

ARTICLE

<https://doi.org/10.1038/s41467-019-13818-7>

OPEN

The ETFL formulation allows multi-omics integration in thermodynamics-compliant metabolism and expression models

Pierre Salvy ¹ & Vassily Hatzimanikatis ^{1*}

Systems biology has long been interested in models capturing both metabolism and expression in a cell. We propose here an implementation of the metabolism and expression model formalism (ME-models), which we call ETFL, for Expression and Thermodynamics Flux models. ETFL is a hierarchical model formulation, from metabolism to RNA synthesis, that allows simulating thermodynamics-compliant intracellular fluxes as well as enzyme and mRNA concentration levels. ETFL formulates a mixed-integer linear problem (MILP) that enables both relative and absolute metabolite, protein, and mRNA concentration integration. ETFL is compatible with standard MILP solvers and does not require a non-linear solver, unlike the previous state of the art. It also accounts for growth-dependent parameters, such as relative protein or mRNA content. We present ETFL along with its validation using results obtained from a well-characterized *E. coli* model. We show that ETFL is able to reproduce proteome-limited growth. We also subject it to several analyses, including the prediction of feasible mRNA and enzyme concentrations and gene essentiality.

¹Laboratory of Computational Systems Biotechnology, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland. *email: vassily.hatzimanikatis@epfl.ch

Metabolic modeling, which helps making sense of the metabolism in a biological network, is an important tool for engineering biocatalysts, with applications in biofuels, drug design, microbial community analysis, and personalized medicine. Model accuracy is instrumental to the success of these applications through an efficient engineering of the host organisms. However, incorporating expression information into metabolic networks poses a significant challenge, and most current models do not even attempt it—effectively excluding an important network in biological systems that can drastically affect results. In metabolic engineering, strains are modified and controlled at the genome level through the transcriptome, and the effects are observed at the fluxome level, which accounts for the range of metabolic reactions in an organism. In between these two levels is the proteome that performs the biochemical transformations according to the genetic template, though it is this middle step in the process that cannot yet be robustly and efficiently incorporated into models of metabolic systems. Because of the complex interplay between these different layers of control, understanding expression and incorporating this into future models is key for improving metabolic engineering.

Classically, model-based strain design has relied on tools that use the DNA sequence of an organism and homology with well-studied organisms to infer a network of metabolic reactions that happen inside a cell of that organism, which is called a genome-scale model (GEM). With current technologies and tools like metagenome sequencing¹, it is possible to generate GEMs for hundreds of different species at a time. GEMs are particularly amenable to flux balance analysis (FBA), which models metabolism at the fluxome level using linear optimization techniques. However, plain FBA has been known to predict biochemically unrealistic solutions like free high-flux cycles or thermodynamically infeasible pathways. It also scales growth linearly with carbon uptake, which is not observed at high-uptake fluxes. FBA also fails to capture growth-dependent and protein-level effects, such as enzyme saturation or proteome-related limitations. Hence, several efforts have been made to supplement FBA with additional constraints to improve its predictive power. For example, thermodynamics-based flux analysis (TFA)^{2,3} uses thermodynamic constraints to enforce thermodynamically consistent reaction directionalities and to allow the integration of metabolomics. Resource balance models add a total proteome capacity constraint, as formulated in Beg et al.⁴, to model the proteome-related limitations of the cell, as enzymes have to compete for the constrained total amount of cellular proteins. Frameworks like GECKO⁵ further build on this resource balance idea and include flux constraints based on proteomics, such as $v \leq V_{\max} = k_{\text{cat}}[E]$ as well as a constraint on the total proteome mass. Finally, metabolomics and expression models (ME-models)^{6,7} were the first to integrate the entirety of the expression mechanisms of the cell from the bottom-up, including mRNA and protein synthesis.

However, simultaneously accounting for all of these constraints is challenging because of the formulation of each method, as TFA models involve integer variables that yield a mixed-integer linear program (MILP), whereas ME-models involve bilinear constraints that require special optimization procedures and a high-precision (quad-precision) solver^{8–10}. Mixing these methods would require the inclusion of integers in ME-models, which is not straightforward and would lead to more complex mixed-integer nonlinear programs (MINLP) that are computationally intensive to solve. In addition, the amount of RNA and protein, the RNA and protein expression rates, and their stabilities are all growth dependent¹¹, and including accurate representations of these variables leads to even more complex, nonlinear models. Meanwhile, although resource balance models such as GECKO

could theoretically be integrated into TFA or ME-models in the current formulations, to the best of our knowledge, no link with TFA or ME-models has been proposed. Therefore, the metabolic engineering community needs a common formulation for these methodologies to build the most accurate models.

We investigated the development of such a framework and propose herein a unified formulation for Expression and Thermodynamics-enabled FLux models (ETFL) that can account for the above integration issues. To our knowledge, ETFL is the first formulation that can account at the same time for expression, thermodynamics, and growth-dependant variables. It is also the first to do so using common double-precision MILP solvers. In ETFL, we address the compatibility of the formulations by expressing the growth rate variable in bilinear products as a piecewise constant function. We also address the issue of solver precision by performing a scaling that reduces the range of orders of magnitude of the variables. This reformulation allows us to transform the problem into a MILP, which we can solve efficiently using common open source or commercial solvers. The resulting model is then effectively able to directly integrate thermodynamic constraints as well as expression constraints and growth-dependent parameters. In this model, metabolite, enzyme, and mRNA concentration levels are explicitly defined to enable fast and easy omics integration: metabolites through their log-concentration variables in thermodynamics constraints, and enzymes and mRNA through their total concentration variables in the expression constraint. Finally, we show an application of this framework to a well-characterized *E. coli* model, iJO1366¹².

Important assumptions are made to derive this formulation. The two most notable ones are (i) we can neglect the dilution rate of metabolites, and (ii) the steady-state approximation holds. While these assumptions are commonly made in FBA, we discuss them in details in the Supplementary Note 3, where we also assess their validity in a context where macromolecules are taken into account. Briefly, these assumptions hold because (i) the dilution rate of the metabolites is negligible in front of their synthesis and consumption rates, and (ii) the dynamics of metabolism (including expression) are faster than that of the environment of the cell.

Results

Formulation of the expression problem. ETFL is an ME-model implementation because it proposes a formulation that both accounts for metabolism and expression constraints. ME-models do not aim to replace kinetic models, but to account for the expression cost of making the enzymes that are necessary to carry a biochemical flux. In ETFL, this includes the cost of peptide and mRNA synthesis, as well as the competition for ribosomes and RNA polymerase in a limited proteome.

To transparently account for expression mechanisms and increase the predictive power of our models, we needed to derive the equations that could bridge the biochemistry with the optimization problem that is ETFL. Here, we present a summary of these equations, and detail their derivation in the Methods section. We derived these equations using assumptions similar to those used in the formulation of the GECKO⁵ and ME-model^{6–8}.

This formulation relies on derivations rooted in the biological mechanism of expression and depends on a number of biochemical parameters related to the cell. In particular, the mass balances of the macromolecules are expressed using concentration variables. Each mass balance will yield an equation where the concentrations of the macromolecules will be variables, thus effectively formulating a new constraint of the model and allowing us to calculate concentration values by solving the model.

We can write the quasi-steady-state mass balance for macromolecules as follows:

$$v_{\star}^{\text{syn}} - v_{\star}^{\text{deg}} - \mu * G_{\star} = 0, \quad (1)$$

where \star represents the indexing of the macromolecule, v_{\star}^{syn} is the synthesis term, v_{\star}^{deg} is the degradation term, and $\mu * G_{\star}$ is the dilution term. The asterisk “*” signifies the product of two variables. The detail of the derivation is available in the Methods section.

Using this formalism, for each macromolecule we can define and link together a synthesis flux, a degradation flux, and the macromolecule’s concentration. Knowing enzyme concentrations allows us to bound the variables representing metabolic reaction fluxes with their maximum catalytic rate according to the classical equation:

$$v \leq k_{\text{cat}} \cdot E, \quad (2)$$

where k_{cat} is the catalytic rate constant of the enzyme E with respect to flux v . The dot product “.” signifies here a product between a parameter value and a variable. In this same fashion, we can also constrain the synthesis flux for the peptides, which are then assembled into enzymes. Peptide synthesis is simply a metabolic reaction that consumes energy (under the form of GTP) and charged tRNAs and produces a peptide and uncharged tRNAs. The catalytic rate of the reaction is proportional to the maximum ribosomal catalytic rate divided by the length of the peptide to be synthesized. The same can be said about mRNA synthesis, which uses nucleoside triphosphates and is catalyzed by the RNA polymerase. The constraints are explained in the Methods section, in which we detail a de novo derivation of the constraint set that describe the expression problem.

The part of the matrix that has been added to the FBA problem to account for expression has been termed the expression problem (EP). Although this initial formulation is bilinear, we detail in the Methods section how we cast it to a MILP.

Biomass reaction synthesis and mass balance. In FBA, the biomass reaction is an artificial, lumped reaction that represents the consumption of metabolites in proportion to the cell growth rate. This consumption reflects nucleoside triphosphate (NTP) requirements for mRNAs, amino acid requirements for proteins, lipid requirements for the cell wall, or metal ion needs. Biomass reaction inclusiveness depends on the modeling assumptions made during the model curation process and can vary significantly among models of the same species. The consumed amount of each metabolite is usually estimated experimentally by measuring the amounts of these metabolites in dried cell mass. Because the stoichiometric ratios of metabolites in the biomass reaction are fixed, the abundance of metabolites is the same for all growth rates. This simplifying assumption, necessary in FBA, goes against experimental evidence. Neidhardt and Curtis¹¹ report for instance that mRNA and protein mass ratios in the cell change with growth rate.

Because ETFL has explicit expression requirements through transcription, translation, and tRNA-charging reactions, it is possible to account for varying ratios of NTPs and amino acids as the growth rate changes, an effect that is captured in experiments¹¹. In this context, the approximation made in FBA can be written using ETFL terms:

$$\forall \text{aa}_i, \quad \eta_{\text{aa}_i}^{\text{biomass}} \cdot \mu \approx v_{\text{aa}_i}^{\text{charging}}, \quad (3)$$

$$\forall \text{NTP}_i, \quad \eta_{\text{NTP}_i}^{\text{biomass}} \cdot \mu \approx \sum_{j \in \mathcal{J}} v_{\text{NTP}_i}^{\text{tcr}_j}, \quad (4)$$

where v^{biomass} represents the flux through the biomass equation, and $\eta_{m_i}^{\text{biomass}}$ is the stoichiometric coefficient of metabolite m_i in the biomass reaction. For each metabolite participating in the

biomass reaction, the expressions above are obtained by equating the corresponding mass balance constraints in ETFL and in FBA. Hence, to avoid accounting for the expression requirements twice (once through the biomass equation, once through the EP), it is necessary to remove the participation of these metabolites linked to expression from the biomass reaction.

Summary of the formulation. Here we show the formulation of the constraints of ETFL. For clarity, we use different indexing sets, each referring to a specific object in the model. The definition of these, as well as that of the variables and the parameters, are detailed in Table 1. The formulation of the following equations and an explanation of the specific cases for RNA polymerase and ribosomes are discussed in details in the Methods section.

Metabolite mass balance

$$S \cdot v = 0 \quad (\text{FBA})$$

Catalytic constraints

$$v_j^+ - k_{\text{cat}}^{j,+} E_j \leq 0 \quad (\text{FC}_j)$$

$$v_j^- - k_{\text{cat}}^{j,-} E_j \leq 0 \quad (\text{BC}_j)$$

Expression mass balance

$$v_l^{\text{tsl}} - \sum_{j \in \mathcal{J}} \eta_l^j \cdot v_j^{\text{asm}} = 0 \quad (\text{PB}_l)$$

$$v_{\text{rRNA}_i}^{\text{tcr}} - v_{\text{rib}}^{\text{asm}} = 0 \quad (\text{RB}_{\text{rRNA}_i})$$

$$v_j^{\text{asm}} - v_j^{\text{deg}} - \mu * E_j = 0 \quad (\text{EB}_j)$$

$$v_l^{\text{tcr}} - v_l^{\text{deg}} - \mu * F_l = 0 \quad (\text{MB}_l)$$

$$-v_{\text{aa}_i}^{\text{charging}} + \sum_{l \in \mathcal{L}} \eta_{\text{aa}_i}^l \cdot v_l^{\text{tsl}} - \mu * T_{\text{aa}_i}^u = 0 \quad (\text{TB}_{\text{aa}_i}^u)$$

$$v_{\text{aa}_i}^{\text{charging}} - \sum_{l \in \mathcal{L}} \eta_{\text{aa}_i}^l \cdot v_l^{\text{tsl}} - \mu * T_{\text{aa}_i}^c = 0 \quad (\text{TB}_{\text{aa}_i}^c)$$

Degradation fluxes

$$v_j^{\text{deg}} - k_{\text{deg}}^j \cdot E_j = 0 \quad (\text{ED}_j)$$

$$v_l^{\text{deg}} - k_{\text{deg}}^l \cdot F_l = 0 \quad (\text{MD}_l)$$

Expression constraints

$$v_l^{\text{tcr}} - \frac{k_{\text{cat}}^{\text{RNAP}}}{L_l^{\text{nt}}} P_l \leq 0 \quad (\text{TR}_1)_l$$

$$v_l^{\text{tsl}} - \frac{k_{\text{cat}}^{\text{rib}}}{L_l^{\text{aa}}} R_l \leq 0 \quad (\text{TR}_2)_l$$

$$R_l - \frac{L_l^{\text{nt}}}{L_l^{\text{rib}}} F_l \leq 0 \quad (\text{EX}_l)$$

Total capacity

$$\sum_{l \in \mathcal{L}} R_l + R_F - E_{\text{rib}} = 0 \quad (\text{TC}_2)$$

$$\sum_{l \in \mathcal{L}} P_l + P_F - E_{\text{RNAP}} = 0 \quad (\text{TC}_1)$$

$$R_F - (1 - \rho) E_{\text{rib}} = 0 \quad (\text{RR})$$

$$P_F - (1 - \pi) E_{\text{RNAP}} = 0 \quad (\text{PR})$$

Table 1 Indices, variables, and parameters used in the formulation.

