Supplementary text

S1 MatLab functions and description

The functions used for this study can be reused as a guideline to link other dynamic models to genome scale reconstructions. Some modifications might be necessary depending on the models used.

Software requirements

The example code requires Matlab (implemented on version R2014b), the COBRA toolbox[1] (opencobra.sourceforge.net) as well as the SBML toolbox[2] (sbml.org/Software/SBMLToolbox).

Function overview

`bktr_AddLeakRxn`	Function to add a leak reaction with the calculated stoichiometry as well as potentially changes stoichiometry of lumped reactions to the dynamic model.
`bktr_addY7LumpedBiomassComponents`	Function to add the lumped biomass precursor pools AA Pool, DNA Desoxyribonucleotide Pool, RNA Ribonucleotide Pool, and Lipid pool to the Y7 model.
`bktr_barPlot`	Plotting function for leak stoichiometries.
`bktr_convertIrrevFluxDistributionFVABounds`	Function to correctly convert FVA bound of an irreversible model back to the bound of the respective reversible model.
`bktr_DistributionFigure`	Plotting function for the leaks distribution if several targets are considered.
`bktr_errorbarPlot`	Plotting function for leak stoichiometries with FVA bounds.
`bktr_Export`	Export function of the amended dynamic model to SBML.
`bktr_LeakFVA`	Flux variability analysis of the leak fluxes.
`bktr_mapping`	Function to map the dynamic model to the genome scale model based on unique identifiers (requirements see below).
`bktr_nAtoms`	Function to extract the number of atoms of a specific kind from the metabolite formula
`bktr_prep_Lump`	Function to produce an irreversible model with amended reversibilities to be used in the bktr_Run function.
`bktr_RefillChebiAndKeggIds`	Function to extract ChEBI and KEGG IDs and metabolite formulas from the imported SBML model (generated with the SBML toolbox for Matlab)
`bktr_Run`	Function to calculate the leaks from the dynamic model.
`bktr_Target`	Function to fix the flux through the selected target metabolite in the genome scale model.

S2 Model requirements

Requirements for dynamic model:

1) Annotations
   Meatbolites: KEGG, ChEBI, Formula
   Reactions: KEGG, genes, EC number

2) Remove all reactions that are

   - secretion reactions
   - biomass drains
   - dummy storage compound production
   - dummy ATP consumers

3) Remove all metabolites that are

   - dummy storage compounds
   - dummy biomass compounds
   - external metabolites

4) Lumped reactions should include annotations for all reactions that are part of the reaction step.

5) Lumped species should be annotated with IDs for all pooled metabolites.

Requirements for GSModel:

1) Set uptake restrictions as desired for dynamic model

2) Set all storage/biomass compounds/external metabolites as target metabolites

Possible Problems during a backtracking run

Biomass backtracking requires in depth knowledge of the used GSM, to judge the biological plausibility of the found flux distributions. Especially ***uptake and secretion rates*** should be checked carefully: They should represent typical cultivation media for the respective species or modeling question and not show unphysiolgically high secretion rates of metabolic intermediates. The flux minimization objective does not account for loss of energy (c.f. ATP yield maximization) such that secretion of high energy compounds is not punished. We especially observed this for the balancing of the single biomass compound as in those simulations the GSM is much less restricted than with the total biomass objective. These effects could be circumvented by changing the objective function, e.g. to also account for ATP turnover, or to restrict critical secretion rates.

The accuracy of the calculated leak flux depends to a large extend on the quality of the available genome scale model. Often the mere flux minimization yields unphysiologial flux distributions as it

does not consider thermodynamic or energetic constraints. The *B. subtilis* model from Ex4 for example secrets almost half of the acquired glucose (11.63 mol) as F6P (5.54 mol, c.f. Figure 5, main text), which is very inefficient from an energetic point of view. Similar observations could be made for the Y7 model that uses 3PG, for which a large flux was found in the backtracking, in a futile cycle to oxidize NADH via 2-oxoglutarate and L-glutamate.

