Package ‘abcdeFBA’

November 5, 2011

Type Package

Title ABCDE_FBA: A-Biologist-Can-Do-Everything of Flux Balance Analysis with this package.

Version 0.1

Date 2010-09-16

Author Abhilash Gangadharan, Neha Rohatgi

Maintainer Abhilash Gangadharan <gangutalk@gmail.com>

Depends Rglpk,rgl,corrplot,lattice,R (>= 2.10)

Suggests LIM

Description Functions for Constraint Based Simulation using Flux Balance Analysis and informative analysis of the data generated during simulation.

License GPL-2

Lazyload yes

Repository CRAN

Date/Publication 2011-11-05 10:48:08

R topics documented:

BFD_Processor *BFD_Processor, a Bi-Flux-Del-Processor function*

Description

BFD_Processor reassembles the "result(n)" files created by Exhaustive_double_deletion, the result files should be a complete set from 1-n and put in a folder called BKO in the current working directory before executing this function. The output is a tab delimited spreadsheet Fatal_Double_knockouts_unique.xls

Usage

```
BFD_Processor(fba_object,EXSDR)
```

Arguments

`fba_object`	is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization,"Maximization" by default".
`EXSDR`	A list of the type generated by the Exhaustive_single_deletion function, this list may be passed to this function or the function will generate the list on its own. It is used for removing the results of lethal single knockouts which will form pairs with all other knockouts leading to false positives of double synthetic lethality.

Examples

```
#Function to process the results of the double knockout
#data(Ecoli_core)
#BFD_Processor(Ecoli_core,EXSDR=Exhaustive_single_deletion_results)
# A prompt will appear asking for the number of simulation pieces
#and also to make sure you have your results filled in a folder called BKO
```

BYPASS_REACTIONS_SUBSTRATE

BYPASS_REACTIONS_SUBSTRATE, to find the influx and efflux reactions available to a particular metabolite/substrate

Description

this function computes the production and consumption fluxes available to a particular metabolite from the fba_object and returns a list of with elements Production and Consumption

Usage

```
BYPASS_REACTIONS_SUBSTRATE(substrate_number,fba_object)
```

Arguments

`fba_object`	is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization, "Maximization" by default".
`substrate_number`	this is the metabolite number of interest which may be found by using the SEARCH_metabolite function

Examples

```
#To find the Bypass Reactions for a Substrate
data(Ecoli_core)
ATP_prod_consump<-BYPASS_REACTIONS_SUBSTRATE(17,Ecoli_core)
```

CHANGE_OBJ_FUNCTION *CHANGE_OBJ_FUNCTION, a function to change the objective for optimization*

Description

a function akin to the COBRA function to change the objective function for FBA

Usage

```
CHANGE_OBJ_FUNCTION(objective_reaction,fba_object,new_obj_weight,old_obj_weight)
```

Arguments

`fba_object` is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization,"Maximization" by default".

`objective_reaction`
a reaction number which is to be made the new objective function; retrieved using the SEARCH reaction function

`new_obj_weight`
the weight of the new objective, defaults to 1 but can be any number from 0~1

`old_obj_weight`
the weight of the old objective, defaults to 0 but can be any number from 0~(1-new_obj_weight)

Examples

```
#To change the objective function of the model.
data(Ecoli_core)
ec_new_obj<-CHANGE_OBJ_FUNCTION(11,Ecoli_core,0.5,0.5)
#ec_new_obj will be identical to the Ecoli_core model except that
#the objective function would change
FBA_solve(ec_new_obj)
```

CHANGE_RXN_BOUNDS *CHANGE_RXN_BOUNDS, Change Reaction Bounds*

Description

this function helps to modify the bounds on the fba_object, it returns an object of the same type as the model, inspired by the COBRA-function of the same name

Usage

```
CHANGE_RXN_BOUNDS(reaction_number,fba_object,lower_bound,upper_bound)
```

Arguments

`fba_object` is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization,"Maximization" by default".

