How to use the BiGGR package

Anand K. Gavai & Hannes Hettling

1 Introduction

The main purpose of this package is to analyze metabolic systems and estimate the biochemical reaction rates in metabolic networks. BiGGR works with the BiGG [1] database and with files encoded in the Systems Biology Markup Language (SBML) from other sources. The BiGG database stores reconstructions of metabolic networks and is freely accessible. BiGGR is an entirely open source alternative for a more extensive software package, COBRA 2.0, which is available for Matlab [2]. BiGGR makes it easy to apply a big variety of open source R packages to the analysis of metabolic systems. Although it contains less functionality than COBRA, BiGGR may be convenient for R users. The BiGG system provides metabolic reconstructions on humans, *M. barkeri*, *S. cerevisiae*, *H. pylori*, *E. coli* and *S. aureus*. BiGGR also works with the new reconstruction of human metabolism Recon 2 [3]. These reconstructions consist of genes, metabolites, reactions and proteins that are identified and connected with each other to form a network structure. The BiGGR package provides various functions to interface to the BiGG database, and to perform flux balance analysis (FBA) after importing selected reactions or pathways from the database. Other functions included in this package allow users to create metabolic models for computation, linear optimization routines, and likelihood based ensembles of possible flux distributions fitting measurement data. To this end, BiGGR interfaces with the LIM package [4]. BiGGR provides models in standard SBML R object format for each organism within the BiGG database as well as the new reconstruction of human metabolism from the Biomodels database [3] (see 'data' directory in the package). This format allows easy construction of the stoichiometric matrix of the entire system which may serve as the core of further computational analysis. Finally, the package allows automatic visualization of reaction networks based on a hypergraph framework using the hyperdraw [5] package.

2 Installation

BiGGR is installed as follows from the R console:

```
> if (!requireNamespace("BiocManager", quietly=TRUE))
+     install.packages("BiocManager")
```

```
> BiocManager::install("BiGGR")
```

BiGGR depends on the Bioconductor packages rsbml [6], hyperdraw [5] (which in turn requires the package hypergraph) and the CRAN package LIM [4]. For detailed installation instructions of the dependencies we refer to the package documentations at `http://www.bioconductor.org/` and `http://www.cran.r-project.org/`.

3 Example: A flux balance analysis with BiGGR

The package is imported as follows:

```
> library("BiGGR")
```

To get an overview about the functions and databases available in the package, you can use:

```
> library(help="BiGGR")
```

The reference manual which describes all functions of BiGGR in detail can be found in the documentation directory ('doc') of the package. In the following we will provide a step-by-step guide demonstrating a flux estimation procedure in a model of glycolysis and TCA cycle. The general work flow using this package consists of the following steps:

- Retrieve a model in SBML object format as provided in the package (alternatively an R object containing the model can be generated from an SBML file)
- Specify optimization objective and model constraints and create a LIM model file as input for the linear programmimg package LIM
- Estimate the reaction fluxes with linear programming
- Visualize the results using the hypergraph framework

3.1 Generate Model

There are several ways to create a model within BiGGR:

- Query one of the databases contained in the BiGGR package (use the command `data()` to see all available databases). You can query with a list of genes (function `buildSBMLFromGenes`), a list of reaction identifiers (`buildSBMLFromReactionIDs`) or for specific pathways (`buildSBMLFromPathways`).

- Alternatively: Retrieve a text file with metabolic reactions from the web interface of the BiGG database (`http://bigg.ucsd.edu/bigg/main.pl`). The user can query and select reactions from BiGG which can then be exported in SBML or text format. BiGG reactions saved in text format can be converted to an internal SBML object by the function `buildSBMLFromBiGG`. An SBML file can be imported using the `rsbml_read` function from the rsbml package.