Ideally, the genome scale model should be ***constrained by experimental data*** (uptake rates or flux measurements) or thermodynamic feasibility to avoid such unphysisological behavior as far as possible.

S3 Summary of the example codes, models & descriptions

Including the following scripts:

Ex1_SimpleBacktracking (c.f. Figure 2A, main text)

Ex2_BacktrackingBiomassComponents (c.f. Figure 2B-D, main text)

Ex3_MoreExampleModels (c.f. Figure 3A, main text and Supplementary Text S5)

Ex4_OxygenAvailability (c.f. Figure 4, main text)

Ex5_DifferentSpeciesGlycolysisBacktracking (c.f. Figure 5, main text)

Ex6_ModelSplitting (c.f. Figure 6, main text)

Including the following genome scale models:

Yeast7.11 [3]	http://sourceforge.net/projects/yeast/files/
iJO1366 [4]	http://msb.embopress.org/content/7/1/535.long#sec-17 (Supplementary file 3)
iNJ661 [5]	http://systemsbiology.ucsd.edu/InSilicoOrganisms/MTB
iBsu1103 [6]	http://genomebiology.com/2009/10/6/R69/additional (data file 4)

The models are provided as SBML files as downloaded from the respective sources and saved as .mat files after the translation and annotation process as described in the function `bktr_generate_GSMModels`.

Including the following topology of the (hypothetical) dynamic models:

```
Hynne_model.mat
Glycolysis_model.mat
Glycolysis_model_ext.mat
```

The 3 models can also be newly generated by the function `bktr_generate_dyn_models`.

The mapping of the dynamic models to the GSM can be carried out using the function `bktr_mapping`. For comparison or immediate use the resulting map structures are stored in the file `maps.mat` generated with the function `bktr_generate_example_maps`.

As overview we provide information about the mapping, models, media constraints in spreadsheet format in SupplementaryTable1.xlsx

S4 Details for the example using the Hynne model

Mapping

In our backtracking workflow, we used the mapping function to identify the corresponding reactions of the Yeast7 model. The Hynne model is well annotated with ChEBI and KEGG compound identifiers and EC-numbers, KEGG reaction and Reactome[7] identifiers for reactions, such that all matches could be easily identified (c.f. Supplementary Table 1). All extracellular compounds, secretion reactions, the storage compound and its producing reaction were removed. We also excluded all reactions modeling the influence of cyanide since the focus of this exemplary study lies on the unperturbed cell state. The uptake rates of the Yeast7 model were set to represent minimal medium with glucose as sole carbon source.

Splitting of the biomass

The Yeast7 biomass was split into its main constituents and each of the resulting species was individually backtracked as described for the whole biomass, using the same mapping but different target compounds. The splitting was based on experimental data from Schulze et al. [8].

Some of the measured compounds were directly available as metabolites in the Yeast7 model (glucan, glycogen, trehalose, mannan). For the remaining entities (lipid, amino acids, DNA, RNA), pooled species were generated based on the stoichiometry in the reaction producing the biomass component in Yeast7 (function bktr_addY7LumpedBiomassComponents.m). The amino acid pool is build up by a mixture of all amino acids with ratios based on their abundance in the proteome. By generating pools of monomers - amino acids, ribonucleotides and desoxy-ribonucleotides, it is possible to discriminate the mass and energy that is needed for their synthesis from the polymerization costs. This is especially helpful if it is intended to model the subsequent transcription and translation in more detail.

For the optimization of the single biomass compounds, the Yeast7 model is overall less constraint, so that slightly different solutions of the overall flux minimization can occur. For example, we now see a leak flux from glycerol to form a part of the amino acid pool, whereas in the backtracking of the complete biomass no leak from this compound could be observed. Hence, the results of the backtracking should be carefully checked for their suitability for the system in question and it might be necessary to add additional constraints on the applied GSM.