Index letter	Type	Refers to	Set or unit
i	Index	Metabolite	\mathcal{I}
aa_i	Index	Amino acid	\mathcal{A}
j	Index	Reaction/flux/enzyme	\mathcal{J}
l	Index	Gene/peptide/mRNA	\mathcal{L}
s	Index	Binary coefficient for growth discretization	$S = \{0..[\log_2 N]\}$
u	Index	Binary coefficient for interpolation discretization	$U = \{0..N\}$
μ	Variable	Growth rate	h^{-1}
v_j^\pm	Variable	j^{th} net positive/negative biochemical flux	$\text{mmol.gDW}^{-1}.h^{-1}$
E_j	Variable	Concentration of the j^{th} enzyme	mmol.gDW^{-1}
F_l	Variable	Concentration of the l^{th} mRNA	mmol.gDW^{-1}
P_l	Variable	Concentration of the RNA polymerase assigned to the l^{th} mRNA	mmol.gDW^{-1}
R_l	Variable	Concentration of the ribosome assigned to the l^{th} peptide	mmol.gDW^{-1}
$T_{aa_i}^u$	Variable	Concentration of the i^{th} uncharged tRNA	mmol.gDW^{-1}
$T_{aa_i}^c$	Variable	Concentration of the i^{th} charged tRNA	mmol.gDW^{-1}
v_l^{trsl}	Variable	Translation rate of the l^{th} gene	$\text{mmol.gDW}^{-1}.h^{-1}$
v_l^{cr}	Variable	Transcription rate of the l^{th} gene	$\text{mmol.gDW}^{-1}.h^{-1}$
v_j^{asm}	Variable	Assembly rate of the j^{th} enzyme	$\text{mmol.gDW}^{-1}.h^{-1}$
v_j^{deg}	Variable	Degradation rate of the j^{th} enzyme	$\text{mmol.gDW}^{-1}.h^{-1}$
v_l^{deg}	Variable	Degradation rate of the l^{th} mRNA	$\text{mmol.gDW}^{-1}.h^{-1}$
$v_{aa_i}^{\text{charging}}$	Variable	Charging rate of the tRNA associated to amino acid aa_i	$\text{mmol.gDW}^{-1}.h^{-1}$
k_{cat}^\pm	Parameter	Forward/backward catalytic rate constant of the j^{th} net biochemical flux	h^{-1}
k_j^{deg}	Parameter	Degradation rate constant of the j^{th} enzyme	h^{-1}
k_l^{deg}	Parameter	Degradation rate constant of the l^{th} mRNA	h^{-1}
η_j	Parameter	Stoichiometry of the l^{th} peptide in the j^{th} enzyme	$[\emptyset]$
$\eta_{aa_i}^j$	Parameter	Stoichiometry of the amino acid aa_i in the l^{th} peptide	$[\emptyset]$
l_j^{aa}	Parameter	Length in amino acids (aa) of the l^{th} peptide	aa
l_l^{nt}	Parameter	Length in nucleotides (nt) of the l^{th} mRNA	b
l_{rib}^{nt}	Parameter	Ribosome footprint size on mRNA, in nucleotides	b
ρ	Parameter	Ribosome occupancy	$[\emptyset]$
π	Parameter	RNA polymerase occupancy	$[\emptyset]$

Recovering the FBA problem. In the ETFL formulation, enzyme synthesis is driven by the coupling between FBA and EP through the catalytic constraints. To carry flux, the cell needs to produce enzymes which will also use the metabolic resources of the cell. If allocation constraints are enforced, the amount of protein and mRNA synthesized must meet predefined mass ratios for the problem to be feasible. Hence, the metabolic requirement terms for the expression machinery (amino acids and NTP) have been removed from the biomass reaction and are accounted for in the tRNA charging and transcription reactions. Thus, the FBA solutions can be recovered from the ETFL formulation by the following routine:

Setting $\forall j, k_{cat}^{j,\pm} = +\infty,$

Constraining $\forall aa_i, v_{aa_i}^{\text{charging}} = \eta_{aa_i}^{v_{biomass}} \cdot \mu,$

Constraining $\forall NTP_i, \sum_{l \in \mathcal{L}} v_{NTP_i}^{\text{trsl}} = \eta_{NTP_i}^{v_{biomass}} \cdot \mu,$

If applicable, relaxing the allocation constraints,

If applicable, relaxing the thermodynamic coupling constraints.

Application: *E. coli* genome-scale model iJO1366. iJO1366¹² is a well-curated and well-studied GEM of *E. coli* that is closely related to the GEM used in developing both ME-models iOL1650-ME⁷ and iJL1678b-ME⁸. In addition, this model has been extensively applied in the literature, and is aligned with a variety of data sets that can be used for data integration. We wanted to subject the model to classical studies that would highlight the power of ETFL, particularly as pertains to proteome-limited growth, macromolecule concentration variability analysis, and gene knockout studies. We also wanted to assess the sensitivity of the model with respect to the presence of

Table 2 Nomenclature of the models used in the study of *E. coli* iJO1366.

	Growth-independent parameters	Growth-dependent parameters
(−) thermodynamics	EFL	vEFL
(+) thermodynamics	ETFL	vETFL

EFL stands for Expression and FLuxes, ETFL for Expression, Thermodynamics, and FLuxes, and the v- prefix indicates the inclusion of growth-dependent parameters (see the section Discretization of mRNA and enzyme content in Methods.)

thermodynamic constraints, as well as growth-dependent parameters.

Thus, we first experimented with four different models using ETFL with or without thermodynamic constraints and growth-dependent protein/RNA/DNA allocation following Table 2 as reported by Neidhardt et al.¹¹. The following Table 2 details the nomenclature used to refer to these different models. The features of the most constrained model containing both thermodynamic and growth-dependent parameters, vETFL, are detailed in Table 3. These four models were optimized for maximal growth at increasing glucose uptake rates to assess their behavior with respect to excess substrate, which will show the non-linearity of the relationship between growth and glucose uptake at high-uptake rates. A plateau in the growth rate was expected, which indicates a proteome-limited phenotype that cannot be observed with FBA. We also subsequently subject vETFL to a variability analysis and gene essentiality analysis, which will, respectively, show us the flexibility of the model and its accuracy in predicting gene knockout behavior.

Growth rate prediction. To study the behavior of the model at different carbon uptake rates, we simulated growth on a minimal medium with only glucose as a carbon source, unlimited oxygen, and some essential inorganic compounds. This would allow us to show that at a higher carbon uptake, the model would predict a limited growth—unlike FBA that would predict an unlimited linear increase.

Figure 1 shows the predicted growth rate of the different (v)E(T)FL models described in Table 2 with respect to the glucose uptake of the cell. As expected and in contrast to current FBA models, all four models plateau after a certain uptake rate, which indicates a proteome-limited phenotype due to the limited capacity of the cells to make more enzymes to metabolize the

glucose. As discussed for the ME-models⁷ and GECKO⁵ formulations, within the context of models accounting for protein usage, this is caused by (i) the protein burden necessary to metabolize higher fluxes; (ii) the increased demand in protein synthesis at higher growth rates; and (iii) for the models with allocation constraints, the allowed protein and RNA mass ratio. We can see that models featuring protein, RNA, and DNA allocation constraints (vE[T]FL) consistently predict a lower growth rate than models without allocation constraints. This is expected, as the data we input requires additional proteins and mRNA to account for non-metabolism-related macromolecules. Models featuring thermodynamic constraints ([v]ETFL) also predict a lower growth rate, consistent with the fact that thermodynamics constrain the model to valid solutions whose flux is in the subspace of the FBA feasible space. The most constrained model (vETFL) consequently has the lowest growth rate at any glucose uptake. This is in accordance with published TFA results that eliminated biologically infeasible flux profiles yielding non-realistic higher growth rates².

We summarize the constraint matrix of the EP of vETFL in Supplementary Table 1, where each line represents a type of constraint and each column represents a type of variable. The blocks of the matrix that are nonzero are colored, and these blocks directly reflect the involvement of the constrained variables.

Modeling missing enzymes. Although we initially focused on including only enzymes for which we had all the necessary information (catalytic rate and peptide constitution), we wanted to assess the robustness of our model when the missing enzymes were modeled as well as check our model's sensitivity to changes in the catalytic rate constants. Thus, we additionally built three more models, based on vETFL, with the following properties: (i) all the missing enzymes were estimated by averaging the properties of the known enzymes based on the curation for the vETFL iJO1366 (333 amino acids long, average $k_{cat}^{\pm} = 172 \text{ s}^{-1}$); (ii) all the enzymes (including the missing enzymes) but the ribosome, RNA polymerase, and ATP synthase were assumed to have an average catalytic rate constant $k_{cat}^{\pm} = 172 \text{ s}^{-1}$; and, for

Table 3 Properties of the vETFL model generated from iJO1366.

Growth upper bound $\bar{\mu}$	3.5 h ⁻¹
Number of bins, N	128
Resolution, $\frac{\bar{\mu}}{N}$	0.0273 h ⁻¹
Number of constraints	68,304
Number of variables	49,207
Number of species	3240
- Metabolites	1806
- Peptides	1431
- rRNA	3
Number of enzymes	562
Number of reactions	8023
- Metabolic	1543
- Transport	733
- Exchange flux	330
- Transcription	1431
- Translation	1431
- Complexation	562
- Degradation	1993
Number of metabolites, $\Delta_r G^{\circ}$	1737
Number of reactions, $\Delta_r G^{\circ}$	1787
Percent of metabolites, $\Delta_r G^{\circ}$	93.9%
Percent of reactions, $\Delta_r G^{\circ}$	68.6%

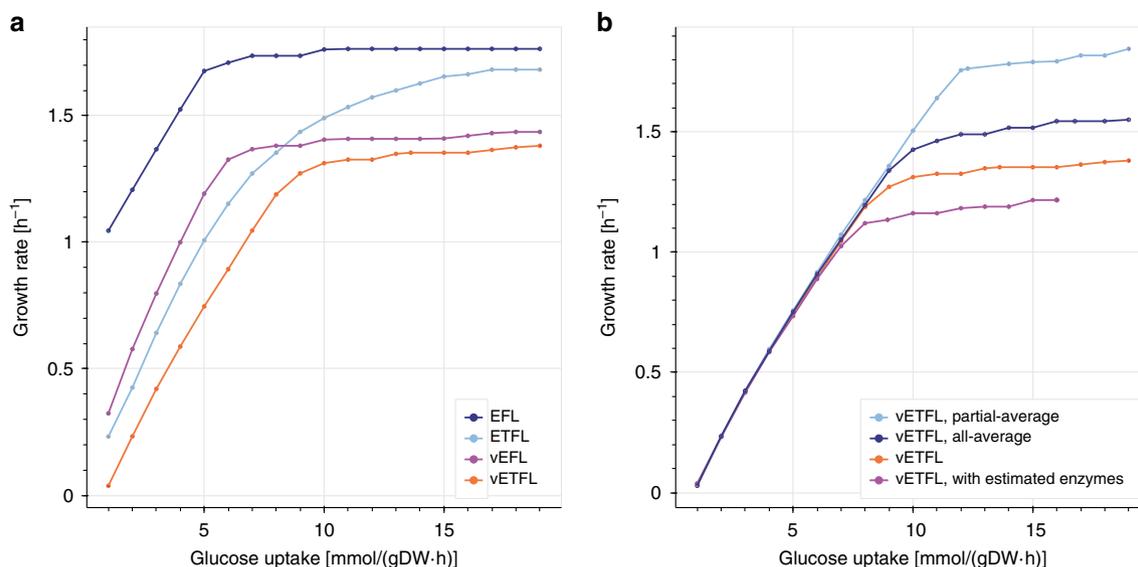


Fig. 1 Growth rate with respect to glucose uptake for differently constrained models in the ETFL framework. Legend in the same order as the height of the right-most point of each curve in each figure. **a** Growth rate predictions using the EFL, ETFL, vEFL, vETFL models (dark blue, light blue, purple, orange); **b** growth rate predictions accounting for missing enzymes using vETFL (orange) and models (i)–(iii) (purple, dark blue, light blue) representing different initial enzyme assumptions, with k_{cat} values obtained from vETFL or $k_{cat} = 172 \text{ h}^{-1}$, and with/without inferred enzymes.

comparison purposes, (iii) all the known enzymes of vETFL except for the ribosome, RNA polymerase, and ATP synthase were assumed to have an average catalytic rate constant of $k_{\text{cat}}^{\pm} = 172 \text{ s}^{-1}$. For clarity, we will refer to these models as (i) the model with estimated enzymes; (ii) the all-average model; and (iii) the partial-average model. The ribosome, RNA polymerase, and ATP synthase were not modified, as their catalytic rates directly and strongly affect the growth of the organism. Any drastic change in these would make changes related to other enzymes negligible in comparison.

Figure 1b shows a comparison of the growth prediction for the model with estimated enzymes (purple), all-average model (dark blue), and partial-average (light blue) models designed to account for the missing enzymes. For a better comparison, we also reproduce the vETFL results in orange on the same graph. The partial-average model (light blue) shows a higher predicted growth than the original vETFL model (orange). This implies that limiting enzymes in the original vETFL model have a k_{cat} parameter lower than the average value of $k_{\text{cat}}^{\pm} = 172 \text{ s}^{-1}$. Both models featuring inferred enzymes, the all-average model (dark blue) and the model with estimated enzymes (purple) show, at a given uptake, a lower growth rate than their counterpart, respectively, the partial-average model (light blue) and the original vETFL (orange). This is expected as fluxes which previously had no enzymes assigned in vETFL are now subject to catalytic constraints, and thus the models are more constrained. In addition, we observe that the model with estimated enzymes (purple) is also below the all-average model (dark blue). Similarly to vETFL and the partial-average model, this shows that the limiting enzymes in the model with estimated enzymes have a k_{cat} parameter lower than the average value. Finally, we observe that the differences between these four models only appear at glucose uptake rates higher than $\approx 6 \text{ mmol}_{\text{glc}} \cdot \text{DW}^{-1} \cdot \text{h}^{-1}$, when the problem switches from being stoichiometry-limited to proteome-limited. Thus, this experiment illustrates the robustness of the formulation in predicting growth-limited phenotypes, but also the importance of well-curated catalytic rate constants for modeling organisms grown in proteome-limited regimens.

These results demonstrate the capability of ETFL to predict different phenotypes depending on growth rate. ETFL is also amenable to hypothesis testing, as evidenced using the models that estimate the missing enzymes. In particular, we showed with ETFL that an uptake increase does not yield a proportional growth rate increase as with FBA and that ETFL provides a maximal uptake rate that is unmodeled in FBA, thus more effectively modeling growth-dependent biomass yield in *E. coli*. This allows for more realistic predictions for phenotypes that are limited by the expression capabilities of the cell as well as captures the variability of the biomass composition in different growth regimens.

Variability analysis. It is also possible to subject the model to a range of variability analyses. These are routinely used in FBA to assess the flexibility of the system and in TFA to find the ranges of allowed metabolite concentrations. In particular, we studied the number of bidirectional reactions in the system. Bidirectional reactions are reactions whose net flux can be either positive or negative. They are an indicator of the flexibility of the system. One of the main results of TFA was to replace ad hoc assumptions on the directionality of the reactions by thermodynamically-based directionality. We show that adding enzymatic constraints with ETFL also reduces the number of bidirectional reactions. The initial iJO1366 formulation with ad hoc directionality assumptions shows 112 bidirectional reactions in FBA, under the constraint of a specific growth rate of 0.79 h^{-1} (TFA prediction). Once TVA is performed on the thermodynamics-enabled model of iJO1366, the number of bidirectional reactions drops to 88. Finally, after the addition of catalytic constraints, this number is reduced to 49 in the vETFL model.

