LGEM⁺: A First-Order Logic Framework for Automated Improvement of Metabolic Network Models Through Abduction

Citation for the original published paper (version of record):
Gower, A., Korovin, K., Brunnsåker, D. et al (2023). LGEM⁺: A First-Order Logic Framework for Automated Improvement of Metabolic Network
Models Through Abduction. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 14276 LNAI: 628-643.
http://dx.doi.org/10.1007/978-3-031-45275-8_42

(article starts on next page)

LGEM$^+$: A First-Order Logic Framework for Automated Improvement of Metabolic Network Models Through Abduction

Alexander H. Gower[1(✉)], Konstantin Korovin[2], Daniel Brunnsåker[1], Ievgeniia A. Tiukova[1,3], and Ross D. King[1,4,5]

[1] Chalmers University of Technology, Gothenburg, Sweden
{gower,danbru,tiukova,rossk}@chalmers.se
[2] The University of Manchester, Manchester, UK
Konstantin.Korovin@manchester.ac.uk
[3] KTH Royal Institute of Technology, Stockholm, Sweden
[4] Cambridge University, Cambridge, UK
[5] Alan Turing Institute, London, UK

Abstract. Scientific discovery in biology is difficult due to the complexity of the systems involved and the expense of obtaining high quality experimental data. Automated techniques are a promising way to make scientific discoveries at the scale and pace required to model large biological systems. A key problem for 21st century biology is to build a computational model of the eukaryotic cell. The yeast *Saccharomyces cerevisiae* is the best understood eukaryote, and genome-scale metabolic models (GEMs) are rich sources of background knowledge that we can use as a basis for automated inference and investigation.

We present LGEM$^+$, a system for automated abductive improvement of GEMs consisting of: a compartmentalised first-order logic framework for describing biochemical pathways (using curated GEMs as the expert knowledge source); and a two-stage hypothesis abduction procedure.

We demonstrate that deductive inference on logical theories created using LGEM$^+$, using the automated theorem prover iProver, can predict growth/no-growth of *S. cerevisiae* strains in minimal media. LGEM$^+$ proposed 2094 unique candidate hypotheses for model improvement. We assess the value of the generated hypotheses using two criteria: (a) genome-wide single-gene essentiality prediction, and (b) constraint of flux-balance analysis (FBA) simulations. For (b) we developed an algorithm to integrate FBA with the logic model. We rank and filter the hypotheses using these assessments. We intend to test these hypotheses using the robot scientist Genesis, which is based around chemostat cultivation and high-throughput metabolomics.

Keywords: Scientific discovery · artificial intelligence · systems biology · metabolic modelling · first-order logic · automated theorem proving

A. Bifet et al. (Eds.): DS 2023, LNAI 14276, pp. 628–643, 2023.
https://doi.org/10.1007/978-3-031-45275-8_42

1 Introduction

An important aspect of modern biology is improving our understanding of cellular processes, and the complex interactions between genes, proteins and chemical species. Systems biology is the research discipline that tackles this complexity. *Saccharomyces cerevisiae*, commonly known as "baker's yeast", is an excellent model organism used for the study of eukaryote biology. This is due to the availability of tools for easy genetic manipulation, and low cultivation cost, enabling targeted experiments to characterise the system. *S. cerevisiae*'s was the first eukaryotic genome to be fully sequenced [10] and there is a wealth of knowledge about the gene functions, many of which are conserved or expected to have equivalents in other eukaryotes, including humans [5]. Metabolic network models (MNMs) represent the cellular biochemistry of an organism and the related action of enzymatic genes; such models which seek to integrate knowledge from the entire organism are known as genome-scale metabolic models (GEMs).

The scientific discovery problem we address is to add knowledge to or reduce *S. cerevisiae* GEMs such that quality is increased. Model quality in GEMs is multi-faceted—desirable properties of a model include: predictive power; metabolic network coverage; and parsimony. There are trade-offs between different desirable properties [11]. Foremost, however, is the predictive power of the GEM. Ultimately the aim is to understand the entities, mechanisms and adaptations that govern yeast growth in different environments.

Given a draft model, improvement consists broadly of three stages: hypothesise refinements to the model; conversion of refined model to a format suitable for simulation; and evaluation based on experimental evidence and internal consistency [24]. Repetition of these stages consists a scientific discovery process. Evaluation is dependent on executing simulations using a mathematical formalism, however optimising a model for a specific formalism is not the objective—any improvements that are made to a GEM within a certain framework should translate to improvements in the underlying knowledge.

Challenges for the future of genome-scale modelling of *S. cerevisiae* include: improving annotation; removing noise from low-confidence components; and adding reactions to eliminate so-called "dead-end" compounds [1]. To multiply the efforts of human researchers, previous work has investigated automating parts of the scientific method. GrowMatch was a technique developed to resolve inconsistencies between predictions and experimental observations of single-gene mutant strains of *Escherichia coli* [15]. Other approaches to metabolic network gap-filling have exploited answer-set programming, the most complete of which is MENECO which is designed to efficiently identify candidate additions to draft network models [19].

Logical inference can be applied to generate and improve metabolic models: induction allows us to generalise models from data; given a theory we can draw conclusions using deduction; and abduction enables us to form hypotheses to improve consistency with empirical data. In this work we use first-order logic (FOL) to simulate the metabolic network, an approach first proposed in 2001 [20]. A FOL model was used to generate functional genomics hypotheses then

tested by a robot scientist [13]; logical induction and abduction was applied to identify inhibition in metabolic pathways after introduction of toxins [23]; and an FOL model constructed in Prolog using the GEM iFF708 [7] as the background knowledge source was used to predict single-gene essentiality [25]. Huginn is a tool that uses abductive logic programming (ALP), and demonstrates the ability to improve metabolic models and suggest *in vivo* experiments [21].