Below we will demonstrate how to build an SBML model from a set of reaction identifiers using the Recon 1 database. The list of reaction IDs can be found in the extdata subdirectory in the package. The model comprises the reactions of glycolysis, pentose-phosphate pathway and TCA cycle [7]:

```
> ##load reaction identifiers from package examples
> file.name <- system.file("extdata",
+                          "brainmodel_reactions.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)
```

The model object `sbml.model` is an rsbml object of class `Model`. It has 92 metabolites participating in 73 reactions in 3 compartments.

3.2 Specify constraints, optimization objective and estimate fluxes

After building the model, we specify additional parameters necessary to run the flux estimation. In the present model, several metabolites are unbalanced because not all the biochemical reactions involving them are represented inside the model. Another unbalanced situation is when metabolites accumulate inside or outside the cell. These metabolites must therefore not be subjected to the equality constraints (i.e. the steady state constraint) of the linear programming routine for flux estimation. These metabolites are termed external metabolites or, in short, externals. The objective of this flux balance analysis is to maximize the net ATP production in the reaction network given the constraints in the model. Note that, of course, also minimizing a linear function of fluxes in the model is possible in BiGGR ('loss' function as opposed to 'profit' function). Below we specify the objective function and the external metabolites.

```
> ##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_e", "M_h2o_e", "M_co2_e",
```

```
+                 "M_o2_e", "M_h_e", "M_o2s_m",
+                 "M_adp_c", "M_atp_c", "M_pi_c",
+                 "M_h_c", "M_nadp_c", "M_nadph_c",
+                 "M_na1_c", "M_na1_e", "M_gln_DASH_L_c",
+                 "M_nh4_c", "M_pyr_e")
```

Additional equality and inequality constraints can be given for fluxes for which the values are known beforehand, e.g. if they rely on experimental measurements. Below we use measurements of cerebral metabolic substrate uptake and release rates in human brain [8]. BiGGR also alows for setting equality constraints on fluxes relative to other fluxes. Based on the observation that the GABA shunt accounts for 32% of the total glucose oxidation in the brain [9] and that in the pentose phosphate pathway flux in brain amounts to 6.9% of glycolysis [10], we constrain fluxes for GABA shunt and the entry reaction into the pentose phosphate pathway accordingly. Equality and inequality constraints are given as lists in the variables `eqns` and `ineq`. Finally a LIM model file is created using the function `createLIMFromSBML`.

```
> ##load lying-tunell data
> data(lying.tunell.data)
> ##set equality constraints
> equation.vars <- c("R_GLCt1r", "R_L_LACt2r", "R_GLNtN1",
+                    "R_PYRt2r", "R_GLUDC", "R_G6PDH2r")
> equation.values <- c(as.character(
+     lying.tunell.data[c("glucose", "lactate", "glutamine", "pyruvate"),
+                       "median"]),
+                      "R_GLCt1r * 0.32", "R_GLCt1r * 0.069" )
> eqns <- list(equation.vars, equation.values)
> ##write LIM file to system's temporary directory
> limfile.path <- tempfile()
> createLIMFromSBML(sbml.model, maximize, equations=eqns,
+                   externals=externals, file.name=limfile.path)
```

3.3 Running simulations to estimate fluxes

BiGGR uses Linear Inverse Models for estimating the fluxes as provided by LIM. All the functionality of this package can be used in this framework. The function `lsei` in LIM provides least squares estimation with equalities and inequalities which is useful to fit the model to biochemical measurements of metabolite exchange. The interface to LIM's `lsei` in BiGGR is `getRates` which takes the model file (or a LIM object) as an input parameter to estimate the fluxes with respect to the objective function.

```
> rates <- getRates(limfile.path)
```

