Chapter 1

Reconstructing High-Quality Large-Scale Metabolic Models with *merlin*

Oscar Dias, Miguel Rocha, Eugénio Campos Ferreira, and Isabel Rocha

Abstract

Here, the basic principles of reconstructing genome-scale metabolic models with *merlin* are described. This tool covers the basic stages of this process, providing several tools that allow assembling models, using the sequenced genome as a starting point.

merlin has two main modules, separating the process of annotating (enzymes, transporters, and compartments) on the genome from the process of model assembly, though information from the former is integrated in the latter after curation. Moreover, *merlin* provides several tools to curate the model, including tools for generating reactions' gene rules and placeholder entities for biomass precursors, such as proteins (e-protein) or nucleotides (e-DNA and e-RNA) among others.

This tutorial covers each feature of *merlin* in detail, including the assessment of experimental data for the validation of the model.

Key words *merlin*, *Genome-scale metabolic models*, *Genome functional annotation*, *Transport proteins annotation*

1 Introduction

Before the dawn of the genomic era, strain evolution was underpinned by random mutagenesis followed by screening and selection of mutants or, later, by manipulation of genes directly associated with the product of interest. The latter strategy was supported by components biology and was often unsuccessful, while the former, though presenting interesting results, could not unveil the mechanisms justifying the outcomes [1].

Instead of exclusively studying each organism's components individually, systems biology studies also the interactions between them, to comprehend and predict the phenotypical behavior on conditions other than ones already characterized [2, 3].

Electronic supplementary material: The online version of this article (https://doi.org/10.1007/978-1-4939-7528-0_1) contains supplementary material, which is available to authorized users.

Marco Fondi (ed.), *Metabolic Network Reconstruction and Modeling: Methods and Protocols*, Methods in Molecular Biology, vol. 1716, https://doi.org/10.1007/978-1-4939-7528-0_1, © Springer Science+Business Media, LLC 2018

According to some definitions, a *gismo* is an advanced technological device which performs a particular task, usually in a new and efficient way [4, 5]. Currently, metabolic systems biology is becoming a standard field of study with its own *g*enome-w*i*de *s*cale *m*etabolic m*o*dels (GiSMos) on the forefront.

The reconstruction of a GiSMo involves four main stages [3] and has been detailed in a protocol with over 100 steps [6]. Those stages are genome annotation, assembling the genome-wide metabolic network, conversion of the network to a stoichiometric model and metabolic model validation.

These models are reconstructed based on the parts of the genome that encode metabolic functions. Hence, a robust and reliable genome functional annotation is of paramount importance. The genome structural annotation process is usually coupled to a draft functional annotation of the coding sequences (CDSs), which may be incorrect or at least incomplete. Thus, several frameworks (e.g., *merlin* [7], Model Seed [8]) provide tools to perform the genome re-annotation and the curation of the outcome.

When integrated with metabolic data, genome annotations are converted into *gen*ome-wide scale metabolic *net*works (GenNets). These are sets of reactions interconnected by metabolites, produced and consumed through reactions promoted by enzymes encoded in the genome. The process of creating a GenNet encompasses data collection from a variety of databases providing such metabolic information, namely the reactions and the enzymes that catalyze them. Curation of the network, such as gap finding and balance validation, should be performed at this stage.

Nevertheless, curation must continue in the third stage as the conversion of the GenNet to a GiSMo involves adding a biomass objective reaction and non-growth ATP requirements to the reaction set. This allows building a stoichiometric matrix that, together with the assumption that the rates of consumption are equal to the rates of production for all metabolites, originates the system of linear equations represented by $S.\ v = 0$, in which v is the flux vector and S is the stoichiometric matrix where the columns represent reactions and the rows the metabolites.

The GiSMo should then be saved in a standard format, i.e., the Systems Biology Markup Language (SBML) [9], to allow importing the models into tools specially developed for operating GiSMos, such as OptFlux [10] (More information on using OptFlux on Chapter XX) or COBRA [11]. Notwithstanding the use of SBML, MIRIAM [12] annotations should also be used for annotating GiSMos to enable, for instance, the comparison of distinct reconstructions of the same organism enhancing the overall comprehension of its metabolism.

merlin is a tool developed for accompanying the reconstruction process throughout all four stages. It is currently composed of two main modules, where the first is developed specifically to help on

the genome annotation stage with dedicated tools and graphical user interfaces (GUIs), while the second module is oriented to the remaining stages, being that the last stage requires also a simulation platform. Moreover, the latter module offers a myriad of operations for curating the network and converting it to a GiSMo.

2 Materials

In this section, a brief description of the resources required for reconstructing GiSMos with *merlin* is provided.

2.1 merlin

merlin is an open source software tool fully implemented in Java™, distributed under the GNU General Public License, available for download in a single multi-platform (tested in Linux, Mac OSX, and Microsoft Windows) version at www.merlin-sysbio.org. The installation process is very straightforward as it only involves decompressing the downloaded .ZIP file into a folder. Whereas *Microsoft Windows* (*MS Windows*) users should use the *merlin.bat* file to run *merlin*, Linux/Mac OSX users must start the *run.sh* file.

In its core, *merlin* is supported by the AIBench [13] software development framework. The latter follows the model-view-controller (MVC) software architecture pattern, allowing the combination of new and existing software components.

It has a bimodular architecture with several subcomponents, as shown in Fig. 1. The three subcomponents of the genome annotation module allow decoding most metabolic capabilities available in the genome. Two of these components identify and select metabolic proteins' functions, whereas the last component predicts the location wherein these proteins operate. The second module is

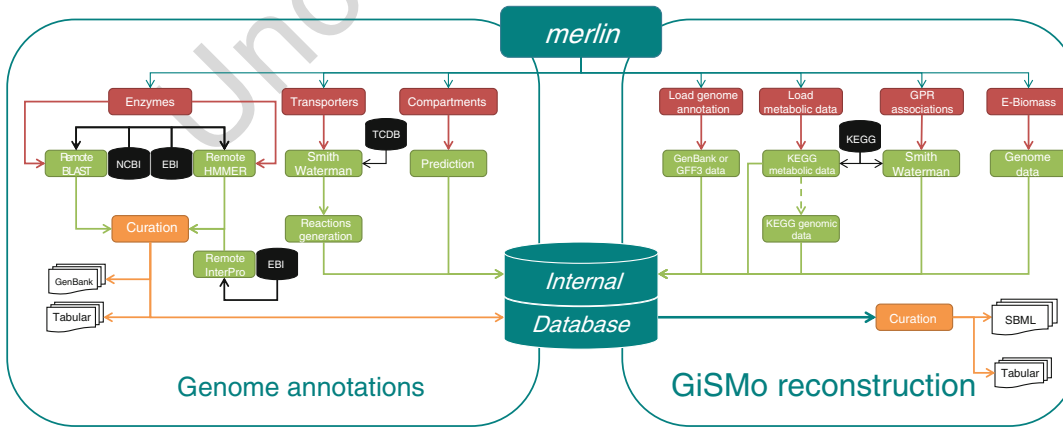

Fig. 1 merlin's architecture. merlin is a bimodular framework with several submodules. The first module (genome annotations) allows decoding the metabolic capabilities of the genome while the latter module (GiSMo reconstruction) provides several operations to assemble the GiSMo

where the GiSMo is assembled. Its subcomponents allow loading existing annotations, loading metabolic data, determining *g*ene-*p*rotein-*r*eaction (GPR) rules, and adding an e-biomass reaction to the core of the GiSMo. All such operations will be detailed in the following sections.

merlin keeps all information in internal databases which are shared between both modules to ease the process of integrating genome annotations with the metabolic data. The information kept in the databases is accessed through *merlin*'s interface (*see* **Note 1** for a detailed description of the interface).

2.2 Online Resources

The reconstruction of metabolic models involves analyzing not only the genome of the organism under study, but also collecting information on proteins' functions, reactions, biomass composition, and other physiological data. Hence, besides organism-specific publications (books and journals' manuscripts), several databases must be accessed throughout the whole process (Table 1).

The Universal Protein Resource (UniProt) [14] is the combination of three databases (UniProt Knowledgebase, UniProt Reference Clusters, and UniProt Archive) and its curated version (UniProt-SwissProt) [15] is one of the best sources of curated protein annotations available to date. The National Center for Biotechnology Information (NCBI) maintains a repository of several biological databases, providing tools for the analysis and retrieval of this information [16]. BRENDA contains information curated by experts, thus being one of the most reliable databases for enzymatic information [17]. The Kyoto Encyclopedia of Genes and Genomes (KEGG) is a knowledge-base that maintains an extensive collection of information on genes, metabolites, reactions, and

Table 1
List of databases with relevant information for the reconstruction of GiSMOs

Name	Contents	Programmatic API	URL	Reference
UniProt	Proteins functions	Available	www.uniprot.org	[14]
NCBI	Genes, protein functions Taxonomic data Genome sequences	Available	https://www.ncbi.nlm.nih.gov/	[16]
KEGG	Genes, protein functions Metabolic data	Available	http://www.kegg.jp/	[18]
BRENDA	Proteins functions	Available	http://brenda-enzymes.org/	[17]
MetaCyc	Genes, protein functions Metabolic data	Not available	https://metacyc.org/	[48]

pathways [18]. All metabolic information used by *merlin* to assemble GiSMos is retrieved from KEGG. Moreover, when available, *merlin* can also retrieve KEGG's genome annotation.

3 Methods

A detailed description of all the steps required for reconstructing GiSMos with *merlin* is provided below. Though most databases and online tools are accessed automatically by *merlin*, some stages may require uploading results to *merlin*'s internal database.

3.1 Reconstruction Project

Starting a project in *merlin* requires the organisms' genome in the FASTA format [19] and its NCBI's taxonomic identifier. Whereas the former is optional, the latter is mandatory.

3.1.1 Download of Genome Sequence

Regarding the genome sequence, users are encouraged to use NCBI's Assembly database (www.ncbi.nlm.nih.gov/assembly) [21] to download the genome sequence, since *merlin* has parsers to process the headers in these files. As shown in Fig. S1 of the supplemental material, the NCBI assembly webpage provides links (green arrows) for downloading the genome in the GenBank and the RefSeq formats (*see* **Note 2** for a detailed explanation of the differences between these formats). These links redirect the user to a File Transfer Protocol (FTP) webpage where the user is presented with a list of several files. All the files are compressed in the gzip (GNU ZIP) format and must be decompressed to access their information. Whereas there are several tools available for *MS Windows* users to decompress these files, such as 7zip and WinRAR, Linux/Mac OSX users can access the compressed files contents using simple commands in the terminal. Files ending with *_protein.faa.gz* (protein products annotated on the genome assembly) and *_cds_from_genomic.fna.gz* (nucleotide sequences corresponding to all coding sequences annotated on the assembly) can be uploaded to *merlin*, after decompression. Other file types, like *_genomic.gbff.gz* (GenBank flat file format of the genomic sequences in the assembly), can also be used by *merlin* (after decompression) in other operations as shown next.

