BMC Systems Biology

Research article

Modeling metabolic networks in *C. glutamicum*: a comparison of rate laws in combination with various parameter optimization strategies

Andreas Dräger*[1], Marcel Kronfeld*[1], Michael J Ziller*[1], Jochen Supper[1], Hannes Planatscher[1], Jørgen B Magnus[2,3], Marco Oldiges[3], Oliver Kohlbacher[1] and Andreas Zell[1]

Address: [1]Center for Bioinformatics Tübingen (ZBIT), Wilhelm-Schickard-Institut für Informatik, Sand 1, 72076 Tübingen, Germany, [2]NNE Pharmaplan, Siemensstraße 21, 61352 Bad Homburg, Germany and [3]Forschungszentrum Jülich, Institut für Biotechnologie 2, 52425 Jülich, Germany

E-mail: Andreas Dräger* - andreas.draeger@uni-tuebingen.de; Marcel Kronfeld* - marcel.kronfeld@uni-tuebingen.de; Michael J Ziller* - michael.ziller@student.uni-tuebingen.de; Jochen Supper - jochen.supper@uni-tuebingen.de; Hannes Planatscher - hannes.planatscher@uni-tuebingen.de; Jørgen B Magnus - jmgs@nnepharmaplan.com; Marco Oldiges - m.oldiges@fz-juelich.de; Oliver Kohlbacher - oliver.kohlbacher@uni-tuebingen.de; Andreas Zell - andreas.zell@uni-tuebingen.de
*Corresponding author

Abstract

Background: To understand the dynamic behavior of cellular systems, mathematical modeling is often necessary and comprises three steps: (1) experimental measurement of participating molecules, (2) assignment of rate laws to each reaction, and (3) parameter calibration with respect to the measurements. In each of these steps the modeler is confronted with a plethora of alternative approaches, e. g., the selection of approximative rate laws in step two as specific equations are often unknown, or the choice of an estimation procedure with its specific settings in step three. This overall process with its numerous choices and the mutual influence between them makes it hard to single out the best modeling approach for a given problem.

Results: We investigate the modeling process using multiple kinetic equations together with various parameter optimization methods for a well-characterized example network, the biosynthesis of valine and leucine in *C. glutamicum*. For this purpose, we derive seven dynamic models based on generalized mass action, Michaelis-Menten and convenience kinetics as well as the stochastic Langevin equation. In addition, we introduce two modeling approaches for feedback inhibition to the mass action kinetics. The parameters of each model are estimated using eight optimization strategies. To determine the most promising modeling approaches together with the best optimization algorithms, we carry out a two-step benchmark: (1) coarse-grained comparison of the algorithms on all models and (2) fine-grained tuning of the best optimization algorithms and models. To analyze the space of the best parameters found for each model, we apply clustering, variance, and correlation analysis.

Conclusion: A mixed model based on the convenience rate law and the Michaelis-Menten equation, in which all reactions are assumed to be reversible, is the most suitable deterministic modeling approach followed by a reversible generalized mass action kinetics model. A Langevin

model is advisable to take stochastic effects into account. To estimate the model parameters, three algorithms are particularly useful: For first attempts the settings-free Tribes algorithm yields valuable results. Particle swarm optimization and differential evolution provide significantly better results with appropriate settings.

Background

The metabolism of whole cells can be described as a network of metabolites and reactions interconverting these metabolites. To understand cellular systems, dynamic modeling of cellular processes has become an important task in systems biology [1-4]. Dynamic models describe the whole system and the state of each reacting species therein in a time-dependent manner [2]. Once such a model is constructed, several network properties can be derived, for instance stability, robustness, or the long-term behavior [5]. Furthermore, a well developed model provides a basis for predictions under different perturbations or varied environmental circumstances and can be applied to enhance the yield of desired metabolic products like certain amino acids [3].

To set up such models, appropriate rate laws have to be assigned to each reaction within the network. From these, a differential equation system that characterizes the rates of change of each reactant can be derived. However, setting up model equations is a difficult task. For many larger networks available in databases like KEGG[6, 7] or METACYC[8] the reaction mechanism remains unknown. In many cases, reliable rate equations for the reactions are not known because these actually have to be derived for each catalyzing enzyme individually [9]. It is therefore a common approach to apply approximative rate laws, which characterize the most important features of the reaction rate. Many rate laws, which are either continuous or discrete, and either deterministic or stochastic [2], have been proposed for this purpose. Several examples of each group exist such as probabilistic [10, 11], phenomenological [5, 12-15], or semi-mechanistic approaches [5, 16].

A second problem arises whenever a dynamic model of biochemical systems is created, because any such model contains a certain number of parameters like the reaction rates, Michaelis constants or the limiting rate as well as constants describing the influence of certain inhibitors [17-19] or, in stochastic systems, the reaction propensity [20]. Except for phenomenological models like power law approximations [21], linlog [12, 22], or loglin kinetics [13, 23], the parameter values can often be measured. However, this procedure is time-consuming, expensive, and often impractical. Online databases like

the Brunswick Enzyme Database (BRENDA) [24-26] provide measured parameter values for many enzymes, but variations in the experimental settings, under which these and the time series measurements for the system under study were obtained, limit the applicability of these values for modeling purposes. In addition, it was observed that there are differences between parameters measured *in vivo* and *in vitro* [[27], p. 461]. The application of computational methods to optimize model parameters regarding the fit error has therefore become an important task in the model identification process [28-31]. In this connection, the optimizer tries to minimize the distance between measured values or values created *in silico* and the simulated time course for each reacting species by varying the model parameters. The smaller this distance is, the higher is the quality of a possible solution for one parameter set. This quality measure is often called the "fitness" of the parameter set. As an exhaustive search for the best solution is computationally not possible, heuristic optimization methods try to find the global optimum of the system. Usually, metabolic systems are analytically hard or infeasible to solve. Often those systems are non-convex or multimodal, i. e., contain numerous local optima, and the gradient cannot be computed easily. Biologically inspired optimization procedures like Evolutionary Algorithms (EAs) are known to handle even highly nonlinear optimization problems [32-34]. Many such optimization algorithms are freely available in several software packages [35-38] or included in commercial toolboxes [39].

During the last few decades, manifold derivatives of EAs have been proposed. Each of them has certain advantages and is therefore more or less appropriate for a special problem. Their development was driven by analogies to natural phenomena such as Darwinian evolution (genetic algorithm [40], evolution strategy [41], differential evolution [42]), hill climbing [43], the formation of crystal structures in metallurgy (simulated annealing [44]), or the swarm intelligence idea (particle swarm optimization [45, 46]). Each one of these optimization procedures provides several settings that influence its performance; for instance, the temperature in simulated annealing, the crossover probability in genetic algorithms, or the population size in particle

swarm optimization. For a detailed introduction to all heuristic optimization procedures used in this work [see Additional file 1].

In several studies, heuristic optimization procedures like EAs have been applied successfully to biochemical systems after these have been translated into sets of differential equations [32-34, 47-50]. Most of the time standard settings for the optimization procedures are used. An analysis of these settings does, in fact, enable enhancement of the performance of the optimization process [33, 47, 48]. In many cases, a lack of time prevents researchers from systematically benchmarking these settings. In order to improve the model quality, however, this would be necessary. The resulting model systems, including the identified parameters, are often used to derive network properties like the long-term behavior or to perform steady-state analyses [49-53] of the system, but only in few cases detailed and specially derived rate laws could be applied to deduce a model system [49, 50]. Many studies are available, in which a single type of approximative rate equation is applied to set up a model – in many cases without a comparison with alternative approaches. Guthke *et al.* modeled the amino acid metabolism of primary human liver cells using a phenomenological approach [51, 53] whereas Liu and Wang used S-systems [21, 54, 55] for their biochemical models [56]. Magnus *et al.* applied linlog kinetics [12, 22] to model the valine and leucine metabolism in *C. glutamicum* [52].

Very few studies compare alternative modeling approaches to investigate their applicability for the specific problem [9, 34]. While Spieth *et al.* studied whether *in silico* time series data generated with certain model systems can be reproduced crosswise with other ones [34], Bulik *et al.* analyzed properties like stability when detailed kinetic equations within a system are replaced by approximative ones [9].

An investigation of both, alternative modeling approaches together with a systematical benchmark of the settings of optimization procedures, was rarely done. The disposability of software tools, which assign rate equations more or less automatically [57-61], requires the user to be especially aware of the properties of different modeling approaches and the possible quality that can be achieved with a certain type of kinetic models. Since automatically created models have already been published [62], and attempts have been made to scale up this modeling process to derive even genome-scale kinetic models automatically [63], these properties are of paramount importance. Summarizing, to construct a mathematical description of a biochemical

reaction system the modeler has to consider at least two central questions:

1. Which rate laws are the most appropriate ones for the specific purpose?

2. Which optimization procedure performs best on the problem class of parameter inference?

Once a model has been created, the choice of initial conditions represents a further important question for an appropriate simulation of the model. Fundamentally, the system can be written as an initial value problem, sometimes referred to as a single-shoot approach [30], or as a multiple-shooting problem [31, 64]. Here, a single-shoot strategy is employed using the steady-state concentrations of the participating metabolites as initial values.

This work addresses both questions and tries to identify an optimal model for a well-studied example network: the metabolism of L-valine (Val) and L-leucine (Leu) in *Corynebacterium glutamicum*, an aerobic gram-positive bacterium which is used to produce about two million tons of amino acids per year [65]. For this reaction network (see Figure 1 and Table 1) we construct seven
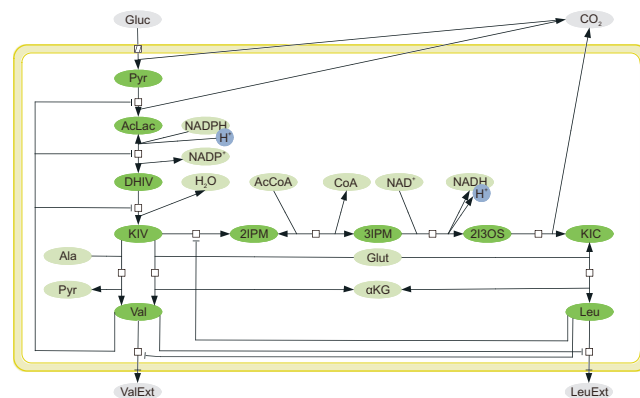

Figure 1
Process diagram of the biosynthesis of valine and leucine in *C. glutamicum*. The pathway shown here is constructed using the information from both the KEGG and the METACYC databases. Metabolites outside the cell are not directly included in the model system. As 2-ketoisovalerate (KIV) is the starting point for both products, its production and degradation are highly regulated. For this purpose four feedback inhibitions control the reactions linked to this intermediate. Val and Leu can be transported out of the cell when not needed thereby competing for a free binding site in the transporting enzyme. Note that enzymes are omitted from this process diagram for the sake of a clear arrangement of the participating species.

Table 1: The reaction system in more detail

No	Reaction	Enzyme	Inhibitor
R_1	$2\ Pyr \rightarrow AcLac + CO_2$	AHAS	Val
R_2	$AcLac + NADPH_2 \rightleftharpoons DHIV + NADP^+$	AHAIR	Val
R_3	$DHIV \rightarrow KIV + H_2O$	DHAD	Val
R_4	$KIV + Glut \rightarrow Val + \alpha KG$		
	$BCAAT_{ValB}$		
R_5	$KIV + Ala \rightarrow Val + Pyr$		
	$BCAAT_{ValC}$		
R_6	$Val \rightarrow Val_{ext}$	$Trans_{Val}$	Leu
R_7	$KIV + AcCoA \rightarrow IPM + CoA$	IPMS	Leu
R_8	$IPM + NAD^+ \rightarrow KIC + NADH_2 + CO_2$	IPMDH	
R_9	$KIC + Glut \rightleftharpoons Leu + \alpha\ KG$		
	$BCAAT_{LeuB}$		
R_{10}	$Leu \rightarrow Leu_{ext}$	$Trans_{Leu}$	Val

The reactions in KEGG are lumped together resulting in this reaction scheme which is in accordance with METACYC except for the question of irreversibility, and apart from R_2 and R_9, which are reversible.

alternative systems of differential equations based on four rate laws. These are the generalized mass action rate law [5], Michaelis-Menten equation, and convenience kinetics [16] as well as the stochastic Langevin equation [10, 11]. To evaluate the influence of irreversible reactions, we construct two models for each deterministic rate law: one in which all reactions are considered reversible and a second one in which only two reactions are considered reversible. In a two-step benchmark we systematically examine eight optimization algorithms to estimate the parameters of all models. In a coarse-grained trial all algorithms are applied to all model systems with standard settings. In the fine-grained benchmark, alternative settings of the algorithms are evaluated to improve their optimization performance on the best models. We focus on nature-inspired heuristic optimization procedures [see Additional file 1], namely, Hill Climber (HC), Simulated Annealing (SA), Genetic Algorithm (GA), Evolution Strategy (ES), Differential Evolution (DE), Particle Swarm Optimization (PSO), and Tribes. A random (Monte Carlo) optimization serves as a general reference algorithm.

