt1r", "R_O2t", "R_L_LACt2r", "R_GLNtm")
equation.values <- c(0.4, 2.4, 0.0, 0.0)
eqns <- list(equation.vars, equation.values)

##create LIM file
limfile.name <- tempfile()
createLIMFromSBML(sbml.model, maximize, equations=eqns,
  externals=externals, file.name=limfile.name)

rates <- getRates(limfile.name)

relevant.species <- c("M_glc_DASH_D_c", "M_g6p_c", "M_f6p_c",
  "M_fdp_c", "M_dhap_c", "M_g3p_c",
  "M_13dpg_c", "M_3pg_c", "M_2pg_c",
  "M_pep_c", "M_pyr_c")
##generate graphical representation
hd <- sbml2hyperdraw(sbml.model, rates=rates,
 relevant.species=relevant.species,
 layoutType="dot", plt.margins=c(20, 0, 20, 0))

##plot hypergraph
plot(hd)
```

`buildSBMLFromBiGG` *Build an SBML model from a given reactions file obtained from the BiGG database*

Description

Creates an SBML model containing all species, reactions and compartments that occur in a reactions file obtained from the BiGG database.

Usage

```
buildSBMLFromBiGG(reactions.filename, model.id=character(0), model.name=character(0))
```

Arguments

reactions.filename	name of the file containing the reactions extracted from BiGG
model.id	id for the SBML model created by the function. Defaults to reactions.filename
model.name	name for the SBML model created by the function. Defaults to reactions.filename

Value

a rsbml Model object containing all reactions, species and compartments that are associated with the reactions in the given input file.

Note

Note that it can be the case that species can be present in multiple compartments. In order to avoid any ambiguities in the model returned by the function, each species identifier is composed of the species identifier given in the reactions file and the compartment identifier, joined by "_". Example: adp in compartment c (cytosol) has the id adp_c.

Author(s)

Anand Gavai, Hannes Hettling

References

Schellenberger, J., Park, J. O., Conrad, T. C., and Palsson, B. , BiGG: a Biochemical Genetic and Genomic knowledgebase of large scale metabolic reconstructions, BMC Bioinformatics, 11:213, (2010). http://bigg.ucsd.edu/

See Also

createLIMFromSBML

Examples

```
##build model from file Reactions.txt from the package examples
path <- system.file("extdata", "Reactions.txt", package="BiGGR")
model <- buildSBMLFromBiGG(path, model.id="myid")
```

---

buildSBMLFromGenes *Build an SBML model for specific genes in a given database*

Description

Creates an SBML model containing all species, reactions and compartments that are associated with (a) specific gene(s) in the database document (e.g. Recon2) passed as an argument.

Usage

```
buildSBMLFromGenes(query, database, logical.fun="any")
```

Arguments

query	a character or a vector or list of character containing the query genes with identifiers as specified in the database.
database	an object of class SBMLDocument
logical.fun	function which specifies the logical relation of the query genes within the reactions (e.g. all or any, see details).

Details

The function all as argument logical.fun would mean that all genes in the query have to be associated with a certain reaction from the database in order to be included in the returned model. The default any means that a reaction is included if any of the query genes are associated with it. Custom functions are possible if they take a vector of type logical as an argument and return a logical. The argument of logical.fun is a vector of type logical having the same length as the query and for each gene the value is TRUE if it is associated with a specific reaction. See 'examples' section for an example of a custom function as logical.fun.

Value

a rsbml Model object containing all reactions, species and compartments that are present in the database and are associated with the query gene(s) or NULL if none of the genes in the database match the query.

Note

If the reactions in the database document provided in the argument database do not contain any "<notes>" with tags with gene information indicated by the string "GENE*ASSOCIATION" (the star stands for any character), no gene association information can be extracted and thus the returned SBML mdel is empty..

Author(s)

Anand Gavai, Hannes Hettling

References

Thiele, I. et al. Nat Biotech, 2013

See Also

buildSBMLFromPathways extractGeneAssociations

Examples

```
##Query genes in Recon 2 database
data("Recon2")
database <- Recon2
m1 <- buildSBMLFromGenes("8884.1", database)
m2 <- buildSBMLFromGenes(c("8884.1", "6509.1"), database)

##different databases
data(H.pylori_ilT341)
```

```
database <- H.pylori_ilT341
m3 <- buildSBMLFromGenes("HP0069", database)

data(M.barkeri_iAF692)
database <- M.barkeri_iAF692
m4 <- buildSBMLFromGenes(c("MBd0456", "MBd4814", "MBd4098"), database)


data(S.aureus_iSB619)
database <- S.aureus_iSB619
m5 <- buildSBMLFromGenes(c("SA0594", "SA1599", "SA0950", "SA0259"), database)

database <- Recon2
query <- c("218.1", "223.1")
m6 <- buildSBMLFromGenes(query, database)
m7 <- buildSBMLFromGenes(query, database, logical.fun="all")
##m6 has more reactions than m7
## because m7 has only reactions which match both genes in the query
length(m6@reactions) > length(m7@reactions)

##Custom logical function: Get model with all reactions
## which are not associated with the query gene
m8 <- buildSBMLFromGenes(query, database, logical.fun=function(x)!any(x))
```

`buildSBMLFromPathways` *Build an SBML model for specific pathway(s) in a given database*

Description

Creates an SBML model containing all species, reactions and compartments that are part of (a) given pathway(s) in the database document (e.g. Recon2) passed as an argument.

Usage

```
buildSBMLFromPathways(query, database, match.exact=TRUE)
```

Arguments

`query`	a `character` or a `vector` or `list` of `character` containing the names of the query pathways
`database`	an object of class `SBMLDocument`
`match.exact`	`logical` whether only the exact pathway name should be matched or whether a pathway should match if one keyword is in the pathway description in the database.

Value

a rsbml `Model` object containing all reactions, species and compartments that are present in the database for the query pathway(s) or NULL if none of the pathways in the database match the query.

Note

If the reactions in the database document provided in the argument `database` do not contain any "<notes>" with tags with pathway information indicated by the string "SUBSYSTEM", no pathway information can be extracted and thus the SBML model returned will be empty.

Author(s)

Anand Gavai, Hannes Hettling

References

Thiele, I. et al. Nat Biotech, 2013

See Also

`extractPathways`

Examples