We can extend the use of variability analyses in ETFL to explore the allowed proteome and transcriptome. For example, we measured the admissible extreme concentrations of each peptide in aerobic growth conditions as described by McCloskey et al.¹³ by performing a variability analysis on the enzyme concentration variables. Figure 2 depicts the admissible peptide concentration upper and lower bounds, sorted by average, for vETFL with a glucose uptake set to $12.5 \text{ mmol}_{\text{glc}} \cdot \text{DW}^{-1} \cdot \text{h}^{-1}$,

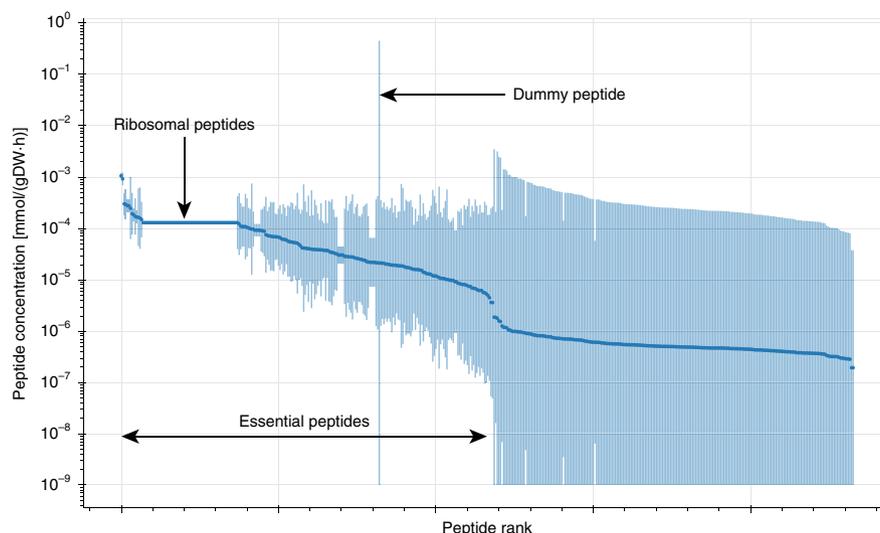


Fig. 2 Concentration variability of peptide species, sorted by average peptide concentration (darker disc). Lower bounds that were 0 were cut off at concentrations of $10^{-9} \text{ mmol}_{\text{glc}} \cdot \text{DW}^{-1}$. The horizontal line on the left side of the figure represents ribosomal peptides, which is narrow due to their instrumental role in making the tightly constrained amount of protein in the cell at a given growth rate. The vertical line in the middle represents the dummy peptide, which accounts for unmodeled peptides (non-metabolic proteins and enzymes with missing information) and therefore is used by the solver as a slack.

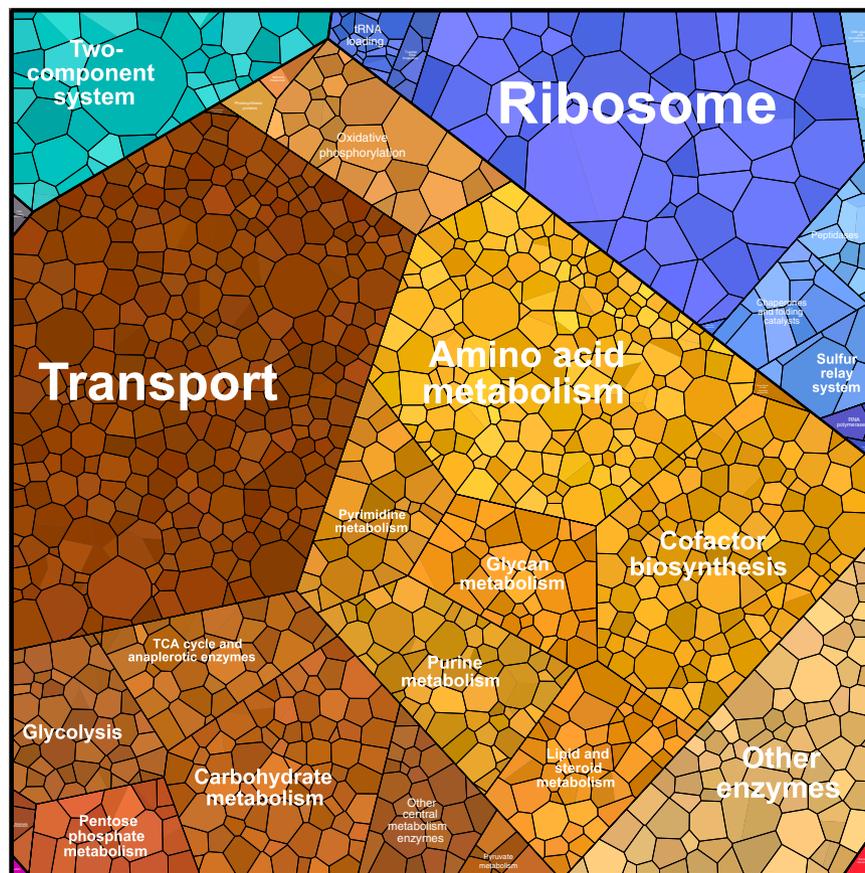


Fig. 3 Voronoi map of the predicted abundances of mRNAs. Each colored patch represents a different mRNA, with its area in proportion to its relative abundance. Genes can be clustered using KEGG Gene Ontology (GO) terms. Colors indicate the clustering.

which yields a proteome-limited phenotype, according to our results in Fig. 1a. It is important to note that all peptides with a nonzero minimal concentration (most of the left of the figure) are, by definition, essential peptides. These are always present at this uptake rate and are hence necessary for the cell to grow at an optimum growth rate. The same study can be performed for mRNA concentrations or even metabolite log-concentrations for models with thermodynamics. This type of study is useful for comparing how the model performs in relation to actual proteomics, transcriptomics, or metabolomics data. The method for running these other types of variability analyses is exactly the same—only the variables subject to the variability analysis are changed.

A specific usage of a variability analysis is the study of the allowed proteome (resp. transcriptome) that is done by performing a variability analysis on the enzyme (mRNA) concentration variables. This type of study can, for instance, be compared with transcriptomics to check if the expression profile of an engineered strain corresponds to what is expected in its corresponding model. A way to visualize the average allowed proteome (transcriptome) is to use the average value of the variability of each enzyme (mRNA) concentration as a feasible observation. Due to the convexity of the solution space, it is a solution to the problem. This observation is then plotted on a finite area, which can be done using the online software Proteomaps^{14,15}. This method and software are often used by biologists to represent protein abundances in the cell, and using the data from ETFL, we can generate similar comparative graphs that can help biologists analyze the variability in the different concentration variables using a visualization they are familiar with.

Figure 3 is an example of such a representation, graphed using the mRNA concentrations corresponding to the solution represented by the darker dots in Fig. 2 as an input. In this figure, mRNAs are clustered using KEGG Gene Ontology (GO) annotations. GO annotations form a tree describing the physiological role of genes, ranging from the least specific (e.g. general metabolism) to most specific (e.g. *araH* gene). The area of each (sub)cluster is proportional to the relative abundance of each (sub)group of mRNAs.

We used the mean of the variability analysis as the observation rather than a single optimal solution because the optimality principle in LP only guarantees a unique global optimum value and not a unique optimal solution. Moreover, solver heuristics give sparse and extreme results (corners of the explored simplex), which do not accurately represent the full extent of the considered solution space.

Essentiality analysis. The ETFL framework can also analyze the essentiality of specific genes by performing single gene knockouts. The growth of models with knocked-out genes can then be compared with the experimental data to assess the quality of the model as a validation.

We performed a gene essentiality analysis using in ETFL and compared it with the results reported in the publication of *iJO1366* by Orth et al.¹². We use the Matthew's correlation coefficient (MCC) as a metric for the quality of the prediction, which is preferred over accuracy as it is not sensitive to the imbalance between the number of essential genes and non-essential genes. The MCC reads like a usual correlation coefficient, with 1 being a perfect correlation, -1 perfect

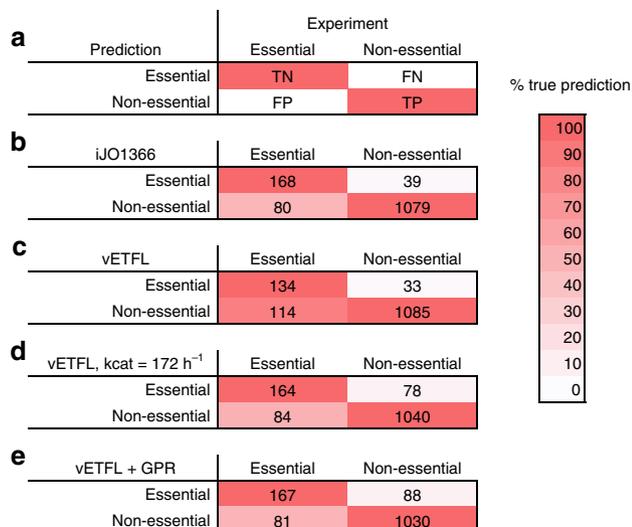


Fig. 4 Confusion matrices for gene essentiality studies. **a** Conventions from Orth et al.¹² for gene essentiality. TN is true negative. FN is false negative. FP is false positive. TP is true positive. The color shading represents how good the classification is. Perfect classification should have a strict red first diagonal, as shown on this example. **b** Gene essentiality prediction for the FBA model iJO1366, yielding a Matthew's correlation coefficient (MCC) of 0.69. **c** Gene essentiality prediction for the vETFL model, yielding a MCC of 0.60. **d** Gene essentiality prediction for the vETFL model with estimated enzymes with all $k_{cat} = 172 \text{ h}^{-1}$, yielding a MCC of 0.59. **e** Gene essentiality prediction for the vETFL model, where genes without enzyme assignment were tested using gene to protein to reaction (GPR) associations from the iJO1366 model, yielding a MCC of 0.59.

anti-correlation, and 0 no correlation. We used the essentiality data and conventions given in the Supplementary Material of Orth et al.¹², as explained in Fig. 4a, b. The results are presented in Fig. 4c, d, e, respectively, for vETFL, the model with estimated enzymes, and vETFL supplemented by iJO1366's Gene-Protein association Rules (GPRs).

Compared with iJO1366, we observe that ETFL predicts fewer false negatives (experimentally non-essential genes predicted as essential). However, ETFL also presents more false positives (experimentally essential genes predicted as non-essential). This indicates that ETFL is less constrained than iJO1366—the cell has more genetic alternatives for growth. This is an artificial effect that stems from the missing enzyme data.

As we add more enzyme information to the model, the false positive rate decreases. This is verified by Fig. 4d, where the addition of enzymes with average characteristics decreased the false positive rate. For comparison purposes, in Fig. 4e, we also computed the gene essentiality using iJO1366 gene to protein to reaction (GPR) associations for the genes which did not have an associated enzyme because of missing data. We show that the false positive rate decreases as well.

A detailed interpretation of the differences between gene knockout in ETFL and FBA is discussed in the Methods section. The Supplementary Data provide more insights on the mismatched between ETFL essentiality results and iJO1366 essentiality results, and indeed shows that 87% of the mismatches are attributed to reactions without enzymatic data. A significant fraction of mismatches (54%) come from the subsystems for the biosynthesis of lipids and cell envelope elements.

Sampling. Sampling the feasible solution space of FBA is a common way to study solution robustness and variability. Since there

are often multiple FBA solutions at the optimal objective value, representative solutions are often sought, and sampling is one way to obtain them. However, because ETFL contains integer variables, it is not compatible with traditional sampling methods in its current formulation. It is possible, though, to make the model convex, and hence amenable to sampling, by fixing the integers to their values at a given growth rate and, if applicable, TFA directionality. This will block the flux directions (if TFA is performed) as well as the growth-dependent parameters. The resulting model is then solely linear, and sampling can be performed with traditional techniques, such as artificially centered hit and run (ACHR)¹⁶, gpSampler¹⁷, or optGpSampler¹⁸. Once it has converged, a sampling should provide a better representation of the center of the solution space than the mean of the variability analysis.

Performance. For robustly reporting solution times of ETFL, we logged solving times each time a model was optimized during the redaction of this article. In that respect, some observations are the result of iterated optimizations, others from different optimization problems. In particular, variability and gene essentiality analyses require thousands of optimizations. We aggregated the solution times report the corresponding histograms, by model type, in Fig. 5. We measured the following metrics of the performance data: (i) arithmetic mean, (ii) geometric mean, and (iii) median. Although the distributions are not log-normal, it is common to report the geometric mean as a measure of the center of the distribution for comparison with other software^{19,20}, as it is more robust to outliers than the arithmetic mean and more sensitive to unevenness than the median.

Using well-established MILP solvers (CPLEX²¹, Gurobi²²), we report a geometric mean solution time of 7.47 s for vETFL, with 95% of the problems solved in <100 s on the test hardware. This is three orders of magnitude better than the reported solution time for O'Brien et al.⁷ (6 h – 2×10^4 s) and between one and two orders of magnitude better than the reported solution time for Lloyd et al. using cobraME⁸ (10 min – 6×10^2 s). It is worth noting that these vETFL optimizations also include thermodynamics constraints, which are absent of the other two formulations.

It is also important to state that although cobraME has an improved solution time over the original ME-model formulation, the formulation trades inequalities in the expression problem for equalities, and hence disregards a whole (non-growth optimal) part of the solution space that might contain physiological phenotypes. In particular, catalytic constraints become equalities, and the flux carried by reactions is set to be proportional to the amount of available enzyme instead of being upper-bounded by it. This gives less flexibility to the cell and prevents the representation of transient phenotypes. As an example, a cell that has been growing on a carbon source (e.g. glucose) will have a proteome suited to utilize this carbon source. However, once exhausted, it will need to reallocate its proteome to a new carbon source (e.g. lactose). In this transient state, some enzymes related to the first carbon source metabolism (e.g. glucose transporters) will carry no flux. In this case, cobraME would predict no flux, and also no enzyme concentration. In contrast, ETFL would allow for non-utilized enzymes and avoids such trade-offs, which is also crucial for accurately integrating proteomics data.

Such performance enhancements allow studies that would have been excessively time consuming using prior ME-model formulations. We show in Table 4 a list of typical completion times for common studies that require multiple optimizations to be carried out.