A core advantage of our model—both over these previous FOL approaches that used Prolog, and over bespoke algorithmic methods such as MENECO—is that we use first order theorem provers (FOTPs) to perform deductive and abductive inference. This removes a large part of the burden of abductive algorithm design and simulation. For the reasoning tasks we use the FOTP iProver [14]. We extended iProver to include abduction inference. iProver is a saturation-based theorem prover that saturates via consequence finding algorithms which are well-suited to abduction [22]. Other declarative programming techniques that we tried, for example Prolog, and SAT solvers based on backtrack search algorithms (e.g. CDCL), lacked certain features that enable abduction. Using FOTPs will also allow us to combine different deduction and abduction strategies.

Furthermore, our model is capable of deductive and abductive reasoning at scales far greater than previous FOL approaches. The ability to reason at scale is particularly important for the automation of scientific discovery in eukaryotic biology where the domain is complex and data are expensive to generate.

One current limitation of our FOL framework is that we do not include information on reaction stoichiometry. To integrate quantitative modelling, we propose in this paper a method to combine flux balance analysis (FBA) and logical inference to validate metabolic pathway configurations found by LGEM$^+$.

The main contributions of LGEM$^+$ as presented in this paper are: (1) a compartmentalised FOL model of yeast metabolism; (2) a two-stage method for the abduction of novel hypotheses on improved models; (3) scalable methods for evaluating these models and hypotheses; and (4) an algorithm to integrate FBA with abductive reasoning.

2 Methods

2.1 The First-Order Logic Framework

We chose FOL as the language to express the mechanics of the biochemical pathways. FOL allows for a rich expression of knowledge about biological processes, such as reactions and enzyme catalysis. We use FOL to express our knowledge about how entities are known to interact, for example that a reaction has substrates and products, and possibly some required enzyme. By contrast, a propositional logic framework would be unable to express these higher level concepts and as such would be less suitable for abduction. The method and model we design is independent of the specific network, meaning that although here we apply LGEM$^+$ to *S. cerevisiae*, this modelling framework could equally well be applied to other organisms.

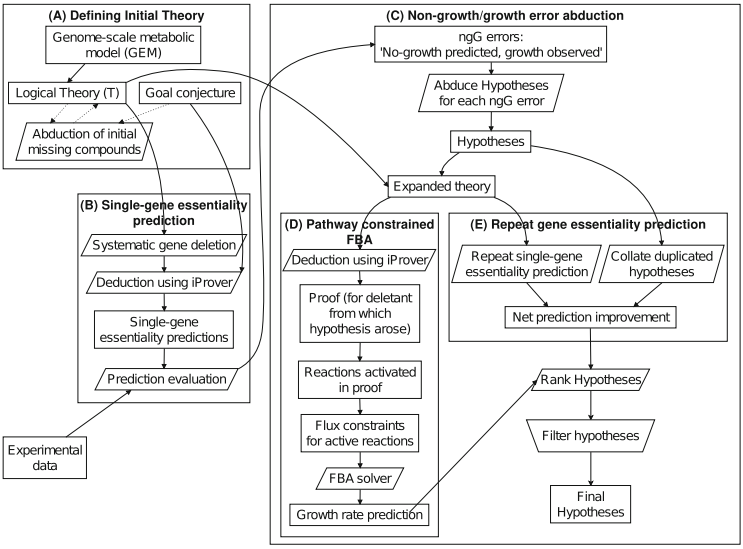

Fig. 1. Processes in LGEM$^+$. **(A)** defining the logical theory, including abduction of missing compounds to enable viability of base strain; **(B)** single-gene essentiality prediction; **(C)** abduction of hypotheses from *ngG* errors; **(D)** using FBA to assess viability of each hypothesis; and **(E)** repeating single-gene deletion to assess viability of each hypothesis.

We define five predicates in the first-order language: met\2, gn\1, pro\1, enz\1, and rxn\1. The semantic interpretation of these predicates is outlined in Table 1. Here a cellular "compartment" refers to a component of the cellular anatomy, e.g. mitochondrion, nucleus or cytosol.

Table 1. Predicates used in the logical theory of yeast metabolism. Forward and reverse reactions are represented separately in the model, thus a "positive flux" through a reversed reaction indicates the reaction flux is negative.

Predicate	Arguments	Natural language interpretation
met\2	metabolite, compartment	"Metabolite X is present in cellular compartment Y"
gn\1	gene identifier	"Gene X is expressed"
pro\1	protein complex identifier	"Protein complex X is available (in every cellular compartment)"
enz\1	enzyme category identifier	"Enzyme category X is available"
rxn\1	reaction	"There is positive flux through reaction X"

Clauses in our model are one of seven types, each expressing relationships between entities in terms of the predicates given above. These types of clauses are listed below, and we provide a graphical overview and example statements in Fig. 2.