`lower_bound` the new value of the lower bound

`upper_bound` the new value of the upper bound

`reaction_number` the reaction number of the reaction the bounds of which have to be changed

Examples

```
#Changing Reaction Bounds, to simulate the deletion/inhibition of a reaction.
data(Ecoli_core)
Ec_mutant<-CHANGE_RXN_BOUNDS(reaction_number=36,fba_object=Ecoli_core,
lower_bound=0,upper_bound=0)
```

DEGREE_MEASURE *DEGREE_MEASURE, measures the in-degree and out-degree of the metabolites in the network.*

Description

this is a simple function to determine the degree measure, it uses the fba_object and computes and writes the measures to disk as a tab-separated spreadsheet

Usage

```
DEGREE_MEASURE(fba_object,file="Degree_measure")
```

Arguments

fba_object	is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization,"Maximization" by default".
file	a filename for the tab delimited output file which is generated by this function

Examples

```
#Determining the Degree Measure of the Core E.coli Metabolic Network
data(Ecoli_core)
DEGREE_MEASURE(fba_object=Ecoli_core)
```

Ecoli_core.rda *E.coli core model 72 reactions and 95 metabolites*

Description

This "list" form of the E.coli core model was created from the S4 object created by readSBMLmod from the package Sybil using the Sybil_2_FBA_obj function.

Usage

```
Ecoli_core
```

Format

A list containing the information required to perform FBA and annotations for intelligible output.

Source

BiGG database

References

Systems Biology Properties of Reconstructed Networks

`Ec_iAF1260.rda` *E.coli model 1260 ORF's*

Description

This "list" form of the E.coli model iAF1260 was created from the S4 object created by readSBMLmod from the package Sybil using the Sybil_2_FBA_obj function.

Usage

```
Ec_iAF1260
```

Format

A list containing the information required to perform FBA and annotations for intelligible output.

Source

BiGG database

References

Feist AM, Henry CS, Reed JL, Krummenacker M, Joyce AR, Karp PD, Broadbelt LJ, Hatzimanikatis V and Palsson BO, A genome-scale metabolic reconstruction for Escherichia coli K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information, Molecular Systems Biology.

`Ec_iAF1260_flux2` *E.coli model 1260 ORF's*

Description

This "list" form of the E.coli model iAF1260 was created from the S4 object created by readSBMLmod function from the package Sybil using the Sybil_2_FBA_obj function.

Usage

```
Ec_iAF1260_flux2
```

Format

A list containing the information required to perform FBA and annotations for intelligible output.

Source

BiGG database

References

Feist AM, Henry CS, Reed JL, Krummenacker M, Joyce AR, Karp PD, Broadbelt LJ, Hatzimanikatis V and Palsson BO, A genome-scale metabolic reconstruction for Escherichia coli K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information, Molecular Systems Biology.

`Ec_iJR904` *E.coli model 904 ORF's*

Description

This "list" form of the E.coli model iJR904 was created from the S4 object created by readSBMLmod from the package Sybil using the Sybil_2_FBA_obj function.

Usage

```
Ec_iJR904
```

Format

A list containing the information required to perform FBA and annotations for intelligible output.

Source

BiGG database

References

Reed JL,Vo TD, Schilling CH and Palsson BO, An expanded genome-scale metabolic reconstruction for Escherichia coli K-12 (iJR904 GSM/GPR) Genome Biology 2003

`Exhaustive_double_deletion` *Exhaustive_double_deletion, a function for computing synthetic double knockouts.*

Description

Exhaustive_double_deletion enables parallel computing of double knockouts by splitting and running the simulation as different instances on the same multi-core machine. The results of the simulation are stored into files named result1, result2 etc.which may be reassembled using the BFDProcessor function

Usage

```
    Exhaustive_double_deletion(fba_object,thread_no,core_number)
```

Arguments

`fba_object` is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization,"Maximization" by default".

`thread_no` This specifies the thread number, it is useful to think of the thread number as a chunk of the complete combination of all pairs of reactions that can be formed. This depends on the number of the cores you are employing which must obviously be a fixed number. For instance if you have a 40 core computer you can divide the double-knockout simulation into 40 chunks. In that case the thread number would refer to the chunks of simulation as 0-39, with thread 0 performing the first 1/40th part of the simulation

`core_number` core_number specifies the number of cores that are available to you for performing double knockout simulations. It should be a fixed number for a particular instance of simulations on one model

Examples

```
#Performing double knockouts in a 2 core computer
data(Ecoli_core)
Exhaustive_double_deletion(Ecoli_core,0,2)
#New instance of R
data(Ecoli_core)
Exhaustive_double_deletion(Ecoli_core,1,2)
```

Exhaustive_single_deletion

Exhaustive_single_deletion, a function that deletes each reaction in the network one at a time and returns results describing reaction lethality

Description

This function takes the argument of type fba_object which would be a FBA object and performs an exhaustive deletion of all the reactions in the reaction network and returns a list of biomass generated for all the deletions, the sub-optimal deletions, the super-optimal deletions and non-lethal deletions, also generates a pdf containing a distribution of the fatal reactions according to their sub-systems and a histogram of the biomass distribution for each deletion.