Finally, ETFL relies on solver-specific MILP algorithms and heuristics, which also means that great variability in performances can be observed depending on the solver parameters. We provide

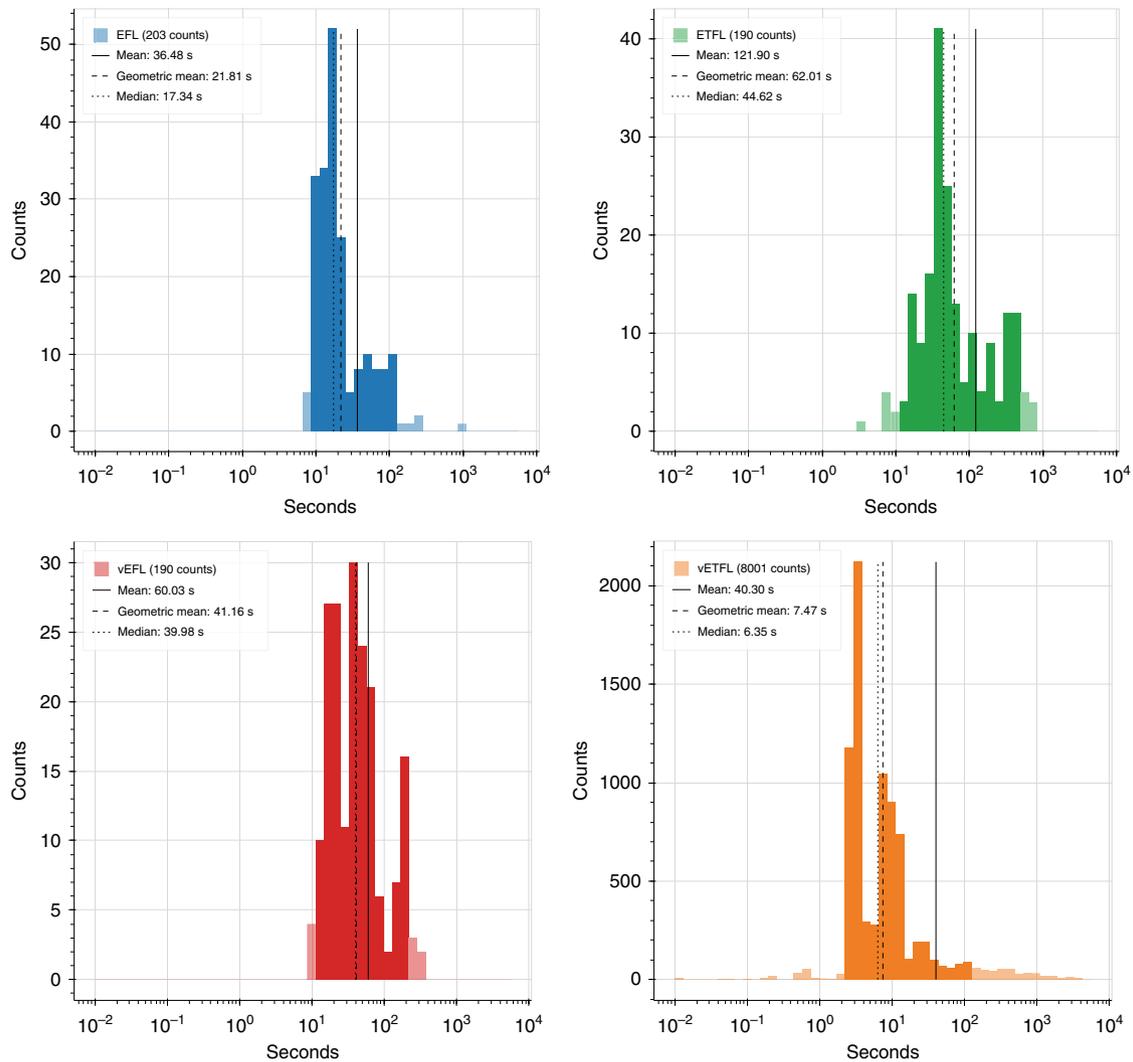


Fig. 5 Histograms displaying the distribution of solving times of each type of model during the data generation for this study. The darker area represents data between the 5th and 95th percentiles.

Table 4 Characteristic completion run times for several types of studies in the vETFL study of iJO1366.

Study type (vETFL)	vETFL characteristic run time (h)
Growth curve (Fig. 1)	1
Enzyme VA	1.5
mRNA VA	2–3
Gene essentiality	10
50-points dETFL (see Dynamic ETFL Method)	1

tuned presets for different tasks (gene knockout, variability analysis, growth maximization) with the package, and recommend that users run their own solver tuning if long run times are observed. We witnessed an up to 10x increase in performance using such tuning.

Adaptation of FBA-based methods to ETFL. The ETFL formulation is amenable to further kinds of analyses. Leveraging both the explicit expression constraints and the MILP nature of the problem, we present several possibilities for future studies using ETFL:

Growth-dependent parameters. It has been reported that several other parameters, such as the ribosome transcription rate constant k_{rib} , are growth dependent¹¹. Although such dependency is not taken into account in the presented results, it is possible to account for this by (i) discretizing k_{rib} following the method used to discretize the mRNA and protein content of the cell, and (ii) using Petersen’s linearization scheme (see the Methods section) on the product $k_{rib} * E_{rib}$. Other parameters that can be transformed in this way include, but are not limited to, the RNAP transcription rate constant k_{trans} , free ribosomes, and the RNAP ratios ρ and π .

Omics integration. Explicit mRNA and enzyme concentrations allow the direct integration of absolute or relative proteomics and transcriptomics by changing the bounds of the corresponding variables in the EP. An additional gauge constraint will be needed for the relative data. Previous transcriptomic integration methods, such as REMI²³, iMAT²⁴, GIMME²⁵, or MINEA²⁶, can also be adequately reformulated for ETFL. Metabolomics can still be integrated using TFA^{2,3}.

Minimization of adjustment. In the original paper, the hypothesis behind the Minimization of Metabolic Adjustment (MOMA) method is that the metabolic fluxes of an organism subject to a

gene knockout show a minimal change compared with the metabolic fluxes of the wild-type organism²⁷. The underlying hypothesis is that the enzyme distribution and assignments remain the same, except for the knocked-out gene. With ETFL, it is possible to directly compute a Minimization of Protein Adjustment (MOPA) by reformulating the objective function as such:

$$\min \sum_{j \in \mathcal{J}} \|E_j - E_j^0\|_p, \quad p \in \{0, 1\} \quad (\text{MOPA})$$

where $\|\cdot\|_p$ is either the Manhattan norm ($p = 1, \ell_1$ -norm) or the Euclidean norm ($p = 2, \ell_2$ -norm), which will require a MIQP solver. In the same fashion, it is also possible to formulate a (weighted) Minimization of mRNA Adjustment (MORA) or even a Minimization of eXpression Adjustment (MOXA) using the following formulations:

$$\min \sum_{l \in \mathcal{L}} \|F_l - F_l^0\|_p, \quad p \in \{0, 1\} \quad (\text{MORA})$$

$$\min \theta \cdot \sum_{j \in \mathcal{J}} \|E_j - E_j^0\|_p + (1 - \theta) \cdot \sum_{l \in \mathcal{L}} \|F_l - F_l^0\|_p, \\ p \in \{0, 1\}, \theta \in [1, 2]. \quad (\text{MOXA})$$

Parsimonious analysis. Parsimonious FBA (pFBA)¹⁷ was developed to address the high fluxes of some of the solutions given by FBA. Although this concern is addressed in ETFL by the combined actions of the EP and thermodynamics, pFBA can be adapted to ETFL to study an organism under parsimonious constraints. For example, it is possible to reformulate it into a parsimonious expression problem to find the minimal expression level required to meet a growth target using objective functions similar to (MOPA), (MORA), and (MOXA). It is also possible to turn the problem around to consider the allowed enzyme amounts under minimal flux constraint obtained by pFBA to assess the metabolic flexibility of an organism.

Dynamic ETFL (dETFL). Dynamic FBA (dFBA)²⁸ is a method that uses FBA to predict the dynamics of a biological system represented with a stoichiometric model. In its original static optimization approach (SOA) formulation, a FBA problem is solved at each time step. The value of boundary fluxes of the FBA problem are updated at each iteration with values produced with a kinetic law, such as Michaelis–Menten glucose uptake and oxygen diffusion. Because ETFL allows direct access to enzyme concentrations, it is possible to use the latter to reformulate dFBA in its SOA. The original SOA approach uses ad hoc constraints on the absolute flux change at each time step. However, in ETFL, it is possible to bound flux changes indirectly by bounding enzyme and mRNA concentration changes in the EP. Effectively, this approach allows the movement from ad hoc constraints to physiological constraints.

Use in kinetic frameworks. Often, kinetic frameworks require a reference flux distribution as an input. ETFL can provide such a distribution, with an increased accuracy as compared with FBA.

Building an ETFL ME-model for other organisms. Building an ETFL model from a genome-scale model follows a detailed procedure, for which a SOP is provided in the Supplementary Note 2. In this procedure, it is the quality of the input data that will determine the accuracy of the model. A well-curated, elementally balanced model is a critical prerequisite. Since ETFL is essentially adding constraints to the FBA problem, it is important as well to ensure the feasibility of the initial model.

In ETFL, and ME-models in general, catalytic constraints are what links the metabolism to the expression problem. Because of this, the accuracy of the ETFL reconstruction is also heavily dependent on the quality of the catalytic rate constants k_{cat}^j . Such information is not always easily accessible. Hence, we recommend to at least manually curate the catalytic rate constants of the key parts of metabolism, namely (i) ATP synthase, (ii) RNA polymerase, and (iii) ribosome. We also advise to pay attention to the pathways of the main carbon source metabolism, as small catalytic rate constants can heavily throttle the rest of the metabolism. For missing catalytic rate constants, a placeholder value can be used. O'Brien et al.⁷ used $k_{\text{cat}}^j = 65 \text{ s}^{-1}$, which is close to the median of the values used in this study. In our comparison with inferred enzymes, we used $k_{\text{cat}}^j = 172 \text{ s}^{-1}$, which is the arithmetic mean of the data we gathered.

Another key component for catalytically constraining the model is to have quality enzyme composition information. Indeed, marking an enzyme as a monomer instead of a dimer halves its synthesis cost. A good source for this information is MetaCyc²⁹, and literature. As explained in the previous paragraph, special attention should be given to the ATP synthase, the RNA polymerase, the ribosome, and the enzymes of the main carbon pathway. Macromolecule degradation rates are less critical and can be averaged. Growth-dependent protein, RNA, and DNA ratios drastically improve the quality of the model, as they allow to account for the expression activity that is related to non-metabolic processes.

In the construction of a model for another organism, approximating parameters based on values from an *E. coli* model should be done with care. Similarly to gap filling and the use of template reactions, conserving parameters across close species is helpful; however, conserving parameters across a large phylogenetic distance is erroneous. An example is the ribosome translation rate, which can vary by one order of magnitude between *S. cerevisiae* and *E. coli*.

Finally, great care should be taken with respect to the units. Different conventions are used across sources. Parameters for which this has been observed include catalytic rate constants, molecular weights, and concentrations.

Conclusions

ETFL is a framework which implements expression and thermodynamic formalism using mainstream double-precision MILP solvers. This could not be previously accomplished using state-of-the-art ME-models, which use specialized quad-precision solvers and do not support integer variables. The formalism itself is based on the explicit and direct relationship with the underlying biochemistry and provides a way to incorporate growth-dependent variables using MILP linearization techniques. These new growth-dependent variables provide a finer modeling of expression because they consider phenotypic differences in different growth regimens, which are key for accurate modeling. ETFL can also compute explicit mRNA and enzyme concentrations as well as perform direct -omics data integration. In this, ETFL complements and extends FBA capabilities by using explicit relationships in lieu of the typical assumptions on the relationships between the transcriptome, proteome, and fluxome. This explicit accounting of expression mechanisms provides a finer level of control and a more relevant prediction of gene-editing outcomes. ETFL is robust to missing data, as missing enzymes and their composition can be approximated using average enzyme characteristics. Because of this and its operational similarity with classic FBA-related analyses, ETFL can be efficiently integrated in standard model-based pipelines. For this intent, we provide in the Supplementary Note 2 a standardized procedure to produce ETFL models from genome-scale models. For example, metagenome-based genome-scale

reconstructions such as published by Magnúsdóttir et al.¹ can be directly fed to the framework to generate models for each of the 773 bacteria they identified. Integration with platforms like KBase³⁰ can also be envisioned to automatically draft EFTL reconstructions parametrized by curated organism-specific data. In a more general way, EFTL can assess the allowed expression profiles of any biological system amenable to genome-scale modeling, such as in the metabolic engineering of biocatalysts, microbial communities, drug design, or personalized medicine.

Methods

Preliminaries, conventions, and notations. The mass balances of the macromolecules in ME-models are written with respect to their concentration variables. If we assume the cell is growing at a specific growth rate μ , we must assume that the volume of cell within which the mass balance is considered varies.

The mass balance of a macromolecule G will be written:

$$\frac{dm_G}{dt} = C_G \frac{dV_c}{dt} + V_c \frac{dC_G}{dt}, \quad (5)$$

$$= v_G^{\text{syn}} \cdot V_c - v_G^{\text{deg}} \cdot V_c, \quad (6)$$

where C_G is the concentration of the macromolecule C_G in the cellular volume V_c , for a total mass m_G in the cell, produced at a rate v_G^{syn} and degraded at a rate v_G^{deg} .

We next combine Eqs. (5) and (6) and divide by V_c (necessarily nonzero) to write the time derivative of the concentration C_G :

$$\frac{dC_G}{dt} = v_G^{\text{syn}} - v_G^{\text{deg}} - \frac{1}{V_c} \frac{dV_c}{dt} \cdot C_G. \quad (7)$$

By definition, $\frac{1}{V_c} \frac{dV_c}{dt} = \mu$ is the specific growth rate of the cell (under the assumption of constant cell density ρ_c), and the term $\mu \cdot C_G$ is called the dilution term, or v_G^{dil} , as per Fredrickson's work on formulating growth models³¹. It is a common assumption that the concentrations inside the cell remain time invariant (quasi-steady-state assumption), effectively yielding the constraint:

$$v_G^{\text{syn}} - v_G^{\text{deg}} - \frac{1}{V_c} \frac{dV_c}{dt} \cdot C_G = 0. \quad (8)$$

It is also understood from the formulation of the FBA that adding a new reaction to the system, such as:



results in adding terms to the mass balances of A and B :

$$\frac{d[A]}{dt} = \dots - \eta_A^j \cdot v_j, \quad (10)$$

$$\frac{d[B]}{dt} = \dots + \eta_B^j \cdot v_j. \quad (11)$$

The further extension of this to reactions of n reactants to m products is trivial.

Several parameter values are taken from the BioNumbers database³². When used, we specify their identification number as well as the original source from which the value was reported. Finally, we will represent products between a parameter value and a variable by the symbol “.” and products between two variables by the symbol “*”.

Hereafter, we propose a detailed top-down approach to formulate the constraints being built for EFTL, starting from the metabolite network and moving down to RNA synthesis. The general organization for each macromolecule is to write down its mass balance, apply assumptions, and then detail its synthesis and consumption mechanisms.

Metabolites. From FBA, the mass-balance relationship for metabolites can be written as:

$$S \cdot v = 0. \quad (\text{FBA})$$

For the rest of the formulation, it is necessary to split the net flux v from each reaction into its forward net component and backward net component:

$$v_j = v_j^+ - v_j^-, \quad v_j^+, v_j^- \geq 0. \quad (12)$$

Biochemical reactions are catalyzed by enzymes. Each enzyme (Enz_j) of concentration E_j can catalyze a flux v_j subject to the enzyme capacity constraint, which is a function of its forward and backward catalytic rate constants $k_{\text{cat}}^{j,+}$ and $k_{\text{cat}}^{j,-}$:

$$0 \leq v_j^+ \leq k_{\text{cat}}^{j,+} E_j, \quad (13)$$

$$0 \leq v_j^- \leq k_{\text{cat}}^{j,-} E_j, \quad (14)$$

$$v_j^+ - k_{\text{cat}}^{j,+} E_j \leq 0, \quad (\text{FC}_j)$$

$$v_j^- - k_{\text{cat}}^{j,-} E_j \leq 0. \quad (\text{BC}_j)$$

The distinction between the bounds of the forward and backward net fluxes is important, as some enzymes have different catalytic activities, depending on the direction of the flux.