- **Reaction activation** clauses state that all substrate compounds for a specific reaction being present in the correct compartments, together with availability of a relevant enzyme, implies the reaction is active.
- **Reaction product** clauses state that a reaction being active implies the presence of a product compound in a given compartment.
- **Enzyme availability** clauses state that the availability of the constituent parts (proteins) of an enzyme imply the availability of the enzyme. Enzymes sometimes act in complexes made up of two or more proteins, and different enzymes that catalyse the same reaction are called isoenzymes.
- **Protein formation** clauses state that the presence in the genome of a gene that codes for a specific protein implies the availability of that protein.
- **Gene presence** clauses are statements expressing either the presence or absence of a particular gene in the genome.
- **Metabolite presence** clauses are statements expressing the presence of a particular compound in a specific compartment.
- **Goal** clauses represent a biological objective, usually the presence in the cytosol of a set of compounds deemed essential for growth, but could also be another pathway endpoint or intermediary compound.

2.2 Assessing Growth and Production of Compounds

Yeast growth is dependent on the production of essential chemical products—intermediary points or endpoints of biochemical pathways within the organism. The core of these biochemical pathways is the enzymatic reactions, and they are facilitated by diffusion of chemicals within cellular compartments, including the cytosol, and passive or active transport across compartment boundaries or the cell membrane. Certain products are deemed essential for growth, so if production of these compounds is inhibited then the organism is inviable.

Logical inference was performed using the automated theorem proving software iProver (v3.7) which was chosen due to its performance and scalability as well as completeness for first-order theorem finding. The general formulation of the problem provided to iProver is to identify whether a theory, T, "entails" a goal, G. In other words that the goal is a logical consequence of the theory ($T \vDash G$). Here T is a set of logical axioms that encode, using the formalism defined in Sect. 2.1: knowledge from the GEM; the medium in which the yeast is growing, represented by axioms in the theory for the presence of compounds in the extracellular space; the availability of ubiquitous compounds in each cellular compartment and the extracellular space; and the presence and expression of genes. Deduction can be used to analyse pathways and reachable metabolites. In the case of growth/no-growth simulations, G represents the availability of all the essential compounds in the cytoplasm. So if $T \vDash G$ we say that there is growth, otherwise not. Other goals used here are the availability of other endpoints of biochemical pathways. T and G are provided to iProver in plain text files and plaintext proofs are output. The logical proofs (that the goal is reachable) found by iProver correspond to detected biochemical pathways.

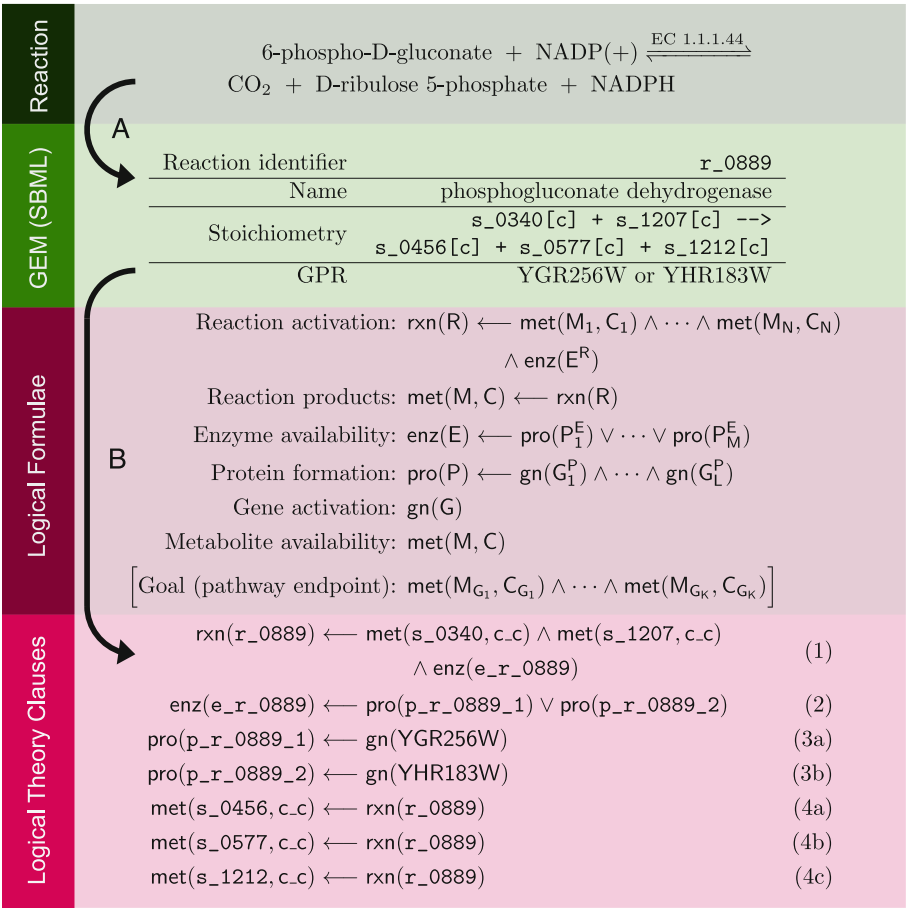

Fig. 2. Conversion of genome-scale metabolic model provided in SBML to logical theory. **(A)** A reaction is encoded in SBML using identifiers to represent the substrates and products, and a logical rule for enzyme availability (GPR = "gene-protein-reaction rule"). **(B)** The information contained on each reaction is encoded using logical formulae into a set of clauses; predicate definitions are provided in Table 1. Here equation (1) is the reaction activation clause. "$\wedge$" is a conjunction symbol ("AND"), meaning all of the literals in the expression must be true for the RHS of the clause to be true; "$\vee$" is a disjunction symbol ("OR"). So we can read (1) as: "reaction `r_0889` is active if all of the metabolites in the set $\{$`s_0340`, `s_1207`$\}$ are present in the cytoplasm and at least one of the isoenzymes is present". Similarly equation (2) describes the condition for a relevant enzyme to be present; equations (3a,b) describe the conditions for each of these isoenzymes to be formed; and equations (4a-c) are the reaction product clauses and state that "if reaction `r_0889` is active then each of its products is present".