```
data("Recon2")
database <- Recon2

##Get Model for specific pathway
m1 <- buildSBMLFromPathways("Arginine and Proline Metabolism", database)

##Get Model for specific pathway "Metabolism": does not exist!
m2 <- buildSBMLFromPathways("Metabolism", database)

##Get model of all pathways which contain keyword "metabolism"
m3 <- buildSBMLFromPathways("Metabolism", database, match.exact=FALSE)

##Multi-query:
query <- c("Transport, endoplasmic reticular", "Arginine and Proline Metabolism")
m4 = buildSBMLFromPathways(query, database)
m5 = buildSBMLFromPathways(query[1], database)
length(m4@species)
length(m5@species)

##different database
data(H.pylori_ilT341)
database <- H.pylori_ilT341
m7 <- buildSBMLFromPathways("Metabolism", database, match.exact=FALSE)
```

---

buildSBMLFromReactionIDs

Build an SBML model for specific reactions in a given database

---

Description

Creates an SBML model containing all species, reactions and compartments that are associated with a number of reaction identifiers in the database document (e.g. Recon2) passed as an argument.

Usage

```
buildSBMLFromReactionIDs(reaction.ids, database)
```

Arguments

reaction.ids	a character or a vector or list of character containing the names of the query reaction IDs
database	an object of class SBMLDocument

Value

a rsbml Model object containing all reactions, species and compartments that are present in the database for the query reaction(s) or NULL if none of the query reaction IDs is found in the database.

Author(s)

Anand Gavai, Hannes Hettling

References

Thiele, I. et al. Nat Biotech, 2013

See Also

buildSBMLFromGenes

Examples

```
##get list of reactions with Recon 2 identifiers from examples
path <- system.file("extdata", "Glycolysis_TCA_recon2_reactionIDs.txt", package="BiGGR")
reaction.ids <- scan(path, what=" ")

data("Recon2")
model <- buildSBMLFromReactionIDs(reaction.ids, Recon2)
```

---

createLIMFromBiGG *Create a LIM model object from a BiGG database file*

Description

Creates a LIM model object from a file containing reactions extreacted from BiGG to be run for simulations of metabolic fluxes

Usage

```
createLIMFromBiGG(reactions.filename, ...)
```

Arguments

reactions.filename	file which contains the reactions extracted from the BiGG database
...	arguments passed to createLIMfromSBML

Note

none

Author(s)

Anand K. Gavai <anand.gavai@bioinformatics.nl>, Hannes hettling <j.hettling@vu.nl>

References

Soetaert K, van Oevelen D (2009). LIM: Linear Inverse Model examples and solution methods. R package version 1.3

See Also

createLIMFromSBML

Examples

```
##maximize flux for reaction R_PYK
maximize <- "R_PYK"

##setting equality constraint R_HEX = 1
equation_var <- "R_HEX1"
equation_value <- 1
eq <- list(equation_var, equation_value)

##range  of  possible  fluxes for R_PYK
constraint <- list("R_PYK", 0, 1000)
externals  <-  c("glc_c",  "pyr_c",  "h_c","nad_c",
 "nadh_c",  "pi_c",  "fad_m",  "fadh2_m",
 "o2_c",  "adp_c",  "atp_c",  "nadp_c",
 "co2_c",  "o2_c",  "gdp_c",  "gtp_c")

##build LIM model from reactions file in package examples
path <- system.file("extdata", "Reactions.txt", package="BiGGR")
limfile.name <- tempfile()
createLIMFromBiGG(path, maximize, equations=eq, constraints=constraint,
  externals=externals, file.name=limfile.name)
```

createLIMFromSBML *Create a LIM model object from an SBML file*

Description

creates a model file to be run for simulations of metabolic fluxes

Usage

```
createLIMFromSBML(model, maximize, equations=NULL, inequalities=NULL,
constraints=NULL, externals=NULL, file.name="model.lim")
```

Arguments

model	an SBML object of reactions/metabolites participating in a metabolic pathway.
maximize	a character vector consisting the tag of the reaction(s) to be maximized or minimized
equations	a list specifying equality constraints on the system. The list must have two entries, the first one being a vector of class character containing the left hand side(s) of the equation(s), the second one being a vector of type character or mumeric with the right hand side(s) of the equation(s). See also 'examples'.
inequalities	a list specifying inequality constraints on the system. The list must have three entries, the first one being a vector of class character containing the left hand side(s) of the inequality equation(s), the second one being a vector of type character or mumeric with the right hand side(s) of the inequality equation(s) and the third one being a vector of class character containing the relational operator of the inequality equations, for example ">" or "<=". See also 'examples'.
constraints	a list specifying constrained on the solution space of the flux vector. The list must have three entries, the first one being a vector of class character with the reaction id(s) to be constrained, the second and third one a numeric vector with the lower and upper flux bounds, respectively, for the reactions to be contrained. is a character vector specifying the minimum and maximum values(boundary) under which the solution for the maximize reaction should fall
externals	a character vector of metabolites as provided by the user for speficific pathways for which FBA (flux balance analysis needs to be performed)
file.name	a character string specifying the name of the LIM file created by the function.

Value

A model file with with extension ".lim" is created

Note

none

Author(s)

Anand K. Gavai <anand.gavai@bioinformatics.nl>, Hannes Hettling <j.hettling@vu.nl>

References

Soetaert K, van Oevelen D (2009). LIM: Linear Inverse Model examples and solution methods. R package version 1.3

Examples