3.1.2 Taxonomic Identifier

The International Nucleotide Sequence Database Collaboration uses NCBI's Taxonomy database (www.ncbi.nlm.nih.gov/taxonomy) [20] as repository for standard nomenclature and classification. Hence, *merlin* requires the identifier provided by this database to univocally identify the organism under study throughout the reconstruction process. The link inside the red ellipse in Fig. S1 of the supplemental material contains a direct link to the taxonomic identifier (blue circle) of the case study organism. Nevertheless, for cases in which the genome sequence is not retrieved from NCBI, this database can be directly accessed to retrieve the taxonomic identifier.

3.1.3 Create Project

After retrieving the taxonomic identifier, and optionally the genome files, a new project can be created in *merlin*. When *merlin*'s interface is opened, users should choose "*project>create*" to start a new project. The minimum requirements to create a new project are setting the project name, the taxonomic identifier, selecting the type (H2 or MySQL, discussed below) and naming the database in which the data will be kept. Setting the FASTA files with genome sequences is optional at this phase.

3.2 Database

merlin supports two different relational database management system (RDBMS) for storing its data, H2 or MySQL. For users assembling GiSMos in personal computers, the H2 RDBMS is recommended, as *merlin* is ready for use with this system. For using MySQL, an external installation is required, which depends on the platform being used to host the database server.

3.2.1 MySQL

MySQL (www.mysql.com) is an open source RDBMS, written in C and C++, which supports the use of Structured Query Language (SQL) to access the information stored in its databases. It runs in practically all platforms, including *MS Windows, Linux and Mac OSX*. MySQL is a free, fast, stable multi-user, and multi-threaded database server [22], which makes it the most popular RDBMS in the world [23]. Hence, *merlin* embedded MySQL in its first versions, providing users with a reliable server for keeping data. Though this RDBMS is not currently included in its releases, *merlin* still supports connections to this server.

3.2.2 H2

H2 Database Engine (www.h2database.com) is an open-source RDBMS written purely in Java, which provides an alternative to traditional RDBMS by offering extremely fast performance and very small disk footprint. Like MySQL, H2 fully supports SQL and Java database connector (JDBC) application programming interface (API). In contrast, unlike MySQL all data are encrypted and each database uses just a couple of files for storage. Other Java-based databases, like Derby (http://db.apache.org/derby) and HSQLDB (http://hsqldb.org), do not support full text search or the Open Database Connectivity (ODBC) driver. Finally, the H2 database engine can be embedded in any Java applications or run in the client-server mode. As *merlin* is a cross-platform tool, which can be stored in an external hard drive and executed in computers with different operating systems, the features offered by this database integrate perfectly with *merlin*'s philosophy.

3.2.3 New Database

merlin is deployed ready to use with an internal database with the default name of "*my_dabatase*," which can be used to start a project (i.e., the reconstruction of a GiSMO). Nevertheless, users can create new databases, with different names, to create other projects for other organisms or just because they want to change the default name, by accessing the "*database>new*" menu.

3.2.4 Clean Database

merlin allows users to clean a project's database without affecting other projects. Users can clean the entire database ("*all information*"), or just specific parts of it, namely the "*model*," the "*enzymes annotation*," the "*transport proteins*," the "*transport annotations*" (TRIAGE), the "*compartments annotation*" or the "*InterPro annotation*." While the former operation is inherently understandable by name and cleans the whole model, the latter operations clean specific annotations. These operations are available at "*database>clean*" and should be handled with care as their execution is permanent.

3.2.5 Delete Project

merlin also allows deleting a project, which involves deleting the entire database from the system. This option is available at "*project>delete*" and (as the previous operation) should be handled with care as the operation is irreversible.

3.3 Genome Annotation

The annotation module is where *merlin* processes genome-wide annotation data. This module is composed of three main procedures, which are discussed below.

3.3.1 Enzymes Annotation

Though not a mandatory step when reconstructing models, it is highly recommended that users perform their own annotation to have maximum confidence on the enzymatic potential of the genome. This unique feature is one of the finest resources offered by *merlin*, easing the process of assigning functions to genes that potentially encode enzymes. The enzymes annotation will be integrated in the model with the following assumption: genes encoding enzymatic proteins should have Enzyme Commission (EC) number assignments, as EC numbers can be mapped to reactions to build the metabolic network.

Similarity Searches

The enzymes functional annotation process in *merlin* relies on similarity searches to remote databases to find homologous protein sequences. The rationale for this approach is that gene or protein sequences with excess similarity (i.e., more similarity than expected by chance) should have arisen from a common ancestral lineage [24].

In *merlin*, users can use the Basic Local Alignment Search Tool (BLAST) [25] web-services from European Bioinformatics Institute (EBI) or the National Center for Biotechnology Information (NCBI), or HMMER remote similarity searches [26] to perform genome-wide sequence alignments. These tools allow configuring several parameters, well known by researchers that commonly use them, including the expected value threshold, the maximum number of hits and the remote database. The first and the second parameters are usually used as thresholds for the similarity search (*see* **Note 3a** for more information on these parameters).

Scoring Algorithm

merlin has a specific algorithm that calculates a score, between 0 and 1, for every possible annotation of each gene and automatically selects the annotation with the highest one. The score assigned to each annotation comprises two factors, being those the frequency and the taxonomy of the homologous protein annotations. As shown in Eq. 1, the first score is obtained by determining the number of times each EC number is found within the homologous gene records annotation (frequency), whereas the latter is associated with the taxonomy of the organisms to which these records belong. More information about this scoring algorithm is provided in [7].

$$\text{Score} = \alpha \times \text{Score}_{\text{f}} + (1 - \alpha) \times \text{Score}_{\text{t}}. \quad (1)$$

The score shown above reflects the degree of confidence of the annotation assigned by *merlin* to a given gene.

Nevertheless, several factors may influence the scoring results, like the quality of the genome assembly (and gene calling), the database selected for the remote similarity search and the taxonomy of the organism. The first factor regards the gene sequence sent to the alignment, as cases in which the sequence was incorrectly assembled or the gene calling is incorrect will affect the number of homologous genes found. In extreme cases, in which contigs have less than 50 nucleotides and these correspond to proteins' conserved domains, the frequency score may be positively affected because of having similarities with many sequences and consequently the calculated scores being very high. It should be noticed that these cases might happen when *merlin* is allowed to adjust the expected value for the alignment of smaller sequences. Regarding the remote database selected for alignment, databases such as "*swissprot*" are very biased, as annotations are propagated through clusters of homologous genes [27]. Thus, the frequency score will be remarkably high. Lastly, the number of bacterial genomes available in databases is much higher than the number of Eukaryotic ones. Moreover, there are quite a few organisms with numerous strains of the same species, which will also bias the results in terms of both frequency and taxonomy. Hence, when gene annotations, such as hypothetical or uncharacterized proteins, are propagated for different strains of the same organism, the similarity search results will also be biased and *merlin* will use these to calculate annotations, which do not provide useful information.

Therefore, *merlin* users should adopt the annotation strategy that best suits their project. For instance, organisms with several sequenced strains should be aligned against curated databases, to increase the confidence of the frequency scores provided by *merlin*. Genomes with many scaffolds and contigs should also be aligned to curated databases initially. Then, for CDSs without hits, a second alignment should be performed against a broader database (like

EBI's TrEMBL or NCBI's *nonredundant database*) to identify CDSs with similarities to known sequences and therefore find other potential enzymes.

Users can automatically accept all annotations with scores above a given threshold and reject the remaining, after an empirical analysis of the best alpha value. Instead, a (complete or partial) curation of the annotations may be performed. Regarding the latter option, selecting the best α value, and an upper and a lower threshold for the EC numbers' scores, allows decreasing the number of records to be manually curated. These parameters can be set after a rational analysis of the results. Usually, this analysis involves performing the manual curation of the annotations of a group of randomly selected genes, following a curation workflow, such as the one described in Subheading 3.3.1.4. The curated annotations are then assessed to the annotations automatically provided by *merlin*, for different α values. This strategy allows selecting the α value that provides the most correct annotations, as well as setting the upper and lower thresholds. All records with scores above the upper threshold should be automatically accepted. Likewise, all entries with scores below the lower threshold should be rejected. Annotations with scores in-between the thresholds should be revised according to the same curation workflow.

Curation Panel

The annotations proposed by *merlin* are based in the similarity searches and are presented in the "*enzymes*" panel of the "*annotation*" module (Fig. 2), which can be accessed from *merlin*'s

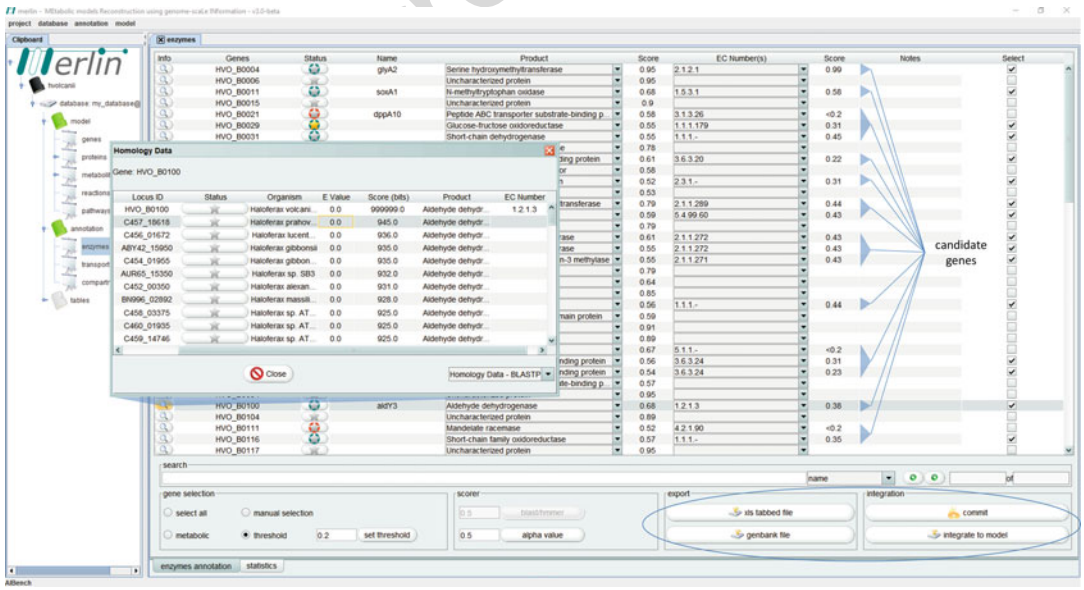

Fig. 2 merlin 's enzymes annotation panel. All the genes with EC numbers are considered candidate genes. merlin allows exporting the annotation or integrating it in the model

clipboard. This panel contains a table with 10 columns and a line per gene. The first column contains buttons with "*magnifying glasses*" which allow accessing the similarity search information, namely the identifiers of the homologous genes, their UniProt status (if available), the similarity search tools' scores (expected value and bit scores), and the functions of the homologous genes. The second column displays the gene identifier (locus tag if available). The third column is very informative, as its buttons provide, when available, the annotation status of each studied gene in UniProt (reviewed entries in UniProt show a gold star and unreviewed a silver star), and the agreement of such annotation with *merlin*'s annotation (a green background reveals concordance and a red one shows divergence; a light green background establishes that *merlin* has identified all annotations available in UniProt plus additional EC numbers not proposed by UniProt, while an orange background represents the opposite, that is, UniProt's annotation provides EC numbers not available in *merlin*'s current assignments). The next column is the common gene name. The fifth and seventh columns (product name and EC number, respectively) are followed by column scores (sixth and eighth columns). These scores are calculated according to the scoring algorithm described below. The last two are: a column in which users can keep notes regarding each gene's annotation, and a column with a check box to select genes to be integrated in the model or exported. This panel also allows searching for text on the table, selecting a specific group of genes, configuring *merlin*'s scorer parameters, exporting the model, and finally saving the annotation data to the database and integrating the annotation in the model.