Single-Gene Essentiality Prediction. Here we seek to predict genes without which *S. cerevisiae* cannot grow. We compare predictions against lists of viable and inviable strains from a genome-wide deletion mutant cultivation for

S. cerevisiae using several media [9]. In particular, we compare with cultivations on a minimal medium with the addition of uracil, histidine and leucine. The strain background used in this study was S288C, which has complete or partial deletions for HIS3, LEU2, LYS2, MET17 and URA3—for our experiments we remove these genes by default. Gene knockouts were performed by negating the gene presence axiom in the logical theory (i.e. $\mathsf{gn}(\mathsf{gene})$ becomes $\neg\mathsf{gn}(\mathsf{gene})$).

There are two basic error types with these predictions. We follow the naming convention as in [15], that we have: (1) *gNG inconsistency:* a prediction of growth when experimental data show no growth; and (2) *ngG inconsistency:* a prediction of no growth when experimental data show growth. Inconsistencies arise from three main sources: deficiencies in the prior knowledge; errors in the prediction process; or conflicting empirical evidence. However it is the deficiencies in the prior knowledge that are of most interest for scientific discovery, which we explore next.

2.3 Abduction of Hypotheses

Abduction is used to suggest hypotheses that resolve inconsistencies between our model and empirical data. As shown in Fig. 1(C) we select a reasonable set of candidate hypotheses through a two-stage process: firstly, we generate hypotheses; and secondly, we rank and filter these according to relevant scientific criteria. Generating hypotheses using an automated theorem prover is general purpose. Ranking and filtering heuristics will be domain-specific; here we describe the heuristics that we used, but others could well be applied. Pseudo-code for the abduction algorithm is provided in Algorithm 1.

Generating Candidate Hypotheses Using iProver. If the goal is not reachable (i.e. $T \nvDash G$) iProver abduces candidate hypotheses: sets H_i such that $\forall i\ (T \wedge H_i \vDash G)$. This is done by reverse consequence finding ($T \wedge \neg G \vDash \neg H_i$). For this project we extended iProver to include these features, which, not being specific to biochemical reaction networks, could be used for automated discovery in other scientific domains by constructing an appropriate FOL model. The form of the hypotheses, H_i, is a set of clauses expressed in terms of the predicates described above in Sect. 2.1. It is possible to restrict or guide the reverse consequence finding algorithm in iProver to seek certain types of hypotheses. For example a hypothesis could be: $\mathsf{met}(\mathsf{compound}, \mathsf{compartment})$, that $\mathsf{compound}$ is available in $\mathsf{compartment}$. Such hypotheses are challenging to discover because of the complexity of interaction in these networks.

None of the logical theories resultant from the conversion from Yeast8, iMM904 and iFF708 was viable given the minimal medium and ubiquitous compounds, even without any gene deletions, meaning one or more of the essential compounds was not produced. iProver abduced hypotheses consisting of combinations of compounds whose presence would enable viability of the base strain (deletions for HIS3, LEU2, LYS2, MET17 and URA3), as shown in Fig. 1(A). We chose the hypothesis with the fewest additional compounds.

For *ngG* inconsistencies there exists a set of essential metabolites not being produced that empirical data indicate will be produced given the specified genotype and conditions—in some sense the pathways in the model are incomplete. Hypotheses in this scenario are those that repair an incomplete pathway: additional reactions; annotation of an isoenzyme for knocked out genes; or removal of reaction annotations. For *gNG* inconsistencies there is a pathway in the model that empirical data suggest should be interrupted but is not. Thus hypotheses in this scenario will be those that interrupt a complete pathway: annotation of a pathway-critical reaction with a gene that is in the set of knocked out genes; removal of an isoenzyme annotation; or removal of reactions.

Heuristics for Ranking and Filtering Hypotheses. We filter hypotheses to only include either: (a) addition of one or more compounds (i.e. containing only atoms using the met predicate); or (b) the presence of one or more particular enzyme groups for a reaction (i.e. containing only atoms using the enz predicate). The motivation is that the subsequent model improvement step (to repair the pathway) for case (a) would be to add reactions to the model that produce the hypothesised metabolites, and for case (b) to either identify an isoenzyme for hypothesised groups or remove the annotation for the deleted gene for one of these reactions. We also remove hypotheses that introduced availability of one or more of the target compounds in the cytosol, as this would directly ensure the goal was reached but is of no scientific value.

We applied two criteria to assess the merit of each hypothesis. Firstly, by using our FBA constraint method, as shown in Fig. 1(D) and described in Sect. 2.4. Around half of the hypotheses resulted in infeasible solutions or very small growth—this means perhaps there might be something else that is missing from the model, and so we have not got a reasonable hypothesis. The second criteria was evaluating the impact each hypothesis had on the overall error in single-gene essentiality prediction, as shown in Fig. 1(E). If the total number of *ngG* errors fixed is greater than the number of *gNG* errors introduced then this is a good hypothesis. Another, more conservative, approach would be to only add hypotheses to the model that do not introduce any *gNG* errors.