3.4 Sampling of feasible flux distributions

Experimentally quantified fluxes are always subject to measurement error. In the above example, the rates for, among others glucose and glutamine uptake (R_GLCt1r and R_GLUDC, respectively) and uptake of lactate and pyruvate (R_L_LACt2r and R_PYRt2r) were fixed. However, it is of interest how the estimated fluxes vary if measurement error on the known fluxes is taken into account. BiGGR provides the functionality to calculate the uncertainty of all estimated fluxes by performing a random walk in the feasible flux space with a Markov chain Monte Carlo (MCMC) method. To this end, BiGGR provides an interface to the `xsample` function from the package limSolve [11]. Ensembles of feasible flux vectors within the precision limits of the known fluxes can be sampled with the function `sampleFluxEnsemble`. As an example, we will sample an ensemble of 10000 flux vectors within the precision limits of the data [8] given as the standard deviation. As 'burn-in', we use 10000 Monte-Carlo steps and we include each 10^{th}. Thus, in total, $2 * 10^7$ steps are taken. Please note that this may take a long time, depending on your machine. Starting point for the random walk is the previously optimized flux vector. For quicker convergence of the MCMC procedure, we set the jump length manually (see `?sampleFluxEnsemble` for details).

```
> ##specify the fluxes with uncertainty given as SD in a data frame
> uncertain.vars <- data.frame(var=equation.vars[1:4],
+                              value=equation.values[1:4],
+                              sd=c(0.058,0.032,0.034,0.004))
> uncertain.vars <- data.frame(var=c(equation.vars[c(1,2,3,4)]),
+                              value=as.numeric(c(equation.values[c(1,2,3,4)])),
+                              sd=lying.tunell.data[c("glucose",
+                                  "lactate", "glutamine", "pyruvate"), "sd"])
> limfile.path.ens <- tempfile()
> ##Create new LIM model
> equations <- list(c("R_G6PDH2r", "R_GLUDC", "R_G3PD2m") ,
+                   c("R_GLCt1r * 0.069", "R_GLCt1r * 0.32", "0"))
> createLIMFromSBML(sbml.model, maximize, externals=externals,
+                   file.name=limfile.path.ens, equations=equations)
> ##sample feasible flux distributions with MCMC
> ensemble <- sampleFluxEnsemble(limfile.path.ens, uncertain.vars,
+                                x0=rates, iter=1e5, burninlength=1e4,
+                                outputlength=1e4, type="mirror", jmp=0.1)
```

The sampled posterior distributions can then simply be plotted as histograms as shown in figure 1 for selected fluxes. Furthermore, it is now possible to assess the effect of possible measurement error in R_GLCt1r and R_O2t on other fluxes present in the system. As an example, we calculate the net rate of ATP production for the whole ensemble from the linear flux combination R_ATPS4m - R_NDPK1m - R_HEX1 - R_PFK - R_PGK + R_PYK. Note that the net ATP production was the profit function of the flux balance analysis presented above.

```
> par(mfrow=c(2,2))
> metab <- c(as.vector(uncertain.vars[1:2,1]), "R_SUCD1m")
> for (m in metab){
+   title <- paste(m, "\n(", sbml.model@reactions[[m]]@name, ")", sep="")
+   myhist <- hist(ensemble[,m], breaks=9, plot=FALSE)
+   plot(myhist, ylim=c(0, max(myhist$counts) + max(myhist$counts / 10)),
+        xlab="flux (mmol/min)",main=title, col="cornflowerblue", cex.lab=1.3,
+        xlim=c(min(myhist$breaks) - sd(myhist$breaks),
+            max(myhist$breaks)+sd(myhist$breaks)))
+   text(mean(myhist$mids), max(myhist$counts) + max(myhist$counts / 10),
+        label=bquote(mu== ~.(round(mean(ensemble[,m]),3)) ~
+            "," ~ sigma== ~.(round(sd(ensemble[,m]),3))), cex=1.2)
+ }
> ## get ensemble of net ATP production
> atp.prod.ens <- eval(parse(text=maximize), envir=data.frame(ensemble))
> ##plot ensemble
> title <- paste("Net ATP production")
> myhist <- hist(atp.prod.ens, breaks=9, plot=FALSE)
> plot(myhist, ylim=c(0, max(myhist$counts) +
+                  max(myhist$counts / 10)), xlab="flux (mmol/min)",
+      main=title, col="cornflowerblue", cex.lab=1.3,
+      xlim=c(min(myhist$breaks) - sd(myhist$breaks),
+          max(myhist$breaks)+sd(myhist$breaks)))
> text(mean(myhist$mids), max(myhist$counts) + max(myhist$counts / 10),
+      label=bquote(mu== ~.(round(mean(atp.prod.ens),3)) ~
+          "," ~ sigma== ~.(round(sd(atp.prod.ens),3))), cex=1.2)
```