Usage

```
Exhaustive_single_deletion(fba_object)
```

Arguments

`fba_object` Is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization,"Maximization" by default.

Examples

```
#Performing an exhaustive single reaction deletion
data(Ecoli_core)
Results<-Exhaustive_single_deletion(Ecoli_core)
```

`FBA_solve` *FBA_solve, a function to solve CBM problems*

Description

This function sugar-coats Rglpk_solve_LP which is a function provided by the Rglpk package, FBA_solve solves FBA problems using Rglpk_solve_LP and gives the solver output in a CBM context including solver error messages and graceful degradation.

Usage

```
    FBA_solve(fba_object, precision,verbosity,maximize)
```

Arguments

`fba_object` is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization,"Maximization" by default".

`precision` A number indicating the precision of the flux solution after the decimal point, defaults to 6

`verbosity` is a Boolean indicating if the verbose output of the LP solver should be displayed during simulation, defaults to FALSE

`maximize` is a Boolean that can over-ride the default mode of optimization (maximization) and minimize if FALSE

Examples

```
# Flux Balance Analysis performed on a core-metabolism model of E.coli
data(Ecoli_core)
FBA_solve(fba_object=Ecoli_core,precision=6,verbosity=FALSE,maximize=TRUE)
```

flux_difference_plotter

flux_difference_plotter, a function to plot two pre-existing flux distributions obtained using FBA_solve

Description

To analyse the effect of reaction deletions on the fluxome requires a contextual visualization, simple plots give little insight on what the results of the simulation mean. This function uses the annotations inherent to SBML models and generates comparative overlapping fluxome bar graphs depicting the overlap/change in flux based on Sub-system wise classification and generates PDF's for the same. The color scheme is green for wild-type and red for mutant. Overlaps of red and green generate brown while overshot mutant fluxes show up as magenta-pink, also separate PDF's are generated for increased and decreased fluxes.

Usage

```
flux_difference_plotter(wt_flux,mut_flux,fba_object,graph_fname)
```

Arguments

`fba_object`	is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization,"Maximization" by default".
`wt_flux`	A list containing the solution to an FBA problem which is returned by the FBA_solve function. By convention the fluxes in this list represent the Wild-type strain and will appear as green bar plots.
`mut_flux`	A list containing the solution to an FBA problem which is returned by the FBA_solve function. By convention the fluxes in this list represent the Mutant-type strain and will appear as red bar plots.
`graph_fname`	A string to name the output files

Examples

```
# A comparison of two flux distributions generated by FBA_solve
data(Ecoli_core)
Ec_Mutant<-Ecoli_core
#The reaction number for O2 exchange is 36 by setting
#the corresponding bounds to zero we make a aerobic
#respiration deficient mutant

Ec_Mutant$bounds$lower$val[36]=0
Ec_Mutant$bounds$upper$val[36]=0
mut_flux<-FBA_solve(Ec_Mutant)
wt_flux<-FBA_solve(Ecoli_core)
flux_difference_plotter(wt_flux,mut_flux,Ecoli_core,graph_fname="Flux_comparison")
```

FLUX_VAR_ANALYSIS *FLUX_VAR_ANALYSIS, a function to perform a flux variability analysis.*

Description

FVA can indicate the decrease in network robustness caused by non-lethal deletions.

Usage

```
FLUX_VAR_ANALYSIS(fba_object, filename)
```

Arguments

fba_object	Is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization,"Maximization" by default.
filename	A string which will be the name of the file containing the output of the FVA simulation

Examples

```
#Simple flux variability Analysis
data(Ecoli_core)
FLUX_VAR_ANALYSIS(Ecoli_core, filename="Wt_FVA.xls")
```

FVA_robustness *FVA_robustness, a function to determine the change in robustness of the network caused by a mutation.*

Description

FVA_robustness builds upon FLUX_VAR_ANALYSIS to give the user a convenient function to examine the change in network robustness caused by a user supplied mutation. Graphical results for absolute flux span comparisons may be expected in the working directory

Usage

```
    FVA_robustness(fba_object, mutation)
```

Arguments

`fba_object`	Is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization,"Maximization" by default".
`mutation`	A reaction number obtained using the SEARCH_reaction, this reaction will be deleted during the FVA analysis

Examples

```
#Mutant network robustness, removing Formate Exchange which has no effect on fluxes
data(Ecoli_core)
FVA_robustness(Ecoli_core,25)
```

H_pylori_iIT341 *Helicobacter pylori by Ines Thiele*

Description

This "list" form of the H. pylori model iIT341 was created from the S4 object created by readSBMLmod from the package Sybil using the Sybil_2_FBA_obj function.