```
##Create a LIM model file from a reactions file in the examples
path <- system.file("extdata", "Glycolysis_TCA_recon1_reactionIDs.txt", package="BiGGR")
reaction.ids <- scan(path, what=" ")
data("H.sapiens_Recon_1")
sbml.model <- buildSBMLFromReactionIDs(reaction.ids, H.sapiens_Recon_1)
maximize <- c("R_ATPS4m - R_NDPK1m - R_HEX1 - R_PFK  - R_PGK + R_PYK")
externals <- c("M_glc_DASH_D_e", "M_lac_DASH_L_e",
   "M_ala_DASH_L_e", "M_gln_DASH_L_c", "M_h2o_e",
   "M_co2_e", "M_o2_e", "M_h_e", "M_pi_c",
   "M_o2s_m", "M_nh4_m", "M_adp_c",
   "M_atp_c", "M_nadp_c", "M_nadph_c", "M_h_c")
equation.vars <- c("R_GLCt1r", "R_O2t", "R_L_LACt2r", "R_GLNtm")
equation.values <- c(0.4, 2.4, 0.0, 0.0)
eqns <- list(equation.vars, equation.values)
constraints <- list(c("R_GLCt1r", "R_CYOOm3"), c(-1000, -1000), c(1000, 1000))
limfile.name <- tempfile()
createLIMFromSBML(sbml.model, maximize, equations=eqns,
  inequalities=list("R_O2t", 2.4, "<="),
  constraints=constraints, externals=externals,
  file.name=limfile.name)
```

E.coli_iAF1260 *Ecoli dataset with ORFs and thermodynamic information*

Description

A genome-scale metabolic reconstruction for Escherichia coli K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information. The dataset was generated by downloading the SBML file of the reconstruction (http://bigg.ucsd.edu/bigg/exportSelect.pl) which was subsequently converted into an object of class SBML using the rsbml_read function from the rsbml package.

Usage

```
data(E.coli_iAF1260)
```

Format

An sbml object of class rsbml

Details

Note that the files in the BiGG database fail the unit consistancy check of the rsbml_read function. To avoid unit checking when creating SBML objects, the substance units in the reaction tags were parsed out from the database SBML files (see example below).

Source

http://bigg.ucsd.edu/bigg/exportSelect.pl

References

Feist, A.M., Henry, C.S., Reed, J.L., Krummenacker, M., Joyce, A.R., Karp, P.D., Broadbelt, L.J., Hatzimanikatis, V., Palsson, B.O., *A genome-scale metabolic reconstruction for Escherichia coli K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information*, olecular Systems Biology, 3:121 (2007)

Examples

```
## Not run:
##The dataset was generated as follows:
##SBML_export.xml was downloaded from http://bigg.ucsd.edu/bigg/exportSelect.pl
##and a newline was added at the end of the file
file <- "SBML_export.xml"
string <- paste(readLines(file), collapse="\n")
##Parse out units to avoid validation error
string <- gsub("units=\".+?\"", "", string)
E.coli_iAF1260 <- rsbml_read(text=string)

## End(Not run)

##load data and get all reaction IDs
data(E.coli_iAF1260)
model <- E.coli_iAF1260@model
##get all reaction identifiers
sapply(model@reactions, id)
```

E.coli_iJR904 *Ecoli genome-scale model*

Description

An expanded genome-scale model of Escherichia coli K-12 (iJR904 GSM/GPR). The dataset was generated by downloading the SBML file of the reconstruction (http://bigg.ucsd.edu/bigg/exportSelect.pl) which was subsequently converted into an object of class `SBML` using the `rsbml_read` function from the `rsbml` package.

Usage

```
data(E.coli_iJR904)
```

Format

An sbml object of class rsbml

Details

Note that the files in the BiGG database fail the unit consistancy check of the `rsbml_read` function. To avoid unit checking when creating `SBML` objects, the substance units in the reaction tags were parsed out from the database SBML files (see example below).

Source

http://bigg.ucsd.edu/bigg/exportSelect.pl

References

Reed, J.L., Vo, T.D., Schilling, C.H., and Palsson, B.O., *An expanded genome-scale model of Escherichia coli K-12 (iJR904 GSM/GPR)*, Genome Biology, 4(9): R54.1-R54.12 (2003).

Examples

```
## Not run:
##The dataset was generated as follows:
##SBML_export.xml was downloaded from http://bigg.ucsd.edu/bigg/exportSelect.pl
##and a newline was added at the end of the file
file <- "SBML_export.xml"
string <- paste(readLines(file), collapse="\n")
##Parse out units to avoid validation error
string <- gsub("units=\".+?\"", "", string)
E.coli_iJR904 <- rsbml_read(text=string)

## End(Not run)

##load data and get all reaction IDs
data(E.coli_iJR904)
model <- E.coli_iJR904@model
##get all reaction identifiers
sapply(model@reactions, id)
```

E.coli_textbook *Ecoli dataset from the BiGG database*

Description

A metabolic reconstruction for Escherichia from text books. The dataset was generated by downloading the SBML file of the reconstruction (http://bigg.ucsd.edu/bigg/exportSelect.pl) which was subsequently converted into an object of class `SBML` using the `rsbml_read` function from the `rsbml` package.

Usage

```
data(E.coli_textbook)
```

Format

An sbml object of class rsbml

Details

Note that the files in the BiGG database fail the unit consistancy check of the `rsbml_read` function. To avoid unit checking when creating `SBML` objects, the substance units in the reaction tags were parsed out from the database SBML files (see example below).

Source

http://bigg.ucsd.edu/bigg/exportSelect.pl

References

Feist, A.M., Henry, C.S., Reed, J.L., Krummenacker, M., Joyce, A.R., Karp, P.D., Broadbelt, L.J., Hatzimanikatis, V., Palsson, B.O., *A genome-scale metabolic reconstruction for Escherichia coli K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information*, olecular Systems Biology, 3:121 (2007)

Examples

```
## Not run:
##The dataset was generated as follows:
##SBML_export.xml was downloaded from http://bigg.ucsd.edu/bigg/exportSelect.pl
##and a newline was added at the end of the file
file <- "SBML_export.xml"
string <- paste(readLines(file), collapse="\n")
##Parse out units to avoid validation error
string <- gsub("units=\".+?\"", "", string)
E.coli_textbook <- rsbml_read(text=string)

## End(Not run)

##load data and get all reaction IDs
data(E.coli_textbook)
model <- E.coli_textbook@model
##get all reaction identifiers
sapply(model@reactions, id)
```

---

extractGeneAssociations

Extract informations on genes from a given database

Description

Extracts all information on genes associated to reactions from an rsbml document containing a metabolic reconstruction database (e.g. Recon2). The associated information is parsed from the "<notes>" tag of each reaction's SBML representation.

Usage