S5 Details for the example the flux calculations in Figure 2 of the main text and biomodels statistics

In addition to the Hynne model, we choose three further dynamic metabolic models and compared their steady state boundary fluxes to the results of the backtracking. We tried to choose detailed models that already had a more detailed boundary description. We choose one additional model of yeast glycolysis and two for E.coli:

BioModels ID	Name	Species	Reference (PMID)	
BIOMD0000000051	Chassagnole2002_Carbon_Metabolism	Ecoli	17590932	[9]
BIOMD0000000221	Singh2006_TCA_Ecoli_acetate	Ecoli	16887020	[10]
MODEL1403250001	vanEunen2012 - Yeast Glycolysis (glucose upshift)	Yeast	22570597	[11]
BIOMD0000000061	Hynne2001_Glycolysis	Yeast	11744196	[12]

Each of the models was downloaded from the Biomodels database and simulated in Copasi [13] until the steady state was reached. For the oscillation Hynne model, the simulation was run until stable oscillations were observed and the steady state boundary fluxes were calculated as the mean of the maximum and minimum of the oscillatory flux.
For the calculation of the leak fluxes with the backtracking workflow, the downloaded models were imported to Matlab and mapped as described in the Online Methods. For the yeast models Yeast7[3] was again used as corresponding genome scale model, for the E.coli models we used iJO1366[14]. The uptake and growth rates were set to correspond to the simulations in the original models. For a detailed summary please also refer to Supplementary Table 2.

We also categorized all metabolic models that were available in the Biomodels database according to how the problem of defining the systems boundary fluxes was solved. The possible categories were either no biomass or other leak reactions at all included in the model (0); no biomass included, but excretion reactions present (1), drain of some intermediates towards the generation of biomass with fixed (mostly arbitrary) stoichiometry (2) and drain of some intermediates with fixed but context dependant stoichiometry (3). The results are also summarized in Supplementary Table 2.

S6 Dynamic simulations with the calculated biomass leak reactions

To test if the addition of a detailed and more precise biomass reaction also improves the dynamic model behavior, we tested the following model against its original data :

BioModels ID	Name	Species	Reference (PMID)	
MODEL1403250001	vanEunen2012 - Yeast Glycolysis (glucose upshift)	Yeast	22570597	[11]

We removed reactions that account for biomass production in the original model:

Tre1, Tre2, Gly, Suc, Ace, Adh

The biomass reaction based on the backtracking was implemented using the following stoichiometry:

0.046 * G6P + 0.254 * TRIO + 2.283 * NADH + 0.956 * P3G + 2.283 * AcAld ↔ 0.1525 * F6P + 2.283 * NAD + 0.956 * PYR

To generate dynamic simulations, we constructed two different artificial biomass production kinetics that scale the growth rate (and the associated effluxes from the model) with substrate availability relative to the initial values:

1) $$k_{growth} = k_{growth,0} \cdot \frac{TRIO(t)}{TRIO(0)} \cdot \frac{G6P(t)}{G6P(0)} \cdot \frac{NADH(t)}{NADH(0)} \cdot \frac{P3G(t)}{P3G(0)} \cdot \frac{AcAld(t)}{AcAld(0)}$$

2) $$k_{growth} = k_{growth,0} \cdot \left(1 + \frac{TRIO(t)-TRIO(0)}{TRIO(0)}\right) \cdot \left(1 + \frac{G6P(t)-G6P(0)}{G6P(0)}\right) \cdot \left(1 + \frac{NADH(t)-NADH(0)}{NADH(0)}\right) \cdot \left(1 + \frac{P3G(t)-P3G(0)}{P3G(0)}\right) \cdot \left(1 + \frac{AcAld(t)-AcAld(0)}{AcAld(0)}\right)$$

Next, we extracted the experimental data given in Figure 4 of van Eunen et al.[11] and fitted the 3 model versions (*i*: without backtracking, *ii*: with backtracking – growth kinetics 1, *iii*: with backtracking – growth kinetics 2) with identical settings with three different methods using Copasi[13].