Usage

```
H_pylori_iIT341
```

Format

A list containing the information required to perform FBA and annotations for intelligible output.

Source

BiGG database

References

Thiele I, Vo TD, Price ND, Palsson BO, Expanded Metabolic Reconstruction of Helicobacter pylori (iIT341 GSM/GPR): an In Silico Genome-Scale Characterization of Single- and Double-Deletion Mutants, Journal of Bacteriology.

`H_sapien_Recon1` *H. sapien Recon1, human metabolic network reconstruction*

Description

This "list" form of the human model Recon1 was created from the S4 object created by readSBMLmod from the package Sybil using the Sybil_2_FBA_obj function.

Usage

```
H_sapien_Recon1
```

Format

A list containing the information required to perform FBA and annotations for intelligible output.

Source

BiGG database

References

Duarte NC, Becker SA, Jamshidi N, Thiele I, Mo ML, Vo TD, Srivas R and Palsson BO, Global reconstruction of the human metabolic network based on genomic and bibliomic data, PNAS.

`M_barkeri_iAF629` *Genome scale metabolic model for the archaeal methanogen M. Barkeri*

Description

This "list" form of the M.Barkeri model was created from the S4 object created by readSBMLmod from the package Sybil using the Sybil_2_FBA_obj function.

Usage

```
M_barkeri_iAF629.rda
```

Format

A list containing the information required to perform FBA and annotations for intelligible output.

Source

BiGG database

References

Modeling methanogenesis with a genome scale metabolic reconstruction of Methanosarcina barkeri by Feist AM, Scholten JCM, Palsson BO

`M_tb_iNJ661` *M. tuberculosis model iNJ661*

Description

This "list" form of the M.tb model iNJ661 was created from the S4 object created by readSBMLmod from the package Sybil using the Sybil_2_FBA_obj function.

Usage

```
M_tb_iNJ661
```

Format

A list containing the information required to perform FBA and annotations for intelligible output.

Source

BiGG database

References

Bordbar A, Lewis NE, Schellenberger J, Palsson BO, Jamshidi N, Insight into human alveolar macrophage and M. tuberculosis interactions via metabolic reconstructions, Molecular Systems Biology.

`PERTURBATION_analysis`

PERTURBATION_analysis, a function for robustness analysis.

Description

Robustness analysis is described procedurally in the COBRA-ToolBox manual. This function encodes the basic principle of the procedure and returns the result as an X versus Y list and generates a plot on successful completion.

Usage

```
PERTURBATION_analysis(reaction_number,fba_object)
```

Arguments

`fba_object`
: is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization,"Maximization" by default".

`reaction_number`
: Reaction number of the reaction to be ramped down. This number may be obtained by using the function SEARCH_reaction for a text search through the FBA_obj-reaction list.

Examples

```
#Perturbation analysis of fluxes
data(Ecoli_core)
Ec_xy_O2<-PERTURBATION_analysis(reaction_number=36,fba_object=Ecoli_core)
```

PHPP *PHPP, phenotypic phase plane analysis*

Description

this function helps in performing a phenotypic phase plane analysis, a visualization of the effect of two input fluxes on the value of the objective function.

Usage

```
PHPP(reaction_number,fba_object,PCS,flux_range)
```

Arguments

`fba_object` is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization, "Maximization" by default".

`PCS` a string to search for the primary carbon source of the model, for example PCS="D glucose" will display all the reactions with that string in them, you need to choose the appropriate reaction number of the Carbon source before continuing. In case you are performing a PhPP for alternate carbon sources then the primary carbon source is supposed to be shut-down to get a correct picture of the flux cone.