The spread in the rates of net ATP production is given in the last histogram in Figure 1. In this way, the uncertainty of the objective function value can be investigated with respect to possible measurement noise of the fluxes in the model.

3.5 Visualization of networks and fluxes

BiGGR provides visualization using hypergraphs. To this end, BiGGR uses the package hyperdraw which in turn uses the Graphviz engine. Hypergraphs are graphs which can connect multiple nodes by one edge. Metabolites are represented by nodes and reactions are represented by edges connecting the nodes. The fluxes of the biochemical reactions can be represented by the width of the edges (a wider edge corresponds to a higher flux value). An SBML model can be converted into a hyperdraw object using the function `sbml2hyperdraw`. Since many models contain numerous metabolites and reactions, a 'human readable' automatic graphical representation of the system in one single plot is often infeasible. Therefore, specific subsets of metabolites and/or reactions can be passed as an argument to the `sbml2hyperdraw` function and only metabsolites or reactions belonging to the specified sets are visualized. Below we will visualize

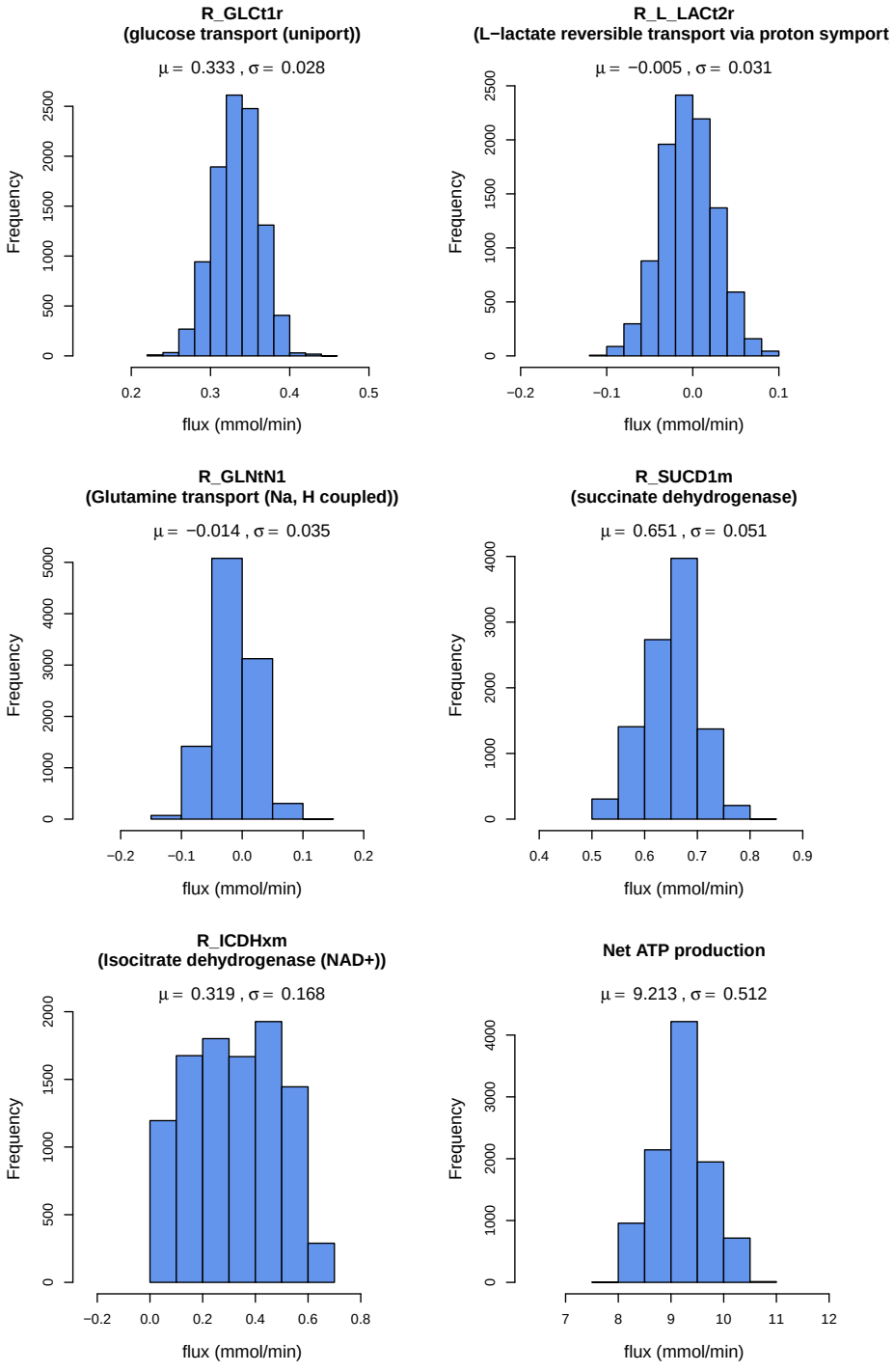

Figure 1: *Posterior distributions of delected fluxes and the net ATP production rate after the sampling with Markov Chain Monte Carlo.*

a subset of metabolites and reactions in the glycolytic pathway and parts of the pentose phosphate pathway, which is a subset of our example model. As a second argument we pass the reaction rates calculated in 3.3 in order to represent the reaction rates by the width of the edges.

```
> 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",
+                       "M_6pgl_c", "M_6pgc_c", "M_ru5p_DASH_D_c",
+                       "M_xu5p_DASH_D_c", "M_r5p_c", "M_g3p_c", "M_s7p_c")