```
extractGeneAssociations(database)
```

Arguments

database	an object of class SBMLDocument

Value

a list with length being the number of reactions in the database passed as argument each entry containing a character containing the assiciated gene identifiers and the reaction IDs as names. For reactions without gene annotation, the list will contain NA.

Note

If the reactions in the database document provided in the argument database do not contain any "<notes>" with tags with gene information indicated by the string "GENE*ASSOCIATION" (the star stands for any character), no gene association information can be extracted and thus the returned SBML mdel is empty..

Author(s)

Anand Gavai, Hannes Hettling

References

Thiele, I. et al. Nat Biotech, 2013

See Also

buildSBMLFromGenes

Examples

```
data("Recon2")
database <- Recon2
gene.info <- extractGeneAssociations(database)
```

---

extractPathways *Extract all pathways from given database*

Description

Extracts all pathway information from an rsbml document containing a metabolic reconstruction database (e.g. Recon2). The pathway information is parsed from the "<notes>" tag of each reaction.

Usage

```
extractPathways(database)
```

Arguments

database	an object of class SBMLDocument

Value

a `list` with length being the number of reactions in the database passed as argument each entry containing a `character` with the pathway information and the reaction IDs as names. For reactions without pathway annotation, the list will contain `NA`.

Note

If the reactions in the database document provided in the argument `database` do not contain any "<notes>" with tags with pathway information indicated by the string "SUBSYSTEM", no pathway information can be extracted.

Author(s)

Anand Gavai, Hannes Hettling

References

Thiele, I. et al. Nat Biotech, 2013

See Also

`buildSBMLFromPathways` `getPathwaysForSBML`

Examples

```
data(Recon2)
pathways.recon2 <- extractPathways(Recon2)
```

---

getPathwaysForSBML *Extract all pathways from a database that are relevant for a given SBML model*

Description

Extracts all pathway information from an rsbml document containing a metabolic reconstruction database (e.g. Recon2) and returns the subset of these pathways that is associated with the reactions in the given model. The pathway information is parsed from the "<notes>" tag of each reaction.

Usage

```
getPathwaysForSBML(model, database)
```

Arguments

`database`	an rsbml object of class `SBMLDocument`
`model`	an rsbml object of class `Model`

Value

A vector of type `character` that contains all the pathways relevant for the given model according to the specified database. Note that duplicate pathways do not appear twice in the return value.

Note

If the reactions in the database document provided in the argument `database` do not contain any "<notes>" with tags with pathway information indicated by the string "SUBSYSTEM", no pathway information can be extracted.

Author(s)

Anand Gavai, Hannes Hettling

References

Thiele, I. et al. Nat Biotech, 2013

See Also

`buildSBMLFromPathways` `buildSBMLFromGenes` `extractPathways`

Examples

```
##Build a model from query genes
data("Recon2")
database <- Recon2
query <- c("218.1", "223.1") ##query gene identifiers
m <- buildSBMLFromGenes(query, database)

##extract all pathways for that model
getPathwaysForSBML(m, database)
```

`getRates` *Get Optimized Rates*

Description

getRates takes the model file as the argument and based on the description of the model file generates flux values for "minimum" or "maximum" reaction rates

Usage

```
getRates(modelFile)
```

Arguments

`modelFile`	The path to a LIM model file as generated for instance from the functions `createLIMFromBiGG` or `createLIMFromSBML`

Value

The value returned is one dimentional numeric vector of flux rates for each reaction

Author(s)

Anand K. Gavai <anand.gavai@bioinformatics.nl>

Examples

```
data("Glycolysis")
rates <-getRates(Glycolysis)
rates
```

Glycolysis *Metabolic reconstruction of Glycolysis pathway*

Description

Model of Glycolysis pathway

Usage

```
Glycolysis
```

Format

A LIM model file created as an example

---

gprMapping *GPR mapping*

Description

Continuous gene expression levels are mapped from genes to reactions using the gene-protein-reaction (GPR) association rules as found in ReconX databases. The expression level of reactions catalyzed by enzyme complexes (and operator) can be set to the minimum,maximum,mean and median functions. Similarly expression level of the associated genes, and the expression level of reactions catalyzed by isoenzymes (or operator) can also be set to either minimum,maximum,mean and median functions for the associated genes. Operator Precedence: "AND" followed by "OR"

Usage

```
gprMapping(gene_express,react_gene_map,OR=c("mean","median","min","max"),AND=c("min","max","mean
```

Arguments

gene_express	The path to a gene expression file with three columns gene_symbol,entrez_id and foldchanges
react_gene_map	Database file created from ReconX database using functions such as rmvSpliceVariant
OR	Takes values from statistical functions such as mean,median,min,max
AND	Takes values from statistical functions such as mean,median,min,max

Value

Returns a dataframe with Reaction_id, GPR formulae and Calculated values

Author(s)

Anand K. Gavai <anand.gavai@bioinformatics.nl>, Hannes Hettling

Examples

```
# Read gene expression data
file <- system.file("extdata", "Gene_Symbol_Entrez_Foldchanges.csv", package="BiGGR")
gene_express<-read.csv(file,header=TRUE)
data(Recon2)
gene.info <- extractGeneAssociations(Recon2)

gene.info<-do.call(rbind.data.frame,gene.info)
colnames(gene.info)<-c("GPR")
gene.info$react_id<-row.names(gene.info)
gene.info<-gene.info[,c(2,1)]
rownames(gene.info)<-NULL
react_gene_map<-rmvSpliceVariant(gene.info)

gpr.map<-gprMapping(gene_express,react_gene_map,OR="mean",AND="min")
```

---

gprMappingAvg *GPR mapping ignoring AND & OR operators*

Description

Continuous gene expression levels are mapped from genes to reactions using the gene-protein-reaction (GPR) association rules as found in ReconX databases. These rules are comprised of `AND` and `OR` operators. This function ignores these rules and take average of all genes

Usage