A final heuristic was whether hypotheses contained compounds that were not produced by any reaction in the GEM, meaning adding a suitable reaction that produces this compound would repair the error. These hypotheses could be tested experimentally by constructing a deletion mutant, cultivating with minimal medium and after observing growth, using metabolomic analysis (e.g. with mass spectrometry) to identify if the hypothesised intermediary metabolite set is present. If there were a reaction already in the GEM that produced the compound there could be other deficiencies in the model that need addressing first, for example gene annotation for those reactions. In this case iProver abduces hypotheses of case (b) above. Currently LGEM$^+$ can hypothesise to remove gene annotation, but this could be extended to include a search for an isoenzyme based on similarity (e.g. sequence similarity) to the knocked out gene.

Algorithm 1. Abduction using LGEM$^+$

procedure AbductionSingleGene
  $\mathcal{H} \leftarrow \emptyset$
  for gene in all genes in theory **do**
    $\widetilde{T} \leftarrow T$ ▷ Make a copy of the base theory
    $\widetilde{T} \leftarrow \widetilde{T} \setminus \{\mathsf{gn}(\mathsf{gene})\} \cup \{\neg\mathsf{gn}(\mathsf{gene})\}$ ▷ Construct deletant
    Use iProver to deduce if goal is reachable by identifying if $\widetilde{T} \vDash G$
    if $\widetilde{T} \vDash G$ **then** ▷ Growth prediction
      continue
    else if $\widetilde{T} \nvDash G$ **then** ▷ Non-growth prediction
      if gene is essential **then** ▷ No growth observed; no error
        continue
      else if gene is not essential **then** ▷ Growth observed; ngG error
        Abduction of potential hypotheses set $\mathcal{H}_{\mathsf{gene}}$ using iProver
        $\mathcal{H} \leftarrow \mathcal{H} \cup \mathcal{H}_{\mathsf{gene}}$
      end if
    end if
  end for
  Filter and rank $\mathcal{H} = \bigcup_{\mathsf{gene} \in \mathrm{theory}} \mathcal{H}_{\mathsf{gene}}$, according to heuristics, e.g. Section 2.3
end procedure

2.4 Constraining Flux Balance Analysis Simulations Using Proofs

Flux balance analysis (FBA) finds a reaction flux distribution, $\boldsymbol{\nu}$, given stoichiometric constraints from the GEM and a biologically relevant optimisation objective, $f(\boldsymbol{\nu})$, for example maximisation of biomass production [8,18]. FBA assumes the metabolism is in steady state, resulting in the constraint $S\boldsymbol{\nu} = \mathbf{0}$, where S is the stoichiometric matrix for the metabolic network and $\boldsymbol{\nu}$ is the reaction flux vector ($S \in \mathbb{Z}^{m \times n}$, where m is the number of compounds and n is the number of reactions in the metabolic network).

$$\begin{aligned} \underset{\boldsymbol{\nu} \in \mathbb{R}^n}{\text{maximize}} \quad & f(\nu_1, \ldots, \nu_n) \\ \text{subject to} \quad & S\boldsymbol{\nu} = \mathbf{0} \\ & \nu_i^{\mathrm{LB}} \leq \nu_i \leq \nu_i^{\mathrm{UB}}, \quad i = 1, \ldots, n. \end{aligned}$$

Whilst the stoichiometric matrix is fixed, the upper and lower bounds for each reaction can be set to achieve relevant results. Existing methods to set these bounds include integrating experimental measurements of fluxes, or using enzyme turnover rates and availability [4]. We use FBA to assess the feasibility of proofs found using iProver by: setting reaction bounds based on pathways activated in the proof; and then solving the resultant optimisation problem. We are able to do this neatly as both use the same GEM as the knowledge source. The procedure is outlined in Algorithm 2.

Flux values are measured in mmol $\mathrm{g}_{\mathrm{DW}}^{-1}\mathrm{h}^{-1}$ and metabolite concentrations vary substantially between compounds, so finding a forcing threshold which is appropriate for all reactions is not straightforward. For our FBA simulations we

used the Python package `cobrapy` (version 0.26.3) [6]; in the absence of relevant documentation on a suitable threshold, we found in a discussion for a MATLAB implementation of COBRA that a suitable threshold should be set at 1×10^{-9} [2].

Algorithm 2. Constraining FBA solution given a logical theory T and a goal G

```
function FBAConstrain(GEM, T, G, ν_0)        ▷ ν_0 is minimum flux threshold for activation
    Use iProver to find proof of T ⊨ G                          ▷ The goal is reachable
    i ← 1
    while i ≤ N do                          ▷ N is the number of reactions in the GEM
        if r_i active in the proof in the forward direction then
            ν_i^LB ← ν_0                                ▷ Force reactions to have positive flux
        else if r_i active in the proof in the reverse direction then
            ν_i^UB ← −ν_0
        end if
        i ← i + 1
    end while
    Solve FBA problem (Sν = 0) with resultant flux bounds
    return (ν, growthValue, solutionStatus) ∈ ℝ^N × ℝ × {optimal, infeasible}
end function
```

2.5 Sources of Knowledge

The primary source of the knowledge about reactions and associated genes is the GEM Yeast8 (v8.46.4.46.2) [16]. This was chosen due to its broad coverage of the reactions and gene associations as well as its specificity to the organism *S. cerevisiae*. The other two GEMs used were: iMM904 [17] and iFF708 [7]. (We include iFF708 as a background knowledge source partly to enable comparison with previous logical modelling approach [25].) The models are stored using Systems Biology Markup Language (SBML). The software written to convert a GEM SBML file to a logical knowledge base is available in the supporting material, and follows the process described below and shown in Fig. 2.