Finally, the statistics panel provides some information regarding the data available in the database, such as: total number of genes, genes without similarities, total number of homologous genes, number of organisms with at least one homologous gene and its division by domain, among others.

InterPro Annotation

The InterProScan annotation can be performed by accessing the "*annotation>enzymes>InterProScan*" menu, though this is not mandatory. As this operation is somewhat slow, it requires entering upper and lower thresholds, to limit the number of annotated records. Unlike BLAST and HMMER annotations, this operation's sole purpose is to provide more information for the curation of the records not automatically annotated by *merlin*. After the remote InterProScan search, *merlin*'s entries with these annotations will show a purple background on the "*magnifying glass*" button. The InterPro information can be accessed on a submenu (red circle) of the information window, as shown in Fig. S2 of the supplemental material.

Curation Workflow

The first step to perform the curation of the annotation is designing a workflow for reviewing the annotations of genes that potentially encode enzymes, i.e., genes with EC number assignments (Fig. 2), the so-called candidate genes (CG). Usually, this involves selecting a well-studied closely related organism to compare annotations. For instance, yeast reconstructions should use *Saccharomyces cerevisiae* as a model organism. The selection of this organism is quite important, as its annotation will be a pseudo-template for the GiSMo.

A typical workflow is presented in Fig. 3. Initially, the user should verify the status of the CG in UniProt and the agreement with *merlin*'s annotation. In case of a gold star and a green background on *merlin*'s status column then the gene's annotation should be accepted and, in the notes' field, the user can add a reference to the level of confidence of this annotation. As this annotation was confirmed with a reviewed entry, the level of confidence can be set to the maximum, i.e., confidence level (CL)-A. The level of confidence will assist in the process of curating the model. If the star is silver, and the background still green, the annotation is also accepted though with a classification of CL-B. When *merlin*'s assignment does not match with UniProt's, the annotation should be confirmed to determine the reason for the discrepancy. If the UniProt's entry is reviewed, *merlin*'s annotation is changed and classified with CL-B. Otherwise, or in the case that the CG does not have a UniProt annotation (no star on the status column), the similarity search results of the CG are analyzed, by seeking a record of the model organism. When such entry exists,

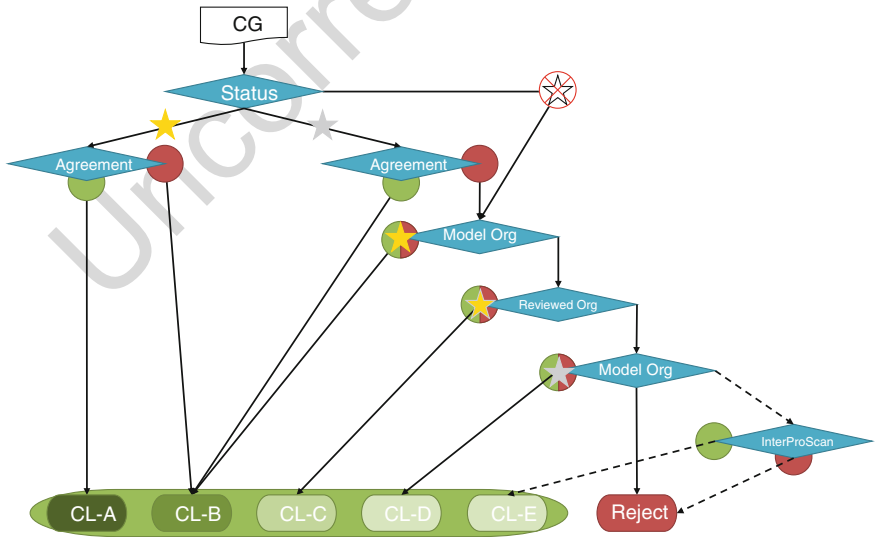

Fig. 3 Typical annotation workflow. CG—candidate gene; CL—confidence level classification. The CG's annotation provided by merlin is compared to its own annotation in UniProt, to a reference organism's annotation and to other reviewed entries in the homolgy search results. When one of the conditions is met, the CG's annotation is accepted and the classification can be added to the CG in merlin

and is reviewed in UniProt, the gene is annotated and classified with CL-B. In the case that the model organism's entry does not exist or exists, but is not reviewed, the CG's annotation is compared with the annotation of the homologous gene with lower expected value and higher score that has its annotation reviewed. In such a case, the classification should be decreased to CL-C. If the similarity search results do not encompass reviewed records but the model organism's annotation is available, this annotation should be considered with the classification of CL-D.

When available, the InterPro annotation results can be used as a final test to find evidences of enzymatic activity on CGs. If the InterPro annotation provides evidences that corroborate *merlin*'s annotation, it should be considered with the classification of CL-E. Alternatively, InterPro results can be used for other purposes, e.g., increasing the confidence level of annotations inferred from unreviewed records.

All other cases not encompassed in this workflow should be rejected, i.e., not considered enzyme-encoding genes, at this stage.

Moreover, all EC numbers verified using this workflow will be confirmed in BRENDA to determine if the function corresponds to the EC code, to prevent cases in which the EC number has been updated (*see* **Note 3b** for the examples of these cases).

Outputs

The enzymes annotation is the backbone of the metabolic model. Thus, a robust, reliable, and traceable genome annotation is mandatory. Annotations performed according to the set of steps described before fulfil these rules.

All the changes performed in the "*enzymes*" panel of the "*annotations*" module are saved to the project file, unless the user clicks the "*commit*" to database button in this panel. This means that, before *committing*, none of the changes is permanent and new projects started with the same database will access raw data. After *committing*, new projects will have access to the curated annotation. Though not required, the curated annotations should be *committed* before the integration to the model. Furthermore, the integration of the enzymes annotation must be preceded by the loading of metabolic data (*see* Subheading 3.4.1 below) in the model's database.

Finally, though usually the objective of performing genome annotations in *merlin* is reconstructing GiSMos, these can serve other purposes. Therefore, *merlin* allows integrating these annotations in existing GenBank format files or simply exporting the annotation in a tabular format, as shown in Fig. 2.

Previous Genome Annotations

There might be cases in which researchers already have access to good quality annotations performed within other contexts and might want to use them within *merlin*. For these cases, *merlin*

provides operations to integrate annotations from GenBank or GFF3 files directly with previously loaded metabolic data. The only requirement is that these files have fields with *e*nzyme *c*ommission (EC) numbers associated with gene entries. These files are not required for creating a project and should be imported later.

3.3.2 Transporters Annotation

The transporters annotation is performed with the tool developed specifically for this purpose, TRIAGE [28]. This tool uses increasingly stringent conditions to predict transport protein encoding genes. The first assumption is that all transporter proteins are in membranes; hence, the initial step is filtering out all genes without transmembrane helices, thus identifying transporter candidate genes (TCGs). The second premise is that transporters should have similarities with proteins available in the Transporters Classification Database (TCDB); thus, TCGs are compared to the whole TCDB database. Lastly, when dealing with compartmentalized models, the first assumption is reinforced by excluding all TCGs not predicted to be in membranes by compartmentation software, such as PSortb 3.0 [29] or LocTree [30].

These proteins are annotated according to the frequency and taxonomy of the homologous proteins annotations in TCDB, in a similar manner to the enzymes annotation. In this case, the score assesses the TC family, metabolites, and transport mechanism of each TCG. A brief description of this process is presented below; for a more detailed explanation please refer to [28].

Transmembrane Helices

Transport proteins are usually located in membranes [31]. Thus, TRIAGE's first step is identifying these proteins. There are several bioinformatics tools such as TMHMM [32], Phobius [33], and several others, which predict the number of transmembrane helices from the amino acid sequence. TMHMM has been considered the best performing tool for this function [34], yet it does not provide an API for remotely accessing the server. Hence, users must perform the search on the website, or install the software locally, and then load the results (format: extensive, no graphics) into *merlin*. Phobius provides a programmatic API for performing remote transmembrane helices predictions, and thus it was possible to embed this tool in *merlin*. The output of either tools is the number of helices per gene, which can be used to filter genes for the next step.

Transporters Classification Database

This database is the most extensive and comprehensive resource of transport proteins available. TCDB implements a classification system analogous to the EC format, though including phylogenetic information that assigns TC numbers to proteins. These identifiers are formed by five components separated by a dot: #.∗.#.#.#, in which # represent numbers and ∗ a letter. TCDB records provide

specific information regarding the transport activities of its entries, namely: TC number and generic description, accession number (UniProt), protein name, length, molecular weight, species (Organism), number of TM's, and location/topology/orientation.

Unfortunately, to date, TCDB does not provide in its records a specific field for reporting transported metabolites nor transport mechanisms. Thus, this information must be manually retrieved from the generic description or, often, the family description. Though containing valuable information, TCDB does not provide data in a readily accessible way. Hence, TRIAGE has an internal Transporters' Annotations Database (TAD), which keeps information, inferred from TCDB's annotations, useful for the reconstruction of GiSMos. Most information, such as UniProt's accession number, protein name, and species (organism), are retrieved from TCDB and stored as are. The remaining data (transport directions, transported metabolites, reversibility, reacting metabolites, and equation) must be extracted from TCDB (*see* **Note 4** for a detailed description of this process). TCDB contains (as of 17/04/2017) 15,021 records, 7383 of which are already curated and available in TRIAGE's TAD. The remaining entries should be curated as needed, following the transporters annotation workflow (*see* **Note 4** for a detailed description of this workflow).

After determining the number of helices in each protein sequence of the genome of the case study, TRIAGE compares each sequence with at least h (a number defined by the user, default = 1) helices to the whole set of proteins available in TCDB to find homologous proteins. This comparison is performed with the Smith-Waterman [35] algorithm, which guaranties optimality and high sensitivity. Overall, the alignment similarity threshold for considering homology between sequences is 10% which can be decreased in special cases [28]. The score for these alignments is calculated by the following equation:

$$\text{Similarity} = \frac{\text{score}_{\text{alignment}} - \text{score}_{\text{minimum}}}{\text{score}_{\text{maximum}} - \text{score}_{\text{minimum}}} \quad (2)$$

In Eq. 2 the minimum score is 0, the maximum score is the maximum between the maximum score of the TCDB protein sequence and the maximum score of the query sequence—and finally, the alignment score is the score of the alignment region between both the sequences. TCGs with at least one homologous protein in TCDB will be stored in TRIAGE's alignments database.