```
gprMappingAvg(gene_express,react_gene_map)
```

Arguments

`gene_express`	The path to a gene expression file with three columns gene_symbol,entrez_id and foldchanges
`react_gene_map`	Database file created from ReconX database using functions such as `rmvSpliceVariant`

Value

Returns a dataframe with Reaction_id, GPR formulae and average values

Author(s)

Anand K. Gavai <anand.gavai@bioinformatics.nl>, Hannes Hettling

Examples

```
# Read gene expression data
file <- system.file("extdata", "Gene_Symbol_Entrez_Foldchanges.csv", package="BiGGR")
gene_express<-read.csv(file,header=TRUE)
data(Recon2)
gene.info <- extractGeneAssociations(Recon2)

gene.info<-do.call(rbind.data.frame,gene.info)
colnames(gene.info)<-c("GPR")
gene.info$react_id<-row.names(gene.info)
gene.info<-gene.info[,c(2,1)]
rownames(gene.info)<-NULL
react_gene_map<-rmvSpliceVariant(gene.info)

gpr.map.avg<-gprMappingAvg(gene_express,react_gene_map)
```

H.pylori_ilT341 *H.pylori in silico genome-scale characterization of single and double deletion mutants*

Description

An Expanded Metabolic Reconstruction of Helicobacter pylori (iIT341 GSM/GPR): An in silico genome-scale characterization of single and double deletion mutants. The dataset was generated by downloading the SBML file of the reconstruction (http://bigg.ucsd.edu/bigg/exportSelect.pl) which was subsequently converted into an object of class `SBML` using the `rsbml_read` function from the `rsbml` package.

Usage

```
data(H.pylori_ilT341)
```

Format

An sbml object of class rsbml

Details

Note that the files in the BiGG database fail the unit consistancy check of the `rsbml_read` function. To avoid unit checking when creating `SBML` objects, the substance units in the reaction tags were parsed out from the database SBML files (see example below).

Source

http://bigg.ucsd.edu/bigg/exportSelect.pl

References

Thiele, I., Vo, T.D., Price, N.D. and Palsson, B., *An Expanded Metabolic Reconstruction of Helicobacter pylori* (iIT341 GSM/GPR): An in silico genome-scale characterization of single and double deletion mutants, Journal of Bacteriology, 187(16): 5818-5830 (2005)

Examples

```
## Not run:
##The dataset was generated as follows:
##SBML_export.xml was downloaded from http://bigg.ucsd.edu/bigg/exportSelect.pl
##and a newline was added at the end of the file
file <- "SBML_export.xml"
string <- paste(readLines(file), collapse="\n")
##Parse out units to avoid validation error
string <- gsub("units=\".+?\"", "", string)
H.pylori_ilT341 <- rsbml_read(text=string)

## End(Not run)

##load data and get all reaction IDs
data(H.pylori_ilT341)
model <- H.pylori_ilT341@model
##get all reaction identifiers
sapply(model@reactions, id)
```

H.sapiens_Recon_1 *Reconstruction of human metabolism from the BiGG database*

Description

The dataset was generated by downloading the SBML file of the reconstruction (http://bigg.ucsd.edu/bigg/exportSelect.pl which was subsequently converted into an object of class SBML using the rsbml_read function from the rsbml package.

Usage

```
data(H.sapiens_Recon_1)
```

Format

An sbml object of class rsbml

Details

Note that the files in the BiGG database fail the unit consistancy check of the rsbml_read function. To avoid unit checking when creating SBML objects, the substance units in the reaction tags were parsed out from the database SBML files (see example below).

Source

http://bigg.ucsd.edu/bigg/exportSelect.pl

References

Duarte, N.D., Becker, S. A., Jamshidi, N., Thiele, I., Mo, M. L., Vo, T. D., Srivas, R., Palsson, B. O., Global reconstruction of the human metabolic network based on genomic and bibliomic data, Proc. Nat Acad. Sci. 104(6):1777-82 (2007)

Examples

```
## Not run:
##The dataset was generated as follows:
##SBML_export.xml was downloaded from http://bigg.ucsd.edu/bigg/exportSelect.pl
##and a newline was added at the end of the file
file <- "SBML_export.xml"
string <- paste(readLines(file), collapse="\n")
##Parse out units to avoid validation error
string <- gsub("units=\".+?\"", "", string)
H.sapiens_Recon_1 <- rsbml_read(text=string)

## End(Not run)

##load data and get all reaction IDs
data(H.sapiens_Recon_1)
model <- H.sapiens_Recon_1@model
##get all reaction identifiers
sapply(model@reactions, id)
```

---

lying.tunell.data *Dataset of in vivo cerebral metabolite uptake and release rates in healthy humans (old subjects)*

Description

These data were taken from a publication of Lying-Tunell et al. (1980) reporting cerebral metabolic uptakes and release rates in older subjects (n=5). The data were published as micromole/kg/min, but converted to mmole/min for this dataset (see details).

Usage

```
data(lying.tunell.data)
```

Format

An object of class data.frame

Details

Data were taken from table 2 (page 271) of the publication. From the given median and range values, mean and standard deviation was estimated using a method by Hozo et al. (2005). Units were converted from micromole/kg/min to mmole/min assuming a brain mass of 1.4kg.

Source

http://www.ncbi.nlm.nih.gov/pubmed/7468149

References

Lying-Tunell U, Lindblad BS, Malmlund HO, Persson B: Cerebral blood flow and metabolic rate of oxygen, glucose, lactate, pyruvate, ketone bodies and amino acids. Acta Neurol Scand 1980, 62:265-75.

Hozo SP, Djulbegovic B, Hozo I: Estimating the mean and variance from the median, range, and the size of a sample. BMC Med Res Methodol 2005, 5:13.

Examples

```
## Not run:
##The dataset was generated as follows:

##Uptake rates given in micromole/kg/min from Lying-Tunell (1980), n=5 old patients
##converted to mmol/min and assuming a brain mass of 1.4 kg
brain.mass <- 1.4 ## in kg
oxygen.median <- 1679 * brain.mass / 1000
oxygen.range <- c(1184, 1872) * brain.mass / 1000
glucose.median <- 203 * brain.mass / 1000
glucose.range <- c(187, 321) * brain.mass / 1000
lactate.median <- -9.2 * brain.mass / 1000
lactate.range <- c(-68, 7.9) * brain.mass / 1000
pyruvate.median <- -2.4 * brain.mass / 1000
pyruvate.range <- c(-10, -brain.mass) * brain.mass / 1000
glutamine.median <- -11 * brain.mass / 1000
glutamine.range <- c(-61, 22) * brain.mass / 1000