We use three reference lists of compounds from [25]; these are shown in the first column of the files on the LGEM$^+$ GitHub repository[1] corresponding to: (1) all compounds deemed essential for growth in *S. cerevisiae*[2]; (2) compounds assumed ubiquitous during growth assumed to be present throughout the cell regardless of initial conditions, such as H_2O and O_2[3]; and (3) the growth media for the experiments, in this case yeast nitrogen base (YNB) with addition of ammonium, glucose and three amino acids (uracil, histidine and leucine)[4].

[1] https://github.com/AlecGower/LGEMPlus.
[2] `src/model-files/essential-compounds-{model}.tsv`.
[3] `src/model-files/ubiquitous-compounds-{model}.tsv`.
[4] `src/model-files/ynb-compounds-{model}.tsv`.

Each compound in these lists has an associated Kyoto Encyclopedia of Genes and Genomes (KEGG) [12] identifier. We matched compounds in the curated GEMs based firstly on KEGG ID, otherwise using the species name or synonyms. Some of the compounds we wish to include do not have corresponding entities in the GEMs used as background knowledge. Therefore there are discrepancies between the reference lists and the compiled lists.

3 Results

Automated Theorem Proving Software can be Used to Estimate Single-Gene Essentiality given a Prior Network Model. Using three GEMs—Yeast8, iMM904 and iFF708—as background knowledge sources we conducted single-gene deletant simulations to assess essentiality of each gene and compared against a genome-wide deletion mutant cultivation [9]. Detailed descriptions of these methods are provided in Sect. 2, and context in the overall method in Fig. 1(B). A summary of the single-gene essentiality prediction results is provided in Table 2.

When compared to previous qualitative methods our method showed state of the art results [25,26]. Yet quantitative prediction using FBA achieves a higher precision and recall. These error rates indicate how much is still to be learnt about yeast metabolism. We also found that gene essentiality predictions vary somewhat depending on the prior.

Simulation times for gene knockouts also appear to scale linearly with the size of the network. Comparing network size to average gene knockout simulation times for the three GEMs tested, we see that the mean (± 1 s.d.) times for one knockout simulation were: 0.52 s $\pm$ 0.09 s for iFF708 (1379 reactions); 0.67 s $\pm$ 0.12 s for iMM904 (1577 reactions); and 1.46 s $\pm$ 0.32 s for Yeast8 (4058 reactions).

Abductive Reasoning Allows for Identification of Possible Missing Reactions. We apply the LGEM$^+$ abduction procedure to model improvement, here demonstrated on the Yeast8 model. For each of the 41 *ngG* errors in the single-gene deletion task, we generated candidate hypotheses according to methods described in Sect. 2.3. In total we generated 2094 unique hypotheses; some hypotheses would result in an error correction for several genes. We ranked and filtered these hypotheses according to domain-specific heuristics, finding 681 of these were valid, i.e. only containing `met` (633) or `enz` (48) predicates. The FBA evaluation outlined in Sect. 2.4 indicated 534 hypotheses that could be balanced by the reactions forced in the model, 118 of which were valid. There were 14 hypotheses that were valid and also resulted in a net improvement on the single-gene prediction task.

Strict Essentiality Criteria and Incomplete Annotation may Explain *ngG* and *gNG* Inconsistencies. If just one essential compound is not produced we have no growth. One result of this setup is a relatively low precision in the single-gene essentiality prediction. Of the 72 deletions predicted inviable

Table 2. Comparative prediction results for single-gene essentiality using LGEM$^+$ across three background knowledge sources: Yeast8 (v8.46.4.46.2); iMM904; and iFF708, with comparison to: (a) an FBA-simulation with a viability threshold on growth rate set at $1\times10^{-6}\text{h}^{-1}$ (according to [16]); and (b) another qualitative prediction method, the "synthetic accessibility" approach taken by Wunderlich et. al. [26]. The empirical data used as truth data for these statistics were taken from a genome-wide screening study using a minimal medium [9]. The FOL model performance represents an improvement on previous qualitative method.

Base GEM	Yeast8	iMM904	iFF708	Yeast8 (FBA)	*Syn. Acc.* [26]
# predictions (#genes in GEM)	1056 (1150)	827 (905)	566 (619)	1068 (1150)	*682*
NG Recall (*ngNG/*NG*)	0.193 (31/161)	0.266 (33/124)	0.140 (14/100)	0.447 (72/161)	*0.119 (14/118)*
NG Precision (*ngNG/ng**)	0.431 (31/72)	0.478 (33/69)	0.778 (14/18)	0.459 (72/157)	*0.292 (14/48)*
gNG Rate (*gNG/*NG*)	0.807 (130/161)	0.734 (91/124)	0.860 (86/100)	0.553 (89/161)	*0.881 (104/118)*
ngG Rate (*ngG/*G*)	0.046 (41/895)	0.051 (36/703)	0.009 (4/466)	0.094 (85/907)	*0.060 (34/564)*
F1 score	0.266	0.342	0.237	0.453	*0.169*

Shorthand: **NG*-observed no growth; **G*-observed growth; *ng**-predicted no growth. (Note that the performance statistics for the synthetic accessibility method are taken directly from the authors' report so there may be a difference in truth data to those used to evaluate our model.)

by our model, 41 of these are shown to result in experimentally viable mutant strains (*ngG* errors).