`reaction_number` two reaction numbers that specify the two input fluxes that make the x and y axes of the PhPP

`flux_range` the range between which the input fluxes are to be varied

Examples

```
#Performing a phenotypic phase plane analysis of
#glucose and oxygen in Core E.coli Metabolism
data(Ecoli_core)
PHPP(reaction_number=c(28,36),fba_object=Ecoli_core,
PCS="glucose",flux_range=c(1,15))
# a menu pops up asking to select the primary carbon source, select D glucose for Ecoli_core
```

SEARCH_metabolite *SEARCH_metabolite, a function to search for metabolites in a model using a simple text query*

Description

this function helps to search for metabolites in a model using a simple text query, it returns a list of possible hits along with their metabolite numbers and locations in the compartments of the model

Usage

```
    SEARCH_metabolite(metabolite_name,fba_object)
```

Arguments

fba_object is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization, "Maximization" by default".

metabolite_name is a character string containing a part or the complete name of the metabolite to be searched for in the fba object

Examples

```
#To search for the metabolite in the model
data(Ecoli_core)
SEARCH_metabolite("ATP",Ecoli_core)
```

SEARCH_reaction *SEARCH_reaction, a function to search for reactions in a model using a simple text query*

Description

this function helps to search for reactions in a model using a simple text query, it returns a list of possible hits along with their reaction numbers, these reaction numbers are to be used with other perturbation/knockout/optimality/robustness analysis functions

Usage

```
SEARCH_reaction(react_name,fba_object)
```

Arguments

`fba_object` is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization,"Maximization" by default".

`react_name` is a character string containing a part or the complete name of the reaction to be searched for in the fba object

Examples

```
#Search for reactions involving glucose
data(Ecoli_core)
SEARCH_reaction("glucose",Ecoli_core)
```

SINGLE_DEL_FLUXOME *SINGLE_DEL_FLUXOME, a function to generate comparative fluxome graphs*

Description

To analyse the effect of reaction deletions on the fluxome requires a contextual visualization, simple plots give little insight on what the results of the simulation mean. This function uses the annotations inherent to SBML models and generates comparative overlapping fluxome bar graphs depicting the overlap/change in flux based on Sub-system wise classification and generates PDF's for the same. The color scheme is green for wild-type and red for mutant. Overlaps of red and green generate brown while overshot mutant fluxes show up as magenta-pink, also separate PDF's are generated for increased and decreased fluxes.

Usage

```
SINGLE_DEL_FLUXOME(fba_object, deletion_number)
```

Arguments

`fba_object` is a list containing the data required to perform flux balance analysis. The elements of the list are mat which is the stoichiometric matrix, dir which gives the direction of the equality constraints, obj specifies the objective function for the simulation, bounds specifies the lower and upper inequality constraints, rhs is the right hand side of the steady state expression, types refers to the numeric nature of the variables which in case of FBA happens to be "Continuous", max is a Boolean specifying the type of optimization, "Maximization" by default".

`deletion_number`
: Reaction number of the reaction to be deleted. This number may be obtained by using the function SEARCH_reaction for a text search through the fba_object-reaction list.

Examples

```
#Compare the flux distributions of the mutant with the wild-type
data(Ecoli_core)
SINGLE_DEL_FLUXOME(Ecoli_core,36)
```

Sybil_2_FBA_obj *Sybil_2_FBA_obj, a function to convert a model generated by Sybil into the one used by abcdeFBA.*

Description

Changes the S4 object read by Sybil which is a non-standard R package into a list usable by abcdeFBA.

Usage

```
    Sybil_2_FBA_obj(Sybil_S4_object)
```

Arguments

`Sybil_S4_object`
: Is an S4 object of class modelorg

Examples

```
#Model conversion
#data(Ecoli_core) - # the Ecoli_core model included in the Sybil package
#FBA_obj<-Sybil_2_FBA_obj(Ecoli_core)
```

S_aureus_iSB619 *Staphylococcus aureus model iSB619 by S.Becker*

Description

This "list" form of the S. aureus model iSB619 was created from the S4 object created by readSBMLmod from the package Sybil using the Sybil_2_FBA_obj function.

Usage

```
S_aureus_iSB619
```

Format

A list containing the information required to perform FBA and annotations for intelligible output.

Source

BiGG database

References

Becker SA, Palsson BO, Genome-scale reconstruction of the metabolic network in Staphylococcus aureus N315: an initial draft to the two-dimensional annotation

---

S_cerevisiae_IND750 *Saccharomyces Cerevisiae iND750*

Description

This "list" form of the yeast model iND750 was created from the S4 object created by readSBMLmod from the package Sybil using the Sybil_2_FBA_obj function.

Usage

```
S_cerevisiae_IND750
```

Format

A list containing the information required to perform FBA and annotations for intelligible output.

Source

BiGG database

References

Duarte NC ,Herrgard MJ , Palsson BO, Reconstruction and Validation of Saccharomyces cerevisiae iND750, a Fully Compartmentalized Genome-Scale Metabolic Model, Genome Research