##This implements eq 4 from Hozo et al. to estimate
##sample mean from median and range
##m: median, a: minimum, b: maximum, n: number of samples
estimate.sample.mean <- function(m, a, b, n)
(a + 2*m + b)/4 + (a-2*m + b)/(4*n)

##This implements eq 16 from Hozo et al. to estimate
##sample standard deviation from median and range
##m: median, a: minimum, b: maximum, n: number of samples
estimate.sample.sd <- function(m, a, b, n)
sqrt((((a - 2*m + b)^2)/4 + (b-a)^2)/12)

##Calculate mean and standard deviation from median and range values using the method of Hoxo et al.
oxygen.mean <- estimate.sample.mean(oxygen.median, oxygen.range[1], oxygen.range[2], 5)
oxygen.sd <- estimate.sample.sd(oxygen.median, oxygen.range[1], oxygen.range[2], 5)

glucose.mean <- estimate.sample.mean(glucose.median, glucose.range[1], glucose.range[2], 5)
glucose.sd <- estimate.sample.sd(glucose.median, glucose.range[1], glucose.range[2], 5)

lactate.mean <- estimate.sample.mean(lactate.median, lactate.range[1], lactate.range[2], 5)
lactate.sd <- estimate.sample.sd(lactate.median, lactate.range[1], lactate.range[2], 5)

pyruvate.mean <- estimate.sample.mean(pyruvate.median, pyruvate.range[1], pyruvate.range[2], 5)
pyruvate.sd <- estimate.sample.sd(pyruvate.median, pyruvate.range[1], pyruvate.range[2], 5)

glutamine.mean <- estimate.sample.mean(glutamine.median, glutamine.range[1], glutamine.range[2], 5)
glutamine.sd <- estimate.sample.sd(glutamine.median, glutamine.range[1], glutamine.range[2], 5)


lying.tunell.data <- data.frame(median=c(oxygen.median, glucose.median, lactate.median, pyruvate.median, glu
```

```
mean=c(oxygen.mean, glucose.mean, lactate.mean, pyruvate.mean, glutamine.mean),
sd=c(oxygen.sd, glucose.sd, lactate.sd, pyruvate.sd, glutamine.sd),
low=c(oxygen.range[1], glucose.range[1], lactate.range[1], pyruvate.range[1], glutamine.range[1]),
high=c(oxygen.range[2], glucose.range[2], lactate.range[2], pyruvate.range[2], glutamine.range[2]),
row.names=c("o2", "glucose", "lactate", "pyruvate", "glutamine"))

## End(Not run)

##load data
data(lying.tunell.data)
##get median value for glucose uptake
lying.tunell.data["glucose", "median"]
```

M.barkeri_iAF692 *Metabolic reconstruction of M.barkeri from the BiGG database*

Description

The dataset was generated by downloading the SBML file of the reconstruction (http://bigg.ucsd.edu/bigg/exportSelect.pl which was subsequently converted into an object of class SBML using the rsbml_read function from the rsbml package.

Usage

```
data(M.barkeri_iAF692)
```

Format

An sbml object of class rsbml

Details

Note that the files in the BiGG database fail the unit consistancy check of the rsbml_read function. To avoid unit checking when creating SBML objects, the substance units in the reaction tags were parsed out from the database SBML files (see example below).

Source

http://bigg.ucsd.edu/bigg/exportSelect.pl

References

Feist, A.M., Scholten, J.C.M., Palsson, B.O., Brockman, F.J., and Ideker, T., "Modeling methanogenesis with a genome-scale metabolic reconstruction of Methanosarcina barkeri", Molecular Systems Biology, 2(1):msb4100046-E1-E14 (2006)

Examples

```
## Not run:
##The dataset was generated as follows:
##SBML_export.xml was downloaded from http://bigg.ucsd.edu/bigg/exportSelect.pl
##and a newline was added at the end of the file
file <- "SBML_export.xml"
```

```
string <- paste(readLines(file), collapse="\n")
##Parse out units to avoid validation error
string <- gsub("units=\".+?\"", "", string)
M.barkeri_iAF692 <- rsbml_read(text=string)

## End(Not run)

##load data and get all reaction IDs
data(M.barkeri_iAF692)
model <- M.barkeri_iAF692@model
##get all reaction identifiers
sapply(model@reactions, id)
```

`M.tuberculosis_iNJ661` *Metabolic reconstruction of M.tuberculosis from the BiGG database*

Description

A metabolic reconstruction for tuberculosis. The dataset was generated by downloading the SBML file of the reconstruction (http://bigg.ucsd.edu/bigg/exportSelect.pl) which was subsequently converted into an object of class `SBML` using the `rsbml_read` function from the `rsbml` package.

Usage

```
data(M.tuberculosis_iNJ661)
```

Format

An sbml object of class rsbml

Details

Note that the files in the BiGG database fail the unit consistancy check of the `rsbml_read` function. To avoid unit checking when creating `SBML` objects, the substance units in the reaction tags were parsed out from the database SBML files (see example below).

Source

http://bigg.ucsd.edu/bigg/exportSelect.pl

References

Feist, A.M., Henry, C.S., Reed, J.L., Krummenacker, M., Joyce, A.R., Karp, P.D., Broadbelt, L.J., Hatzimanikatis, V., Palsson, B.O., *A genome-scale metabolic reconstruction for Escherichia coli K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information*, olecular Systems Biology, 3:121 (2007)

Examples

```
## Not run:
##The dataset was generated as follows:
##SBML_export.xml was downloaded from http://bigg.ucsd.edu/bigg/exportSelect.pl
##and a newline was added at the end of the file
file <- "SBML_export.xml"
string <- paste(readLines(file), collapse="\n")
##Parse out units to avoid validation error
string <- gsub("units=\".+?\"", "", string)
M.tuberculosis_iNJ661 <- rsbml_read(text=string)