For several genes in the L-arginine biosynthesis pathway the only essential metabolite not reachable in the model was L-arginine. These resulted in *ngG* errors despite the pathway structure and previous empirical evidence showing that null mutants for genes in this pathway (e.g. for *ARG1* [3]) are auxotrophic for L-arginine (i.e. L-arginine was not produced). These results demonstrate that the model can successfully identify behaviour of the metabolic network consistent with other experimental evidence and not the genome-wide screen results [9]. These cases are candidates for experimental testing, and highlight the potential of such models to inform laboratory experimental design and research direction.

In the Yeast8 model there are 4058 reactions, 1425 (35%) of which have no enzyme annotation and 540 (13%) are annotated with a set of isoenzymes that do not have a specific gene in common. Thus nearly half of all reactions will not be affected by single-gene deletions, which is likely to account for a portion of the 130 *gNG* inconsistencies in LGEM$^+$ single-gene essentiality predictions.

Pathways Output from LGEM$^+$ Overlap with FBA Simulations. In the case of predicting growth, LGEM$^+$ outputs reaction pathways. FBA simulations output a reaction flux distribution, and from this we can use a flux threshold for reaction activate to obtain reaction pathways. When comparing reaction pathways obtained from both methods, for each deletant simulation just over 50% of reactions in the LGEM$^+$ derived pathways are also active in the FBA pathways. However, only around 30% of reactions in FBA derived pathways are also active in the LGEM$^+$ derived pathways.

Using pathways derived from the FBA constraint method described in Sect. 2.4, we investigated the *gNG* errors. Of the 130 errors, 50 of them resulted in

pathways that the FBA method indicated were unfeasible (i.e., they resulted in low or zero growth). This would mean that by including this constraint method in the LGEM$^+$ framework we could eliminate these errors. However doing so would also falsely predict 56 viable deletant strains as inviable (new *ngG* errors).

4 Discussion and Conclusion

Scientific discovery in biology is difficult due to the complexity of the systems involved and the expense of obtaining high quality experimental data. Automated techniques that make good use of background knowledge, of which GEMs are prime examples, will have a strong starting point. LGEM$^+$ seeks to do just that by using FOL combined with a powerful theorem prover, iProver.

We efficiently predicted single-gene essentiality in *S. cerevisiae* using a first-order logic (FOL) model. Our method showed state of the art results compared to previous qualitative methods, yet quantitative prediction using FBA achieves a higher precision and recall.

We designed and implemented an algorithm for the abduction of hypotheses for improvement of a GEM. We found 633 hypotheses proposing availability of compounds in specific compartments, and therefore indicate possible missing reactions, 118 of which were validated through FBA constraint and 14 of which resulted in improvements in the single-gene essentiality prediction task. These heuristics help to select more promising hypotheses for experimentation; further selection will be informed by viability or cost of experiment design. We intend to test these hypotheses using the robot scientist Genesis, which is based around chemostat cultivation and high-throughput metabolomics. As we scale the system we can adjust parameters in the heuristics, or introduce new heuristics, to return only the most promising hypotheses.

Measuring performance statistics relative to the number of genes in a model, rather than the number of genes in the organism, presents some challenges when designing a learning process to improve this performance (e.g. GrowMatch [15]). This highlights the need for better model assessment criteria to drive abduction. We have attempted here to provide an example with the constraint of FBA solutions. Future work could certainly be directed to defining such criteria and integrating them into LGEM$^+$.

The logical theory developed here was focused on efficient inference on biochemical pathways. A challenge for future development is to extend the first-order vocabulary to improve the power and performance of LGEM$^+$. Extending the vocabulary could mean: including more predicates, increasing the arity (number of arguments) of predicates, and introducing other logical clause forms. All to better encode biological processes, for example more detail regarding enzyme availability, integration of gene regulation and signalling or introducing time-dependent processes. Aligning the logic more closely with existing ontologies, for example the Systems Biology Ontology (SBO), would ensure the theory remains useful and semantically precise as it is extended. This is a common challenge across the scientific discovery community as we move further toward joint teams

of human and robot scientists—ontologies provide a common language. Using FOL allows us to work toward connecting LGEM$^+$ with external knowledge bases.

The best way to test hypotheses is through *in vivo* experimentation. Integrating LGEM$^+$ into an automated experimental design process would enable the next generation of robot scientists.

Acknowledgements. The authors thank the King Group at Chalmers University of Technology for valuable discussion and feedback. This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Alice Wallenberg Foundation. Funding was also provided by the Chalmers AI Research Centre and the UK Engineering and Physical Sciences Research Council (EPSRC) grant nos: EP/R022925/2 and EP/W004801/1, as well as the Swedish Research Council Formas (2020-01690).

Code and Data Availability. Code and data used in this study, including the tables for essential compounds, ubiquitous compounds and minimal media, are available at https://github.com/AlecGower/LGEMPlus.