Afterward, transport reactions will be created to annotate the TCGs. This process is performed by retrieving TAD's annotations of each TCG's similarity hits. TAD's annotations are then used to calculate which metabolites and transport mechanisms should be assigned to each TCG. As stated before, TAD contains 7383 TCDB entries, yet TCDB is always adding new records. Hence, the

similarity alignments may always find entries unavailable in TAD. The absent records should be annotated and included in TAD. Every user is invited to contribute to this effort, which already encompasses hundreds of person-hours, by following the transporters annotation workflow (*see* **Note 4** for a detailed description of this process).

After performing the similarity alignments, TRIAGE allows generating transport reactions associated to the genes with similarities to TCDB.

Membrane Locations

LocTree3 can be used to assign subcellular localizations both for eukaryotes and prokaryotes, whereas PSortb 3.0 can only be used for the latter. The integration of these predictions will be discussed in Subheading 3.3.3. Nevertheless, these locations will be used to filter and compartmentalize the transport reactions generated in the previous step.

Curation Panel

The annotation of transport proteins is initiated by running the transport proteins identification operation available at "*annotation>TRIAGE>transport proteins identification*." The transporters' annotations are presented in the "transporters" panel of the "annotation" module.

The panel shown in Fig. S3 of the supplemental material provides information on the transported metabolites and the type of transport associated with each gene. This panel allows tracing back the reason for associating a gene to the transport of a given metabolite and contains a table with seven columns and a line per gene. The first column contains buttons with "*magnifying glasses*," which when clicked open a smaller information window that provides information on the similarities, metabolites, and reactions associated with selected gene. The second column displays the gene identifier. The third column shows the number of transmembrane domains predicted to be associated with each gene. The fourth and fifth columns are only filled in after clicking the "*add TRIAGE data*" button (red ellipse in Fig. S3 of the supplemental material) and show the calculated TC family number and the number of metabolites associated with each gene. The sixth column shows the number of transport reactions generated after pressing the "*create transport reactions*" button (blue ellipse in Fig. S3 of the supplemental material). The last column allows selecting which genes and corresponding transport reactions will be integrated in the model. Lastly, the main table can be exported into an excel file and the transport reactions with gene associations integrated into the model (green ellipse).

The reactions will be created considering the parameters set in the panel. The metabolites associated with each gene will be classified with a score between 0 and 1. The α will weigh the frequency

and the taxonomy of the homologous proteins, whereas the threshold will determine if the metabolite is selected or not. Compounds labeled as currency symport metabolites will not be classified in such reactions.

As shown in Fig. S3 of the supplemental material, the information window presents three types of information. The first table shows which TCDB entries are similar (and the similarity score) to the selected gene. The second table, which may be selected in the drop-down box (black circle), allows visualizing which metabolites are associated with each gene, their KEGG identifiers, their score, the direction and reversibility associated with each metabolite, and the transport types and corresponding scores. The information in the metabolites table is available after the integration with TRIAGE's database. Finally, the last table in these windows shows the generated transport reactions, and whether these originated from primary annotations or from ChEBI's ontologies.

3.3.3 Compartments

LocTree3 [30] and PSORTb 3.0 [29] are used to predict subcellular localizations of all proteins encoded in the genome. The former was selected because its previous version (LocTree2) behaved equal or better than all state-of-the-art location prediction tools [30]. The current version is an improvement of LocTree2 by adding homology-based inference. PSORTb 3.0 is the most widely employed localization prediction software for bacteria [36] and was already supported by *merlin*, thus it can still be used for predicting protein locations in bacteria and Archaea. Although none of these tools provides a web-service, both offer whole genome/proteome predictions, which can be loaded into *merlin*. If the case study organism is not available in the database, users may use email mode for submitting the genomes/proteomes and receive the results by emails. Examples of reports from these tools in formats supported by *merlin* are available at *merlin*'s homepage.

Compartmentalization is also related to the energetic requirements of the cells. The ATP synthase harnesses the energy of an unfavorable proton gradient, via allowing hydrogen cations to cross the membrane and using this drive for coupling inorganic phosphate to ADP. Transport reactions, associated with the oxidative phosphorylation and proton pumps, are often generated, based in the proteome similarities, by TRIAGE and the compartmentalization should be verified to determine the direction of the reaction.

Curation Panel

As shown in Fig. S4 of the supplemental material, this panel allows visualizing the compartment predictions. Though LocTree3 reports assign genes to a single location, PSORTb 3.0 often identifies several locations, with different scores, for each gene. Hence, *merlin* allows setting a percentage difference limit for considering alternative predictions. For instance, if the main compartment has a

score of 0.4 and the secondary compartment a score of 0.35, the difference percentage between both compartments is 5%. If the difference percentage threshold for considering secondary compartments is set to 10%, the second compartment would be accepted. Similarly, as shown in Fig. S4 of the supplemental material, gene "*ADE01272.1*" the main location is *cytoplasmic* with a score of 0.75. The secondary compartment for this gene is *cytoplasmic membrane* with a score of 0.1. Hence, the acceptable difference between the main compartment prediction would have to be set to 66%, to accept the *cytoplasmic membrane*, which would be imprudent. Hence, the *cytoplasmic membrane* would not be considered when assigning gene products to compartments.

Clicking a given gene's "*magnifying glass*" provides information on the scores obtained with the compartments prediction tool.

3.4 Assembling the GiSMo

A curated genome annotation provides a good basis for reconstructing a GiSMo. However, metabolic information is also required for developing these models. The reconstruction of a GiSMo begins with the assembly of the genome-scale metabolic network (GenNet). This network represents the set of biochemical reactions encoded in the case study's genome. Though metabolic reactions are available in several databases, *merlin* retrieves them from KEGG as it provides a simple web-accessible API. Nevertheless, other metabolic data sources may also be included in upcoming versions of *merlin*.

3.4.1 Load Metabolic Information

Retrieving and loading metabolic data to *merlin*'s internal database is very simple, as it only requires accessing the "*model>load>load metabolic data*" menu. This operation retrieves all metabolic information from KEGG, which includes: compounds, glycans, drugs, reactions, enzymes, and pathways. Moreover, KEGG's genome annotation of the case study organism (if available) can also be retrieved, though *merlin* provides the enzymes annotation framework and supports loading annotations from different sources. This operation will load KEGG's information into *merlin*'s internal model database, setting all spontaneous reactions as integral to the model, thus launching the GenNet. The network will be further enlarged, when annotations are integrated to the GiSMo database, in the next step.

3.4.2 Integrate Annotations

The annotations performed earlier increase the reliability of the reconstructed GiSMo. However, if users have a former curated annotation, *merlin* provides, as mentioned earlier, operations for loading annotations from other sources in specific formats.

Enzymes Annotation

The enzymes annotation will define the topology and connectivity of the GenNet. The enzymatic activities (EC numbers) encoded in

the genome will activate reactions according to the associations provided by the KEGG mappings. For this, *merlin* will include all reactions associated by KEGG to such EC numbers, provided that the reactions and enzymes are included in the same KEGG pathway or that an EC number is associated with a single reaction (more information on this principle on [7]). The resulting GenNet should then be curated and converted to a GiSMo.

Load External Annotations

merlin supports loading annotations from GFF (version 3) and GenBank file formats. Whereas the former does not natively provide a field for EC numbers, though these can be annotated in the notes' field, the latter has a specific field for this purpose. These operations can be performed by accessing the "*model>load*" menu and can only be performed if an annotation was not previously loaded in the model.

merlin's Annotation

The "*enzymes*" panel in the "*annotation*" module (Fig. 2) has a button ("*integrate to model*") for performing the integration of the annotation in the model database. Though it is possible to set a threshold and integrate the results directly in the model, users are encouraged to perform, at least, a partial curation of the annotations. This button opens a window, which offers different options for the integration process. The first option regards the genes' names, determining whether the given gene's name in the annotation should precede, be merged as a synonym, or rejected if the gene is already labeled in the model. The second is the most important as it decides if *merlin*'s EC numbers annotation should be favored, merged, or rejected concerning EC numbers already available in the model. Finally, the latter integration option is not mandatory, as every EC number is associated with a recommended name for the respective enzyme. Yet, *merlin* allows the integration of the enzymes' names in the same conditions as the previous options, by being preferred, merged into synonyms or rejected, regarding the existing annotations.

Transporters Annotation

The transporters annotation will set the GenNet's borders by restricting the number of metabolites that can be exchanged between compartments or with the exterior. The "*transporters*" panel in the "*annotation*" module (Fig. S3 of the supplemental material) provides a button ("*integrate to model*") for performing the integration of the annotation in the model database. This operation will integrate the transport reactions, gene annotations, and TRIAGE's transport proteins identifiers in the model.

Compartments Annotation

The compartments annotation will define the number of compartments and further restrict the connectivity of the GenNet. The same metabolite in different compartments is considered a distinct

metabolite species in each compartment. Hence, the connectivity of the network may be impaired if reactions in the same pathway are allocated to different compartments.

The "*compartments*" panel in the "*annotation*" module (Fig. S4 of the supplemental material) provides a button ("*integrate to model*") for performing the integration of the compartments in the model database. This operation will reject the compartments set to be ignored in the main panel. Genes encoding proteins predicted to go to these compartments will be assigned to the internal compartment instead, cytosol and cytoplasm for prokaryotes and eukaryotes, respectively. There is also the option to compartmentalize the biochemical and/or the transport reactions.

This operation will create a new compartmentalized model. Reactions connected to distinct genes, which encode the same protein, assigned to different locations and genes assigned to multiple compartments will be replicated and versions of the same reaction in different locations will be included in the new model. This change in the internal model is transparent for the user and only the final model will be presented in the "*reactions*" panel of the "*model*" module.

3.4.3 Model Curation

The GenNet curation can start before the integration of the transporters and compartments. Indeed, the transport reactions can be added manually, and the compartments may be set to *inside* (pseudo-compartment representing the *interior* of the cell) and *outside* (representing the exterior of the cell). However, when added manually, transport reactions are seldom associated with genes. Moreover, a gene encoding a critical enzymatic activity inside the cell may have a paralogous gene encoding a similar protein in a different compartment. Thus, the final model will be impaired, as essentiality predictions will be affected. Nevertheless, the model must be iteratively curated to be able to mimic the organism phenotypical behavior. The curation phase comprises several steps, including its conversion to a GiSMo, which are described next.

Correct Reactions Reversibility and Direction

All the reactions in KEGG are in the canonical format, i.e., all the reactions are theoretically reversible. However, the reversibility and direction of the reactions can be determined by calculating the Gibbs free energy. Cases in which the free energy of the reactants is much greater than that of the products, the reaction takes place as written. Conversely, if the free energy of the products exceeds a lot that of the reactants, the reaction will tend to proceed in the reverse direction. Finally, if the free energy of the reactants is similar to the one of the products, the reaction may be reversible.

There are online tools to calculate the Gibbs free energy, such as the eQuilibrator (http://equilibrator.weizmann.ac.il). These

tools allow establishing a procedure to determine the reactions direction and reversibility, through the analysis of the reactions' Gibbs energy, at 1 mM concentration. Additionally, databases like the ones developed by Ma and Zengh [37] or Stelzer et al. [38] provide information on the reversibility of reactions available in KEGG. Hence, these curated databases were used for correcting *merlin*'s internal metabolic database reactions. Nevertheless, KEGG is frequently updated and new reactions are frequently added, thus requiring the curation of the reactions not available in the aforementioned studies.