Results and discussion
Mathematical models
Figure 1 shows the biosynthesis of Val and Leu in *C. glutamicum*, starting with pyruvate (Pyr). The reactions of this pathway are summarized in Table 1. In R_3, one substrate (DHIV) turns into one product (KIV) when neglecting the influence of the second product, water, which is plentiful in the cell. Therefore, we can apply a Michaelis-Menten equation to model the kinetics of this reaction. The transport of Val and Leu through the cell wall (R_6 and R_{10}) has exactly one substrate and one product and can, therefore, also be modeled using the Michaelis-Menten approach. All other reactions in

the network cannot be modeled using this classical approach because multiple substrate or product molecules are involved. Therefore, approximative rate laws can be applied to these reactions. Here we employ one stochastic and three deterministic rate equations. Approximative rate laws can be used for the three reactions with a Michaelis-Menten mechanism as well.

The reversibility of the reactions constitutes another important question to be solved before modeling. As transport through the cell wall removes both products from the cellular system, we assume that there is no reverse reaction (an uptake of Val or Leu) and, hence, consider both reactions irreversible. The two reactions R_2 and R_9 are reversible [6-8, 52]. We let the optimization procedure "decide" if the remaining reactions should be modeled in a reversible or irreversible way. To this end, we construct one "reversible" and one "irreversible" alternative model for each rate law, keeping only the two known reactions R_2 and R_9 reversible.

In this way we derive the following seven models based on four approaches for the reaction velocity on this pathway. Details and the formulas can be found in Methods and [see Additional file 2].

GMAKr Pure generalized mass action kinetics, in which all reactions apart from R_6 and R_{10} are modeled reversibly, with 24 parameters.

GMAKi Pure generalized mass action kinetics, in which only the two reactions R_2 and R_9 are considered reversible, with 18 parameters.

GMMr Like GMAKr but with three Michaelis-Menten equations for reactions R_3, R_6 and R_{10}. This model contains 31 parameters. In the GMAKr model the influences of all enzymes are neglected and hidden in the rate constants, which is an oversimplification of the biochemical process. The model comes closer to the biochemical process when inserting Michaelis-Menten equations for the three reactions R_3, R_6, and R_{10}.

GMMi Like GMMr but with only two reversible reactions R_2 and R_9, leading to 24 parameters.

CKMMr Convenience kinetics with three Michaelis-Menten equations as in GMMr. All reactions apart from R_6 and R_{10} are considered reversible leading to 59 parameters. The convenience rate law is also an approximation of the biochemical process. Therefore, we do not construct a pure convenience kinetics model of the whole system but apply Michaelis-Menten kinetics whenever this is possible (R_3, R_6 and R_{10}).

CKMMi Convenience kinetics with three Michaelis-Menten equations as in model GMMi, in which only the two reactions R_2 and R_9 are considered reversible, with 41 parameters.

LANG To demonstrate the possibility of large-scale parameter optimization, even for stochastic models, and to model the effects of random fluctuations in the metabolite concentrations, we consider a stochastic description as well. Based on the Langevin equation, this system contains 24 parameters.

In a glucose stimulus-response experiment 47 measurements are taken for 13 metabolites on this pathway (for details see Methods). The parameters of all models are calibrated with regard to these data.

Fine-tuned optimization algorithms and models

In the next step the parameters of all seven mathematical models have to be estimated. In this process the relative distance between simulated model data and experimental data serves as a quality measure (fitness) of an estimated parameter set. Note that the error between the measurements and the model simulation is to be minimized, so the quality of the solutions increases with decreasing error values. A random optimization (Monte Carlo search) of the models (Figure 2) yields relative differences between measurements and model systems that are at least three times higher than the difference between the measurements and uncoupled cubic approximation splines. Therefore, we apply the nature-inspired heuristic optimization procedures HC, SA, GA, ES, DE, PSO, and Tribes to all seven models with standard settings (Table 2; details in Methods section). Apart from only some minor exceptions for the two irreversible models, five algorithms turn out to be especially useful (Figure 3 and Table 3). These are the binary-valued Genetic Algorithm (binGA), Evolution Strategy with covariance matrix adaptation [66] (cmaES) with elitism (plus strategy), PSO, DE, and Tribes. The performance of the lattermost procedure cannot be further improved as this
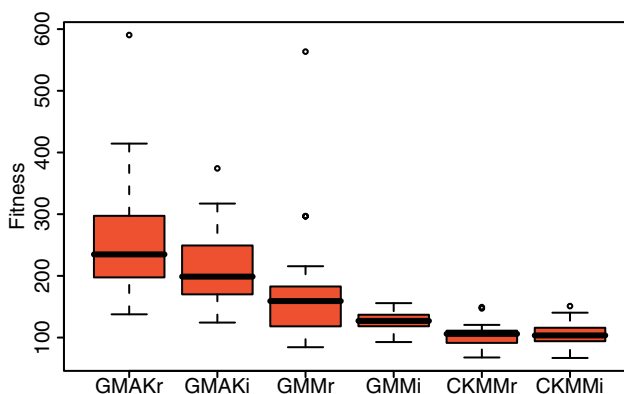

Figure 2
Parameter identification using random optimization. The Monte Carlo method constitutes the simplest way to optimize the parameters within a model. Thereby this method "dices" a random solution within the search space and logs the best solution found. This procedure is repeated twenty times in fifty multi-starts using 100,000 evaluations. The more complex the model the better is the solution found by the Monte Carlo method (RSE 66.920 for the CKMMi model), but still this method cannot approach the quality of independently generated splines (RSE 19.670).

algorithm is a settings-free derivative of PSO. Hence, we study the influences of various settings on the capabilities of the other four procedures aiming to improve the fit of the model to the data for each deterministic model.

Due to the generally better performance of the three reversible models, we examine alternative settings for the optimization procedures only for these models and subsequently apply the best setting found to each alternative irreversible model. The most promising settings are used to optimize the stochastic model as well.

Taking a closer look at the effects of alternative mutation and crossover operators on binGA and ES (Figure 4) reveals

Table 2: Settings for the standard algorithms in detail

Algorithm	Population	Mutation	Crossover	Selection
Monte Carlo	50	no	no	Best
Hill Climber	1, 10, 25, 50, 100, 250	Fixed Step $\sigma = 0.2$, $p_m = 1$	no	Best
binGA	250	one-point, $p_m = 0.1$	one-point, $p_c = 0.7$	Tournament, group of 8
realGA	250	global, $p_m = 0.1$	UNDX, $p_c = 0.8$	Tournament, group of 8
stdES	(5, 25)	global, $p_m = 0.8$	discrete one-point, $p_c = 0.2$	Best
cmaES	(5,$^+$25)	CMA, $p_m = 1$	no	Best
SA	250	linear annealing schedule, $\alpha = 0.1$, initial $T = 5$		Best
DE	100	current-to-best/1, $\lambda = F = 0.8$, $CR = 0.5$		
PSO	100	star topology, $\varphi_1 = \varphi_2 = 2.05$, $\chi = 0.73$		

This overview lists the standard settings for the optimization procedures used to infer the parameters in the differential equation systems. The algorithms DE and PSO do not follow the general scheme of mutation and crossover. The Tribes algorithm is not listed here as it was designed to be a settings-free derivative of PSO. The cmaES is used with and without elitism (plus or comma strategy). For details, see Methods and [see Additional file 1].
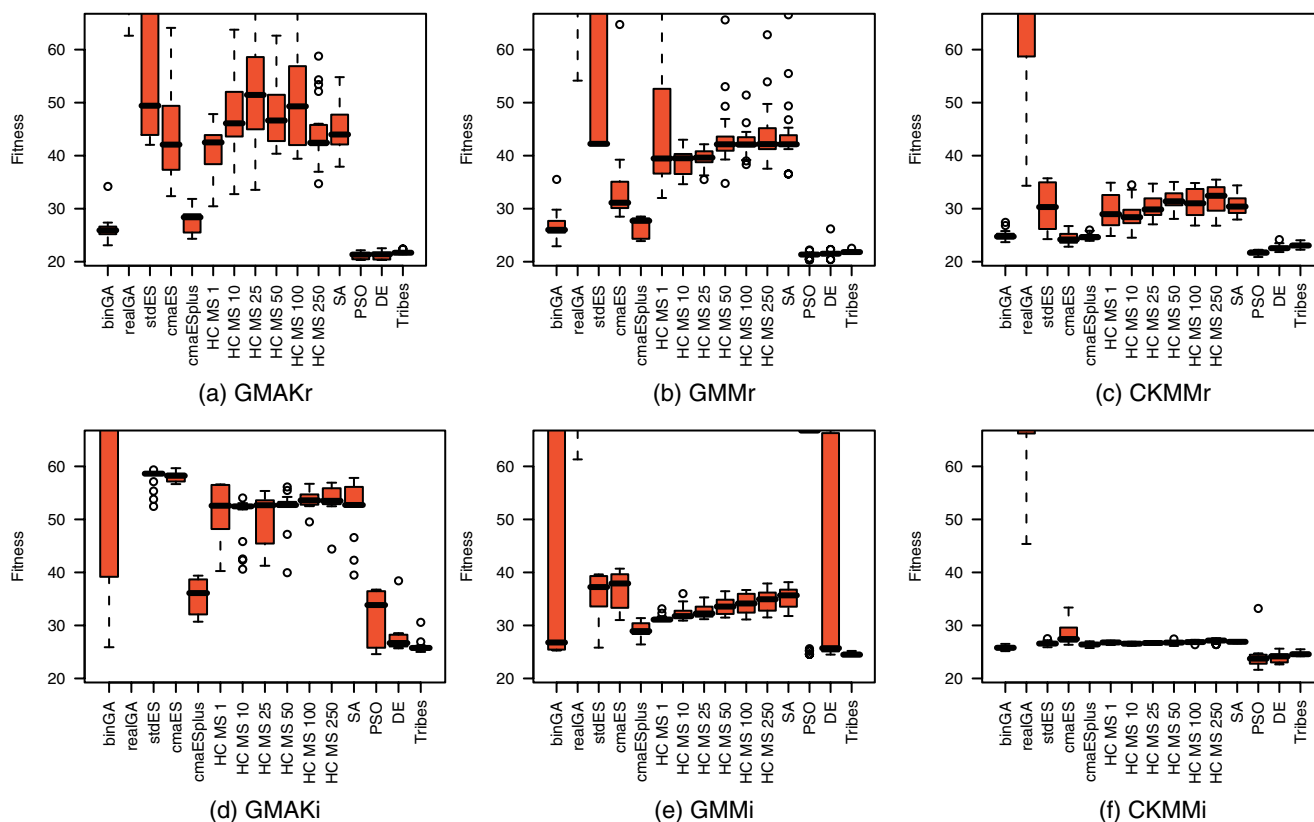
Figure 3
Comparison of the standard optimization algorithms applied to each deterministic model. Five algorithms show outstanding results on the reversible models. These are binGA, cmaESplus, DE, PSO, and Tribes. On the irreversible models, binGA does not yield such good average results, but is still capable of finding good local optima. For these models, cmaESplus and Tribes also find pre-eminent results, whereas PSO and DE are among the best algorithms, except for the GMM models. The real-valued GA performs worst in all cases. The hill climbers show differences in their effectiveness when invoked with various numbers of multi-starts but cannot compete with the other algorithms. The term "fitness" is used to define a quality measure of possible solutions, which is minimized.

that the more detailed CKMMr model can be fitted to the data with almost any combination of both operators, whereas the other two reversible models show larger differences. BinGA especially provides good performance with almost every operator combination. The only two exceptions are no mutation combined with one- or *n*-point crossover. The influence of these operators on ES is much stronger and more problem-specific. Some settings improve the performance of ES, but most result in significantly worse fitness values. In contrast to binGA, for which the combination of adaptive mutation with one-point or bit-wise crossover or adaptive mutation without crossover provides an improved average performance, the plots in Figure 4(d) through Figure 4(f) do not show such a general trend for ES. Thus, we evaluate the influence of the mutation and crossover probabilities p_m and p_c on only binGA to identify the best ratio. The plot of the resulting fitness landscape for the GMAKr model was limited to a fitness of 28 (Figure 5(a)). Hence, areas with a worse performance are

shown in white. The best combination $p_m = 0.2$ with $p_c = 1.0$ is the starting point for investigating the influence of different population sizes. We vary the population sizes from 50 to 2,000 (Figure 5(b)). The larger the population, the smaller is the variance within this population. However, if the population is too large, this variance increases again. Although a larger population leads to smaller variances, statistically, it does not help to find a better total solution for the optimization problem.