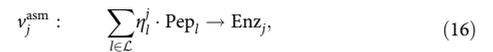
General constraints for enzymes. Each enzyme Enz_j is represented by its total concentration, the variable E_j . It is subject to mass balance, which can be written:

$$\frac{d}{dt} E_j = v_j^{\text{asm}} - v_j^{\text{deg}} - v_j^{\text{dil}}, \quad (15)$$

which reads under quasi-steady-state assumption (QSSA):

$$v_j^{\text{asm}} - v_j^{\text{deg}} - \mu * E_j = 0, \quad (\text{EB}_j)$$

where v_j^{asm} is the formation rate of the enzyme by the assembly of its constituent peptides, v_j^{deg} is the degradation rate, v_j^{dil} is the dilution rate, and μ is the growth rate of the cell. The formation rate of the enzyme describes the assembly of free peptides, hence it is necessary to add the peptide assembly reaction to the stoichiometric matrix:



where η_l^j is the stoichiometric coefficient of peptide Pep_l for the formation of the complex of enzyme Enz_j . This reaction is assumed to happen spontaneously by default.

We model the degradation reaction of the enzyme in the following manner:



where $\eta_{\text{aa}_i}^j$ is the number of amino acids aa_i in the enzyme. It is obtained from the composition of the constituent peptides. For this degradation reaction, the rate is known:

$$v_j^{\text{deg}} - k_{\text{deg}}^j \cdot E_j = 0, \quad (\text{ED}_j)$$

where k_{deg}^j is the degradation rate constant of the enzyme. The reaction is added to the model, and the Eq. ED_j is added as a constraint.

Constraints specific to ribosomes. Like any other enzyme, ribosomes verify the mass balance:

$$v_{\text{rib}}^{\text{asm}} - v_{\text{rib}}^{\text{deg}} - \mu * E_{\text{rib}} = 0 \quad (\text{EB}_{\text{rib}}).$$

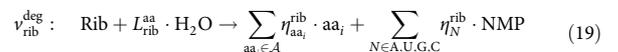
E_{rib} denotes the total concentration of ribosomes in a cell. It accounts for R_j , the ribosomes assigned to the translation of Pep_j , as well as the free ribosomes in the cell, R_F .

The ribosome differs from other enzymes in that it takes ribosomal peptides rPep_j , as well as ribosomal RNA rRNA_j for its assembly. Hence, its assembly reaction is:



As explained earlier, the stoichiometric coefficients η_k^{rib} will appear in the mass balances of each of the compounds of the reaction. This reaction is also assumed to happen spontaneously by default.

When ribosomes are degraded, their constituting amino acids and ribonucleotides are recovered:



The degradation rate is constrained in a manner similar to the constraint (ED_j) .

Finally, we can then write the total ribosome capacity constraint:

$$\sum_{l \in \mathcal{L}} R_j + R_F - E_{\text{rib}} = 0. \quad (\text{TC2})$$

If we know the ratio ρ of occupied vs free ribosomes, we can enforce it:

$$R_F - (1 - \rho) E_{\text{rib}} = 0. \quad (\text{RR})$$

Constraints specific to RNA polymerase. RNAP is an enzyme, and hence it also satisfies mass balance:

$$v_{\text{RNAP}}^{\text{asm}} - v_{\text{RNAP}}^{\text{deg}} - \mu * E_{\text{RNAP}} = 0, \quad (\text{EB}_{\text{RNAP}})$$

where E_{RNAP} is the total amount of RNAP, which also accounts for free RNAP P_F .

Its synthesis and degradation follow equations similar to other enzymes:

$$v_{\text{RNAP}}^{\text{asm}} - v_{\text{RNAP}}^{\text{deg}} - \mu * E_{\text{RNAP}} = 0, \quad (EB_{\text{RNAP}})$$

with the same conventions as in Eq. (EB₁). As for a generic enzyme, RNAP is assembled from free peptides, which adds the peptide assembly reaction to the stoichiometric matrix:

$$v_{\text{RNAP}}^{\text{asm}} : \sum_{l \in \mathcal{L}} \eta_l^{\text{RNAP}} \cdot \text{Pep}_l \rightarrow \text{RNAP}, \quad (20)$$

again with the same conventions as in the section General constraints for enzymes. This reaction is also assumed to happen spontaneously by default. The degradation reaction is also modeled similarly, with the same conventions:

$$v_{\text{RNAP}}^{\text{deg}} : \text{Enz}_j + L_j^{\text{aa}} \cdot \text{H}_2\text{O} \rightarrow \sum_{\text{aa}_i \in \mathcal{A}} \eta_{\text{aa}_i}^j \cdot \text{aa}_i. \quad (21)$$

The degradation rate is constrained in a manner similar to the constraint (ED₁).

In addition, the total capacity of RNAP follows a capacity constraint similar to that of ribosomes:

$$\sum_{l \in \mathcal{L}} P_l + P_F - E_{\text{RNAP}} = 0. \quad (\text{TC1})$$

As we did with the ribosomes, if we know the ratio of occupied RNAP, π , we can enforce it:

$$P_F - (1 - \pi)E_{\text{RNAP}} = 0. \quad (\text{PR})$$

Constraints for peptides. The peptide concentrations obey the mass-balance equation:

$$\frac{d}{dt} \text{Pep}_l = v_l^{\text{isl}} - \sum_{j \in \mathcal{J}} \eta_l^j \cdot v_j^{\text{asm}} - v_l^{\text{deg}} - v_l^{\text{dil}}. \quad (22)$$

We assume in the current model that the protein assembly rates are much faster than dilution and degradation, and thus simplify this mass balance to:

$$\frac{d}{dt} \text{Pep}_l = v_l^{\text{isl}} - \sum_{j \in \mathcal{J}} \eta_l^j \cdot v_j^{\text{asm}}, \quad (23)$$

which, under QSSA, can be written:

$$v_l^{\text{isl}} - \sum_{j \in \mathcal{J}} \eta_l^j \cdot v_j^{\text{asm}} = 0. \quad (\text{PB}_l)$$

In this context, the peptides are treated just like regular metabolites in the system. This assumption in (PB_l) can be relaxed without a loss of generality by introducing a dilution and a degradation term, thus introducing a bilinearity.

The synthesis of peptides consumes charged tRNAs, which are subsequently uncharged during the current peptide synthesis by a ribosome. The process consumes two GTPs and releases two GDPs and two Pi per amino acid:

$$v_l^{\text{isl}} : \sum_{\text{aa}_i \in \mathcal{A}} \eta_{\text{aa}_i}^l \cdot \text{tRNA}_{\text{aa}_i}^{\text{charged}} + 2L_l^{\text{aa}} \cdot (\text{GTP} + \text{H}_2\text{O}) \rightarrow \text{Pep}_l + \sum_{\text{aa}_i \in \mathcal{A}} \eta_{\text{aa}_i}^l \cdot \text{tRNA}_{\text{aa}_i}^{\text{uncharged}} + 2L_l^{\text{aa}} \cdot (\text{GDP} + \text{Pi} + \text{H}^+), \quad (24)$$

where aa_i denotes the *i*th amino acid, $\eta_{\text{aa}_i}^l$ its stoichiometric coefficient (count) in the sequence of Pep_l, tRNA_{aa_i}^{*} the (un)charged tRNAs for each amino acid, and $L_l^{\text{aa}} = \sum_{\text{aa}_i \in \mathcal{A}} \eta_{\text{aa}_i}^l$ is the length of the amino acid sequence of Pep_l.

As explained in the section Preliminaries, conventions, and notations, this reaction adds a supplementary term in the mass balances of the metabolites (GTP, GDP, Pi, H₂O, H⁺), the peptide, and the tRNAs (see Constraints specific to tRNAs for the latter). This term is what connects the expression requirements to the metabolic network defined in the FBA.

The peptides are the product of a translation reaction that is catalyzed by a ribosome. As we did with the catalytic constraints for general biochemistry reactions, we can apply the ribosome maximum catalytic rate as an upper bound to its translation rate v_l^{isl} :

$$v_l^{\text{isl}} - \frac{k_{\text{cat}}^{\text{rib}}}{L_l^{\text{aa}}} R_l \leq 0, \quad (\text{TR}_2l)$$

where $k_{\text{cat}}^{\text{rib}}$ is the maximum ribosomal translation rate constant (10 – 12aa.s⁻¹ for *E. coli*, BioNumbers ID [BNID] 100059³³), L_l^{aa} is the amino acid length of the peptide *l*, and R_l is the concentration (in mmol.gDW⁻¹) of ribosomes assigned to the translation of this peptide. This way, the ratio R_l/Pep_l is effectively the number of ribosomes, or average polysome size, translating the peptide *l*.

Constraints for mRNAs. During the translation, an mRNA is read to produce a peptide. mRNAs are subject the same mass-balance constraints:

$$v_l^{\text{tcr}} - v_l^{\text{deg}} - \mu * F_l = 0, \quad (\text{MB}_l)$$

where F_l is the total concentration of the *l*th mRNA (mRNA_l), v_l^{deg} is its degradation rate, and v_l^{tcr} is its transcription (synthesis) rate. F_l is variable that represents

the concentration of (RNA_l). The transcription reaction is modeled as follows:

$$v_l^{\text{tcr}} : \eta_A^l \cdot \text{ATP} + \eta_U^l \cdot \text{UTP} + \eta_C^l \cdot \text{CTP} + \eta_G^l \cdot \text{GTP} \rightarrow (\eta_A^l + \eta_U^l + \eta_C^l + \eta_G^l) \text{PPi} + \text{mRNA}_l. \quad (25)$$

Again, the stoichiometric coefficients will appear in the mass balances of each of the metabolites and macromolecules involved. The transcription process is catalyzed by RNA polymerase (RNAP). For each transcription of mRNA, we can put an upper bound on the transcription rate v_l^{tcr} in the same way as for translation:

$$v_l^{\text{tcr}} - \frac{k_{\text{cat}}^{\text{RNAP}}}{L_l^{\text{nt}}} P_l \leq 0, \quad (\text{TR1}_l)$$

where L_l^{nt} is the length in nucleotides of the mRNA sequence, $k_{\text{cat}}^{\text{RNAP}}$ is the catalytic rate constant of RNAP (85 nt.s⁻¹ for *E. coli*, BNID 100060³³), and P_l the concentration of RNAP assigned to the transcription of this mRNA.

We must also take into account the relationship between ribosome assignment and mRNA concentration. On each strand of mRNA_l, there can be only a finite number ρ_l of ribosomes translating at the same time. This number is given by the ratio of the footprint size of the ribosome $L_{\text{rib}}^{\text{nt}}$ and the length of the mRNA strand L_l^{nt} . This effectively yields the number of ribosomes that can be present at the same time on a given mRNA strand:

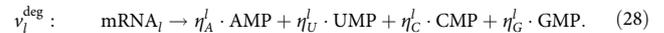
$$\rho_l = \frac{L_l^{\text{nt}}}{L_{\text{rib}}^{\text{nt}}}. \quad (26)$$

For *E. coli*, $L_{\text{rib}}^{\text{nt}}$ is ~20 nm (BNID 102320³⁴, 100121³⁵), which amounts to ~60 base pairs (the length of a nucleotide is ~0.3 nm; BNID 103777³⁶). From there we can get the additional constraint:

$$R_l \leq \rho_l \cdot F_l, \quad (27)$$

$$R_l - \frac{L_l^{\text{nt}}}{L_{\text{rib}}^{\text{nt}}} F_l \leq 0. \quad (\text{EX}_l)$$

We consider the following degradation reaction for mRNAs:



And, again, we know the degradation rates:

$$v_l^{\text{deg}} - k_{\text{deg}}^l \cdot F_l = 0. \quad (\text{MD}_l)$$

Constraints specific to rRNAs. rRNAs are used in the ribosome assembly reaction. According to the definition of $v_{\text{rib}}^{\text{asm}}$ in the section Constraints specific to ribosomes, their mass balance can be written:

$$\frac{d}{dt} [\text{rRNA}_l] = 0 = v_{\text{rRNA}_l}^{\text{tcr}} - v_{\text{rib}}^{\text{asm}} - v_{\text{rRNA}_l}^{\text{deg}} - v_{\text{rRNA}_l}^{\text{dil}}. \quad (29)$$

We neglect their dilution and degradation under the hypothesis that free rRNAs are scarce and stable³⁷. Thus, their mass balance in the model reads:

$$v_{\text{rRNA}_l}^{\text{tcr}} - v_{\text{rib}}^{\text{asm}} = 0. \quad (\text{RB}_{\text{rRNA}_l})$$

The degradation reaction is the same as for mRNA, and is part of the total degradation of the ribosome.

Constraints specific to tRNAs. Since tRNAs are relatively stable molecules³⁷, we neglect their degradation. Let $T_{\text{aa}_i}^u$ (resp. $T_{\text{aa}_i}^c$) represent [tRNA_{aa_i}^{uncharged}] (resp.

[tRNA_{aa_i}^{charged}]). Then, we can write the following constraints:

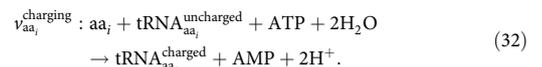
$$\frac{d}{dt} T_{\text{aa}_i}^u = 0 = -v_{\text{aa}_i}^{\text{charging}} + \sum_{l \in \mathcal{L}} \eta_{\text{aa}_i}^l \cdot v_l^{\text{isl}} - \mu * T_{\text{aa}_i}^u, \quad (30)$$

$$\frac{d}{dt} T_{\text{aa}_i}^c = 0 = v_{\text{aa}_i}^{\text{charging}} - \sum_{l \in \mathcal{L}} \eta_{\text{aa}_i}^l \cdot v_l^{\text{isl}} - \mu * T_{\text{aa}_i}^c, \quad (31)$$

$$-v_{\text{aa}_i}^{\text{charging}} + \sum_{l \in \mathcal{L}} \eta_{\text{aa}_i}^l \cdot v_l^{\text{isl}} - \mu * T_{\text{aa}_i}^u = 0, \quad (\text{TB}_{\text{aa}_i}^u)$$

$$v_{\text{aa}_i}^{\text{charging}} - \sum_{l \in \mathcal{L}} \eta_{\text{aa}_i}^l \cdot v_l^{\text{isl}} - \mu * T_{\text{aa}_i}^c = 0. \quad (\text{TB}_{\text{aa}_i}^c)$$

tRNAs are produced with a charging reaction and consumed by peptide synthesis. We use the following charging reaction:



By default, this reaction is assumed to happen spontaneously, but catalytic constraints can be applied if the adequate catalytic rate constants and enzyme compositions are known. Once again, the stoichiometric coefficients of each reactant will appear in the stoichiometric matrix in the column corresponding to this reaction.