Unbalanced Reactions

A GiSMo must be balanced to yield reliable predictions. The number of atoms of each element in the input environmental conditions must be the same as in the output (including biomass). A model unbalanced at the stoichiometric level may sink elements or excrete compounds because of having unbalanced reactions, thus impairing simulations' results. Hence, *merlin* provides a feature designed for highlighting these reactions in bold and italics. After executing the operation available at "*model>unbalanced reactions>find*" a new tab becomes available in the reaction's properties panel ("*magnifying glass*"), which shows the sum of all elements in the reactants, in the products and finally the net result that should be 0. This way it is possible to pinpoint the possible errors in the reaction and correct them. There are several types of stoichiometric errors, including:

(a) Metabolites without formulae;

(b) Missing protons or water;

(c) Reactions for synthesis/hydrolysis of macromolecules.

The first case can be verified by determining the missing metabolites formulae. The second by adding/removing protons/water to the reaction, after confirmation in other data sources, whereas the third is usually fixed by eliminating the polymer from the reactants or products according to the reaction stoichiometry (e.g., reaction R01762> Arabinan + H_2O ⇔ Arabinan + Arabinose becomes Arabinan + H_2O ⇔ Arabinose).

Either way, there are no strict rules for dealing with these cases. Each situation should be assessed and other databases/tools like MetaCyc, BRENDA, and even eQuilibrator should be checked to determine the correct stoichiometry. Reactions whose stoichiometry cannot be verified should be removed from the model. It is worth re-emphasizing that this step is critical and every reaction marked as unbalanced by *merlin* must be corrected or removed.

Unconnected Reactions

There are cases of reactions in the GenNet that convert metabolites, not available in the remaining network, into metabolites not consumed by other reactions in the GenNet. Though the inclusion of

these reactions does not usually impair model predictions, they can result from errors in the annotation, compartmentalization or simply be unnecessary to the GiSMo (although being indeed encoded in the genome). Hence, as before, *merlin* offers an operation ("*model>unconnected reactions>find*") aimed at identifying these reactions, by coloring the reactions' font in red. The analysis of these reactions may help in filling gaps in the GenNet or correcting mistakes in the annotation, improving the overall model performance. Alternatively, for cases in which there are too many of these reactions, they may be automatically removed ("*model>unconnected reactions>remove*").

GPR Associations

The gene-protein-reaction (GPR) associations are one of the most important aspects of a GiSMo. These rules make direct associations between genes and reactions in the model, and are important to correctly predict mutant phenotypes. These rules are usually added manually to models, yet *merlin* provides a unique operation that accesses information available in the KEGG BRITE [39] database, including their subunits and stoichiometry, to automatically create these rules.

KEGG BRITE is a collection of manually created hierarchical entries, which include structural complex modules. These modules are collections of manually defined functional units, including protein structural complexes. Each module is composed by an identifier, name, class, a set of reactions associated with the module, sets of KEGG orthologous (KO) genes, and a definition, which describes the relationship between the KOs, i.e., the GP rule. The rules can be very complex with several alternative subunits and connected with AND/OR operators. Thus, *merlin* includes a mathematical parser for the rules, which calculates all possible KO combinations. Each KO comprises a set of genes with similar roles belonging to distinct organisms and horizontally conserved across several species.

The modus operandi for finding GPRs in *merlin* is briefly described next. For each EC number annotated in the case study, *merlin* scans KEGG BRITE searching for complex modules. The KO of the taxonomically closest organism is then aligned using SW, initially only to genes annotated with such an EC number, and if no hits are found, with the whole genome. Whenever *merlin* finds all KO comprised in the module definition it creates the gene rule and associates the rule to the reactions present in the KEGG BRITE module. Other reactions associated with genes identified as part of a rule, yet not available in the KEGG BRITE module, may be automatically removed from the model. Nevertheless, the user can insert/edit/remove any reaction to/from the model.

Adding GPRs to the GiSMo involves running the corresponding operation ("*model>gene-protein-reaction rules*")

and setting a few parameters. This operation requires setting distinct threshold for ortholog and paralog genes, as the operation will select KEGG's closest orthologues to compare against the case study genome. The default values are the result of empirical tests, yet users may prefer being either more or less restrictive. In addition, the operation can automatically integrate the rules after generating or just save them to a file on a report file to allow a previous inspection of the rule. Other options include whether to remove reactions not available in KEGG BRITE associated with enzymes assigned with GPR rules, or removing reactions with notes or manually inserted.

After determining the GPR rules, a new menu called "*GPRs*" becomes available in the dropdown box of the pop-up window shown when the "*magnifying glass*" of a reaction assigned with rules is clicked. These reactions are labeled on the notes field with the message "*merlin GPR*." Each line presented in the GPR table contains one or more gene subunits (AND rules) required for creating a protein complex that promotes the reaction of interest. Different lines offer alternative combinations of genes (OR rules).

The rules automatically generated by *merlin* can be edited or removed by using the edit reaction option on the "*reactions*" panel of the "*model*" module. Likewise, GPR rules can be manually added/edited/removed to/from the reaction using the same option.

Add Biomass Formation Reaction

This is the first step in converting the GenNet to a GiSMo. The equation of the biomass formation reaction denotes the necessary amount of each biomolecule (e.g., amino acids, lipids, etc.) required for cell replication [3]. This equation should also include growth-associated energy requirements, characterized by the number of ATP molecules required for synthesizing a gram of biomass. Though previous studies [40] propose that using the biomass equation of a related organism does not introduce significant errors in GiSMos, a recent study demonstrates that such an approach may be incorrect [41]. Moreover, according to [42], estimation of the average protein and (deoxy)ribonucleotides contents from the genome sequence (gene translated amino acid sequences for the former; mRNA, tRNA, and rRNA for ribonucleotides; gene nucleotide sequences for deoxyribonucleotides) provides better results than using the biomass equation from related species.

Hence, *merlin* provides a tool that allows adding an e-biomass equation to the GiSMo using the latter approach. Besides estimating the average protein and (deoxy)ribonucleotides contents *merlin*'s "*model> e-biomass equation*" operation adds several reactions to the model, representing the assembly of pseudo-compound macromolecules for the biomass, namely: e-Protein, e-Nucleotides, e-Lipids, e-Carbohydrates, and e-Cofactors. To perform this

Table 2 t.1
Relative contents of several macromolecules in different organisms

Organisms	Source	RNA % (g/g_{RNA})			RNA % (g/g_{DW})	Protein % (g/g_{DW})	DNA % (g/g_{DW})	Lipids % (g/g_{DW})	Carbohydrates % (g/g_{DW})	Cofactors % (g/g_{DW})	
		mRNA	tRNA	rRNA							t.2 t.3
Yeast	exp	0.05	0.15	0.8	0.049	0.401	0.007	0.045	0.497	n/a	t.4
	literature	0.05	0.15	0.8	0.067	0.425	0.026	0.055	0.353	0.074	t.5
Archaea	exp									n/a	t.6
	literature	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	t.7
Gram positive	exp	0.05	0.2	0.75	0.077	0.539	0.02	0.125	0.239	n/a	t.8
	literature	0.05	0.2	0.75	0.075	0.531	0.023	0.162	0.148	0.062	t.9
Gram negative	exp	0.05	0.15	0.8	0.154	0.582	0.021	0.096	0.147	n/a	t.10
	literature	0.05	0.15	0.8	0.152	0.591	0.025	0.094	0.084	0.054	t.11
Total Bacteria	exp	0.05	0.15	0.8	0.115	0.561	0.02	0.111	0.193	n/a	t.12
	literature	0.05	0.15	0.8	0.118	0.565	0.024	0.124	0.112	0.057	t.13

operation the relative amounts of proteins and nucleotides must be identified. Table 2 shows the relative amounts of several macromolecules, retrieved from the literature and experimentally determined in our group according to the methodology defined in [42], in different types of organisms. These values were determined by averaging the experimentally determined quantities of different organisms. Moreover, gene expression data can be used to adjust the protein contents to the ones expressed de facto. Besides the gene expression data, files required by this operation are the genome protein fasta file (.faa) for the amino acids, the whole genome sequence fasta file (.fna) for the DNA, genomic coding sequences fasta file (.fna) for the mRNA, and the RNA sequences fasta file (.fna) for rRNA and tRNA. The latter file will have to be preprocessed to extract the rRNA sequence into a file and the tRNA sequences into a separated file. All other RNA sequences are discarded. These files can be obtained on the same assembly FTP links shown in Fig. S1 of the supplemental material.

e-Protein

The e-Protein pseudo-compound is calculated by determining the frequency of each amino acid in the translated coding sequences. In this approach, the frequency of each amino acid, normalized by the total number of residues, will be equivalent to the mass fraction of each amino acid. This mass fraction is then converted to mol_{aa} per $g_{protein}$. Gene expression data can be used to correct this estimate for proteins effectively expressed in specific conditions. The only condition is that this data used the same gene identifiers as the genome protein fasta file (.faa). This calculation considers the amount of H_2O formed during protein polymerization, which is included in the e-Protein equation.

e-Nucleotides

The rationale for calculating the RNA and DNA composition is the same as before. The whole genome sequence is used to estimate the amount of each deoxyribonucleotide and the mRNA, rRNA, and tRNA are used to estimate the moles of total RNA in the cell. The contribution of each different type of RNA, for different types organisms, is shown in Table 2. As before, the amount of diphosphate formed for the polymerization of the nucleotides is considered and included in the e-DNA and e-RNA pseudo-compounds equations.

Cofactors

Cofactors are important in models as, though their biosynthesis is not a burden to the cell since these are recycled, the apparatus for their production must be available in the GiSMo. The inclusion of these compounds in the biomass forces the GiSMo to produce them. Hence, the cofactors pseudo-compound macromolecular composition includes trace amounts of all metabolites identified as essential in [43] and is included in the biomass. Nevertheless, the reactions' stoichiometries can be changed and elements added/removed to/from the reaction.

Polysaccharides, Lipids, and Fatty Acid Formulation

The polysaccharides and lipids pseudo-compounds composition must be determined experimentally. Alternatively, since it cannot be inferred from the genome sequencing data, the composition should be set to the composition of a closely related organism. The fatty acid pseudo-compound, though not usually directly included in the biomass reaction, is a precursor of the lipids. This average fatty acid composition should also be determined experimentally or alternatively from a closely related organism. A new reaction should be inserted in the model indicating the contribution (stoichiometry) of each fatty acid (e.g., Octanoic acid—C8, Decanoic acid—C10, Octadecenoic acid—C18, etc.).

Phosphorus to Oxygen Ratio

The phosphorus to oxygen (P/O) ratio is an indicator of the relationship between ATP and oxygen, which specifies the number of orthophosphate molecules used for ATP biosynthesis per atom of oxygen reduced through the electron transport chain in aerobic organisms. This ratio is directly proportional to the number of hydrogen atoms transported outward against the electrochemical gradient and inward through the ATP synthase. These reactions often involve the oxidation of ferrocytochrome to ferricytochrome with the reduction of oxygen and the reduction ubiquinol to ubiquinone to recycle the ferricytochrome. As before, the absence of specific studies to characterize the P/O ratio in the case studies should be overcome by assuming the same P/O ratio as of closely related organisms.