The parameters fitted and their boundaries were as follows:

Parameter Name	lower bound	original value	upper bound
K_ace	0.25	0.694405	1
K_gly	8	11.68832	35
K_suc	0.45	1.8	1.8
kgrowth0	1.00E-06	0.01094	1.00E+06
Vmax_adh_r	450	1327.833	1800

Vmax_ald	150	331.5573	620
Vmax_eno	175	715	715
Vmax_gapdh_f	700	1484.85	3000
Vmax_gapdh_r	426	734.883	1705
Vmax_glt	80	640	640
Vmax_gpm	215	582.691	1700
Vmax_hk	100	136.864	425
Vmax_pdc	200	800	800
Vmax_pfk	50	97.44973	400
Vmax_pgi	150	150	1500
Vmax_pgk_r	1250	1250	5024
Vmax_pyk	210	1640	1640
ADP	0.75	0.75	1.25
AMP	0.2	0.314287	0.4
ATP	1.2	1.401208	1.8
F26P	0.007	0.014	0.028
T6P	0.1	0.2	0.4
ETOH	12	25	50

Additionally, to prevent accumulation of metabolites during simulation, we imposed upper boundaries for the measured metabolites:

Model species	**upper bound**
F16P	20
F6P	4
G6P	50
Glci	24
P2G + P3G	8

All 3 models were fitted to the experimental data using the following methods in Copasi:

1. Scatter Search, 200 iterations
2. Evolutionary Programming, 500 Generations, Population Size 50
3. Simulated Annealing, Start Temperature 1, Cooling Factor 0.85, Tolerance 1e-6

Fit results are reported in the table in Supplementary Figure 1A. On average, the objective value of the fits of backtracked models is about 5% better (about 45 versus about 47) than that of non-backtracked models. Because most parameter estimation algorithms apply heuristic methods, the average best fits are usually not the optimal ones. For backtracked models, evolutionary programming found a second parameter set that is more than 15% better than the best objective value of non-backtracked models found with simulated annealing (compare to full distributions in Supplementary Figure 1B).

The two different modes of scaling k_{growth} do, in this case, not affect the fitting results. We note however, that such scaling factors can greatly influence the dynamics of models and need to be carefully tuned to the problem at hand. The two methods employed here are simple but effective examples.

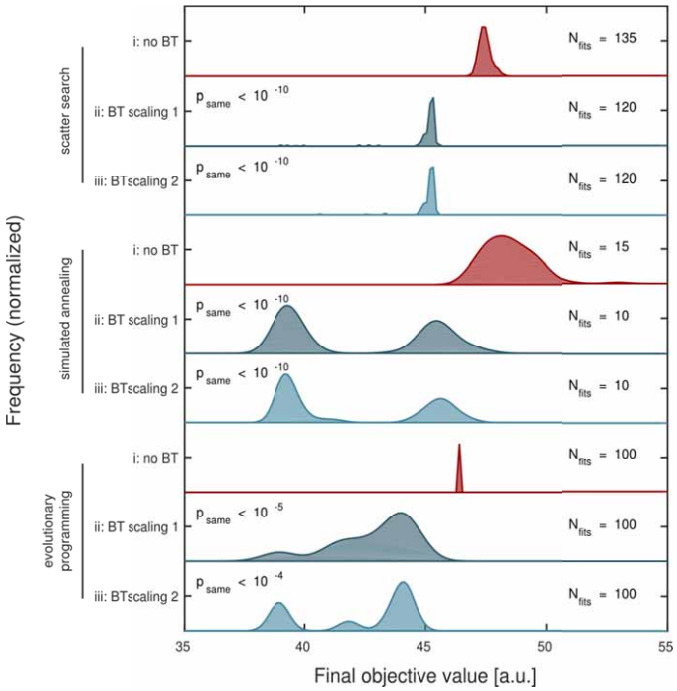
A

	scatter search			
	min	**m**	**σ**	**N_{fits}**
i: no BT	46.89	47.44	0.2	135
ii: BT scaling1	40.59	45.25	0.5	120
iii: BT scaling2	39.00	45.24	1.1	120

	simulated annealing			
	min	**m**	**σ**	**N_{fits}**
i: no BT	46.40	46.41	0.0	15
ii: BT scaling1	38.93	43.95	2.4	10
iii: BT scaling2	38.93	43.51	1.8	10

	evolutionary programming			
	min	**m**	**σ**	**N_{fits}**
i: no BT	46.65	48.36	1.0	100
ii: BT scaling1	39.00	39.68	3.1	100
iii: BT scaling2	38.98	39.96	3.2	100