## End(Not run)

##load data and get all reaction IDs
data(M.tuberculosis_iNJ661)
model <- M.tuberculosis_iNJ661@model
##get all reaction identifiers
sapply(model@reactions, id)
```

P.putida_iJN746 *Metabolic reconstruction of P.putida from the BiGG database*

Description

A metabolic reconstruction for P. putida. The dataset was generated by downloading the SBML file of the reconstruction (http://bigg.ucsd.edu/bigg/exportSelect.pl) which was subsequently converted into an object of class SBML using the rsbml_read function from the rsbml package.

Usage

```
data(P.putida_iJN746)
```

Format

An sbml object of class rsbml

Details

Note that the files in the BiGG database fail the unit consistancy check of the rsbml_read function. To avoid unit checking when creating SBML objects, the substance units in the reaction tags were parsed out from the database SBML files (see example below).

Source

http://bigg.ucsd.edu/bigg/exportSelect.pl

References

Feist, A.M., Henry, C.S., Reed, J.L., Krummenacker, M., Joyce, A.R., Karp, P.D., Broadbelt, L.J., Hatzimanikatis, V., Palsson, B.O., *A genome-scale metabolic reconstruction for Escherichia coli K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information*, olecular Systems Biology, 3:121 (2007)

Examples

```
## Not run:
##The dataset was generated as follows:
##SBML_export.xml was downloaded from http://bigg.ucsd.edu/bigg/exportSelect.pl
##and a newline was added at the end of the file
file <- "SBML_export.xml"
string <- paste(readLines(file), collapse="\n")
##Parse out units to avoid validation error
string <- gsub("units=\".+?\"", "", string)
P.putida_iJN746 <- rsbml_read(text=string)

## End(Not run)

##load data and get all reaction IDs
data(P.putida_iJN746)
model <- P.putida_iJN746@model
##get all reaction identifiers
sapply(model@reactions, id)
```

`Recon2` *Human metabolic reconstruction Recon2*

Description

The dataset was generated by downloading the SBML file of the reconstruction (http://www.ebi.ac.uk/biomodels-main/MODEL1109130000) which was subsequently converted into an object of class `SBML` using the `rsbml_read` function from the `rsbml` package.

Usage

```
data(Recon2)
```

Format

An sbml object of class rsbml

Source

http://www.ebi.ac.uk/biomodels-main/MODEL1109130000

References

Thiele I, Swainston N, et al.,"A community-driven global reconstruction of human metabolism", Nature Biotechnology 31, 419-425 (2013), doi:10.1038/nbt.2488

Examples

```
## Not run:
##The dataset was generated as follows:
##MODEL1109130000.xml was downloaded from http://www.ebi.ac.uk/biomodels-main/MODEL1109130000
##Recon2 <- rsbml_read("MODEL1109130000.xml")

## End(Not run)

##load data and get all reaction IDs
data(Recon2)
model <- Recon2@model
##get all reaction identifiers
sapply(model@reactions, id)
```

`rmvSpliceVariant` *Remove splicing variants from the database.*

Description

Removes alternative splicing information from the database.

Usage

```
rmvSpliceVariant(gene.info)
```

Arguments

`gene.info`	A reaction gene maping from the ReconX database created from functions `extractGeneAssociatior`

Value

A n x 2 dimentional dataframe of Reaction-Gene(Entrez number) mapping from ReconX database

Author(s)

Anand Gavai <anand.gavai@bioinformatics.nl>, Hannes Hettling

References

Thiele, I. et al. Nat Biotech, 2013

Examples

```
data(Recon2)
gene.info <- extractGeneAssociations(Recon2)

gene.info<-do.call(rbind.data.frame,gene.info)
colnames(gene.info)<-c("GPR")
gene.info$react_id<-row.names(gene.info)
gene.info<-gene.info[,c(2,1)]
```

```
rownames(gene.info)<-NULL

react_gene_map<-rmvSpliceVariant(gene.info)
```

S.aureus_iSB619 *Metabolic reconstruction of S.aureus from the BiGG database*

Description

The dataset was generated by downloading the SBML file of the reconstruction (http://bigg.ucsd.edu/bigg/exportSelect.pl which was subsequently converted into an object of class SBML using the rsbml_read function from the rsbml package.

Usage

```
data(S.aureus_iSB619)
```

Format

An sbml object of class rsbml

Details

Note that the files in the BiGG database fail the unit consistancy check of the rsbml_read function. To avoid unit checking when creating SBML objects, the substance units in the reaction tags were parsed out from the database SBML files (see example below).

Source

http://bigg.ucsd.edu/bigg/exportSelect.pl

References

Becker, S.A. and Palsson, B.O., Genome-scale reconstruction of the metabolic network in Staphylococcus aureus N315: an initial draft to the two-dimensional annotation, BMC Microbiology, 5(1):8 (2005)

Examples

```
## Not run:
##The dataset was generated as follows:
##SBML_export.xml was downloaded from http://bigg.ucsd.edu/bigg/exportSelect.pl
##and a newline was added at the end of the file
file <- "SBML_export.xml"
string <- paste(readLines(file), collapse="\n")
##Parse out units to avoid validation error
string <- gsub("units=\".+?\"", "", string)
S.aureus_iSB619 <- rsbml_read(text=string)

## End(Not run)
```

```
##load data and get all reaction IDs
data(S.aureus_iSB619)
model <- S.aureus_iSB619@model
##get all reaction identifiers
sapply(model@reactions, id)
```

S.cerevisiae_iND750 *Metabolic reconstruction of S.cerevisiae from the BiGG database*

Description

The dataset was generated by downloading the SBML file of the reconstruction (http://bigg.ucsd.edu/bigg/exportSelect.pl which was subsequently converted into an object of class SBML using the rsbml_read function from the rsbml package.

Usage