> relevant.reactions <- c("R_HEX1", "R_PGI", "R_PFK", "R_FBA", "R_TPI",
+                         "R_GAPD", "R_PGK", "R_PGM", "R_ENO", "R_PYK",
+                         "R_G6PDH2r", "R_PGL", "R_GND", "R_RPE", "R_RPI", "R_TKT1")
> hd <- sbml2hyperdraw(sbml.model, rates=rates,
+                      relevant.species=relevant.species,
+                      relevant.reactions=relevant.reactions,
+                      layoutType="dot", plt.margins=c(20, 0, 20, 80))
```

The hypergraph object can then simply be plotted using the plot function:

```
> plot(hd)
```

The resulting plot is shown in Figure 2. Flux values are displayed following each reaction identifier. The forward direction is defined in the BiGG database according to biochemical conventions, but if the actual calculated flux is backwards accroding to the definition the arrow is colored red. Additional graphical arguments are documented in the help file (see `?sbml2hyperdraw`).

Below, we give various reactions and metabolites in the TCA cycle which are present in our example model and plot all components using a circular layout (see Figure 3):

```
> relevant.species <- c("M_cit_m", "M_icit_m" , "M_akg_m",
+                       "M_succoa_m", "M_succ_m", "M_fum_m",
+                       "M_mal_DASH_L_m", "M_oaa_m")
> relevant.reactions <- c("R_CSm", "R_ACONTm", "R_ICDHxm",
+                         "R_AKGDm", "R_SUCOAS1m", "R_SUCD1m",
+                         "R_FUMm", "R_MDHm", "R_ICDHyrm", "R_ME1m",
+                         "R_ME2m", "R_ASPTAm","R_AKGMALtm", "R_GLUDym",
+                         "R_ABTArm", "R_SSALxm","R_CITtam")
> hd <- sbml2hyperdraw(sbml.model, rates=rates,
+                      relevant.reactions=relevant.reactions,
+                      relevant.species=relevant.species,
+                      layoutType="circo", plt.margins=c(150, 235, 150, 230))
> dev.new() ##Open a new plotting device
> plot(hd)
```