Due to the fact that cmaESplus leads to reasonable results on each model and also that there is no general trend for alternative combinations of other mutation and crossover operators, we pick the CKMMr model to examine the influence of population size ($\mu + \lambda$). The value of μ represents the number of parents in the population from which, in each generation, λ offsprings are created. Figure 6 depicts the resulting fitness landscape. Combinations of μ and λ with $\mu > \lambda$ are left out and occur as a white

Table 3: Preliminary test results

GMAK, reversible					GMAK, irreversible				
Min.	Algorithm	Average	Std. Dev.	Algorithm	Min.	Algorithm	Average	Std. Dev.	Algorithm
20.334	PSO	21.190	0.576	PSO	24.587	PSO	25.967	1.171	Tribes
20.335	DE	21.228	0.756	DE	25.006	Tribes	29.502	9.610	DE
21.401	Tribes	21.725	0.275	Tribes	25.683	DE	33.169	10.143	PSO
23.097	binGA	26.106	2.204	binGA	25.981	binGA	35.670	3.125	cmaESplus
24.321	cmaESplus	27.598	2.091	cmaESplus	30.704	cmaESplus	50.663	4.138	HC MS 10
GMM, reversible					GMM, irreversible				
Min.	Algorithm	Average	Std. Dev.	Algorithm	Min.	Algorithm	Average	Std. Dev.	Algorithm
20.312	PSO	21.272	0.461	PSO	24.477	Tribes	24.654	0.282	Tribes
20.407	DE	21.711	1.153	DE	24.499	DE	37.696	19.374	DE
21.590	Tribes	21.887	0.243	Tribes	24.553	PSO	41.529	21.768	binGA
22.913	binGA	26.742	2.711	binGA	25.266	binGA	31.136	9.052	cmaESplus
23.890	cmaESplus	26.624	1.851	cmaESplus	25.812	stdES	31.338	0.546	HC MS 1
CKMM, reversible					CKMM, irreversible				
Min.	Algorithm	Average	Std. Dev.	Algorithm	Min.	Algorithm	Average	Std. Dev.	Algorithm
20.882	PSO	21.773	0.352	PSO	21.632	PSO	23.968	0.931	DE
21.821	DE	22.633	0.562	DE	22.651	DE	24.624	0.315	Tribes
22.258	Tribes	23.079	0.464	Tribes	24.191	Tribes	25.761	0.331	binGA
22.829	cmaES	24.341	1.026	cmaES	25.152	binGA	26.434	0.339	cmaESplus
23.687	binGA	24.736	0.557	cmaESplus	25.738	cmaESplus	26.539	0.210	HC MS 10

For all deterministic models (reversible models on the left, irreversible models on the right) the five absolute best algorithms and the five average best algorithms with standard deviations are listed.

area within the landscape. Larger population sizes lead to better average fitness values. ES achieves its best average and total performance for (25 + 125). The combination (50 + 75) leads to only a slightly worse average and total fitness. These two settings clearly outperform all other combinations. By varying the values for *F*, *λ*, and *CR* we test how to improve the performance of DE on each reversible model. Generally, the choice of *F* = 0.8 leads to a better performance than *F* = 0.5 (Figure 7). The influence of the remaining two parameters is less clear. Hence, we pick the best setting for each model and evaluate the influence of different population sizes (Figure 8). A population with 100 individuals performs best on each model according to the median.

To hone the performance of PSO we alter both strategy parameters φ_1 and φ_2 on the star topology and apply a grid 3 and linear 3 topology using the standard values for φ_1 and φ_2 (Figure 9). On the three reversible models the grid 3 topology performs slightly better than all other settings. Hence, we test how an alternative population size influences its capacity. Figure 10 depicts the results of this experiment where we vary the size of the population in the intervall from 25 to 500. A larger size lowers the quality and a small population of 25 individuals is confirmed to be the best choice.

Comparison of the performance of the optimization algorithms

An overview of the most successful optimization algorithms together with their best suited settings can be found in Table 4. Tribes is not the very best optimization algorithm but yields meaningful results for all models. As a settings-free procedure, Tribes is a good choice if there is no time to examine alternative adjustments. The standard PSO algorithm yields the best median fitness for the CKMMr model with 21.687. On the GMAKr model, DE with *F* = 0.8, *λ* = 0.5, and *CR* = 0.3, and a population size of 100 gives the best median fitness of 20.369 for this model. This is almost 0.9 better than standard PSO. DE yields a median fitness of 22.196 on the CKMMr model when set to *F* = 0.8, *λ* = 0.5, *CR* = 0.5, and a population size of 100. Both algorithms also perform well on the GMMr and the Langevin model (in total, average, and median). Hence, they are an advisable choice when optimizing the parameters of various mathematical models of biological systems.

Comparison of the modeling approaches

Figure 11 depicts the measurement data together with the best simulation results of each reversible deterministic and the Langevin model.
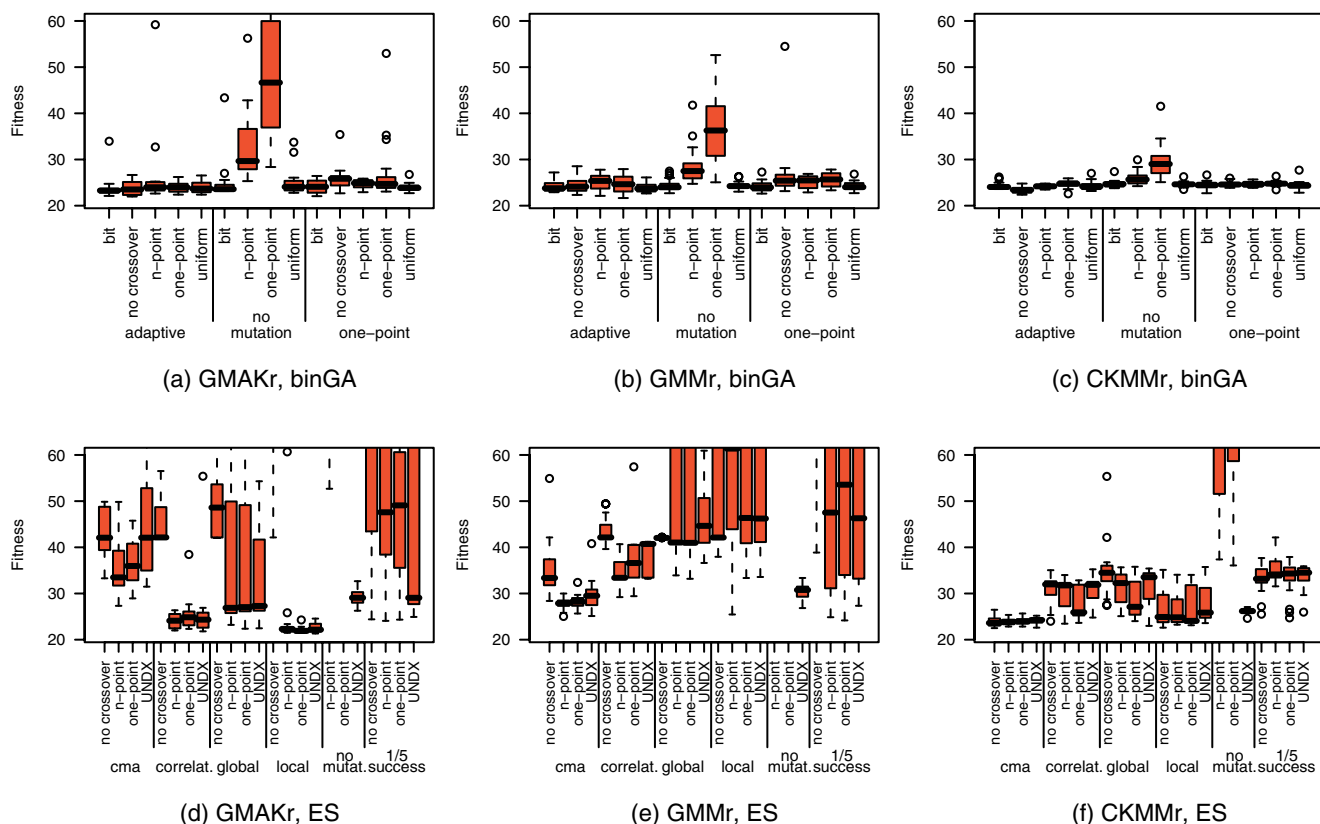
Figure 4
Benchmark of various combinations of mutation and crossover operators. The six box plots depict the dependency of fitness on a multitude of combinations of mutation and crossover operators available for binGA (a-c) and ES (d-f). All plots are limited to a fitness of 60. The more detailed the model the better the overall fitness that can be obtained. For binGA almost all combinations of mutation and crossover operators yield good fitness values. The only two exceptions are no mutation with one- or *n*-point crossover, which shows the worst fitness on all three models. The ES is more sensitive to alternative combinations of mutation and crossover operators. The abbreviations cma, correlat., no mutat., and 1/5 success stand for covariance matrix adaptation, correlated, and no mutation and for the 1/5th success rule. The elaborated comparison shows that some settings of the ES lead to equally good or even better results than that of binGA. However, most ES-settings cannot compete with the fitness values found for most binGA settings.

To evaluate which model is the most promising one we consider the following three criteria:

1. Fit of the model to the data

2. Number of model parameters

3. Computational time for simulation (computational complexity, weak criterion)

Each reversible deterministic model can be fitted to the data with a similar deviation from the measurements. The irreversible alternatives show a significantly higher deviation. Only the irreversible CKMM model is able to fit the data almost as well as the reversible models. However, most curves resulting from all of the irreversible models tend to become straight lines through the measurements, and thus behave in a biologically implausible manner [see Additional file 2]. This suggests that the irreversible models are unable to follow the dynamics of the system due to their fewer degrees of freedom. The rather abstract reversible models are able to tackle possible side effects of reactions not included in this reaction system and simulate them in terms of the reverse reaction. These models also consider the fact that, in biological systems, reaction products are normally not completely absent. Their concentration may be low, but they still take part in the reaction, in some cases giving a kind of feedback to the reactants [[19], pp. 312–313]. Therefore, the irreversible models are generally not competitive with respect to the data fit. From the three remaining reversible models, the CKMMr model achieves the best fit to the data, with 20.100, followed by GMMr model at 20.280 (worse by 0.180), and the
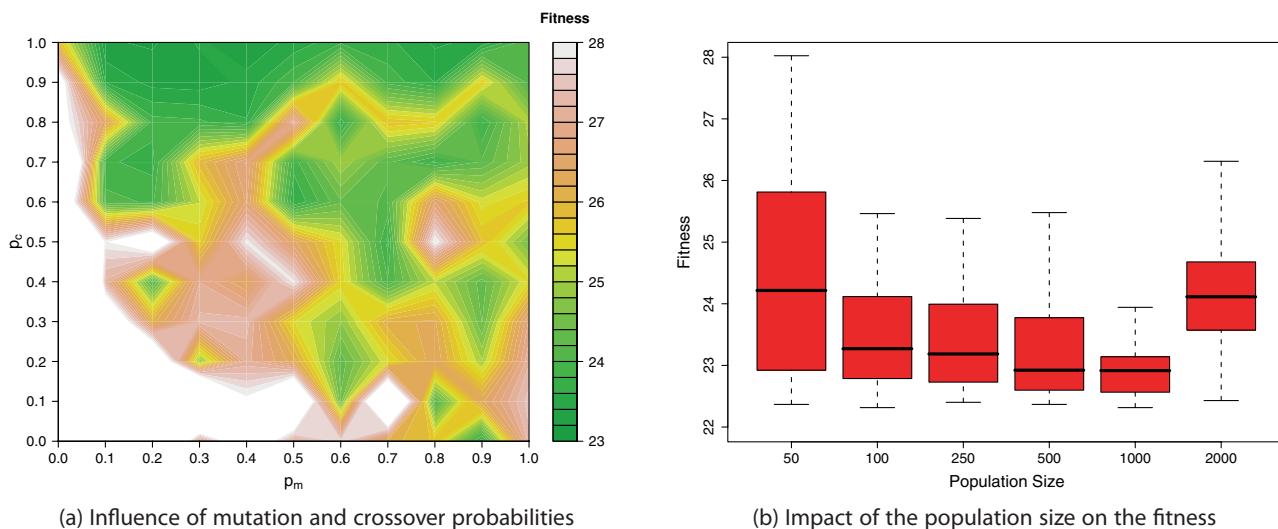
(a) Influence of mutation and crossover probabilities

(b) Impact of the population size on the fitness

Figure 5
Optimizing the settings for binGA on the reversible GMAK model. The best combination of mutation and crossover operators for binGA on the reversible GMAK model is adaptive mutation and bit-simulated crossover. For this pair the influence of the mutation and crossover probabilities p_m and p_c is examined and plotted in (a). Each experiment is repeated 20 times. The combination $p_m = 0.2$ and $p_c = 1.0$ improved the average fitness most successfully. Figure 5(b) depicts the impact of the population size for this setting. The variance decreases with an increasing population size and reaches its minimum at 1,000. However, the best single result cannot be surpassed.
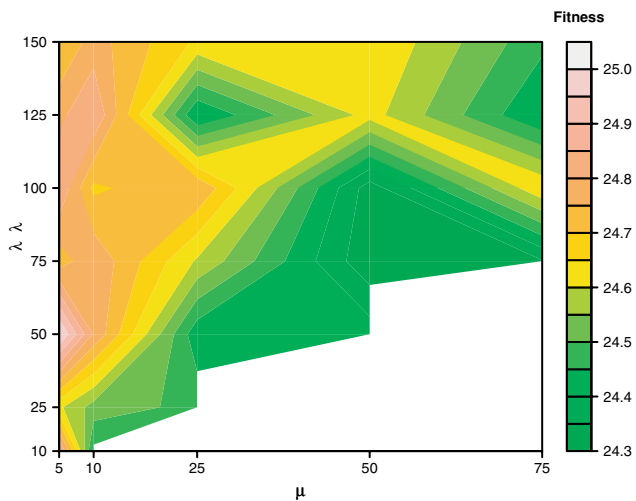

Figure 6
Influence of the population size on the performance of cmaESplus for the reversible CKMM model. The Evolution Strategy with covariance matrix adaption and elitism is one of the most promising optimization strategies among the standard algorithms. Here, we examine whether its capabilities can be improved by another choice of the parameters $(\mu + \lambda)$. All experiments are repeated 20 times. The fitness landscape shows a minimum in the average values at (50 + 75).