References

1. Chen, Y., Li, F., Nielsen, J.: Genome-scale modeling of yeast metabolism: Retrospectives and perspectives. FEMS Yeast Res. **22**(1), foac003 (2022). https://doi.org/10.1093/femsyr/foac003
2. Cobra-Toolbox: what is the minimum flux computed by flux balance analysis or the accuracy of FBA? https://groups.google.com/g/cobra-toolbox/c/9xmP1VcrWL0
3. Crabeel, M., Seneca, S., Devos, K., Glansdorff, N.: Arginine repression of the Saccharomyces cerevisiae ARG1 gene. Comparison of the ARG1 and ARG3 control regions. Curr. Genet. **13**(2), 113–124 (1988). https://doi.org/10.1007/BF00365645
4. Domenzain, I., Sánchez, B., Anton, M., et al.: Reconstruction of a catalogue of genome-scale metabolic models with enzymatic constraints using GECKO 2.0. Nat. Commun. **13**(1), 3766 (2022). https://doi.org/10.1038/s41467-022-31421-1
5. Dujon, B.: Yeast evolutionary genomics. Nat. Rev. Genet. **11**(7), 512–524 (2010). https://doi.org/10.1038/nrg2811
6. Ebrahim, A., Lerman, J.A., Palsson, B.O., Hyduke, D.R.: COBRApy: constraints-based reconstruction and analysis for python. BMC Syst. Biol. **7**(1), 74 (2013). https://doi.org/10.1186/1752-0509-7-74
7. Förster, J., Famili, I., Fu, P., Palsson, B.Ø., Nielsen, J.: Genome-scale reconstruction of the Saccharomyces cerevisiae metabolic network. Genome Res. **13**(2), 244–253 (2003). https://doi.org/10.1101/gr.234503
8. García Sánchez, C.E., Torres Sáez, R.G.: Comparison and analysis of objective functions in flux balance analysis. Biotechnol. Prog. **30**(5), 985–991 (2014). https://doi.org/10.1002/btpr.1949
9. Giaever, G., Chu, A.M., Ni, L., et al.: Functional profiling of the Saccharomyces cerevisiae genome. Nature **418**(6896), 387–391 (2002). https://doi.org/10.1038/nature00935
10. Goffeau, A., Barrell, B.G., Bussey, H., et al.: Life with 6000 genes. Science **274**(5287), 546–567 (1996). https://doi.org/10.1126/science.274.5287.546

11. Heavner, B.D., Price, N.D.: Comparative analysis of yeast metabolic network models highlights progress, opportunities for metabolic reconstruction. PLoS Comput. Biol. **11**(11), e1004530 (2015). https://doi.org/10.1371/journal.pcbi.1004530
12. Kanehisa, M.: KEGG: Kyoto encyclopedia of genes and genomes. Nucleic Acids Res. **28**(1), 27–30 (2000). https://doi.org/10.1093/nar/28.1.27
13. King, R.D., Whelan, K.E., Jones, F.M., et al.: Functional genomic hypothesis generation and experimentation by a robot scientist. Nature **427**(6971), 247–252 (2004). https://doi.org/10.1038/nature02236
14. Korovin, K.: iProver – an instantiation-based theorem prover for first-order logic (system description). In: Armando, A., Baumgartner, P., Dowek, G. (eds.) IJCAR 2008. LNCS (LNAI), vol. 5195, pp. 292–298. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-71070-7_24
15. Kumar, V.S., Maranas, C.D.: GrowMatch: an automated method for reconciling in silico/in vivo growth predictions. PLoS Comput. Biol. **5**(3), e1000308 (2009). https://doi.org/10.1371/journal.pcbi.1000308
16. Lu, H., Li, F., Sánchez, B.J., et al.: A consensus S. cerevisiae metabolic model Yeast8 and its ecosystem for comprehensively probing cellular metabolism. Nat. Commun. **10**(1), 3586 (2019). https://doi.org/10.1038/s41467-019-11581-3
17. Mo, M.L., Palsson, B., Herrgård, M.J.: Connecting extracellular metabolomic measurements to intracellular flux states in yeast. BMC Syst. Biol. **3**, 1–17 (2009). https://doi.org/10.1186/1752-0509-3-37
18. Orth, J.D., Thiele, I., Palsson, B.Ø.: What is flux balance analysis? Nat. Biotechnol. **28**(3), 245–248 (2010). https://doi.org/10.1038/nbt.1614
19. Prigent, S., Frioux, C., Dittami, S.M., et al.: Meneco, a topology-based gap-filling tool applicable to degraded genome-wide metabolic networks. PLoS Comput. Biol. **13**(1), e1005276 (2017). https://doi.org/10.1371/journal.pcbi.1005276
20. Reiser, P.G.K., King, R.D., Muggleton, S.H.: Developing a logical model of yeast metabolism. Electron. Trans. Artif. Intell. **5**(B), 223–244 (2001)
21. Rozanski, R., Bragaglia, S., Ray, O., King, R.: Automating the development of metabolic network models. In: Roux, O., Bourdon, J. (eds.) CMSB 2015. LNCS, vol. 9308, pp. 145–156. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23401-4_13
22. Simon, L., del Val, A.: Efficient consequence finding. In: Nebel, B. (ed.) Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence, IJCAI 2001, Seattle, Washington, USA, 4–10 August 2001, pp. 359–370. Morgan Kaufmann (2001)
23. Tamaddoni-Nezhad, A., Chaleil, R., Kakas, A., Muggleton, S.: Abduction and induction for learning models of inhibition in metabolic networks. In: Fourth International Conference on Machine Learning and Applications (ICMLA 2005), p. 6 (2005). https://doi.org/10.1109/ICMLA.2005.6
24. Thiele, I., Palsson, B.Ø.: A protocol for generating a high-quality genome-scale metabolic reconstruction. Nat. Protoc. **5**(1), 93–121 (2010). https://doi.org/10.1038/nprot.2009.203
25. Whelan, K.E., King, R.D.: Using a logical model to predict the growth of yeast. BMC Bioinf **9**, 97 (2008). https://doi.org/10.1186/1471-2105-9-97
26. Wunderlich, Z., Mirny, L.A.: Using the topology of metabolic networks to predict viability of mutant strains. Biophys. J . **91**(6), 2304–2311 (2006). https://doi.org/10.1529/biophysj.105.080572