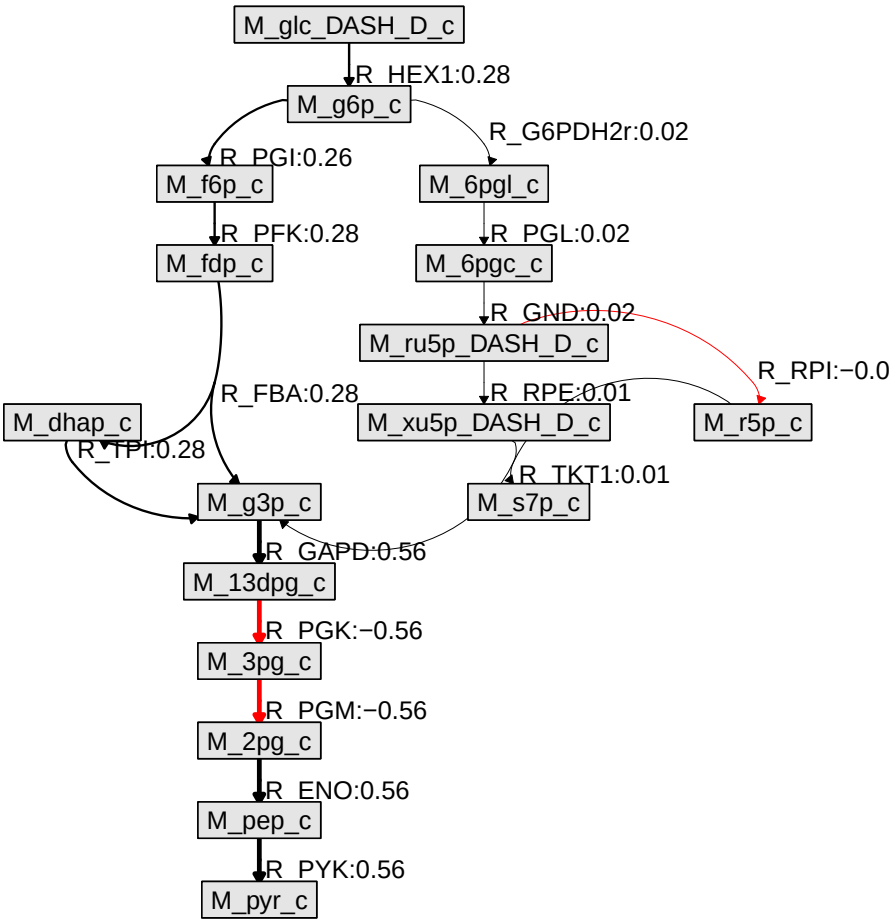

Figure 2: *Estimated fluxes in the glycolytic pathway and parts of the pentose phosphate pathway. For each reaction, the arrow points in the direction of the calculated flux. If that is backward relative to the direction defined as forward in the metabolic reconstruction, the arrow is colored red. Note that only a subset of all metabolites and reactions is plotted.*

In this example, reactions with a flux equal to zero are displayed in grey. Note that metabolites which are not specified are not plotted, even if reactions in which they participate are drawn. This is for instance the case for the exchange reaction below:

```
M_akg_m + M_mal_DASH_L_c -> M_akg_c + M_mal_DASH_L_m
```

The visualization function `sbml2hyperdraw` is not restricted to FBA models, but `sbml2hyperdraw` can be used as a generic plotting function for SBML models.