In *merlin*, this ratio is translated by curating a set of reactions that consume oxygen to transport hydrogen outward and a reaction that transports hydrogen inside, producing ATP. TRIAGE's

analyses often detect genes associated with the oxidative phosphorylation pathway and create reactions involved in this process, as shown in the example in Table S1 of the supplemental material. Nevertheless, when assembling aerobic organisms' GiSMos, users should curate the stoichiometries of these reactions to guarantee the correct P/O ratio.

Growth and Maintenance ATP

The GiSMo should also include an equation that represents ATP depletion for cellular conservation and preservation, the maintenance ATP. Likewise, the ATP spent in cellular replication efforts should also be included in the model. The growth ATP flux includes all cellular processes involved in cell replication, such as proteins and nucleotide synthesis. There are some enzymes involved in the translation process, by spending energy to ligate amino acids to tRNA molecules. These reactions are now often included in GiSMos and should be considered when performing the calculations of the growth ATP flux.

The maintenance and growth ATP fluxes should be determined experimentally by plotting ATP metabolomics analyses against growth data from of chemostat growth experiments, as shown in Fig. S5 of the supplemental material. The slope of the linear regression of the ATP flux measurements vs. growth rate indicates the amount of ATP spent for cellular growth, whereas the *y*-intercept represents the amount of ATP consumed for cellular maintenance functions. Alternatively, the ATP fluxes can be determined by comparing the simulation results, with different maintenance and/or ATP flux values, to in vivo data. Simulations should be performed with the same environmental conditions as the experiments, namely the limiting carbon source flux. The linear regression slopes and *y*-intercept values of the simulated vs. experimental growth rates are then evaluated [44]. The better ATP flux values are provided by the simulation that yields the *y*-intercept value closest to zero and the slope value closest to the unit. Lastly, when experimental data is not an option, the ATP flux values of closely related organisms can be used.

In *merlin*, the maintenance ATP can be included in the GiSMo by setting the lower and upper thresholds of a clone of KEGG's reaction R00086. The original reaction is promoted by several enzymes identified with different EC numbers, but in such cases the reaction may serve specific purposes and a gene knockout may silence it. Thus, *merlin*'s duplicate function can be used to clone the reaction and the same, above zero, flux value should be set in both thresholds. The purpose of fixing the thresholds is guaranteeing that every simulation with the GiSMo produces enough ATP for cellular maintenance, besides growth.

The growth ATP requirements are added to the GiSMo by including the hydrolysis of ATP in the biomass equation with

predetermined stoichiometric values, proportional to the requirements of the macromolecules biosynthesis.

Curation per Pathway

The reconstruction of a functional GiSMo must guarantee that routes from known carbon, nitrogen, sulfur, and phosphorous sources to the biomass' constituents and other identified products are present and gapless. The curation of the draft model can be performed while performing several of the steps described above. The first step of the curation should be analyzing the main pathways, to ensure the connectivity of the GiSMo. The main pathways may vary with the case study organism, depending on carbon sources and auxotrophies. Still, the following pathways should be examined, when available, as they are associated with the production of important metabolites: glycolysis, pentose phosphate, citrate cycle, (every) amino acid biosynthesis, pantothenate and CoA biosynthesis, fatty acid biosynthesis, glycerolipid metabolism, glycerophospholipid metabolism, purine metabolism, pyrimidine metabolism, folate biosynthesis and ubiquinone, and other terpenoid-quinone biosynthesis. Other pathways that may be of interest are: biotin metabolism, riboflavin metabolism, lipopolysaccharide biosynthesis, and steroid biosynthesis. As every GiSMo is distinct, pathways not listed here will also have to be curated.

The curation of the pathways should focus on finding a metabolic path from a starting substrate to one or more final products, within the several alternate *viae* in a metabolic pathway. For instance, in glycolysis the most common substrate is glucose, and a set of reactions converting this metabolite into pyruvate should be mapped. The path from pyruvate onward will highly depend on the type of organism. Though a part of the flux will in most cases be redirected to the citrate cycle, possible routes for the remainder include conversion to and excretion of lactate, ethanol, or acetate. Other curation actions include confirmation of the reversibility and direction of each reaction in the selected *viae*.

merlin allows selecting a KEGG pathway and using the "*draw in browser*" button to open a browser window where the pathway is opened and the enzymes and/or reactions available for such pathways in the GiSMo are colored. If the EC number, in the browser window, is written in green then the reaction is included in the GiSMo and the enzyme promoting the reaction in such a pathway is the same as the one highlighted. If it is written in dark blue then, though the reaction is present in the model, it is being promoted by an enzyme other than the one available for this via. Moreover, if the "*find unconnected reactions*" tool was used, then dead-end reactions and metabolites, in the browser window, will be colored (as in the reactions' panel) in red. Likewise, EC numbers associated to reaction connected to dead-end reactions will be colored in light blue.

These tools allow easily detecting gaps in the GiSMo which should be examined. A gap in the most obvious route of the pathway is, in most cases, probably related to annotation errors. Hence, other sources (such as MetaCyc or KEGG's own annotation) should be verified and annotations proposed by these analyzed to identify potential gap-filling options. Literature should also be considered as metabolites identified as dead-ends may be auxotrophic needs of the organisms and should be added to environmental conditions.

Users are encouraged to remove "*Metapathways*," such as "*Metabolic Pathways*," "*Biosynthesis of secondary metabolites*" and similar, from the GiSMo as the contribution of these to comprehension of the organisms' metabolism and the curation efforts is reduced. Other pathways that should be removed to keep the GiSMo as simple as possible include all other *viae* not available in the case study, like the "*Carbon fixation in photosynthetic organisms*" pathway for prokaryotes. This operation is easily performed in *merlin* by clicking the "*remove*" button in the "pathway" tab available in the "*reactions*" panel.

Drains

The last step of converting a GenNet to a GiSMo is adding drain boundaries to the model. Drains are exchange constraints set to mimic the environment conditions in which organisms live and grow. Usually, these constraints are set for external metabolites, thus allowing the GiSMo to control the uptake and excretion of metabolites. The constraints are pairs of lower and upper restrictions that limit the input and output fluxes of selected metabolites. To mimic the environmental conditions in which the organisms are grown, the drains of the metabolites that compose the growth media should be set to the fluxes measured during the experiment, yet with negative signs to label metabolites as consumed.

merlin provides an operation to automatically add drains to every external metabolite, through the creation of a reaction with the compound as a reactant, no products and setting the lower boundary to 0 and the upper boundary to 999999. Hence, drains are clustered in a pseudo-pathway to ease the process of finding drains for metabolites that compose the growth media.

3.4.4 Export

In *merlin*, GiSMos are kept in a database and should be exported to be validated and utilized in other platforms. Reactions can be exported in tabular format by pressing the "*xls tabbed file*" button in the "*reactions*" panel, but this operation will only export the panel contents. To export a fully formatted model in the SBML format with identifiers for reactions and metabolites, GPR rules, and MIRIAM annotations, users should go to "*model>export to SBML*" menu. *merlin* allows setting various options for exporting the model, such as whether generating or not a file with the

metabolites formulae, validating the model online (may delay the operation), the biomass reaction name, the level and version of the SBML file, and export folder and file name.

3.5 Validation

The SBML file containing the GiSMo can be imported into several platforms like OptFlux and COBRA [11] to perform simulations and validate the GiSMo, by assessing the simulation results to experimental data. If the results of the growth rate and byproducts fluxes do not match, these steps should be repeated iteratively.

Validation tests that should be performed include, but should not be limited to, the following:

1. Spontaneous growth;
2. Auxotrophies;
3. Gene essentiality;
4. Alternative basic elements sources;
5. Assess growth rate.

These tests can be classified as qualitative and quantitative. The first four tests belong to the qualitative tests category, while the last one is quantitative and evaluates the GiSMo's ability to predict flux rates.

The first test is assessed by performing simulations with null environmental conditions that is by setting the lower boundary of all drains to zero. This test will evaluate if the GiSMo is able to grow without in silico growth media. The second test is performed by comparing simulations, in which the first is performed defining a minimal medium in the environment conditions and the second using the same medium except for the removal of a metabolite for which the case study is known to be auxotrophic. The third test involves mimicking growth for conditions in which genes are essential for growth, namely media that do not provide a certain metabolite, which must be biosynthesized. The fourth test depends on the availability of studies or tests such as Biolog phenotype microarrays that test for different conditions, providing data on which metabolites can be used as sources of different elements, namely carbon, nitrogen, sulfur, and phosphorous. Finally, the last test involves comparing experimental to in silico fluxes. This evaluates the quality of the model predictions, for the tested experimental conditions and is dependent on the availability of experimental data.

3.5.1 Experimental Data

These data should be, preferably, obtained from chemostat and fluxes normalized to $\frac{\text{mmol}}{h \times g_{\text{DW}}}$. Fluxes should be determined for the carbon, nitrogen, sulfur, and phosphorous sources (e.g., glucose, ammonium, oxygen, etc.) and for all fermentation byproducts (or at least the main ones such as carbon dioxide, ethanol, acetate, formate, etc.). Finally, the biomass growth rate is also required for the validation of the model.

Though the normalization process is straightforward for chemostat experiments, these data can also be obtained from batch experiments as shown in Sauer et al. [45] (*see* **Note 5** for a summarized description of this process).

4 Notes

1. *merlin*'s interface

   merlin's interface has three main components, as shown in Fig. S6 of the supplemental material. The first one (blue square—Fig. S6 of the supplemental material) is the *merlin*'s operations bar, where most procedures are called and which is divided into four tabs. The "*project*" operations' tab is where a project can be created, loaded, saved, and deleted from the database. The "*database*" operations tab allows creating new internal databases or clean existing ones. The "*annotation*" tab includes three main sub-tabs. The first one, "*enzymes*," offers several options for identifying and annotating enzymes on a genome. Likewise, "*TRIAGE*" does the same but for transport proteins and transport reactions. The third sub-tab, "*compartments,*" allows loading and processing compartments' prediction reports from different tools. Finally, the "*model*" operations tab offers several tools for developing and curating a GiSMo.

   merlin's clipboard (green square—Fig. S6 of the supplemental material) offers an intuitive schema for accessing the internal database data. It clusters information per project and within each project it further groups it into "*model*" and "*annotation*" data. Whereas the latter group provides instances for curating enzymes, transporters, and compartments annotations, the former is used to assemble the GiSMo by editing its subcomponents, namely genes, proteins, reactions, metabolites, and pathways.

   Lastly, the visualization area (red square—Fig. S6 of the supplemental material) is where the components selected in the clipboard are shown and in most cases, operations are provided for editing data in the database for curation of the annotation or the model itself. Most "*views*" allow inserting, editing, and removing information into/from the database. Furthermore, annotations' "*views*" also have operations for integrating these data with the model database.