Reformulation of the bilinearity of the problem. The main issue with the EP formulation presented previously lies in the continuous bilinear terms that describe the dilution of the macromolecules, $G_* \in \{E_j\} \cup \{F_i\} \cup \{\text{tRNA}_{aa}^{\text{(un)charged}}\}$. We use $*$ as a placeholder for the indexing of G . Using previous notations for the synthesis, degradation, and growth rate:

$$v_*^{\text{syn}} - v_*^{\text{deg}} - \mu * G_* = 0. \quad (33)$$

In this state, the dilution term is bilinear, and the formulation requires a bilinear solver or potentially a mixed-integer bilinear solver if thermodynamics are to be added. The original ME-model formulation has similar terms as we are presenting here^{6,7}. As such, its recent adaptation in Lloyd et al.⁸ uses the two-level iterative algorithm SolveME⁹ that requires a dedicated nonlinear solver. In this fashion, iterative approaches which try to sequentially improve a value of the growth are a way to deal with the bilinearity. We present instead a MILP approximation of the problem that makes it compatible and solvable with mainstream MILP solvers in a single optimization formulation. We achieve this through the discretization and linearization of the bilinear products. This operation can be understood as locally approximating the bilinear problem by several linear subproblems and choosing the best approximation.

Using a MILP approximation rather than an iterative scheme has two clear advantages. First, it allows to simulate growth-dependent parameters (such as RNA/protein mass ratios) with guarantees on convergence and global optimality directly inherited from the MILP nature of the problem. In the case of parameters that are monotonically increasing or decreasing with respect to growth rate, guarantees exist in quadratically-constrained programming (QCP), such as showed in SteadyCom³⁸. However, in the case of non-monotonically increasing or decreasing parameters with respect to the growth rate, such guarantees are harder to prove in general QCP cases, and thus MILP provides a strong framework, with global optimality guarantees and enumeration of alternative solutions. Second, by displacing the solving complexity to the solver, it also allows us to rely on the latest advances in MILP solving, which is a very dynamic field, with new solver releases every 6–12 months.

Approximation of the growth rate. In ETFL, we approximate the growth rate μ in bilinear products with a piecewise-constant function $\hat{\mu}$ (0th order approximation). A zeroth-order approximation is an approximation by a piecewise-constant function. If $\hat{\mu}$ is piecewise-constant, then the product $\hat{\mu} * G_*$ is piecewise-linear. This can be represented in a MILP form, and allows us to transform the continuous bilinear terms into mixed (integer \times continuous) bilinear terms. This simplifies the problem, as these mixed bilinear terms can be linearized in a MILP setting using the Petersen linearization scheme³⁹, a particular case of the Glover linearization scheme⁴⁰ that was previously used in metabolic engineering by Hatzimanikatis et al.^{41,42}.

Let $\bar{\mu}$ be an upper bound to μ , $(p, N) \in \mathbb{N}^2, p \leq N$. We can approximate μ with the following 0th order approximation:

$$\forall \mu \in [0, \bar{\mu}], \quad \mu \approx \hat{\mu} = p \cdot \frac{\bar{\mu}}{N}. \quad (34)$$

With this notation, $\frac{\bar{\mu}}{N}$ is, in fact, the resolution of the approximation. N is the number of bins in which μ has been discretized, and p allows to choose which bin is selected in the solution. For the linearization of the problem, we will need to express p using only binary variables. To this effect, we can perform its binary expansion:

$$p = \sum_{s=0}^{\lceil \log_2 N \rceil} 2^s \cdot \delta_s, \quad (35)$$

where $\lceil \log_2 N \rceil$ denotes the smallest majoring integer to $\log_2 N$, and $\delta_s \in \{0, 1\}$ is r^{th} digits from the right of the binary notation of p .

The model needs two more constraint to ensure that $\mu \in [\hat{\mu} - \frac{p}{N}, \hat{\mu} + \frac{p}{N}]$ and that p does not exceed N , which would result in $\hat{\mu} > \bar{\mu}$:

$$0 \leq \sum_{s=0}^{\lceil \log_2 N \rceil} 2^s \cdot \delta_s \leq N \quad (36)$$

$$-\frac{p}{N} \leq \mu - \hat{\mu} \leq \frac{p}{N}. \quad (37)$$

As an example, let us consider modeling an organism whose growth rate does not exceed $\mu_{\text{max}} = 2.3 \text{ h}^{-1}$. To do this, we can set $\bar{\mu} = 2.5 \geq \mu_{\text{max}}$. Let us choose a resolution of 0.25 h^{-1} , which gives $N = 10$. Then, $\log_2 N \approx 3.32$, and $\lceil \log_2 N \rceil = 4$. A growth rate $\mu = 1.4$ will be approximated by:

$$\hat{\mu} = 1.5 = 6 \cdot \frac{\bar{\mu}}{10},$$

$$\hat{\mu} = (\delta_0 \times 2^0 + \delta_1 \times 2^1 + \delta_2 \times 2^2 + \delta_3 \times 2^3 + \delta_4 \times 2^4) \cdot \frac{\bar{\mu}}{10},$$

$$\hat{\mu} = (0 \times 1 + 1 \times 2 + 1 \times 4 + 0 \times 8 + 0 \times 16) \cdot \frac{\bar{\mu}}{10}.$$

The values of δ_s are obtained by the solver upon optimization. This example is illustrated in Fig. 6a. To maximize the resolution of the model, and minimize the

associated computational cost (under the form of three additional constraints for each linearization to be performed, see Petersen linearization in the Methods section), the user should ideally choose N as a power of two.

MILP solvers use a variety of algorithms and heuristics to solve MILP problems. In this case, the difficulty lies in the fact that the EP and the FBA are almost independent and linked through a limited number of equations and variables. Even though the automated solving methods of the solver might seem obscure to a human, we thought useful to provide a human-understandable heuristic for solving a formulation such as ETFL. It might prove useful in the case where one needs to find an initial non-optimal solution, which sometimes greatly improve solver performances. Thus, conceptually, a heuristic for solving an ETFL problem would be:

1. Solve the FBA for μ
2. Select the corresponding, closest value of $\hat{\mu}$
3. Apply it to compute dilution values
4. Solve the EP with fixed dilution
5. Apply the catalytic constraints to the FBA
6. Recalculate the FBA under catalytic constraints
7. If $\mu \notin \{\hat{\mu} \pm \frac{\bar{\mu}}{N}\}$, go back to 3, else, end.

Linearizing the bilinearity. In the previous derivation, we replaced the growth rate variable by a discrete number of acceptable values. We can approximate the continuous product $\mu * G_*$, which represents the dilution, as follows:

$$\mu * G_* \approx \hat{\mu} * G_*, \quad (38)$$

$$\hat{\mu} * G_* = \sum_{s=0}^{\lceil \log_2 N \rceil} \frac{2^s}{N} \bar{\mu} \cdot \delta_s * G_*, \quad (39)$$

The product $\delta_j * G_*$ is then still bilinear, but one of its variables is binary. Assuming a constant $M > G_*$, We can use Petersen's linearization theorem^{39,40} to replace the product $\delta_s * G_*$ with a single nonnegative variable z_s^* , as described in the section Petersen linearization.

Because of the binary expansion, the complexity of the model grows only as $\mathcal{O}(\log_2 N) = \mathcal{O}(\log_2 \frac{1}{\epsilon})$, where $\epsilon = 1/N$ is proportional to the resolution of the approximation (which is $\frac{\bar{\mu}}{N}$). This means that the linearization part of a model with a resolution of 0.01 h^{-1} is only around twofold bigger than that of a model with a resolution 0.04 h^{-1} , while resolution has been improved fourfold.

Petersen linearization. After discretization of the growth rate, the dilution term for the macromolecule G_* will consist of a sum of products of the binary variables δ_s and the continuous variable G_* . We can use the Petersen linearization scheme³⁹ to transform this product into an equivalent system of one new variable and three new constraints:

$$\begin{aligned} z_s^* &= \delta_s * G_*, \\ &\iff \begin{cases} G_* + M \cdot \delta_s - M \leq z_s^* \leq M \cdot \delta_s, \\ z_s^* \leq G_*, \end{cases} \\ &\iff \begin{cases} G_* + M \cdot \delta_s - z_s^* \leq M, \\ z_s^* - M \cdot \delta_s \leq 0, \\ z_s^* - G_* \leq 0. \end{cases} \end{aligned} \quad (40)$$

With this method, we can directly reformulate generalized mass balances as described in Eq. (33) for mRNAs, enzymes, uncharged tRNAs, and charged tRNAs:

$$v_j^{\text{asm}} - v_j^{\text{deg}} - \sum_{s=0}^{\lceil \log_2 N \rceil} \frac{2^s}{N} \bar{\mu} \cdot z_j^s = 0, \quad (\text{EB}_j^{\text{u}})$$

$$v_j^{\text{tr}} - v_j^{\text{deg}} - \sum_{s=0}^{\lceil \log_2 N \rceil} \frac{2^s}{N} \bar{\mu} \cdot z_j^s = 0, \quad (\text{MB}_j^{\text{u}})$$

$$-v_{aa_i}^{\text{charging}} + \sum_{aa_i \in \mathcal{A}} \eta_{aa_i}^{\text{I}} \cdot v_i^{\text{I}} - \sum_{s=0}^{\lceil \log_2 N \rceil} \frac{2^s}{N} \bar{\mu} \cdot z_{aa_i}^{u,s} = 0, \quad (\text{TB}_{aa_i}^{\text{u}})$$

$$v_{aa_i}^{\text{charging}} - \sum_{aa_i \in \mathcal{A}} \eta_{aa_i}^{\text{I}} \cdot v_i^{\text{I}} - \sum_{s=0}^{\lceil \log_2 N \rceil} \frac{2^s}{N} \bar{\mu} \cdot z_{aa_i}^{c,s} = 0. \quad (\text{TB}_{aa_i}^{\text{c}})$$

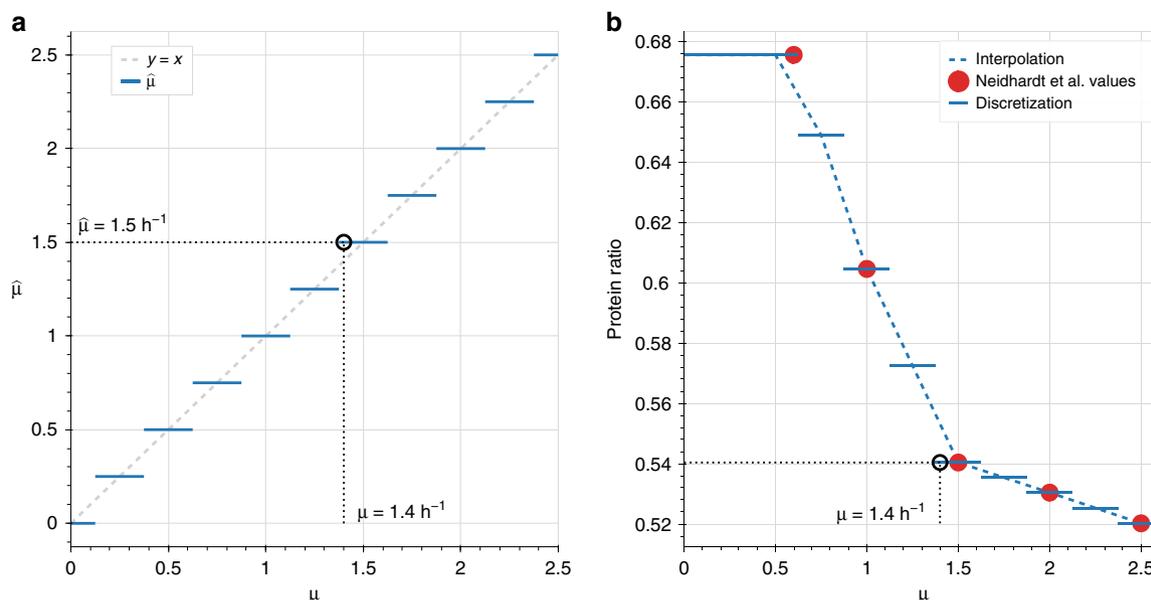


Fig. 6 Discretization example for specific growth rate and growth-dependent parameters. **a** Discretization of μ into $\hat{\mu}$. The step approximation transforms the continuous interval $[0, 2.5]$ into the discrete set $\{0, 0.25, \dots, 2.5\}$. **b** Example of piecewise linear interpolation and discretization of the protein mass ratio from Neidhardt et al.¹¹. Red circles represent the values reported. The dashed line is the piecewise linear interpolation. The solid line is its discretization.

And we get the additional linearization constraints:

$$\sum_{s=0}^{\lfloor \log_2 N \rfloor} \frac{2^s}{N} \bar{\mu} \leq \bar{\mu}, \quad (GR)$$

$$-\frac{\bar{\mu}}{2N} \leq \mu - \sum_{s=0}^{\lfloor \log_2 N \rfloor} \frac{2^s}{N} \bar{\mu} \leq \frac{\bar{\mu}}{2N}. \quad (GC)$$

Discretization of mRNA and enzyme content. Since the growth has been discretized, it is now possible to also directly discretize other growth-dependent parameters of the problem, regardless of whether they are in a linear or nonlinear relationship with growth. This is a direct consequence of the formulation of ETFL, which allows some flexibility in the modeling assumptions of the user. As an example, we described the relationship between growth and protein and mRNA mass ratios, P^m and R^m , in the cell as reported in Neidhardt et al.¹¹. We thus aim to approximate the nonlinear function $P^m(\mu)$ (resp. $R^m(\mu)$) over the interval $[0, \bar{\mu}]$ with a piecewise-constant function \hat{P}^m (resp. \hat{R}^m). We perform this approximation by interpolating and discretizing the protein ratio and mRNA ratio as functions of the growth rate so that:

$$\hat{P}^m = \sum_{u \in \mathcal{U}} \lambda_u \cdot P_u^m, \quad (41)$$

$$\hat{R}^m = \sum_{u \in \mathcal{U}} \lambda_u \cdot R_u^m, \quad (42)$$

where $P_u^m = P^m(u \cdot p \frac{\bar{\mu}}{N})$ (resp. $R_u^m = R^m(u \cdot p \frac{\bar{\mu}}{N})$). λ_u are binary variables, and only one can be active at a time, since we are choosing exactly one value per function. To enforce this behavior, we used a special ordered set constraint of type 1 (SOS1):

$$\sum_{j \in \mathcal{J}} MW_j \cdot E_j - \sum_{u \in \mathcal{U}} \lambda_u \cdot P_u^m = 0, \quad (IC1)$$

$$\sum_{l \in \mathcal{L}} MW_l \cdot F_l - \sum_{u \in \mathcal{U}} \lambda_u \cdot R_u^m = 0, \quad (IC2)$$

$$\sum_{u \in \mathcal{U}} \lambda_u = 1. \quad (SOS1)$$

P_u^m and R_u^m are growth-dependent, interpolated protein and RNA mass ratios (in $\text{g} \cdot \text{g}^{-1}$). Given a growth rate, they define the relative mass of the cell that is protein or RNA. MW_u represents the molar weight of the corresponding enzyme or RNA, and this their product with macromolecules concentrations (in $\text{mmol} \cdot \text{ggDW}^{-1}$) will result in mass ratios as well, in grams per gram of dry cell weight. The first two constraints enforce equality between the interpolated data and the model production. The last line is the SOS1 constraint that forces only one of the λ_u to be active.