GMAKr model at 20.326 (slightly worse by a further 0.046). The difference in the best model fit between the CKMMr and the GMAKr models is only 0.303. Hence, all three reversible models can be fitted to the data with a similarly small relative squared error (RSE, see Methods). A comparison of the best model fit 20.100 (CKMMr) to the fitness of the independently computed splines (19.670) evinces a difference of only 0.430. When considering the number of parameters (criterion 2), the GMAKr model shows a clear advantage with its 24 parameters compared to the 31 of the GMMr model or even 59 of the CKMMr model.

When choosing the parameter values for the GMAKr model completely randomly, this system can hardly be integrated without step size adaptation. Therefore, it is necessary to identify meaningful ranges of kinetic parameters within BRENDA [24-26]. A low-value initialization is necessary to assure numerically stable initial populations for the optimization procedures. For the other models this happens only if the parameters are chosen from implausibly large ranges.

The last criterion, the computation time, also depends on the complexity of the model but not necessarily on the number of parameters. The GMAKr model requires
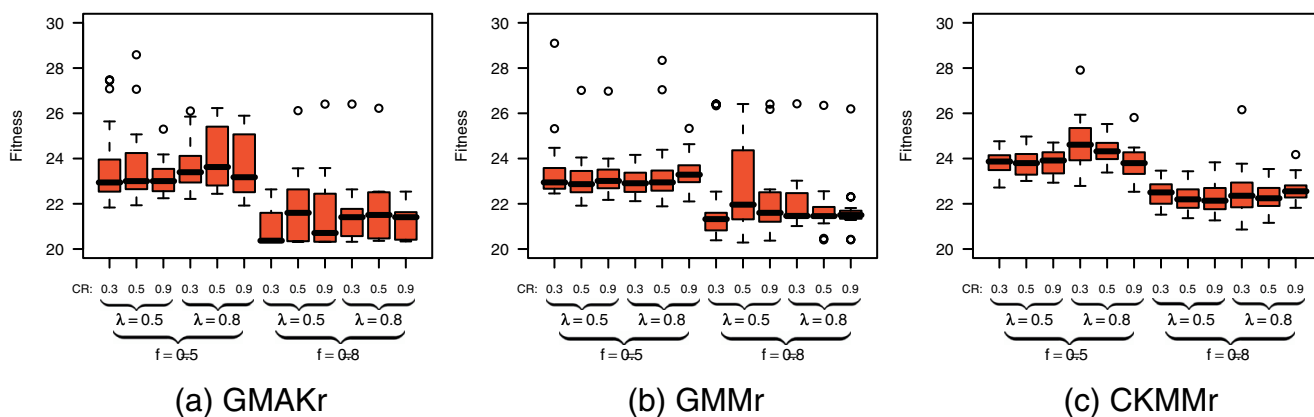
(a) GMAKr   (b) GMMr   (c) CKMMr

Figure 7
Comparison of a multitude of settings for differential evolution on the three reversible models. The settings of DE are studied on the reversible models. All plots suggest that the choice $F = 0.8$ is more appropriate than $F = 0.5$, whereas CR and λ influence the results less clearly.

the smallest number of mathematical operations, followed by the GMMr model. The most complicated model is the CKMMr model. The average evaluation time over 100,000 repeats with randomly chosen parameters is 49 ms for the GMAKr model, 101 ms for the GMMr model and 151 ms for the CKMMr model. For hardware details see the Hardware Configuration section.

In order to take the effects of random fluctuations into account, one has to use stochastic models of the chemical reaction system. However, the most general approach, the chemical master equation [67], can hardly be solved numerically for larger systems [20]. Taking the number of parameters (criterion 2) and the

computational costs (criterion 3) into account, the Langevin model is the most suitable formalism to consider the effects of random fluctuations while still providing an acceptable performance. Since the model can be stated in a way that allows it to be integrated with standard solvers for ordinary differential equations, the computational costs are of the same order of magnitude as the solution of the GMAKr model. However, care must be taken with respect to justification of the underlying simplifying assumptions [10]. The stochastic model equations of the biological system under consideration show no qualitatively different behavior in comparison to the deterministic model. The main reasons for this observation are found in the rather large molecule
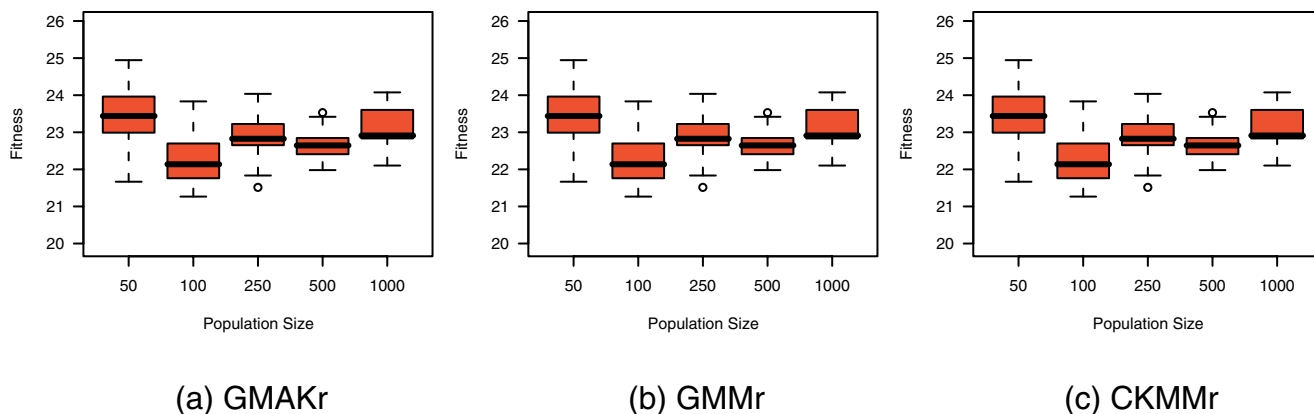

(a) GMAKr   (b) GMMr   (c) CKMMr

Figure 8
Influence of various population sizes on the performance of differential evolution. The settings $F = 0.8$, $\lambda = 0.5$, and $CR = 0.3$ are found to be most suitable for the GMAKr and GMMr model, whereas $CR = 0.9$ performs slightly better on the CKMMr model. Using these settings, the influence on the population size is studied. The boxplots suggest that a population size of 100 yields the best median results over 20 repeats.
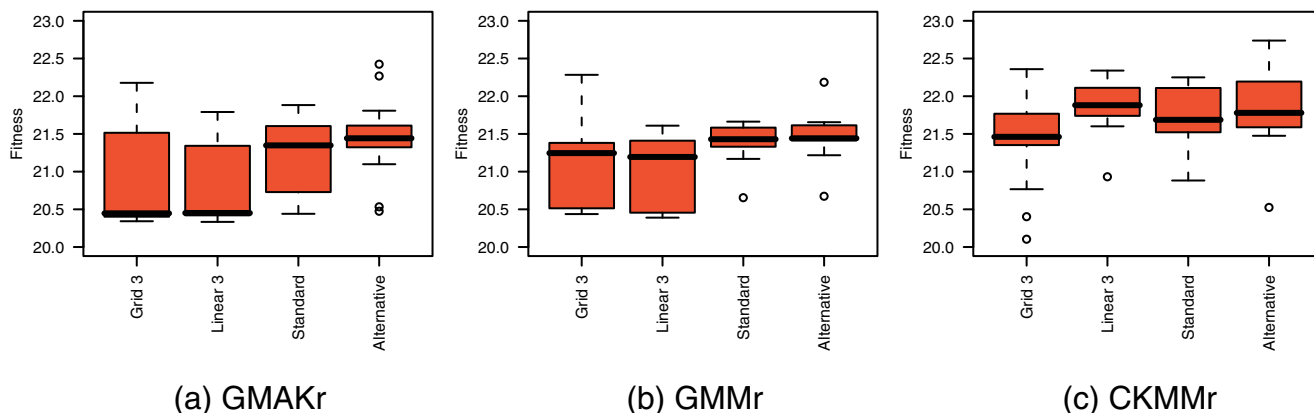
(a) GMAKr   (b) GMMr   (c) CKMMr

Figure 9
Influence of the settings on particle swarm optimization. Besides the standard star topology, a grid 3 and a linear 3 topology is tested on the three reversible models with $\varphi_1 = \varphi_2 = 2.05$. Furthermore, the alternative setting $\varphi_1 = 2.8$ and $\varphi_2 = 1.3$ is applied to the star topology. The grid topology performs best according to the median for the GMAKr and the CKMMr models, but is slightly worse than the linear topology on the GMMr model. All experiments are repeated 20 times.

populations and the absence of points of instability in the allowed phase-space region.

Parameter space analysis
Once a model has been optimized several times and locally optimal parameter sets for the model are available, an analysis of the space of potential solutions becomes possible. This allows deducing characteristics of the solution space aiming to reduce the model complexity and enhance the optimization performance. If, for instance, two parameters show a linear dependency or correlation to each other, one of these can be removed from the model. Another interesting experiment would be to determine new ranges for each parameter. This can be done if a certain parameter varies only in a very small range compared to its maximal possible range.

For each of the three models, GMAKr, GMMr, and CKMMr, we gather all parameter values from all optimization runs that lead to a fitness less than 25. In this way, we obtain one parameter matrix for each model, in which each column corresponds to one optimization run and each row stands for one parameter. We conduct three analyses on the best parameters on each reversible model (see Methods):

1. *Clustering* to identify groups of similar ranges or almost constant values and to visualize the values of each parameter.

2. *Variance analysis* to visualize the scattering of each parameter.
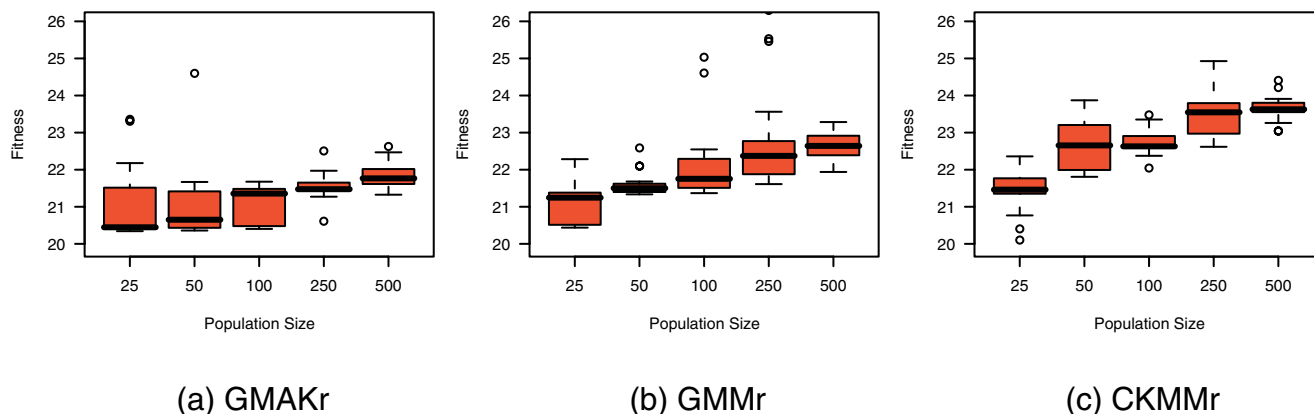

(a) GMAKr   (b) GMMr   (c) CKMMr

Figure 10
Influence of the population size on the performance of PSO with grid 3 topology. An increasing population size cannot improve the performance of PSO.