```
data(S.cerevisiae_iND750)
```

Format

An sbml object of class rsbml

Details

Note that the files in the BiGG database fail the unit consistancy check of the rsbml_read function. To avoid unit checking when creating SBML objects, the substance units in the reaction tags were parsed out from the database SBML files (see example below).

Source

http://bigg.ucsd.edu/bigg/exportSelect.pl

References

Duarte, N.C., Herrgard, M.J., and Palsson, B.O., " Reconstruction and Validation of Saccharomyces cerevisiae iND750, a Fully Compartmentalized Genome-scale Metabolic Model", Genome Research, 14: 1298-1309 (2004)

Examples

```
## Not run:
##The dataset was generated as follows:
##SBML_export.xml was downloaded from http://bigg.ucsd.edu/bigg/exportSelect.pl
##and a newline was added at the end of the file
file <- "SBML_export.xml"
string <- paste(readLines(file), collapse="\n")
##Parse out units to avoid validation error
string <- gsub("units=\".+?\"", "", string)
S.cerevisiae_iND750 <- rsbml_read(text=string)
```

```
## End(Not run)

##load data and get all reaction IDs
data(S.cerevisiae_iND750)
model <- S.cerevisiae_iND750@model
##get all reaction identifiers
sapply(model@reactions, id)
```

`sampleFluxEnsemble` *Sample a posterior ensemble of feasible flux configurations within the precision limits of given fluxes.*

Description

This function uses a Markov chain Monte Carlo algorithm to sample an ensemble of flux vectors that satisfy the constrained posed by the model. To account for inaccuracy in certain fluxes, the user can specify uncertain fluxes and provide standard deviations. The function uses the `xsample` function from the package `limSolve`.

Usage

```
sampleFluxEnsemble(model, uncertain.vars=NULL, iter=3000, ...)
```

Arguments

`model`	Either an object of class LIM as generated by `createLIMFromBiGG` or `createLIMFromSBML`, a `character` with the full path to a LIM model file or an object of class `Model` from the package `rsbml`
`uncertain.vars`	An object of class `data.frame` containing three columns: 1. The identifier for the flux(es) to be constrained within its uncertainty limits (linear combinations of fluxes e.g. F1 + F2 - F3 are also allowed), 2. the value of the constrained flux and 3. its standard deviation. If uncertain.vars is NULL, the ensemble is sampled without approximate equality constraints
`iter`	Number of iterations in the Monte Carlo procedure
`...`	Additional arguments to `xsample`

Value

A `matrix` with the posterior flux ensemble. The number of columns is equal to the number of fluxes in the provided model, the number of rows is equal to `iter`.

Note

This function is a wrapper for the function `xsample`.

Author(s)

Hannes Hettling

References

K. V. den Meersche, K. Soetaert, and D. V. Oevelen: xsample(): An R function for sampling linear inverse problems,Journal of Statistical Software, Code Snippets, vol. 30, pp. 1-15, 4 2009.

See Also

xsample

Examples

```
##get example model file of glycolysis and TCA cycle
limfile.path <- system.file("extdata", "Glycolysis_TCA.LIM",
package="BiGGR")

##Specify uncertainty of fluxes "R_GLCt1r", "R_O2t"
uncertain.vars <- data.frame(var=c("R_GLCt1r", "R_O2t"), value=c(0.4, 2.4), sd=c(0.08, 0.48))
##sample ensemble
ensemble <- sampleFluxEnsemble(limfile.path, uncertain.vars)

##Example in which linear combination of fluxes is constrained
atp.reacs <- "R_ATPS4m - R_NDPK1m - R_HEX1 - R_PFK - R_PGK + R_PYK"
uncertain.vars <- data.frame(var=atp.reacs, value=10, sd=1)
ensemble <- sampleFluxEnsemble(limfile.path, uncertain.vars)
```

---

sbml2hyperdraw *Returns a graph representation of an SBML model*

Description

Convert an SBML model to a RagraphBPH using hypergraph. Metabolites are displayed as nodes and reactions are displayed as directed edges connecting the nodes. If a vector of rates is given, edge widths are weighted according to the rates. For negative rates, edges are drawn in red and the arrow between the metabolites is reversed to represent the correct direction of the flux.

Usage

```
sbml2hyperdraw(sbml.model, rates ,relevant.species,
relevant.reactions,layoutType, lwd.max, lwd.min, plt.margins)
```

Arguments

sbml.model	an rsbml Model object
rates	a named vector with the rates of the reactions in the model. The names of the rates must agree with the reaction identifiers in the sbml.model
relevant.species	a vector of type character defining a subset of species in the sbml.model to be plotted. Defaults to all species identifiers in the sbml.model.
relevant.reactions	a vector of type character defining a subset of reactions in the sbml.model to be plotted. Defaults to all reactions identifiers in the sbml.model.

layoutType	is a character string representing the layout engine to be used for visualization. Current supported layouts are "dot", "twopi","neato","fdp","sfdp" and "circo". Defaults to "dot". See ?GraphvizLayouts for further documentation.
lwd.max	a numeric given the maximum edge width. Defaults to 3.
lwd.min	a numeric given the minimum edge width. Defaults to 0.5.
plt.margins	A numerical vector of the form c(bottom, left, top, right) giving additional white space around the graph (in case long node or edge labels fall outside the plotting region). Defaults to c(150,150,150,150).

Value

Object of class RagraphBPH with the hypergraph representation of the SBML object.

Author(s)

Hannes Hettling <j.hettling@vu.nl>, Anand K. Gavai <anand.gavai@bioinformatics.nl>

See Also

RagraphBPH hyperdraw

Examples

```
##Generate an example model
path <- system.file("extdata", "Glycolysis_TCA_recon2_reactionIDs.txt", package="BiGGR")
reaction.ids <- scan(path, what=" ")

data("Recon2")
model <- buildSBMLFromReactionIDs(reaction.ids, Recon2)

##Plot ATP and ADP in cytosol and mitochondrion in model without rates
rel.sp <- c("M_adp_c", "M_atp_c", "M_adp_m", "M_atp_m")
hd <- sbml2hyperdraw(model, relevant.species=rel.sp)
plot(hd)

##Plot model with random rates
rates <- rnorm(length(model@reactions))
names(rates) <- sapply(model@reactions, id)
hd <- sbml2hyperdraw(model, rates=rates, relevant.species=rel.sp, lwd.max=4)
plot(hd)
```

Index