B



Supplementary Figure 1: Fitting results with and without backtracking. **A** Summary of the fitting results for the 3 model topologies and the 3 different fitting algorithms; min – minimum, m – median, σ – standard deviation of all found final objective values, N_{fits} – Number of fits run for each method. **B** Kernel denisty plots for the full distributions of the found final objective values, normalized to the maximum density value. P-values refer to a Wilcoxon ranksum test ("Do the two samples originate from distributions with equal medians?") of the found results with backtracking vs. the original model for each fitting algorithm tested.

Supplementary References

1. Becker, S. a *et al.* Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox. *Nat. Protoc.* **2,** 727–38 (2007).

2. Keating, S. M., Bornstein, B. J., Finney, A. & Hucka, M. SBMLToolbox: an SBML toolbox for MATLAB users. *Bioinformatics* **22,** 1275–7 (2006).

3. Aung, H. W., Henry, S. a & Walker, L. P. Revising the Representation of Fatty Acid, Glycerolipid, and Glycerophospholipid Metabolism in the Consensus Model of Yeast Metabolism. *Ind. Biotechnol. (New Rochelle. N. Y).* **9,** 215–228 (2013).

4. Archer, C. T. *et al.* The genome sequence of E. coli W (ATCC 9637): comparative genome analysis and an improved genome-scale reconstruction of E. coli. *BMC Genomics* **12,** 9 (2011).

5. Jamshidi, N. & Palsson, B. Ø. Investigating the metabolic capabilities of Mycobacterium tuberculosis H37Rv using the in silico strain iNJ661 and proposing alternative drug targets. *BMC Syst. Biol.* **1,** 26 (2007).

6. Henry, C. S., Zinner, J. F., Cohoon, M. P. & Stevens, R. L. iBsu1103: a new genome-scale metabolic model of Bacillus subtilis based on SEED annotations. *Genome Biol.* **10,** R69 (2009).

7. Croft, D. *et al.* The Reactome pathway knowledgebase. *Nucleic Acids Res.* **42,** D472–7 (2014).

8. Schulze, U. Anaerobic Physiology of Saccharomyces Cerevisiae. (1995).

9. Chassagnole, C., Noisommit-Rizzi, N., Schmid, J. W., Mauch, K. & Reuss, M. Dynamic modeling of the central carbon metabolism of Escherichia coli. *Biotechnol. Bioeng.* **79,** 53–73 (2002).

10. Singh, V. K. & Ghosh, I. Kinetic modeling of tricarboxylic acid cycle and glyoxylate bypass in Mycobacterium tuberculosis, and its application to assessment of drug targets. *Theor. Biol. Med. Model.* **3,** 27 (2006).

11. Van Eunen, K., Kiewiet, J. a L., Westerhoff, H. V. & Bakker, B. M. Testing biochemistry revisited: How in vivo metabolism can be understood from in vitro enzyme kinetics. *PLoS Comput. Biol.* **8,** (2012).

12. Hynne, F., Danø, S. & Sørensen, P. G. Full-scale model of glycolysis in Saccharomyces cerevisiae. *Biophys. Chem.* **94,** 121–163 (2001).

13. Hoops, S. *et al.* COPASI--a COmplex PAthway SImulator. *Bioinformatics* **22,** 3067–74 (2006).

14. Orth, J. D. *et al.* A comprehensive genome-scale reconstruction of Escherichia coli metabolism--2011. *Mol. Syst. Biol.* **7,** 535 (2011).