Table 4: Statistics on the most successful runs of each main optimizer

Model	reversible			irreversible			Algorithm	Population
	Min	Avg	Std. Dev.	Min	Avg	Std. Dev.		
GMAK	20.326	20.742	0.501	25.745	34.472	15.285	PSO linear 3, $\varphi_1 = \varphi_2 = 2.05$	25
	20.403	21.787	1.297	25.183	31.694	13.255	DE, $f = 0.8$, $\lambda = 0.5$, $CR = 0.3$	100
	21.975	23.812	1.604	24.741	49.045	21.285	binGA, adaptive MUT, no CROSS	250
	24.321	27.598	2.091	30.704	35.670	3.125	cmaES	(5+25)
GMM	20.280	22.818	2.186	24.857	27.978	9.146	DE, $f = 0.8$, $\lambda = 0.5$, $CR = 0.5$	100
	20.312	21.272	0.461	24.553	58.957	17.253	PSO star, $\varphi_1 = \varphi_2 = 2.05$	25
	21.649	24.628	1.801	24.616	40.896	25.881	binGA, adaptive MUT, one-point CROSS	250
	23.890	26.624	1.851	26.414	31.136	9.052	cmaES	(5+25)
CKMM	20.100	21.434	0.563	21.511	26.077	7.729	PSO grid3, $\varphi_1 = \varphi_2 = 2.05$	25
	20.862	22.499	1.119	21.763	23.603	1.268	DE, $f = \lambda = 0.8$, $CR = 0.3$	100
	21.431	23.222	1.066	25.632	28.055	2.697	cmaES	(10, 50)
	22.092	23.353	0.666	25.040	25.346	0.176	binGA, adaptive MUT, no CROS	100

For each model the minimal fitness and the corresponding standard algorithm are listed. The algorithm that achives the best average fitness and the corresponding average fitness are written in the last two columns together with the standard deviation. On the Langevin model, PSO with star topology, $\varphi_1 = \varphi_2 = 2.05$ and a population size of 100 is the most successfull algorithm with a fitness of 20.716.

3. *Multiple correlation analysis* aiming at finding highly correlated parameters, some of which can be eliminated.

For Histograms showing the parameter distribution of the GMAKr, GMMr, and CKMMr models [see Additional file 2].

Clustering groups similar parameters and similar optimization runs together. First, all parameters are reordered so that within the parameter vector all parameters with similar values over all optimization runs are placed next to each other. In the second step, all parameter vectors from every optimization run are swapped so that similar parameter vectors are located next to each other. Figure 12 graphically displays the results of the clustering approach (1). The heatmaps show all parameters on the y-axis and all optimization runs on the x-axis. The lighter the color, the lighter the value e of the parameter in the respective optimization run, with black representing values close to zero. As can be seen, there is a very flat hierarchy so there are no groups of parameters showing a similarity, but there are many parameters which are similar to their neighbor. A similar flat hierarchy can also be seen for the optimization runs. There is almost no relationship between the values of the parameters with respect to the experiments. This means that each parameter was optimized independently by the analyzed procedures. Some parameters show stripes within the heatmap of Figure 12. Stripes like these mean that the corresponding parameter barely varies in its value over all experiments. Note that these more or less constant parameters do not occur within the same cluster. All parameters of this type can either be replaced by their median thus reducing the complexity of the system, or

the ranges of these parameters can be set to more restrictive values. However, the experiments are broken into two groups in all three models: The first group shows homogeneously distributed parameter values whereas the second one contains more differences. The level of differences in the second group rises with the complexity of the model.

These results are confirmed by the *variance analysis* (2), whose results are shown in Figure 13. As can be seen, the higher the dimension of the optimization problem, i. e., the more complex the model is, the higher are the variances among the parameter set. This indicates that all parameters in the CKMMr model are allowed to vary within a rather large range. Such behavior is often referred to as a multimodal optimization problem. In contrast, the less complex models, GMAKr and GMMr, showed several dimensions of almost no variance. This corresponds to the observation made from the heatmaps that certain parameter values can vary only within a small range of values or even stay constant over multiple optimizations. Thus, the probability of finding multiple local optima increases with the model complexity. Particularly, all parameters which represent the impact of inhibitors exhibit noticeably large variances. The biological interpretation of this is that variations in strength and concrete mechanism of inhibition in one reaction can be balanced in terms of other reactions because this pathway contains four feedback inhibition mechanisms for this purpose (structural robustness).

In order to identify linear dependencies between model parameters, we perform a *multiple correlation analysis* (3).
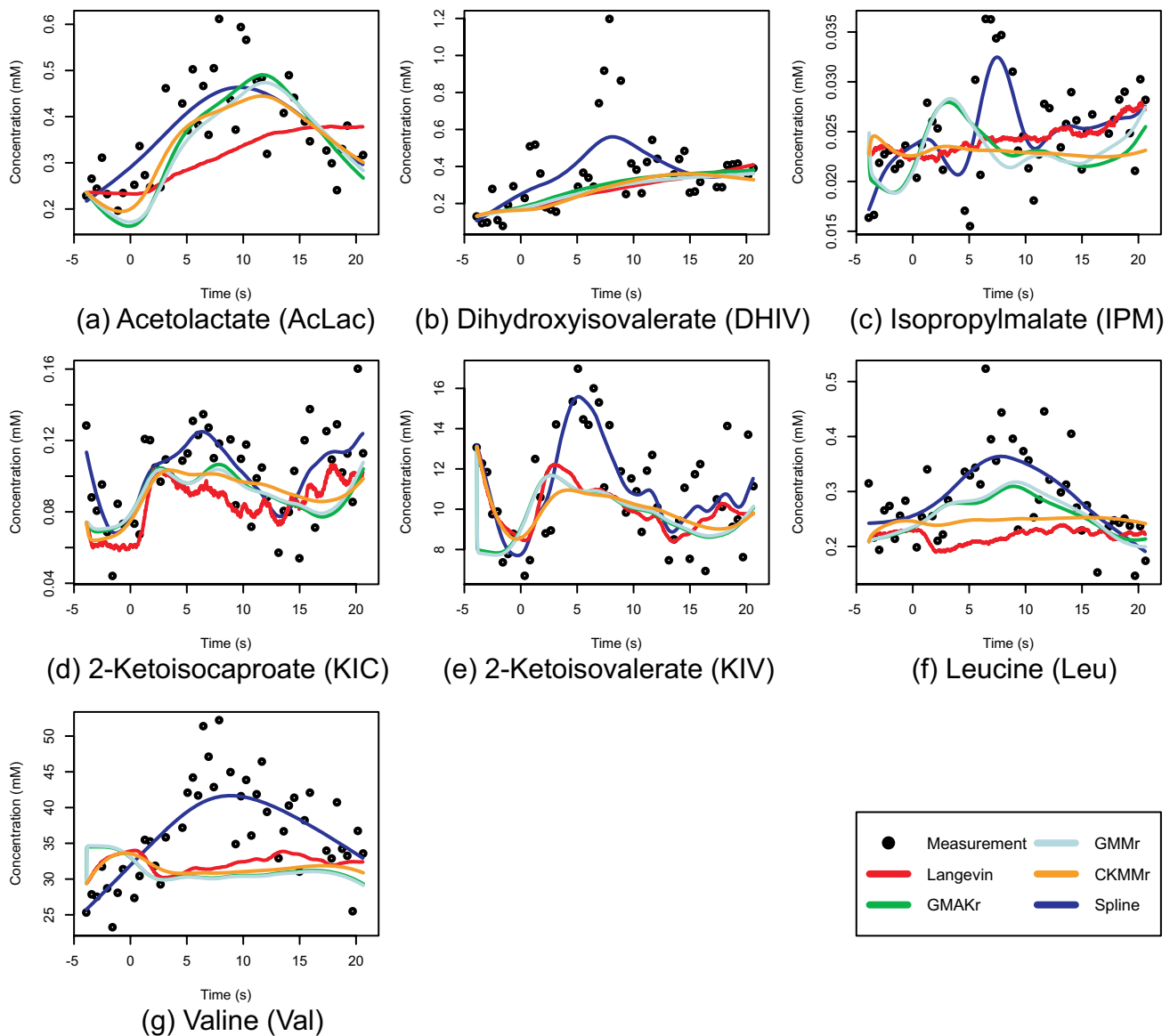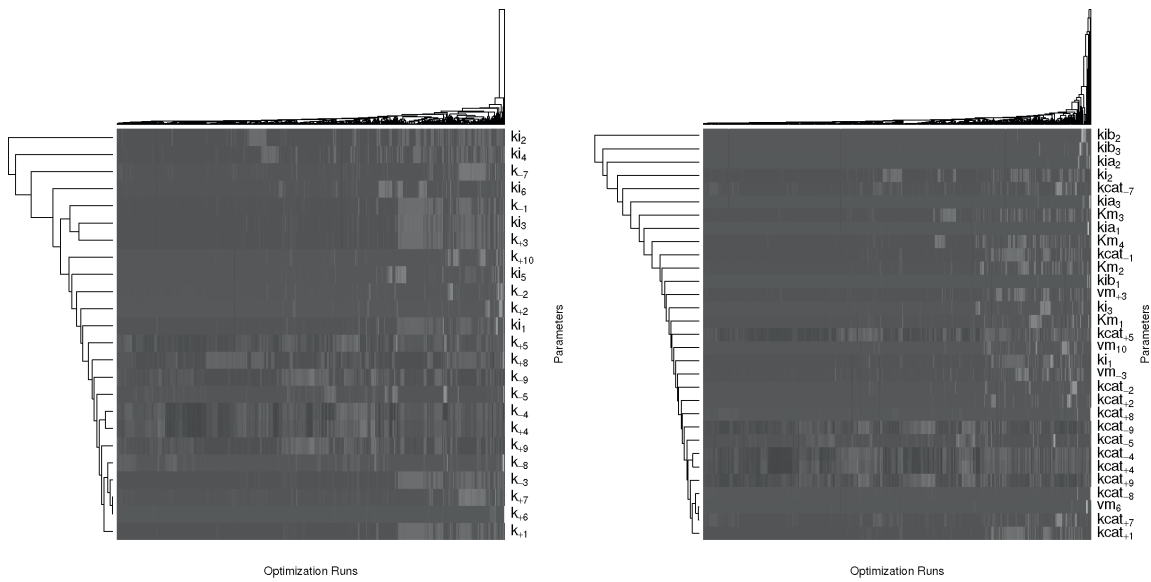
Figure 11

The best fit of all reversible deterministic and the Langevin models. Shown are the Langevin model, the measurements, and all reversible deterministic models. For a better visualization, splines are added to the data as well to help imagin how a perfect model curve would look. However, as can be seen, biological measurement data always shows fluctuations. The splines shown here are uncoupled, one individual spline for each metabolite. Thus, there is no underlying biological model motivating these curves. The irreversible models mostly result in straight lines through the data and are hence biologically implausible. For the sake of a clearer recognition of the reversible models, we omit these curves from the plot, [see Additional file 2] for the irreversible models.

For maximum generality, each parameter is assumed to possibly correlate with all other parameters of the system. Several highly correlated parameters are found in each model system. The correlation results are shown in Figure 14. All highly correlated parameters found exhibited significant variances as can be seen in Table 5 and Figure 13. We select a subset of parameters to be replaced by a linear regression model of highly correlated parameters. In this way the number of parameters is reduced by seven in the GMAKr model, by five in the GMMr model and by four in the CKMMr model. Subsequently, each model is optimized with the reduced parameter set, using the linear regression model for the non-optimized parameters. For this optimization, PSO is
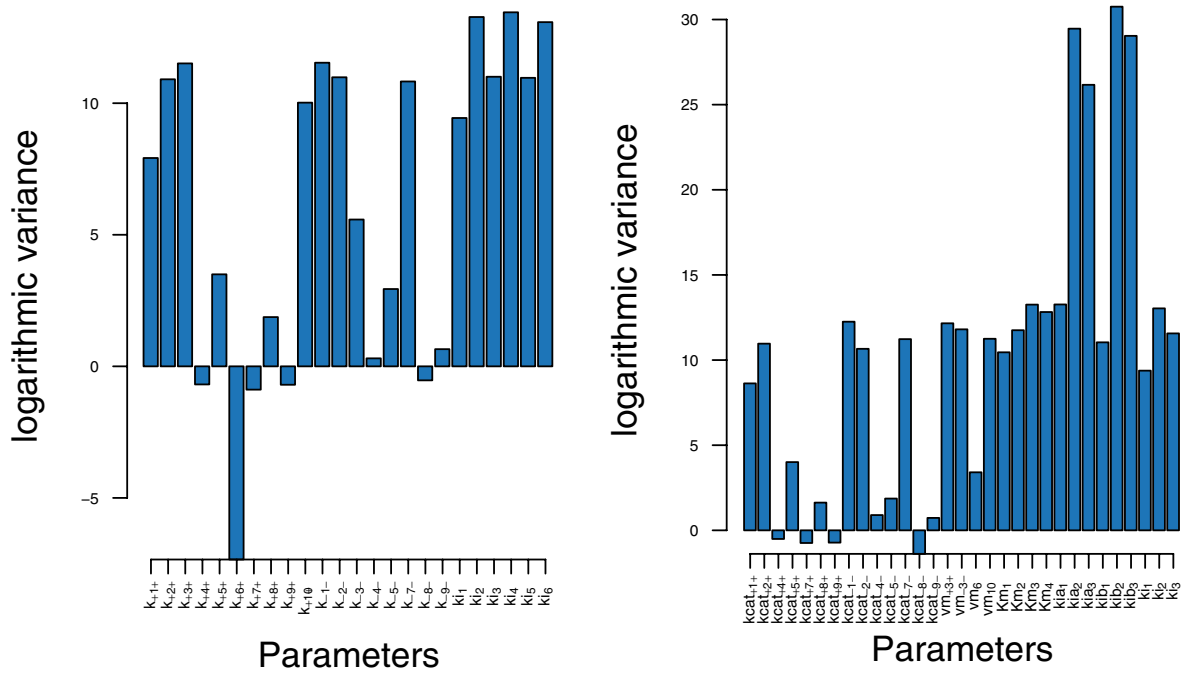
(a) GMAKr