In addition, it is necessary to have the integer index of λ_u equal to the index of the growth rate. This is obtained through the constraint:

$$\sum_{u \in \mathcal{U}} u \cdot \lambda_u - \sum_{l \in \mathcal{L}} 2^l \cdot \delta_l = 0. \quad (EQI)$$

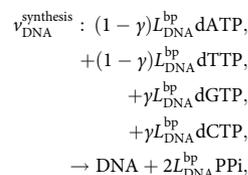
The first term represents the growth integer index (which discrete value of $\hat{\mu}$ to use for choosing P_u^m), and the second represents its binary expansion (which discrete value of $\hat{\mu}$ to use for μ). The constraint makes sure they are equal.

Imposing such mass ratios requires the addition of a dummy mRNA as well as a dummy protein to represent the part of the transcriptome/proteome that is either missing from the expression model or altogether unrelated to metabolic function. We use average amino acid frequencies and GC content to model this. Explicit interpolation functions can also be used, such as the growth-dependent functions given by Pramanik et al.⁴³.

The simultaneous use of catalytic constraints on metabolic reactions (Eq. FC_j, BC_j) and maximal enzyme load (Eq. IC3) effectively implements allocation constraints like in GECKO⁵, although in ETFL, the enzyme concentrations are also directly linked to the metabolism. In GECKO, the metabolic cost of building the enzymes is not taken into account.

Figure 6b shows an example piecewise linear interpolation of the growth-dependent protein mass ratio in *E. coli* according to Neidhardt et al.¹¹. The reported values (red circles) are interpolated using a piecewise linear function (dashed line), which is then discretized (full line). Using the integer constraints described above, the model can be forced to display a protein content that corresponds to its growth. We apply the same techniques to mRNA and DNA content.

Discretization of DNA content. To further increase the scope of macromolecules covered by the model, it is also possible to add growth-dependent DNA content. DNA mass ratios at specific growth rates are reported in Neidhardt et al.¹¹. We model the DNA reaction synthesis as follows:



where γ is the GC content of the cell, and $L_{\text{DNA}}^{\text{bp}}$ is the total length in base pairs of the DNA. As with mRNA_l and Enz_j, DNA has a mass-balance equation of the following shape:

$$\frac{d}{dt} [\text{DNA}] = 0 = v_{\text{DNA}}^{\text{synthesis}} - v_{\text{DNA}}^{\text{degradation}} - v_{\text{DNA}}^{\text{dilution}}, \quad (43)$$

$$v_{\text{DNA}}^{\text{synthesis}} - v_{\text{DNA}}^{\text{deg}} - \mu * \text{DNA} = 0. \quad (\text{DB}_{\text{DNA}})$$

We consider that the DNA does not degrade, meaning the only source of DNA consumption is dilution caused by the growth of the cell and $k_{deg}^{DNA} = 0$. We then define the molar weight of DNA MW_{DNA} and enforce the DNA mass ratio D^m as we did with both proteins and mRNA:

$$MW_{DNA} = (1 - \gamma)L_{DNA}^{bp}(MW_{dATP} + MW_{dTTP}), \quad (44)$$

$$+ \gamma L_{DNA}^{bp}(MW_{dGTP} + MW_{dCTP}),$$

$$MW_{DNA} \cdot DNA - \sum_{u \in \mathcal{U}} \lambda_u \cdot D_u^m = 0. \quad (IC3)$$

Scaling. A critical issue in the formulation of this problem is that the variables are different orders of magnitude. Fluxes are typically between $10^{-3} - 10^1$ mmol.gDW⁻¹.h⁻¹, whereas protein concentrations are around $10^{-6} - 10^{-3}$ mmol.gDW⁻¹ and mRNA concentrations are $10^{-10} - 10^{-6}$ mmol.gDW⁻¹. The relationship between these scales is given by the catalytic rate constant of enzymes and expression machinery, which spans from $10^3 - 10^6$ h⁻¹. In particular, the ribosome rate constant for translation (~ 12 aa.s⁻¹ = 43 200 aa.h⁻¹) as well as the RNA polymerase rate constant of transcription (~ 85 nt.s⁻¹ = 306 000 nt.h⁻¹) are responsible for strong differences in the concentrations and fluxes between transcription- and translation-related parts of the problem. Consequently, the constraint matrix becomes ill-conditioned, and the solver has to operate close to, or sometimes beyond, its maximal solving accuracy (usually around 10^{-9} for commercial solvers such as ILOG CPLEX or Gurobi).

To circumvent these limitations, we scale the EP, which will reduce the numerical difficulty of the problem, using nondimensionalization. We create nondimensionalized variables by dividing the variables of the initial problem by an estimated upper bound. For example, by definition, macromolecule concentrations cannot exceed 1 g.gDW⁻¹, and the following constrains the transformed macromolecule variables between 0 and 1:

$$\hat{X} = \frac{X}{\sigma_X}, \sigma_X \geq \sup(X) \Rightarrow 0 \leq \hat{X} \leq 1. \quad (45)$$

In this scheme, σ_X is an upper bound to X . In particular, if we consider σ_X to be the concentration of 1 g.gDW⁻¹:

$$\begin{aligned} \sigma_X &= 1 \text{ g.gDW}^{-1}, \\ &= 1 \text{ g.gDW}^{-1} \times \frac{1}{MW(X)} \text{ mmol.g}^{-1}, \\ &= \frac{1}{MW(X)} \text{ mmol.gDW}^{-1}, \end{aligned} \quad (46)$$

where $MW(X)$ denotes the molecular weight of the macromolecule in SI units (kg.mol⁻¹ \equiv g.mmol⁻¹), and \hat{X} represents the mass fraction of the molecule in the cell. We scale the fluxes using a method derived from this, detailed in the supporting file Supplementary Note 1. It is also possible to further refine this upper bound by performing a variation analysis on X and re-generating a model using the newly estimated upper bound.

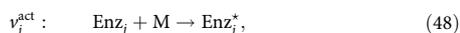
For the sake of clarity, all problem formulations will be kept in their dimensionalized form in the subsequent equations although the implementation is in fact nondimensionalized. The nondimensionalized problem is described further in Supplementary Note 1.

Advanced modeling. ETFL is amenable to modeling more intricate expression processes. A short selection of these is detailed below.

Enzyme-mediated complex assembly: By default, all the peptides are assumed to assemble spontaneously, without an enzyme. However, in the case of an enzyme-mediated assembly, it is possible to limit the assembly rate by a catalytic constraint if needed, in a fashion similar to Eq. (13). If we denote A the total concentration of assembling enzyme, and k_{asm}^A the catalytic rate constant of assembly, we can constraint v_j^{asm} the assembly rate of the j^{th} enzyme:

$$v_j^{asm} \leq k_{asm}^A \cdot A. \quad (47)$$

Enzyme activation and posttranslational modifications: Some enzymes require to be modified in order to be active, and sometimes by metabolites of the cell. This can be captured by adding a new species representing the active enzyme, and an activation reaction transforming the inactive enzyme to the active form. If the metabolite M is required to activate enzyme Enz_j into Enz_j^* , then the following activation reaction is added to the model:



The mass balances of Enz_j and M will be supplemented by a term $-v_j^{act}$, and the mass balance of Enz_j^* by $+v_j^{act}$. Finally, the catalytic constraint of the reaction v_j catalyzed by Enz_j^* at concentration E_j^* shall be:

$$v_j \leq k_{cat}^j \cdot E_j^* \quad (49)$$

This activation reaction can itself be catalytically limited if needed (see previous paragraph), and require the participation of metabolites. Thus, ETFL allows to capture protein-metabolite interactions.

Enzyme association: It is also possible to model the partition between free enzymes and associated enzymes. In that case, we simply need to operate the following adaptations: (i) replace the E_j term in any catalytic constraint by a new variable E_j^f , which represents the enzymes participating in the catalysis of the j^{th} reaction; (ii) add a variable E_j^f which represents the free enzymes of the system; and (iii) add the enzyme usage constraint:

$$E_j^f + E_j - E_j = 0 \quad (EU_j)$$

Dilution and degradation assumptions: In the current formulation, some species have their dilution or degradation neglected because of high reactivity or slow degradation rate constants. This can be relaxed by simply editing the mass balance reaction according to the assumption to be relaxed. In particular, enzyme-mediated degradation can be modeled by adding suitable catalytic constraints on the degradation reactions. In addition, the dilution term for metabolites can be taken into account if needed, in a manner similar to what Benyamini et al. describe in their method for FBA accounting for dilution⁴⁴.

MILP-based gene knockout strategies for strain design: The ETFL formulation of gene knockout using an upper bound on the translation rate allows to directly formulate MILP-based gene knockout strategies for strain design. Indeed, for each j^{th} gene, we can enforce the constraint:

$$v_j^{tsl} \leq M \cdot b_j, \quad (50)$$

with v_j^{tsl} the gene's transcription rate, b_j a binary variable and M a big-M constant. With that kind of constraint, if $b = 1$, the gene is active, while if $b = 0$, the gene is knocked-out. It is hence possible to formulate an objective function to optimize the number of KO while fulfilling a metabolic objective, for instance.

Thermodynamics-based constraints. Thermodynamics flux analysis (TFA)^{2,3} imposes constraints on a FBA problem to couple reaction directionality to the standard free energy of reactions and metabolite concentrations. We also introduce constraints that couple the sign of the Gibbs energy of a reaction to its directionality through the use of integer variables and a mixed-integer linear coupling formulation. This framework reduces the feasible flux space and improves the predictive power of FBA by removing thermodynamically invalid flux profiles.

Considering c_i is the concentration of i^{th} metabolite, we define C_i as its scaled logarithm with respect to c_0 so that in standard conditions $c_0 = 1$ M:

$$\forall i, \quad C_i = \ln\left(\frac{c_i}{c_0}\right). \quad (51)$$

We use the group contribution method⁴⁵ to directly calculate $\Delta_r G_j^o$, the Gibbs energy in solution of the j^{th} reaction. The calculated energy is the net change in the energies of formation of the compounds, which is simply the algebraic sum of the energies of bonds that are broken and formed. This allows to minimize the estimation error of $\Delta_r G_j^o$, as there is no error coming from the groups that do not react. Hence, we obtain the additional variables:

$$C_i^{\min} \leq C_i \leq C_i^{\max}, \quad (52)$$

$$\Delta_r G_{j,\min}^o \leq \Delta_r G_j^o \leq \Delta_r G_{j,\max}^o, \quad (53)$$

$$\Delta_r G_{j,\min}^o \leq \Delta_r G_j^o \leq \Delta_r G_{j,\max}^o. \quad (54)$$

Some metabolites are not fully characterized, e.g. metabolites with -R groups such as fatty acids, or metabolites attached to a Coenzyme A or acyl-carrier protein. In these cases, the group contribution method allows to directly calculate the net change in the standard Gibbs energy. Since these -R groups are often conserved in the reaction, their contribution terms cancel out when calculating the Gibbs energy of the reaction.

The concentration variables are bounded by experimental measurements or physiological assumptions, and the standard Gibbs energies are bounded by the measurement or estimation error. Since the net flux of each reaction has already been split between forward flux (v_j^+) and backward flux (v_j^-), (see Eq. (12)), we can directly add the constraints described in ref. 2:

$$\Delta_r G_j^o - RT \sum_{i=1}^m \eta_i^j C_i - \Delta_r G_i^o = 0, \quad (55)$$

$$\Delta_r G_j^o - K + K \cdot b_j^+ \leq 0, \quad (56)$$

$$-\Delta_r G_j^o - K + K \cdot b_j^- \leq 0, \quad (57)$$

$$v_j^+ - K \cdot b_j^+ \leq 0, \quad (58)$$

$$v_j^- - K \cdot b_j^- \leq 0, \quad (59)$$

$$b_j^+ + b_j^- \leq 1. \quad (60)$$

R denotes the ideal gas constant, T is the temperature in Kelvin, and η_i^j represents the stoichiometry of the metabolite i in the reaction j . K is a big-M constant (bigger than all upper bounds), and b_j^\pm are binary variables. Equation (55)

defines the actual Gibbs energy of the reaction as a function of its standard Gibbs energy and the scaled logarithms of metabolite concentrations. Equations (56) and (57) ensure that $\Delta_r G_j^* \leq 0 \iff b_i^+ = 1$ and $\Delta_r G_j^* \geq 0 \iff b_i^- = 1$. These binary variables are used to block flux in Eqs (58) (59) if the thermodynamics do not favor it. Finally, Eq. (60) is added to enforce that only one direction is chosen.

Data. mRNA degradation rates constants k_{deg} were taken from Bernstein et al.⁴⁶ We converted the reported half lives into rate constants using the classical relationship $k = \frac{\ln(2)}{t_{1/2}}$. Proteins were approximated to have a half life of 20 h (BNID 111930,⁴⁷).

Catalytic rate constants k_{cat} were obtained from Davidi et al.⁴⁸ for homomeric enzymes. Complex formation reactions for non-homomer enzymes were taken from the supplementary information of O'Brien et al.⁷ and Lloyd et al.⁸. EC numbers were obtained from BiGG⁴⁹ and the iJO1366 publication¹². Their corresponding k_{cat} values were assigned using conservative (max) values from SabioRK⁵⁰.

Homomer compositions were obtained from Davidi et al.⁴⁸. Other peptide compositions of enzymes were taken from the Supplementary Information of O'Brien et al.⁷ and Lloyd et al.⁸. Additional information was obtained from the Metacyc/Biocyc database^{29,51} using specialized SmartTables queries⁵².

Model modification. The initial model was subjected to minor changes to accommodate for ETFL modeling. In particular, we added:

Selenocysteine as a metabolite.

Cysteine to selenocysteine conversion as a pseudo reaction.

Replacements for the tRNA metabolites and their charging reaction, as dilution has to be considered.

We also modified the biomass reaction by removing its nucleotide and amino acid components, since they are already taken into account by the expression problem as explained in the section Biomass reaction synthesis and mass balance.

Enzyme estimation. Given a reaction in the model, if no enzyme is supplied but the reaction possesses a gene reaction rule, it is possible to infer an enzyme from it. The rule expression is expanded, and each term separated by an OR boolean operator is interpreted as an isozyme, while terms separated by an AND boolean operator are interpreted as unit peptide stoichiometric requirements. The enzyme is then assigned an average catalytic rate constant and degradation rate constant.