2. GenBank versus RefSeq annotations

   The main difference between these is the identification of the genes products, in the FASTA format amino acids (.faa) files, which in RefSeq may contain records identified by WP_ + 9 digits + version number (e.g., WP_000000001.1). These records represent single, nonredundant, protein sequences

annotated on several different RefSeq genomes from different or the same (different strains) species, as shown in Fig. S7 of the supplemental material. Hence, the identifiers in these CDSs are generic and cannot be mapped to a specific genome.

3. Enzymes annotation

   (a) *Similarities Searches*

   Though gene or protein sequences with high similarity may have arisen from a common ancestor, the opposite is not always true, that is, homologous sequences do not always exhibit significant sequence similarity. Hence, it is possible to determine if two sequences are homologous by performing similarity searches, but it is not possible to sustain that two sequences are not homologous based on similarity searches to databases alone [24].

   When performing BLAST searches, each data source (NCBI or EBI) has its own databases with different accession numbers. Thus, when databases from the same data source are used in different tools, *merlin* may allow combining the results of both tools. For instance, if the database used in the NCBI BLAST is "*swissprot*," *merlin* allows performing a HMMER search to the same database and combine the results for annotating the genome. Likewise, any database selected for performing the EBI BLAST would be able to combine with the HMMER search to "*swissprot*," as the accession numbers are compatible. A list of compatibilities can be found in Table S2, of the supplemental material. Another important parameter (for BLAST) is the substitution-scoring matrix (matrix containing values relative to the probability that a given amino acid mutates into another amino acid for all pairs of amino acid). The matrix used in the similarity search may affect the results as the selection of the best matrix depends on the size of each sequence.

   (b) *EC Numbers Update*

   Genes initially annotated with partial EC numbers (e.g., 1.1.-.-) may eventually be assigned with complete EC numbers. These annotations may happen due to known enzymatic activities not yet assigned with EC codes or sequences showing evidence of generic enzymatic activities. In the former case, the function of the gene may be already known, but a complete EC number, at the time of the annotation, is not available. In the latter case, although the specific function (for instance a specific substrate) is not known, the protein sequence shows signs indicating that the gene encodes an enzyme that should belong to a given EC family.

Nonetheless, complete EC numbers can later be assigned with new codes. These are cases in which an EC number is discontinued and the enzymatic activity(ies) moved into a (sometimes more than one) new code, e.g., EC 1.1.1.128—created 1972, modified 1976, deleted 2012, L-idonate 2-dehydrogenase. The reaction described is now covered by EC 1.1.1.264.

4. *TAD Annotations*

Whenever TRIAGE's alignments are integrated with TAD, *merlin* creates a report at the "*temp*" folder, within *merlin*'s main folder, with the extension "*.out*." The first two records of the report are examples of annotated entries. The following rows are TCDB records unavailable in TAD, which should be curated for posterior inclusion in the internal database.

(a) *Report Structure*

This report is organized in 18 columns (UniProt ID, TCDB ID, TCDB family, TCDB description, affinity, Transport type, TCDB location, YTPDB gene, YTPDB description, YTPDB type, YTPDB metabolites, YTPDB location, **TC #, direction, metabolite, reversibility, reacting metabolites, and equation**). Information in columns 1–7 (UniProt and TCDB identifiers, protein family and description, transport affinity, type, and location) is retrieved from TCDB. Columns 8–12 contain information retrieved from the YTPDB [46] (http://ytpdb.biopark-it.be/ytpdb/index.php/Main_Page), a database specialized in the classification and annotation of yeast transporters. This resource is very useful because, unlike TCDB, it has a dedicated field for the substrate being transported. Hence, organisms with similarities to *Saccharomyces cerevisiae* will have another resource for inferring transport annotations. TRIAGE retrieves information on the yeast gene, protein, transport type, metabolites, and location from this database. Whereas the purpose of these columns (1–12) is to provide information, the bold columns (columns 13–18) must be filled out for adding new entries in TRIAGE's TAD.

The *TC #* column is automatically populated by TRIAGE. The next column (direction) has controlled vocabulary, that is, there are a finite number of options to fill this column depending on the type of transport, being those: *in* or *out* (for uniport), *in:in* or *out:out* (for symport), and *in // out* (for antiport). If a protein has different behavior depending on the substrate, these can be combined by adding two vertical bars (||) between them (e.g., in || in:in).

Column 15 holds the metabolites names. In this case, the vocabulary is not controlled, though using KEGG or

ChEBI nomenclature is encouraged, as TRIAGE uses an internal algorithm to assign KEGG and ChEBI identifiers to the metabolites. This algorithm uses ChEBI hierarchy to identify metabolites' second-generation elements. For instance, α-D-glucose (CHEBI:17925) and β-D-glucose (CHEBI:15903) are both second-generation elements of D-glucose (CHEBI:4167). For a detailed description of the algorithm, *see* [28]. Each metabolite transported by a given protein is separated by a semicolon (;) and (in cases where it applies) by the same symbol of the transport type. For instance, the symport of glycine with hydrogen and the symport of alanine with glycine with hydrogen by the 2.A.25.1.1 "*Alanine (or glycine):Na+ symporter*" (P30144) is denoted as "*alanine; glycine : Na+*." In these cases, every element on the left of the colon will be co-transported with every element on the right. Likewise, the antiport of oxalate with formate by the 2.A.1.11 oxalate formate antiporter (Q51330) is set as "oxalate // formate." Since the direction was set as "*in // out*", the oxalate will go in and the formate outward. However, as the next column identifies this transport as reversible, the metabolites' directions can be in fact reversed. Moreover, due to format parsing limitations (metabolite names often contain numbers and other characters) stoichiometry is represented by symport of the same molecule. For instance, the transport of two protons has a direction of *in* and the metabolites are *proton: proton*. Additionally, cases in which transporters may function with different mechanisms (||) should also reflect this separation in the metabolites column. For instance, 2.A.29.10.5 (P40556) may "*import NAD^+ into mitochondria by unidirectional transport or by exchange with intramitochondrially generated (d)AMP and (d)GMP.*" In TAD, the direction for this transport will be in || in // out and the metabolites entry will be NAD || NAD// AMP; GMP. The subsequent column determines the reversibility of the transport reaction. It should be considered true by default, except in some cases discussed later, unless it is determined otherwise by the transporters annotation pipeline.

Some transport reactions may require energy provided by the chemical reactions, such as ATP hydrolysis. This information is added to TAD in column 17, using a specific notation. A exergonic reaction, such as ATP + water ⇒ ADP + P_i is represented as "*1:ATP; 1:water || 1:ADP; 1:orthophosphate.*" In this case, the limited number of metabolites allows a straightforward representation of the stoichiometry and compound names. These reactions are often irreversible and the vertical bars separate the reactants from products. Moreover, some transport reactions involve

modifications of substrates like, for instance, the phosphorylation of L-ascorbate to L-ascorbate-6-phosphate. In these cases, the transported metabolites field should be set to two hyphens ("--").

The last column is the family (or subfamily) equation. This field can be very informative, as in some cases (e.g., Quinol + 1/2 O_2 + $\underline{4H^+}$ [in] ⇒ Quinone + H_2O + $\underline{4H^+}$ [out]) it determines the direction, reversibility, and metabolites transported by a specific family of transporters (e.g., irreversible export of hydrogen promoted by the oxidation of a quinol to quinone).

(b) *Workflow*

A detailed explanation of the flowchart developed for annotating the TAD report and presented in Fig. S8 of the supplemental material is provided next. The first step is analyzing the TCDB entry description (Fig. S8 of the supplemental material—Level A) to identify the transported metabolite(s), transport type, and reversibility. Usually, this description is copied from publications, hence other information like reacting metabolites and equation is not available in this field. Information retrieved from this field is specific for each protein and should be favored regarding evidence collected from the next steps. The next step is checking the UniProt entry (Fig. S8 of the supplemental material—Level B) to find information on the reacting metabolites and fields unsuccessfully annotated in the previous step (metabolite(s), transport type, and reversibility). Likewise, after analyzing the UniProt entry, the TCDB sub-family (or family) (Fig. S8 of the supplemental material—Level C) should be checked to retrieve the transport equation. If this information cannot be retrieved from the sub-family (or family [47]), it can be searched in the level above (Fig. S8 of the supplemental material—Level D)—the TC family (or superfamily or class, according to the hierarchy of the TC number [47]). The default values for these fields, which should be set whenever data cannot be found, are direction -> in, metabolite -> unknown, reversibility -> true, reaction metabolites -> -- and equation -> --.

5. Determination of physiological parameters from batch experiments

Estimating physiological parameters from batch data involves identifying the exponential growth phase. This step can be performed by performing the log-linear regression of the biomass concentration vs. time, in which the growth rate (μ) is the regression coefficient. The regression should only include the phase in which growth is exponential, which is the phase wherein

the plot produces a straight line as shown in Fig. S9 of the supplemental material (a).

The specific consumption/production rates (q_S/q_P) are the differential variation of the substrate/product with time normalized to the biomass concentration. Hence, these are calculated by performing the linear regression of the substrate/product concentration vs. the biomass concentration divided by μ $\left(\frac{X}{\mu}\right)$ in which the rate q_S/q_P is the regression coefficient, as shown in Fig. S9 of the supplemental material (b). Substrates will have negative values whereas products will have positive rates.

The consumption rates should be set in the environmental conditions before simulations (to override the default values) and the simulation output assessed for the production rates to evaluate the GiSMo.