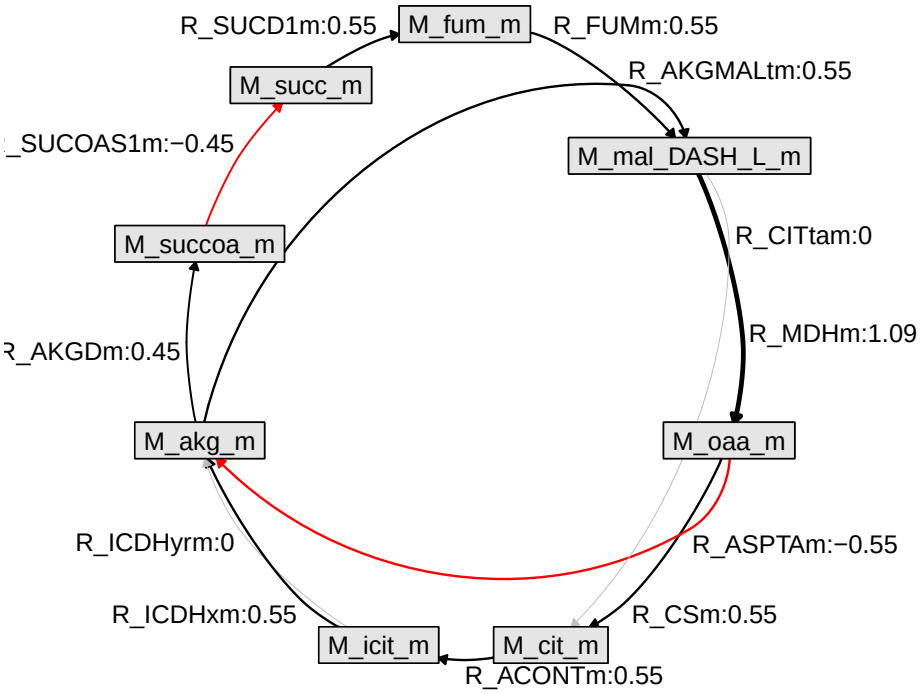

Figure 3: *Estimated fluxes in the citric acid cycle in the mitochondrion.*

To this end, in case that no reaction rates are given as argument, all edges are plotted with the same width and in the same color.

4 Troubleshooting BiGGR

Model building is an iterative process and requires careful selection of parameters and arguments. Some of the most common problems and solutions are described below:

- **Infeasible solution:** This problem can be encountered when using the `linp` method form the LIM package. This problem occurs when the constraints provided by the user for the model are conflicting. (A trivial ex-

ample is that a constraint says that a specific flux is greater than 5 units and another constraint says the same flux is smaller than 4. Such conflicts can be much more subtle). The reactions in the model file may sometimes be defined incorrectly, for instance with regard to their reversibility.

- **Visualizing too many metabolites and reactions:** If the plotting area is too small to fit all boxes for metabolites, the following error is produced by the hyperdraw package:

  ```
  Error in `[.unit`(pts$x, ref + step) :
  Index out of bounds (unit subsetting)
  ```

  In case you encounter this error when plotting your model, you can consider several possibilities:

  - Increase the size of the plotting area: When plotting to the screen, width and height of the plotting window can be set with the `x11()` command. Type `?x11` for more information. Similarly, figure dimensions can be set when plotting to a jpeg, png, pdf, eps etc. device. Type for instance `?pdf` for the documentation.
  - Consider plotting only a subset of the metabolites and reactions in the model. It is possible to pass a list or vector of relevant species and/or relevant reactions to the function `sbml2hyperdraw`. See `?sbml2hyperdraw` for more information.

- **Resizing the plotting window:** Resizing the plotting window after plotting a model can cause the edges to get distorted. We advice not to manually resize the plotting window. Instead, if a larger plotting area is desired, the dimensions of the plotting area can be set as described above.