(b) GMMr
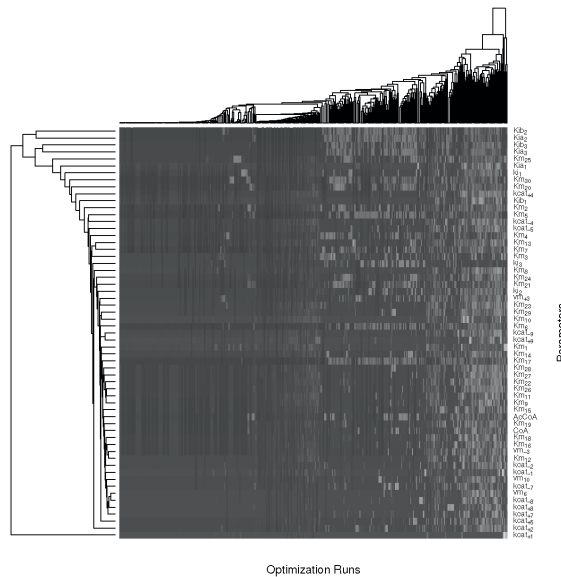
(c) CKMMr

Figure 12
Clustering of the best parameter values of all reversible models. A cluster analysis is carried out on all parameter values that have a fitness of less than 25. Each row of the above heatmaps corresponds to one parameter of the respective model and each column gives one parameter set that has been obtained in one of the optimization runs. The cluster algorithm swaps rows and columns to group similar parameters and parameter sets next to each other. This procedure leads to stripes from the left to the right, meaning that many parameters are often set to similar values in all optimization runs. If there were rectangular blocks within the heatmaps, this would mean that some parameters are correlated, thus showing a similar behavior, but because this is not the case, all parameters are distinct from each other, which can also be seen from the flat dendrograms at the side of each heatmap. Also, the dark-colored figures show that most parameters are set to low values because zero corresponds to black. The lighter the color the higher the value. For details, histograms showing the parameter distribution for the three models [see Additional file 2].

(a) GMAKr

(b) GMMr
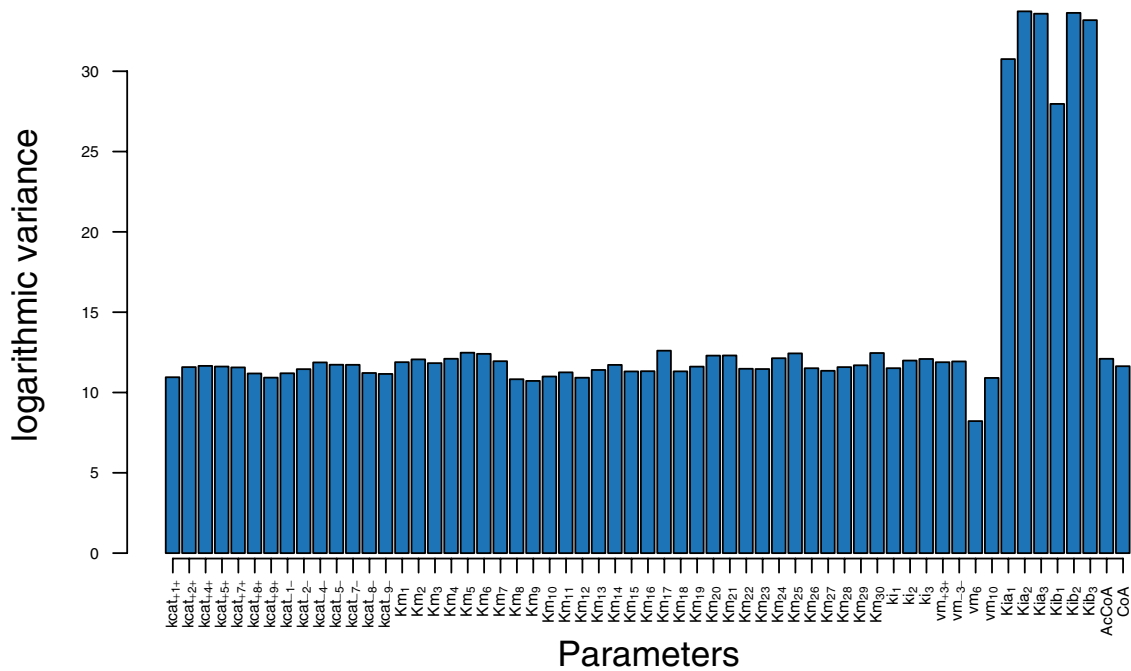
(c) CKMMr

Figure 13
Variance analysis of the best parameter values of all reversible models. Each bar plot shows the variances of every parameter among the best optimization runs for the three reversible deterministic models. For the sake of a better visualization these are plotted with a logarithmic scale.
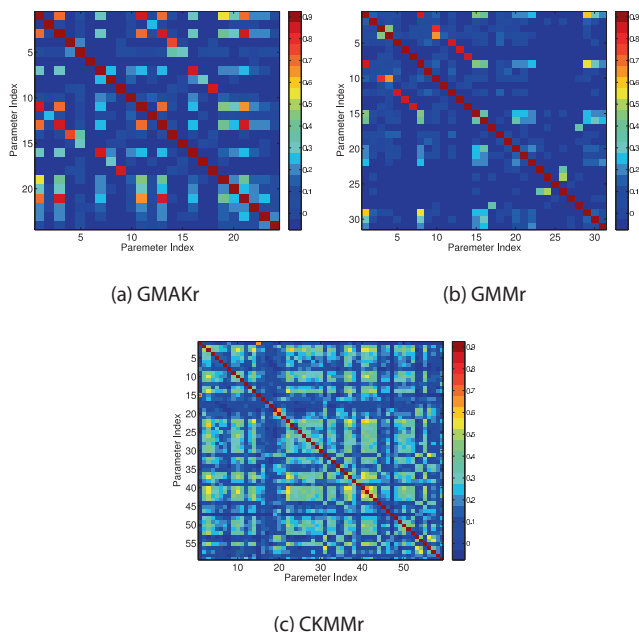
(a) GMAKr

(b) GMMr

(c) CKMMr

Figure 14
The correlation among the parameters in all reversible models. Each heat map shows to which extent each parameter is correlated with all other ones. The main diagonal shows the self correlation, which equals one. The correlation was computed using a multiple correlation analysis based on the Pearson correlation coefficient. If two parameters are highly correlated, one of both can be replaced by the other one hence reducing the system's dimensionality.

used because it is the the best-performing procedure in our benchmark. The results are shown in Table 5. The parameter reduction induced only a small loss of performance in each model, indicating that the original number of parameters does not reflect the true degrees of freedom of the system.

Conclusion

The purpose of this study is to identify both the most suitable modeling approach and the best-performing optimization algorithm to calibrate the parameters contained in metabolic network models. To this end, we constructed one probabilistic and six deterministic mathematical descriptions of valine and leucine biosynthesis in *C. glutamicum*. The parameters of each model were optimized with respect to *in vivo* measurements for the reacting species within the system. In this way we compared eight optimization procedures. We systematically benchmarked both the algorithms and the alternative models to highlight their advantages and drawbacks. In the following paragraphs, we draw several conclusions from the comparison of these seven variants of a realistic reaction system, and we assume them to hold for similar systems.

Table 5: Reduced reaction system and optimization results

Model	Replaced Parameter	Linear Regression Model	Mean Fitness ± Std. Dev.
GMAKr	k_{+1}	$a_1 \cdot K_3^{\mathrm{I}}$	21.4782 ± 0.3254
	k_{+3}	$a_2 \cdot K_3^{\mathrm{I}}$	
	k_{-1}	$a_3 \cdot K_3^{\mathrm{I}}$	
	k_{-3}	$a_4 \cdot K_3^{\mathrm{I}}$	
	k_{+4}	$a_5 \cdot k_{-4}$	
	k_{+7}	$a_6 \cdot k_{-7}$	
	k_{+9}	$a_7 \cdot K_{-9}$	
GMMr	k_{-4}	$b_1 \cdot k_{+4} + b_2 \cdot k_{+5}$	21.9538 ± 0.0832
	k_{+1}	$b_3 \cdot k_{-1} + b_4 \cdot K_1^{\mathrm{I}}$	
	k_{+7}	$b_5 \cdot k_{-7}$	
	k_{+8}	$b_6 \cdot k_{-8}$	
	k_{+9}	$b_7 \cdot k_{-9}$	
CKMMr	k_{+1}^{cat}	$c_1 \cdot K_{[\mathrm{Pyr}]_1}^{\mathrm{M}}$	22.7219 ± 0.6626
	$K_{[\mathrm{DHIV}]_1}^{\mathrm{M}}$	$c_2 \cdot K_{[\mathrm{NADP^+}]_1}^{\mathrm{M}}$	
	k_{+3}^{cat}	$c_3 \cdot K_{[\mathrm{KIV}]_1}^{\mathrm{M}}$	
	$K_{[\mathrm{KIC}]_2}^{\mathrm{M}}$	$c_4 \cdot K_{[\mathrm{KIV}]}^{\mathrm{M}} + c_5 \cdot K_{[\mathrm{Glut}]_2}^{\mathrm{M}}$	

The table lists the parameters that are replaced by linear regression models with coefficients a_i, b_i, c_i of other highly correlated model parameters.

Thus, if no prior knowledge about a comparable metabolic system is available, our results can serve as a starting point for model construction and calibration.

Let us consider the capabilities of the modeling approaches in more detail, when taking into account the ability to approximate measured data, the hybrid model for the reversible system based on convenience rate laws and Michaelis-Menten equations (CKMMr) has the best performance. At the same time, this is the most complex model with respect to the number of parameters and computational costs for each simulation. The acceptable parameter values for this model, found by multiple optimization runs, varied over several orders of magnitude. This corresponds to the fact that the optimization problem shows a large number of local optima, which is often referred to as multimodal behavior. Furthermore, this model is integrable with parameter values selected by chance from an almost arbitrarily wide range. This means that no preliminary data analysis in enzyme databases is required to obtain an integrable start population for the CKMMr model.

On the other hand, a simplified deterministic description of the reaction system based on the generalized

mass action rate law (GMAKr) yields good performance as well. Its advantage over the CKMMr model is its small number of parameters. Its major disadvantage is the strong tendency to become non-integrable when selecting parameter values by chance from a larger range. A restriction of the parameter space is required to ensure numerical stability when integrating metabolic network models. Some of the parameters showed almost no variance among the results of multiple optimization runs.

For models of biochemical systems with low metabolic concentrations or systems operating close to the point of instability, stochastic effects should be considered. Generally, simulating large stochastic models is computationally not feasible. The Langevin approach is a simplified stochastic description which facilitates taking these effects into account at acceptable computational costs. For the specific biochemical example network considered in this study, the stochastic effects are negligible as the behavior of the Langevin model is similar to the GMAKr model due to the rather large molecular populations. However, the benchmark showed that this approach is suitable for large-scale parameter optimization and model inference.

Modeling certain reactions in a non-reversible way as was done in all remaining models leads to a significantly worse ability to fit the measured data. We conclude that possible side effects are compensated by means of the reverse reaction. When modeling multi-enzyme systems all reactions should be treated reversibly, unless there is significant biological evidence to introduce irreversible rate laws. If neither the kinetic equations nor meaningful ranges of the parameter space are known, the model should be constructed using convenience kinetics. If the parameter space can be restricted using prior knowledge and the number of parameters matters, a model based on generalized mass-action rate laws constitutes an appropriate choice.

The second aim of this study is to identify the best-performing optimization algorithm for parameter estimation. The ability to find good local optima for the parameter values is the first quality measure for the algorithms. All five evolutionary algorithms tested yielded reasonable performance. From a user perspective, these algorithms differ in the number of settings which influence their behavior and are therefore more or less easy to apply. Hence, the effort to find a good configuration for an algorithm constitutes the second criterion of quality. The Tribes algorithm was among the best-performing algorithms in our benchmarks. As a settings-free optimization procedure, it is the most user-friendly method. However, other algorithms are able to

yield even better results after fine-tuning. Particularly, DE and PSO provided the best performance while keeping the effort necessary for their fine-tuning within reasonable limits. ES and binGA are also able to identify valuable local optima for all systems but require examining a large number of well-established alternative operators for their crossover and mutation steps.

For first optimization attempts, the easy to use Tribes algorithm is a good choice. With slightly more effort, the user can adjust the algorithms PSO and DE to yield even better results. Combined with the convenience kinetics modeling approach, these algorithms provide a suitable choice to model unknown systems of metabolic reactions.