References

1. Otero JM, Nielsen J (2010) Industrial systems biology. Biotechnol Bioeng 105:439–460. https://doi.org/10.1002/bit.22592
2. Kitano H (2002) Systems biology: a brief overview. Science 295:1662–1664. https://doi.org/10.1126/science.1069492
3. Dias O, Rocha I (2015) Systems biology in fungi. In: Paterson R (ed) Mol. Biol. Food water borne mycotoxigenic mycotic fungi. CRC Press, Boca Raton, FL, pp 69–92
4. gismo Meaning in the Cambridge English Dictionary. http://dictionary.cambridge.org/dictionary/english/gismo#translations. Accessed 13 Apr 2017
5. Gizmo definition and meaning | Collins English Dictionary. https://www.collinsdictionary.com/dictionary/english/gizmo. Accessed 13 Apr 2017
6. Thiele I, Palsson BØ (2010) A protocol for generating a high-quality genome-scale metabolic reconstruction. Nat Protoc 5:93–121. https://doi.org/10.1038/nprot.2009.203
7. Dias O, Rocha M, Ferreira EC, Rocha I (2015) Reconstructing genome-scale metabolic models with merlin. Nucleic Acids Res 43:3899–3910. https://doi.org/10.1093/nar/gkv294
8. Henry CS, DeJongh M, Best AA, Frybarger PM, Linsay B, Stevens RL (2010) High-throughput generation, optimization and analysis of genome-scale metabolic models. Nat Biotechnol 28:977–982. https://doi.org/10.1038/nbt.1672
9. Hucka M, Finney A, Sauro HM, Bolouri H, Doyle JC, Kitano H, Arkin AP, Bornstein BJ, Bray D, Cornish-Bowden A, Cuellar AA, Dronov S, Gilles ED, Ginkel M, Gor V, Goryanin II, Hedley WJ, Hodgman TC, Hofmeyr J-H, Hunter PJ, Juty NS, Kasberger JL, Kremling A, Kummer U, Le Novère N, Loew LM, Lucio D, Mendes P, Minch E, Mjolsness ED, Nakayama Y, Nelson MR, Nielsen PF, Sakurada T, Schaff JC, Shapiro BE, Shimizu TS, Spence HD, Stelling J, Takahashi K, Tomita M, Wagner J, Wang J (2003) The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. Bioinformatics 19:524–531. https://doi.org/10.1093/bioinformatics/btg015
10. Rocha I, Maia P, Evangelista P, Vilaça P, Soares S, Pinto JP, Nielsen J, Patil KR, Ferreira EC, Rocha M (2010) OptFlux: an open-source software platform for in silico metabolic engineering. BMC Syst Biol 4:45. https://doi.org/10.1186/1752-0509-4-45
11. Schellenberger J, Que R, Fleming RMT, Thiele I, Orth JD, Feist AM, Zielinski DC, Bordbar A, Lewis NE, Rahmanian S, Kang J, Hyduke DR, Palsson BØ (2011) Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox v2.0. Nat Protoc 6:1290–1307. https://doi.org/10.1038/nprot.2011.308
12. Le Novère N, Finney A, Hucka M, Bhalla US, Campagne F, Collado-Vides J, Crampin EJ, Halstead M, Klipp E, Mendes P, Nielsen P, Sauro H, Shapiro B, Snoep JL, Spence HD, Wanner BL (2005) Minimum information requested in the annotation of biochemical models (MIRIAM). Nat Biotechnol 23:1509–1515. https://doi.org/10.1038/nbt1156
13. Glez-Peña D, Reboiro-Jato M, Maia P, Rocha M, Díaz F, Fdez-Riverola F (2010) AIBench: a rapid application development

framework for translational research in biomedicine. Comput Methods Programs Biomed 98:191–203. https://doi.org/10.1016/j.cmpb.2009.12.003

14. UniProt Consortium (2015) UniProt: a hub for protein information. Nucleic Acids Res 43: D204–D212. https://doi.org/10.1093/nar/gku989
15. Boutet E, Lieberherr D, Tognolli M, Schneider M, Bansal P, Bridge AJ, Poux S, Bougueleret L, Xenarios I (2016) UniProtKB/Swiss-Prot, the Manually Annotated Section of the UniProt KnowledgeBase: how to use the entry view. Methods Mol Biol 1374:23–54. https://doi.org/10.1007/978-1-4939-3167-5_2
16. Sayers EW, Barrett T, Benson DA, Bryant SH, Canese K, Chetvernin V, Church DM, DiCuccio M, Edgar R, Federhen S, Feolo M, Geer LY, Helmberg W, Kapustin Y, Landsman D, Lipman DJ, Madden TL, Maglott DR, Miller V, Mizrachi I, Ostell J, Pruitt KD, Schuler GD, Sequeira E, Sherry ST, Shumway M, Sirotkin K, Souvorov A, Starchenko G, Tatusova TA, Wagner L, Yaschenko E, Ye J (2009) Database resources of the National Center for Biotechnology Information. Nucleic Acids Res 37:D5–15. https://doi.org/10.1093/nar/gkn741
17. Schomburg I, Chang A, Schomburg D (2002) BRENDA, enzyme data and metabolic information. Nucleic Acids Res 30:47–49
18. Ogata H, Goto S, Sato K, Fujibuchi W, Bono H, Kanehisa M (1999) KEGG: Kyoto encyclopedia of genes and genomes. Nucleic Acids Res 27:29–34. https://doi.org/10.1093/nar/27.1.29
19. Lipman DJ, Pearson WRW (1985) Rapid and sensitive protein similarity searches. Science 227:1435–1441. PMID: 2983426
20. Federhen S (2012) The NCBI Taxonomy database. Nucleic Acids Res 40:D136–D143. https://doi.org/10.1093/nar/gkr1178
21. Kitts PA, Church DM, Thibaud-Nissen F, Choi J, Hem V, Sapojnikov V, Smith RG, Tatusova T, Xiang C, Zherikov A, DiCuccio M, Murphy TD, Pruitt KD, Kimchi A (2016) Assembly: a resource for assembled genomes at NCBI. Nucleic Acids Res 44: D73–D80. https://doi.org/10.1093/nar/gkv1226
22. mysql-server - Linux Mint Community. https://community.linuxmint.com/software/view/mysql-server. Accessed 13 Apr 2017
23. MySQL :: About MySQL. https://www.mysql.com/about/. Accessed 13 Apr 2017
24. Pearson WR (2013) An introduction to sequence similarity ("Homology") searching. In: Curr. Protoc. Bioinforma. John Wiley & Sons, Inc., Hoboken, NJ, pp 3.1.1–3.1.8
25. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ (1990) Basic local alignment search tool. J Mol Biol 215:403–410. https://doi.org/10.1016/S0022-2836(05)80360-2
26. Finn RD, Clements J, Eddy SR (2011) HMMER web server: interactive sequence similarity searching. Nucleic Acids Res 39: W29–W37. https://doi.org/10.1093/nar/gkr367
27. Magrane M, Consortium UP (2011) UniProt Knowledgebase: a hub of integrated protein data. Database. https://doi.org/10.1093/database/bar009
28. Dias O, Gomes D, Vilaca P, Cardoso J, Rocha M, Ferreira E, Rocha I (2017) Genome-wide semi-automated annotation of transporter systems. IEEE/ACM Trans Comput Biol Bioinforma 14:443. https://doi.org/10.1109/TCBB.2016.2527647
29. Yu NY, Wagner JR, Laird MR, Melli G, Rey S, Lo R, Dao P, Sahinalp SC, Ester M, Foster LJ, Brinkman FSL (2010) PSORTb 3.0: improved protein subcellular localization prediction with refined localization subcategories and predictive capabilities for all prokaryotes. Bioinformatics 26:1608–1615. https://doi.org/10.1093/bioinformatics/btq249
30. Goldberg T, Hecht M, Hamp T, Karl T, Yachdav G, Ahmed N, Altermann U, Angerer P, Ansorge S, Balasz K, Bernhofer M, Betz A, Cizmadija L, Do KT, Gerke J, Greil R, Joerdens V, Hastreiter M, Hembach K, Herzog M, Kalemanov M, Kluge M, Meier A, Nasir H, Neumaier U, Prade V, Reeb J, Sorokoumov A, Troshani I, Vorberg S, Waldraff S, Zierer J, Nielsen H, Rost B (2014) LocTree3 prediction of localization. Nucleic Acids Res 42:W350–W355. https://doi.org/10.1093/nar/gku396
31. Saier MH (2000) A functional-phylogenetic classification system for transmembrane solute transporters. Microbiol Mol Biol Rev 64:354–411
32. Sonnhammer EL, von Heijne G, Krogh A (1998) A hidden Markov model for predicting transmembrane helices in protein sequences. Proc Int Conf Intell Syst Mol Biol 6:175–182
33. Käll L, Krogh A, Sonnhammer ELL (2004) A combined transmembrane topology and signal peptide prediction method. J Mol Biol 338:1027–1036. https://doi.org/10.1016/j.jmb.2004.03.016

34. Moller S, Croning MDR, Apweiler R, Möller S (2001) Evaluation of methods for the prediction of membrane spanning regions. Bioinformatics 17:646–653. https://doi.org/10.1093/bioinformatics/17.7.646
35. Smith TF, Waterman MS (1981) Identification of common molecular subsequences. J Mol Biol 147:195–197. https://doi.org/10.1016/0022-2836(81)90087-5
36. Gardy JL, Brinkman FSL (2006) Methods for predicting bacterial protein subcellular localization. Nat Rev Microbiol 4:741–751. https://doi.org/10.1038/nrmicro1494
37. Ma H, Zeng A-P (2003) Reconstruction of metabolic networks from genome data and analysis of their global structure for various organisms. Bioinformatics 19:270–277. https://doi.org/10.1093/bioinformatics/19.2.270
38. Stelzer M, Sun J, Kamphans T, Fekete SP, Zeng A-P (2011) An extended bioreaction database that significantly improves reconstruction and analysis of genome-scale metabolic networks. Integr Biol (Camb) 3:1071–1086. https://doi.org/10.1039/c1ib00008j
39. Tanabe M, Kanehisa M (2012) Using the KEGG database resource. Curr Protoc Bioinformatics Chapter 1:Unit1.12. doi: https://doi.org/10.1002/0471250953.bi0112s38
40. Varma A, Palsson BO (1993) Metabolic capabilities of Escherichia coli II. Optimal growth patterns. J Theor Biol 165:503–522. https://doi.org/10.1006/jtbi.1993.1203
41. Santos ST (2013) Development of computational methods for the determination of biomass composition and evaluation of its impact in genome-scale models predictions. Universidade do Minho
42. Santos S, Rocha I (2016) Estimation of biomass composition from genomic and transcriptomic information. J Integr Bioinform. https://doi.org/10.2390/biecoll-jib-2016-285
43. Xavier JC, Patil KR, Rocha I (2017) Integration of biomass formulations of genome-scale metabolic models with experimental data reveals universally essential cofactors in prokaryotes. Metab Eng 39:200. https://doi.org/10.1016/j.ymben.2016.12.002
44. Dias O, Pereira R, Gombert AK, Ferreira EC, Rocha I (2014) iOD907, the first genome-scale metabolic model for the milk yeast Kluyveromyces lactis. Biotechnol J 9:776–790. https://doi.org/10.1002/biot.201300242
45. Sauer U, Lasko DR, Fiaux J, Hochuli M, Glaser R, Szyperski T, Wuthrich K, Bailey JE (1999) Metabolic flux ratio analysis of genetic and environmental modulations of escherichia coli central carbon metabolism. J Bacteriol 181:6679–6688
46. Brohée S, Barriot R, Moreau Y, André B (2010) YTPdb: a wiki database of yeast membrane transporters. Biochim Biophys Acta 1798:1908–1912. https://doi.org/10.1016/j.bbamem.2010.06.008
47. Saier MH, Reddy VS, Tamang DG, Västermark A (2014) The transporter classification database. Nucleic Acids Res 42:D251–D258. https://doi.org/10.1093/nar/gkt1097
48. Caspi R, Altman T, Dreher K, Fulcher CA, Subhraveti P, Keseler IM, Kothari A, Krummenacker M, Latendresse M, Mueller LA, Ong Q, Paley S, Pujar A, Shearer AG, Travers M, Weerasinghe D, Zhang P, Karp PD (2012) The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. Nucleic Acids Res 40:D742–D753. https://doi.org/10.1093/nar/gkr1014

Author Queries

Chapter No.: 1 417588_1_En

Query Refs.	Details Required	Author's response
AU1	Please provide appropriate chapter cross reference instead of "Chapter XX".	
AU2	Please check and confirm the captions and artwork of figures 1–3.	
AU3	Please note that subheadings are not allowed under Notes heading. So please check the following list presentation for correctness.	
AU4	Please check and confirm the captions and artwork of supplement figures 1–9.	