Essentiality analysis. The method for testing gene essentiality in FBA is to evaluate for each reaction the gene-protein-reaction association rules (GPRs) containing the gene of interest. The GPR is a boolean expression where the symbols represent whether a gene is expressed. OR operators represent isozymes, and AND operators the assembly of several peptides in a complex. To knock a gene out, its symbol in each GPR is simply assigned the value `False`. The GPR of all reactions is subsequently evaluated, and the reactions whose GPR evaluates to `False` are set to have a net flux of 0. Knocking a gene out in ETFL works differently: we replaces GPRs with mass balances, and the direct interaction between gene transcription, peptide translation, enzyme assembly, and metabolism. In this context, knocking-out a gene is done by forcing its transcription rate to 0. Indeed, gene-reactions relationships are conveyed directly through the direct contribution of the relevant peptides either as components of the enzyme complex (AND operator in GPRs) or as isozymes (OR operator). An advantage of this formulation is that it can be used in strain design strategies to optimize directly for knockouts in a single optimization problem.

If a knocked-out gene does not have enzyme associated with it (because of the lack of composition or k_{cat} information), there will be no catalytic constraint associated with the corresponding enzyme. The absence of catalytic constraint will prevent the reaction to be knocked-out. Hence, because of the missing information, gene essentiality information will be lost. An example is the essential reaction Sulfite reductase NADPH2 (SULR). iJO1366 provides a GPR describing a complex needing b2763 and b2764. The ETFL source (the cobraME model and BioCyc) could not provide the stoichiometry of the peptides to form the complex, and thus no enzyme is associated to this reaction in the vETFL model. iJO1366 correctly predicts the genes b2763 and 2764 as essential, but ETFL fails because these genes are not associated to any enzyme. As more enzyme data are added to the model, the false positive rate decreases, as we show in the section Essentiality analysis.

For increased performance, the essentiality analysis was cast into a feasibility problem. We put a lower bound on growth equal to 10% of the predicted ETFL growth and set the objective to 0. With this method, essential genes will cause the problem to be infeasible, while non-essential genes will return a feasible solution satisfying at least 10% of the growth. This method achieved up to a fivefold reduction in solving time on the most complex models.

Hardware. Computations were done on a 64-bit Ubuntu 18.04.1 LTS (Bionic Beaver); $2 \times$ Intel(R) Xeon(R) CPU E5-2667 v3 @ 3.20 GHz (8 cores, 16 threads per socket); 4×16 Go @ 2400 MHz RAM. Code was run on Python 3.6 on Docker (18.09.0) containers based on the official python 3.6-stretch container, available on ETFL GitHub and ETFL GitLab.

Reporting summary. Further information on research design is available in the Nature Research Reporting Summary linked to this article.

Data availability

All the data used to conduct this study are available in the `organism_data` subfolder of the repositories. Some of the data has been obtained from publications, for which all the references are provided in the main text, and a copy has been included in our repositories that mentioned above. The code also contains comments crediting the publications from which data sets and values have been obtained.

Code availability

The code has been implemented as a plug-in to pyTFA⁵³, a Python implementation of the TFA method. It uses COBRAPy⁵⁴ and Optlang⁵⁵ as a backend to ensure compatibility with several open source (GLPK, scipy, ...) as well as commercial (CPLEX, Gurobi, ...) solvers. We rely on the Python package Biopython⁵⁶ for transcribing and translating sequences of nucleotides and amino acids. The code used to generate the models is freely available under the APACHE 2.0 license at <https://github.com/EPFL-LCSB/etfl> and <https://gitlab.com/EPFL-LCSB/etfl>.

Received: 30 April 2019; Accepted: 28 November 2019;

Published online: 13 January 2020

References

- Magnúsdóttir, S. et al. Generation of genome-scale metabolic reconstructions for 773 members of the human gut microbiota. *Nat. Biotechnol.* **35**, 81 (2017).
- Henry, C. S., Broadbelt, L. J. & Hatzimanikatis, V. Thermodynamics-based metabolic flux analysis. *Biophysical J.* **92**, 1792–1805 (2007).
- Soh, K. C. & Hatzimanikatis, V. Constraining the flux space using thermodynamics and integration of metabolomics data. *Methods Mol. Biol.* **1191**, 49–63 (2014).
- Beg, Q. K. et al. Intracellular crowding defines the mode and sequence of substrate uptake by *Escherichia coli* and constrains its metabolic activity. *Proc. Natl Acad. Sci. USA* **104**, 12663–12668 (2007).
- Sanchez, B. J. et al. Improving the phenotype predictions of a yeast genome-scale metabolic model by incorporating enzymatic constraints. *Mol. Syst. Biol.* **13**, 935 (2017).
- Lerman, J. A. et al. In silico method for modelling metabolism and gene product expression at genome scale. *Nat. Commun.* **3**, 929 (2012).
- O'Brien, E. J., Lerman, J. A., Chang, R. L., Hyduke, D. R. & Palsson, B. O. Genome-scale models of metabolism and gene expression extend and refine growth phenotype prediction. *Mol. Syst. Biol.* **9**, 693 (2013).
- Lloyd, C. J. et al. Cobrame: A computational framework for genome-scale models of metabolism and gene expression. *PLoS Computational Biol.* **14**, e1006302 (2018).
- Yang, L. et al. solveme: fast and reliable solution of nonlinear me models. *BMC Bioinforma.* **17**, 391 (2016).
- Ma, D. et al. Reliable and efficient solution of genome-scale models of metabolism and macromolecular expression. *Sci. Rep.* **7**, 40863 (2017).
- Neidhardt, F. C. & Curtiss, R. *Escherichia Coli and Salmonella: Cellular and Molecular Biology* Vol. 2 (ASM Press, Washington, DC, 1999).
- Orth, J. D. et al. A comprehensive genome-scale reconstruction of *Escherichia coli* metabolism-2011. *Mol. Syst. Biol.* **7**, 535 (2011).
- McCloskey, D. et al. A model-driven quantitative metabolomics analysis of aerobic and anaerobic metabolism in *E. coli* K-12 mg1655 that is biochemically and thermodynamically consistent. *Biotechnol. Bioeng.* **111**, 803–815 (2014).
- Liebermeister, W. et al. Visual account of protein investment in cellular functions. *Proc. Natl Acad. Sci. USA* **111**, 8488–8493 (2014).
- Otto, A. et al. Systems-wide temporal proteomic profiling in glucose-starved *Bacillus subtilis*. *Nat. Commun.* **1**, 137 (2010).
- Schellenberger, J. & Palsson, B. O. Use of randomized sampling for analysis of metabolic networks. *J. Biol. Chem.* **284**, 5457–5461 (2009).
- Schellenberger, J. et al. Quantitative prediction of cellular metabolism with constraint-based models: the cobra toolbox v2.0. *Nat. Protoc.* **6**, 1290–1307 (2011).
- Meghelenbrink, W., Huynen, M. & Marchiori, E. optgmsampler: an improved tool for uniformly sampling the solution-space of genome-scale metabolic networks. *PLoS ONE* **9**, e86587 (2014).
- Lee, J., Lam, W. & Dechter, R. Benchmark on daoopt and gurobi with the pascal2 inference challenge problems. (2013). <https://www.ics.uci.edu/~dechter/publications/r202.pdf>.
- Lodi, A. & Tramontani, A. Performance variability in mixed-integer programming. In *Theory Driven by Influential Applications*, 1–12 (INFORMS, 2013). <https://pubsonline.informs.org/doi/pdf/10.1287/educ.2013.0112>.

21. CPLEX, I. I. I. High-performance mathematical programming engine. *Int. Business Machines Corp.* (2010). <http://www.ibm.com/software/integration/optimization/cplex>.
22. Gu, Z., Rothberg, E. & Bixby, R. *Gurobi Optimizer Reference Manual, Version 8.0* (Gurobi Optimization Inc., Houston, 2018).
23. Pandey, V., Hadadi, N. & Hatzimanikatis, V. Enhanced flux prediction by integrating relative expression and relative metabolite abundance into thermodynamically consistent metabolic models. *PLoS Computational Biol.* **15**, e1007036 (2019).
24. Zur, H., Rupp, E. & Shlomi, T. imat: an integrative metabolic analysis tool. *Bioinformatics* **26**, 3140–3142 (2010).
25. Becker, S. A. & Palsson, B. O. Context-specific metabolic networks are consistent with experiments. *PLoS Computational Biol.* **4**, e1000082 (2008).
26. Pandey, V. & Hatzimanikatis, V. Investigating the deregulation of metabolic tasks via Minimum Network Enrichment Analysis (MiNEA) as applied to nonalcoholic fatty liver disease using mouse and human omics data. *PLoS Computational Biol.* **15**, e1006760 (2019).
27. Segre, D., Vitkup, D. & Church, G. M. Analysis of optimality in natural and perturbed metabolic networks. *Proc. Natl Acad. Sci. USA* **99**, 15112–15117 (2002).
28. Mahadevan, R., Edwards, J. S. & Doyle, F. J. Dynamic flux balance analysis of diauxic growth in *Escherichia coli*. *Biophys. J.* **83**, 1331–1340 (2002).
29. Caspi, R. et al. The metacyc database of metabolic pathways and enzymes and the biocyc collection of pathway/genome databases. *Nucleic Acids Res.* **36**, D623–D631 (2007).
30. Arkin, A. P. et al. Kbase: the united states department of energy systems biology knowledgebase. *Nat. Biotechnol.* **36**, 566–569 (2018).
31. Fredrickson, A. Formulation of structured growth models. *Biotechnol. Bioeng.* **18**, 1481–1486 (1976).
32. Milo, R., Jorgensen, P., Moran, U., Weber, G. & Springer, M. Bionumbers—the database of key numbers in molecular and cell biology. *Nucleic Acids Res.* **38**, D750–3 (2010).
33. Bremer, H. & Dennis, P. P. Modulation of chemical composition and other parameters of the cell by growth rate. *Escherichia coli Salmonella: Cell. Mol. Biol.* **2**, 1553–1569 (1996).
34. Schuwirth, B. S. et al. Structures of the bacterial ribosome at 3.5 Å resolution. *Science* **310**, 827–834 (2005).
35. Zhu, J., Penczek, P. A., Schroder, R. & Frank, J. Three-dimensional reconstruction with contrast transfer function correction from energy-filtered cryoelectron micrographs: procedure and application to the 70S *Escherichia coli* ribosome. *J. Struct. Biol.* **118**, 197–219 (1997).
36. Gilbert, R. Physical biology of the cell, by Rob Phillips, Jane Kondev and Julie Theriot, *Crystallography Reviews* **15**, 285–288 (2009).
37. Neidhardt, F.C., 1964. The regulation of RNA synthesis in bacteria. In *Progress in nucleic acid research and molecular biology* (Vol. 3, pp. 145–181). Academic Press.
38. Chan, S. H. J., Simons, M. N. & Maranas, C. D. Steadycom: predicting microbial abundances while ensuring community stability. *PLoS Computational Biol.* **13**, e1005539 (2017).
39. Petersen, C. C. *A Note on Transforming the Product of Variables to Linear Form in Linear Programs* (Diskussionspapier, Purdue University, 1971).
40. Glover, F. Improved linear integer programming formulations of nonlinear integer problems. *Manag. Sci.* **22**, 455–460 (1975).
41. Hatzimanikatis, V., Floudas, C. A. & Bailey, J. E. Analysis and design of metabolic reaction networks via mixed-integer linear optimization. *AIChE J.* **42**, 1277–1292 (1996).
42. Hatzimanikatis, V., Floudas, C. A. & Bailey, J. E. Optimization of regulatory architectures in metabolic reaction networks. *Biotechnol. Bioeng.* **52**, 485–500 (1996).
43. Pramanik, J. & Keasling, J. Stoichiometric model of *Escherichia coli* metabolism: incorporation of growth-rate dependent biomass composition and mechanistic energy requirements. *Biotechnol. Bioeng.* **56**, 398–421 (1997).
44. Benyamini, T., Folger, O., Rupp, E. & Shlomi, T. Flux balance analysis accounting for metabolite dilution. *Genome Biol.* **11**, R43 (2010).
45. Jankowski, M. D., Henry, C. S., Broadbelt, L. J. & Hatzimanikatis, V. Group contribution method for thermodynamic analysis of complex metabolic networks. *Biophysical J.* **95**, 1487–1499 (2008).
46. Bernstein, J. A., Khodursky, A. B., Lin, P. H., Lin-Chao, S. & Cohen, S. N. Global analysis of mRNA decay and abundance in *Escherichia coli* at single-gene resolution using two-color fluorescent DNA microarrays. *Proc. Natl Acad. Sci. USA* **99**, 9697–9702 (2002).
47. Moran, M. A. et al. Sizing up metatranscriptomics. *ISME J.* **7**, 237 (2013).
48. Davidi, D. et al. Global characterization of in vivo enzyme catalytic rates and their correspondence to in vitro k_{cat} measurements. *Proc. Natl Acad. Sci. USA* **113**, 3401–3406 (2016).
49. King, Z. A. et al. Escher: a web application for building, sharing, and embedding data-rich visualizations of biological pathways. *PLoS Computational Biol.* **11**, e1004321 (2015).
50. Wittig, U. et al. SABIO-RK—database for biochemical reaction kinetics. *Nucleic Acids Res.* **40**, D790–D796 (2011).
51. Keseler, I. M. et al. Ecocyc: a comprehensive database resource for *Escherichia coli*. *Nucleic Acids Res.* **33**, D334–D337 (2005).
52. Travers, M., Paley, S. M., Shrager, J., Holland, T. A. & Karp, P. D. Groups: knowledge spreadsheets for symbolic biocomputing. *Database* **2013**, bat061 (2013).
53. Salvy, P., Fengos, G., Ataman, M., Pathier, T., Soh, K. C. & Hatzimanikatis, V. pyTFA and matTFA: a Python package and a Matlab toolbox for Thermodynamics-based Flux Analysis. *Bioinformatics* **35**, 167–169 (2018).
54. Ebrahim, A., Lerman, J. A., Palsson, B. O. & Hyduke, D. R. Cobrapy: constraints-based reconstruction and analysis for python. *BMC Syst. Biol.* **7**, 74 (2013).
55. Jensen, K., Cardoso, J. & Sonnenschein, N. Optlang: An algebraic modeling language for mathematical optimization. The Journal of Open Source Software, **2**, 139. <https://doi.org/10.21105/joss.00139> (2016).
56. Dalke, A. et al. Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics* **25**, 1422–1423 (2009).

Acknowledgements

The authors would like to thank Prof. Jens Nielsen and Dr. Ibrahim El-Semman for the valuable discussions about the formulation; and Dr. Kaycie Butler for her valuable input on the wording and structure of this paper. This work has received funding from the European Union's Horizon 2020 Research and Innovation Programme under the Marie Skłodowska-Curie grant agreement No. 722287, the European Union's Horizon 2020 Research and Innovation Programme under grant agreement No. 686070, and the Swiss National Science Fund (SNSF) under the grant agreement No. 200021_188623.

Author contributions

P.S. and V.H. designed the formulation and the studies. P.S. wrote the ETFL code, curated the models, and performed the studies. P.S. and V.H. wrote the paper.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information is available for this paper at <https://doi.org/10.1038/s41467-019-13818-7>.

Correspondence and requests for materials should be addressed to V.H.

Peer review information *Nature Communications* thanks Ali Zomorodi and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permission information is available at <http://www.nature.com/reprints>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2020