Methods
The biochemical example network
Figure 1 illustrates the biosynthesis of Val and Leu in a process diagram [68, 69] according to the METACYC [8] and KEGG[6, 7] databases. The pathway starts with pyruvate (Pyr), from which Val and Leu are produced. Both products are used for biomass production or can be transported out of the cell if not needed. It is important to note that this pathway is regulated by both products in six feedback inhibition mechanisms. The transport of Leu and Val across the cell wall is actually performed by the same enzyme, so that both substrates compete with each other. However, for modeling purposes two distinct reactions are necessary in which the competition is included as inhibition. Some reactions are lumped together (Table 1) as suggested by Magnus *et al.* [52]. Since the reaction 2 IPM) $\rightleftharpoons$ 3 IPM is fast, it is assumed to be in equilibrium. This and the two following reactions 3 IPM + NAD$^+$ $\rightarrow$ 2 I$_3$OS + NADH$_2$ as well as (2S)-2-isopropyl-3-oxosuccinate (2 I$_3$OS) $\rightarrow$ 2-ketoiso-caproate (KIC) +CO$_2$ that only depend on the concentration of 2 IPM are lumped together, introducing the symbol IPM for both derivatives.

Glucose stimulus–response experiment
After a 10 min starvation period, a glucose pulse was added to the culture medium increasing the glucose concentration from 0 to 3.5 g/l. This glucose step-function induced a dynamic response from the metabolic intermediates linked to this central nutrient. Over a time span of 25 s, beginning 4 s before the glucose pulse, 47 samples were taken for 13 metabolites on the pathway starting at the state of Pyr, which is generated during phosphotransferase system-mediated glucose uptake and is also the final product of glycolysis. Immediate quenching and cooling with methanol to -30°C prevented the metabolites from further reactions. Mass spectrometry (HPLC MS/MS) was used to quantify

the metabolite concentrations in the probes. For details of this experiment we refer to Magnus *et al.* [52]. For technical reasons, $NADH_2$ and $NADPH_2$ as well as acetylCoA and CoA could not be measured with a high degree of exactness. Thus, Magnus *et al.* [52] suggested taking into account that both couples, NAD^+ and $NADH_2$ as well as $NADP^+$ and $NADPH_2$, follow a conservation relation so that the total amount of both coupled metabolites remains constant during the 25 s of interest. Thus, $NADH_2 = 0.8$ mM - $[NAD^+]$ and $[NADPH_2] = 0.04$ mM - $[NADP^+]$. We assume a constant pool of the other two central metabolites that does not vary over the considered time span. The steady-state concentrations (Table 6) of the seven metabolites to be simulated serve as initial values for the models.

Mathematical models

The rates of change of each metabolite's concentration over time can be calculated by linear combination of the stoichiometric matrix N describing the structure of the reaction system, i. e., its topology, with the vector of reaction velocities v that depends on the vector of reacting species S, the parameter vector p, and may also explicitly depend on time t:

$$\frac{d}{dt} S = Nv(S(t), t, p). \tag{1}$$

For the resulting seven-dimensional differential equation system [see Additional file 2]. The databases KEGG[6, 7] and METACYC[8] do not indicate if the reaction network pictured in Figure 1 contains irreversible reactions besides the draining of both products as listed in Table 1. One way to study the influence of the existence or non-existence of reverse reactions on the dynamics of the whole system is to derive alternative models and investigate their ability to approximate the data. The simpler irreversible reactions are favored if they are able to fit the data. The following paragraphs present the general equations for all rate laws, which are inserted for each v_i and investigated in this study. For the resulting differential equation systems [see Additional

Table 6: Steady-state concentrations of the reacting species

Metabolite	Concentration (mM)
AcLac	0.236
DHIV	0.132
IPM	0.0227
KIC	0.0741
Leu	0.209
Val	29.4

The data in this table serve as the initial values of all differential equation systems derived herein.

file 2]. An implementation of the fourth-order Runge-Kutta method [70] solves the ordinary differential equation systems. The stochastic Langevin system is adapted to be integrated using the MATLAB™ integrator ode15s [71, 72].

Generalized Mass Action Kinetics (GMAKr)

The simplest rate law is mass action kinetics, in which the effects of the participating enzymes are hidden in the rate constants. To include inhibition effects, we apply an inhibition function that fits the generalized mass action rate law [5, 73]:

$$v_j(S, p) = F_j(S, p) \left(k_{+j} \prod_i S_i^{n_{ij}^-} - k_{-j} \prod_i S_i^{n_{ij}^+} \right). \tag{2}$$

The function $F_j(S, p)$ must be a positive function of the substrate concentrations S and the parameter vector p to introduce saturation or inhibition effects to the common mass action rate law written in brackets [5]. For convenience of notation, the matrices $N^{\pm}$ are introduced, whose elements $n_{ij}^{\pm}$ express the absolute values of the positive or negative stoichiometric coefficients. Feedback inhibition loops are included using the following approaches:

$$F_j(S, p) = \frac{1}{1 + K_j^I \cdot [I]} \tag{3}$$

$$F_j(S, p) = \exp(-K_j^I \cdot [I]) \tag{4}$$

with $K_j^I \geq 0$. While Equation (4) is derived intuitively, driven by the assumption that the exponential function constitutes an important growth and shrinkage function in biology, Equation (3) can be derived from the competing reactions of the enzyme with its substrate or inhibitor. The first equation can be derived using the equilibrium constant for the inhibition reaction and the conservation law of the enzyme as well as the enzyme-inhibitor complex concentrations. Applying Equation (2) combined with Equation (3) to reaction system R_1 through R_{10} leads to an Ordinary Differential Equation (ODE) system with 24 parameters $k_{\pm j}$, K_j^I.

Irreversible GMAK with exp inhibition (GMAKi)

By setting all product concentrations apart from R_2 and R_9 to zero and applying Equation (4) to Equation (2) we obtain the irreversible version of this equation system with 18 parameters $k_{\pm j}$, K_j^I.

Michaelis-Menten equations (GMMr)

Three reactions of the system (R_3, R_6, and R_{10}, Table 1) follow a Michaelis-Menten mechanism. The

corresponding rate law is given by Equation (5), where S forms product P and the catalyst E is inhibited by I:

$$v_j = \frac{\dfrac{v_+^{\mathrm{m}}}{K_{\mathrm{S}}^{\mathrm{M}}}[\mathrm{S}] - \dfrac{v_-^{\mathrm{m}}}{K_{\mathrm{P}}^{\mathrm{M}}}[\mathrm{P}]}{1 + \dfrac{[\mathrm{I}]}{K^{\mathrm{I}a}} + \left(\dfrac{[\mathrm{S}]}{K_{\mathrm{S}}^{\mathrm{M}}} + \dfrac{[\mathrm{P}]}{K_{\mathrm{P}}^{\mathrm{M}}}\right)\left(1 + \dfrac{[\mathrm{I}]}{K^{\mathrm{I}b}}\right)}. \tag{5}$$

In the case of R_3 there might be a reverse reaction. R_6 and R_{10} are assumed to be irreversible because they describe the transport of Val and Leu out of the cell. We further assume that the constants v^{m} in R_6 and R_{10} are allowed to be zero so that there is no need to export Val or Leu if it is needed for biomass formation. All other reactions are modeled using the GMAKr approach including Equation (3) for inhibition. The complete GMMr model contains 31 parameters to be estimated. To avoid numerical problems, the inhibition constants in Michaelis-Menten kinetics are transformed into their reciprocals $K^{\mathrm{I}a|b\prime} = \frac{1}{K^{\mathrm{I}a|b}}$. This modification allows the optimization procedure to "decide" which kind of inhibition occurs [5] [see Additional file 2].

Irreversible Michaelis-Menten Model (GMMi)
From the GMMr model an irreversible alternative is established by setting all product concentrations to zero. The resulting system contains 24 parameters $K_j^{\mathrm{I}a|b}$, $k_{\pm j}$, K_{ij}^{M} to be estimated.

Reversible Convenience Kinetics (CKMMr)
The general equation of the convenience rate law for reaction j reads

$$v_j = [\mathrm{E}_j]\prod_m h_{\mathrm{A}}(S_m, K_{jm}^{\mathrm{A}})^{w_{jm}^+} h_{\mathrm{I}}(S_m, K_{jm}^{\mathrm{I}})^{w_{jm}^-} \cdot \frac{k_{+j}^{\mathrm{cat}}\prod_i\left(\dfrac{S_i}{K_{ji}^{\mathrm{M}}}\right)^{n_{ij}^-} - k_{-j}^{\mathrm{cat}}\prod_i\left(\dfrac{S_i}{K_{ji}^{\mathrm{M}}}\right)^{n_{ij}^+}}{\prod_i \sum_{m=0}^{n_{ij}^-}\left(\dfrac{S_i}{K_{ji}^{\mathrm{M}}}\right)^m + \prod_i \sum_{m=0}^{n_{ij}^+}\left(\dfrac{S_i}{K_{ji}^{\mathrm{M}}}\right)^m - 1} \tag{6}$$

with h_{A} and h_{I} being functions for activation or inhibition, respectively, the turnover rates $k_{\pm j}^{\mathrm{cat}}$ and the matrices $\mathbf{W}^{\pm}$ containing positive entries for the connectivity of the modulating metabolites as well as K_{ji}^{M} being a constant analogous to the Michaelis constant K^{M} [16]. For inhibition, which plays an important role in Val and Leu biosynthesis of *C. glutamicum*,

$$h_{\mathrm{I}}(S_i, K^{\mathrm{I}}) = \frac{K^{\mathrm{I}}}{K^{\mathrm{I}} + S_i} = \frac{1}{1 + \dfrac{S_i}{K^{\mathrm{I}}}} = \frac{1}{1 + K^{\mathrm{I}\prime}S_i} \tag{7}$$

has been suggested and herein applied. Besides the reciprocal constant this approach equals Equation (3).

The product $[\mathrm{E}_j]\, k_{\pm j}^{\mathrm{cat}}$ is lumped into one parameter $V_{\pm j}^{\mathrm{m}}$ for all j assuming that all enzyme concentrations remain constant during the 25 s. No enzyme concentrations have been measured, so that an optimizer cannot distinguish between the product of two parameters or one parameter. The three reactions that follow the Michaelis-Menten mechanism are modeled using Equation (5). The reactions R_6 and R_{10} are considered irreversible as described before. Applying Equation (6) to all remaining reactions in the system R_1 through R_{10} yields an equation system with 59 parameters. The stoichiometric matrix has full column rank. Hence, the parameters $k_{\pm j}^{\mathrm{cat}}$ can be estimated directly without violating thermodynamic constraints [16].

Irreversible Convenience Kinetics (CKMMi)
By setting all product concentrations apart from R_2 and R_9 to zero, we obtain an irreversible version of this model containing 41 parameters.

Stochastic modeling based on the Langevin equation (LANG)
The concentration variables S_i are replaced by the random variables $X_i(t) \equiv$ number of S_i molecules in the system at time t, $i = 1,...,N$ in an enclosing reaction volume V, where the N species interact through M reaction channels R_j, $j = 1,...,M$. Each reaction is characterized by a stochastic rate constant c_j depending only on the physical properties of the reacting molecules [11]. In the case of large systems with high metabolite concentration these simulation strategies are highly computationally intensive and therefore unsuited for large-scale parameter optimization. However, for macroscopic systems it is possible to directly approximate the time evolution of the stochastic state variables by the chemical Langevin equation [10, 11], which reads

$$\mathrm{d}x_i(t) = \sum_{j=1}^{M} n_{ij}a_j(\mathbf{x}(t)) + \sum_{j=1}^{M} n_{ij}\sqrt{a_j(\mathbf{x}(t))}\mathrm{d}W_j, \; i = 1,...,N \tag{8}$$

when rewritten using the Wiener process [74] for easier numerical treatment. Here n_{ij} represents the stoichiometric coefficient of the ith metabolite in the jth reaction. The propensity a_j is defined as: $a_j\mathrm{d}t = c_jh_j\mathrm{d}t \equiv$ probability that an R_j reaction will occur in V in $(t, t + \mathrm{d}t)$, given that the system is in state $(X_1,...,X_N)$ at time t. The function h_j gives the number of distinct R_j molecular reactant combinations available in the state $(X_1,...,X_N)$, $j = 1,..., M$. The discrete variables $(X_1,...,X_N)$ are replaced by the continuous molecule concentrations $(x_1,...,x_N)$. In order to numerically integrate the Langevin equation with standard ODE solvers, the equation is split into a stochastic and a deterministic term. The deterministic

term and the deterministic part of the stochastic term are treated like ODEs, as suggested by Bentele *et al.* [75]:

$$\Delta\hat{x}_i(t) = \sum_{j=1}^{M} n_{ij} a_j(\mathbf{x}_t)\Delta t \qquad (9)$$

$$\Delta\tilde{x}_i(t) = \sum_{j=1}^{M} n_{ij} \sqrt{a_j(\mathbf{x}(t))\Delta t}. \qquad (10)$$

The latter term is then multiplied by a normal random variable $n_i = \mathcal{N}$ in analogy to the finite Wiener increments used in the Euler-Maruyama method [74]. After each time-step, both terms are added to give the full state-variable change:

$$\Delta x_i(t) = \Delta x_i(t) + \Delta\tilde{x}_i(t) \cdot n_i \qquad (11)$$

allowing adaptive step-size control of the ODE solver. The reaction propensities are calculated according to Gillespie [11]. This leads to an equation system with 24 parameters.