References

[1] J. Schellenberger, J. O. Park, T. M. Conrad, and B. O. Palsson, "BiGG: a Biochemical Genetic and Genomic knowledgebase of large scale metabolic reconstructions.," *BMC Bioinformatics*, vol. 11, p. 213, Apr. 2010. *Database available at* ***http://bigg.ucsd.edu***.

[2] J. Schellenberger, R. Que, R. M. T. Fleming, I. Thiele, J. D. Orth, A. M. Feist, D. C. Zielinski, A. Bordbar, N. E. Lewis, S. Rahmanian, J. Kang, D. R. Hyduke, and B. O. Palsson, "Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox v2.0.," *Nature Protocols*, vol. 6, pp. 1290–307, Sept. 2011. *Software available at* ***http://opencobra.sourceforge.net***.

[3] I. Thiele, N. Swainston, R. M. T. Fleming, A. Hoppe, S. Sahoo, M. K. Aurich, H. Haraldsdottir, M. L. Mo, O. Rolfsson, M. D. Stobbe, S. G.

Thorleifsson, R. Agren, C. Bölling, S. Bordel, A. K. Chavali, P. Dobson, W. B. Dunn, L. Endler, D. Hala, M. Hucka, D. Hull, D. Jameson, N. Jamshidi, J. J. Jonsson, N. Juty, S. Keating, I. Nookaew, N. Le Novère, N. Malys, A. Mazein, J. A. Papin, N. D. Price, E. Selkov, M. I. Sigurdsson, E. Simeonidis, N. Sonnenschein, K. Smallbone, A. Sorokin, J. H. G. M. van Beek, D. Weichart, I. Goryanin, J. Nielsen, H. V. Westerhoff, D. B. Kell, P. Mendes, and B. O. Palsson, "A community-driven global reconstruction of human metabolism.," *Nature Biotechnology*, vol. 31, pp. 419–25, May 2013. *Database available at* `http://humanmetabolism.org`.

[4] D. Oevelen, K. Meersche, F. J. R. Meysman, K. Soetaert, J. J. Middelburg, and A. F. Vézina, "Quantifying Food Web Flows Using Linear Inverse Models," *Ecosystems*, vol. 13, pp. 32–45, Nov. 2009.

[5] P. Murrell, *hyperdraw: Visualizing Hypergaphs.* R package version 1.6.0. *Package available at* `http://www.bioconductor.org/packages/release/bioc/html/hyperdraw.html`.

[6] M. Lawrence, *rsbml: R support for SBML, using libsbml*, 2013. R package version 2.12.0. *Package available at* `http://www.bioconductor.org/packages/release/bioc/html/rsbml.html`.

[7] J. H. G. M. van Beek, F. Supandi, A. K. Gavai, A. A. de Graaf, T. W. Binsl, and H. Hettling, "Simulating the physiology of athletes during endurance sports events: modelling human energy conversion and metabolism," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 369, pp. 4295–4315, Oct. 2011.

[8] U. Lying-Tunell, B. S. Lindblad, H. O. Malmlund, and B. Persson, "Cerebral blood flow and metabolic rate of oxygen, glucose, lactate, pyruvate, ketone bodies and amino acids.," *Acta neurologica Scandinavica*, vol. 62, pp. 265–75, Nov. 1980.

[9] A. B. Patel, R. a. de Graaf, G. F. Mason, D. L. Rothman, R. G. Shulman, and K. L. Behar, "The contribution of GABA to glutamate/glutamine cycling and energy metabolism in the rat cortex in vivo," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, pp. 5588–93, Apr. 2005.

[10] J. R. Dusick, T. C. Glenn, W. N. P. Lee, P. M. Vespa, D. F. Kelly, S. M. Lee, D. A. Hovda, and N. A. Martin, "Increased pentose phosphate pathway flux after clinical traumatic brain injury: a [1,2-13C2]glucose labeling study in humans.," *Journal of cerebral blood flow and metabolism : official journal of the International Society of Cerebral Blood Flow and Metabolism*, vol. 27, pp. 1593–1602, 2007.

[11] 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.