Representing external metabolites with splines

As suggested by Magnus *et al.* [52], metabolites whose concentrations cannot be explained in terms of the model are considered external, i. e., they are an input to the model but involved in numerous other reactions that are not reflected by this system (Figure 1). We approximate these using cubic splines which smooth the measurements. To weight all measurements equally, all ω_i are set to 1. Due to the different ranges of the concentrations of the six metabolites, it is not possible to find one appropriate degree of smoothness λ that leads to equally smooth curves. Hence, we transform all concentrations into the range [0, 1], set $\lambda = 1$, compute the spline coefficients and re-transform them back into the original range of the specific metabolite. For details and a figure showing the resulting splines [see Additional file 2].

Fitness function and search-space restrictions

Since many distance measurements have been defined, the choice of the most appropriate one is an important step for the parameter estimation process. Due to the differences in the concentrations of certain metabolites the Euclidian distance between the model values and the measurements is not applicable: metabolites in higher concentration would dominate in fitness over those in lower concentration. Minimizing the relative squared error (RSE, see Equation (12)) overcomes this limitation. The outer sum runs over all dimensions of $\hat{\mathbf{x}}$ describing the model output at each sample time τ_t. T is the number of measurements taken and $\mathbf{X} = (x_{ti})$ is the given data matrix.

$$f_{\mathrm{RSE}}(\mathbf{x}, \mathbf{X}) = \sum_{i=1}^{\dim(\mathbf{x})} \sum_{t=1}^{T} \left(\frac{x_i(\tau_t) - x_{ti}}{x_{ti}} \right)^2. \qquad (12)$$

This fitness function was used in several publications for similar problems [33, 47, 48].

To restrict the search space for the optimizers, we limit parameter values to the range [0, 2000], covering 98.748% of all known kinetic parameters in BRENDA [24-26], as suggested in [47, 48]. All known parameters in BRENDA are greater than or equal to zero. To avoid division by zero in some parameters, the range is set to [ε, 2000] with $\varepsilon = 10^{-8}$. For parameters transformed in Michaelis-Menten equations (as described above), the range is limited to [0, 10^8], resulting in a search space from 10^{-8} through ∞. Only 0.962% of all K^{I} and 0.004% of all K^{M} values in BRENDA are reported to be lower than 10^{-8}. This ε is chosen to guarantee numerical stability. These restrictions are applied to all parameters. In cases where no division by the parameter value is necessary, $\varepsilon = 0$ is allowed to be a lower bound of the parameter range.

All parameters are initialized with low values to avoid obtaining unstable initial populations and it is assumed that large parameter values are rather infrequent in nature [47, 48]. A Gaussian distribution with $\mu = \sigma = 1$ guarantees low initial values and ensures stable initial populations. Each parameter is set to the boundary values if it breaks any of the search-space restrictions. We limit the initialization procedure for all models to a low value initialization to ensure equal conditions in all comparisons.

Systematically improving the performance of the optimization procedures

Standard settings for the optimization algorithms

For a comprehensive introduction to all optimization algorithms used in this study [see Additional file 1], in which we also explain the specific settings and operators for each method in detail.

Using the open-source framework EVA2 for nature-inspired optimization procedures [37, 38], we test the following standard settings of the algorithms (Table 2) on the inference problem, of which the following are evolutionary optimization procedures:

• Binary Genetic Algorithm (binGA) with one-point mutation, $p_m = 0.1$, and one-point crossover, $p_c = 0.7$.

• Real-valued Genetic Algorithm (realGA) with global mutation, $p_m = 0.1$, and UNDX crossover, $p_c = 0.8$. Both GAs use tournament selection with a group size of 8 in a population of 250 individuals.

• Standard Evolution Strategy (stdES) as (5,25)-ES with global mutation, p_m = 0.8, and discrete one-point crossover, p_c = 0.2.

• Evolution Strategy with covariance matrix adaption with and without elitism, i. e., "plus" strategy, (cmaES/cmaESplus) with μ = 5 and λ = 25, p_m = 1.0, and no crossover. All ESs use deterministic best-first selection to choose the next generation.

• Differential Evolution (DE) with the scheme DE/current-to-best/1 [42] setting λ = F = 0.8, CR = 0.5 and a population size of 100.

Two optimization strategies are swarm intelligence optimization procedures:

• Constricted Particle Swarm Optimization (PSO) [45, 76]: setting φ_1 = 2.05, φ_2 = 2.05, χ = 0.73 using star topology and a population size of 100 as well as its derivative

• Tribes [46]

We also study the following classical non-evolutionary methods:

• Monte Carlo Optimization (MCO) with 50 multi-runs

• (Multi-start) Hill Climber (HC), the number of starts varying from 1, 10, 25, 50, 100 to 250 using Gaussian mutation with a fixed standard deviation of σ = 0.2 and a mutation probability p_m = 1.0.

• Simulated Annealing (SA) with α = 0.1 and an initial temperature of T = 5 using a linear annealing schedule and a population size of 250.

For all algorithms with population sizes lower than 250 individuals, a pre-population with 250 parameter vectors is generated and the best are selected to create the initial population. This step is crucial to obtain comparable results for algorithms with different population sizes [33]. Every setting described in this and all following paragraphs is repeated 20 times with 100,000 fitness evaluations per run on the deterministic models. The results of this preliminary trial are shown in Table 3 and Figure 3. The methods found to be successful in these analyses are utilized to optimize the Langevin model as well.

Alternative settings for the best optimization algorithms
We first study the influence of various mutation and crossover operators on the three deterministic reversible models for binGA and ES. In a grid search, all combinations of the following operators are evaluated,

and we exclude the combinations of no crossover with no mutation. For binGA, no mutation, one-point, and adaptive mutation, an operator which modifies individual mutation probabilities similar to ES step-size adaptation, is tested, paired with one- and n-point (n = 3), uniform, and bit-simulated crossover, each with p_m = 0.1, p_c = 0.7. On ES we use the mutation operators covariance matrix adaptation (CMA), the $1/5^{th}$ success rule as well as correlated, global, local, and no mutation paired with the crossover operators one- and n-point (n = 3), UNDX, and no crossover, each with p_m = 0.8, p_c = 0.2. To study the influence of the mutation or the crossover operator alone, the values for p_m and p_c are set to zero or one, depending on which influence is investigated. The population size for binGA is set to 100 and for ES (5, 25) (Figure 4). For the most successful operator combination (adaptive mutation with bit-simulated crossover) on the GMAKr model, we study the influence of the probabilities p_m and p_c, with which the respective operator is invoked by binGA. All pairs of pm and pc are evaluated from 0.0 through 1.0 in 0.1 steps and a population size of 100, excluding p_m = p_c = 0.0 (Figure 5(a)). Subsequently, the population sizes 50, 100, 250, 500, 1,000, and 2,000 are tested for binGA with p_m = 0.2 and p_c = 1.0 (Figure 5(b)). For the cmaESplus, we evaluate all combinations of $\mu \in$ {5, 10, 25, 50, 75}, and $\lambda \in$ {10, 25, 50, 75, 100, 125, 150}, excluding cases where $\mu > \lambda$ and keeping p_m at 1.0, and p_c at 0.0 (Figure 6).

For the DE approach, another grid search is performed on the three reversible deterministic models, altogether testing values for F, $\lambda \in$ {0.5, 0.8}, and CR $\in$ {0.3, 0.5, 0.9} (Figure 7). For the most promising parameter set (F = 0.8, λ = 0.5, and CR = 0.3 for GMAKr and GMMr, and CR = 0.9 for the CKMMr), the population size is additionally varied over {50, 250, 500, 1000} for each model (Figure 8).

The PSO with star topology and standard settings for φ_1 = φ_2 = 2.05 is compared with a star topology and settings φ_1 = 2.8, φ_2 = 1.3, as suggested in [77] as well as a linear 3 and grid 3 topology with standard parameters on all three reversible deterministic models (Figure 9). The population size is set to 25. Additional population sizes, namely {50, 250, 500}, are evaluated using a grid 3 topology and standard values for φ_1 and φ_2 on these models (Figure 10).

Parameter-space analysis
The completion of the large-scale parameter optimization study in the first part yields a large set of different optimal parameters. We select all parameter sets with a fitness less than 25 for each reversible deterministic model (GMAKr, GMMr, and CKMMr).

Cluster analysis
To cluster these best parameter sets (Figure 12), we apply the agglomerative nesting algorithm (AGNES) [[78], pp. 199–251], using a Euclidean metric, which is implemented in the cluster package [79] of the R-project [80].

Variance analysis to visualize the scattering of each parameter
The variances of each parameter among the best optimization results of the three reversible deterministic models are calculated and plotted on a logarithmic scale due to the large differences in their orders of magnitude (Figure 13).

Multiple correlation analysis
Multiple correlation $r_{Y,(X_1,...,X_p)}$, measures the dependency of one model parameter Y on p other parameters $X_1,...,X_p$ of the model. Here, $r_{Y,(X_1,...,X_p)}$ is defined as the largest simple correlation among the correlations of Y and all linear combinations of the X_i, that is $\sum_{i=1} a_i X_i$, with arbitrary weights a_i. A large value of $r_{Y,(X_1,...,X_p)}$ indicates a strong dependence of Y on other model parameters and suggests that Y is not a genuine degree of freedom of the model. In order to calculate the multiple correlation of Y and $X_1,...,X_p$ within a sample of size n different runs for each parameter, all simple correlations r_{Y,X_i} must be determined using the Pearson correlation coefficient r_{Y,X_i} [81]:

$$r_{Y,X_i} = \frac{\sum_{i=1}^{n} x_i y_i - n \bar{x} \bar{y}}{\sqrt{\left(\sum_{i=1}^{n} x_i^2 - n \bar{x}^2\right)\left(\sum_{i=1}^{n} y_i^2 - n \bar{y}^2\right)}}. \quad (13)$$

A correlation of ± 1 means that there is a perfect positive/negative linear relationship between the parameters Y and X. In that case, the parameter Y can be explained by parameter X and therefore be omitted. The multiple correlation is then established from all of these simple correlations [81]:

$$r_{Y,(X_1,...,X_p)} = \sqrt{r_{YX}^T R_{(XX)}^{-1} r_{YX}} = \left([r_{YX_1},...,r_{Y,X_p}] \begin{bmatrix} 1 & r_{X_1X_2} & \cdots & r_{X_1X_p} \\ r_{X_1X_2} & 1 & \cdots & r_{X_2X_p} \\ \vdots & \vdots & \ddots & \vdots \\ r_{X_1X_p} & r_{X_2X_p} & \cdots & 1 \end{bmatrix}^{-1} \begin{bmatrix} r_{YX_1} \\ r_{YX_2} \\ \vdots \\ r_{YX_p} \end{bmatrix} \right)^{\frac{1}{2}}. \quad (14)$$

The coefficient of determination

$$B = r_{Y,(X_1,...,X_p)}^2 \quad (15)$$

measures how well the paremeter Y can be explained in a linear sense by the other model parameters $X_1,...,X_p$. After identification of highly correlated parameters, a subset of replacement candidates is selected. A parameter X is considered a replacement candidate for parameter Y if $r_{X,Y} \geq 0.7$ and the p-value of the correlation, computed using a

t-test, is close to zero. The degrees of freedom selected for replacement are subsequently substituted by a linear regression model of their correlated parameters.

Hardware configuration
All experiments are run on a cluster with 16 AMD dual Opteron CPUs with 2.4 GHz, 1 MB level 2 cache, and 2 GB RAM per node under the Sun Grid Engine, and JVM 1.5.0 with Scientific Linux 5 as operating system. An optimization of one model in 20 parallel multi-runs takes approximately 1.5 h.

Availability of models and optimization procedures
All models investigated in this study are included as optimization problems in EvA2, a Java™-based workbench for heuristic optimization [37, 38], which can be downloaded at http://www.ra.cs.uni-tuebingen.de/software/EvA2.

Authors' contributions
AD developed the conceptual idea, created all deterministic models, and wrote Additional file 2 together with MJZ. MK is the main developer of EvA2 and performed the optimization parts of this study together with AD. MK selected and implemented the algorithms to be tested, suggested alternative settings for the optimizers, and wrote Additional file 1. MJZ created and optimized the stochastic model, and performed all parameter space analyses. AD, MK, MJZ, and JS wrote this manuscript. HP implemented the integrator and parts of EvA2, and supported the optimization. JM and MO performed the *in vivo* measurements for the model system. OK and AZ supervised the work. All authors read and approved the final manuscript.

Additional material

Additional file 1
Nature-inspired heuristics for optimization. This document provides a detailed introduction to the basic concepts of all evolutionary and swarm intelligence algorithms that are used in this study.
Click here for file
[http://www.biomedcentral.com/content/supplementary/1752-0509-3-5-S1.pdf]

Additional file 2
Modeling the valine and leucine metabolism in Corynebacterium glutamicum. This document gives a comprehensive introduction to mathematical modeling of biochemical systems and all four kinds of rate laws that are used in this study. Additionally, the method to introduce inhibition effects to the generalized mass action rate law is derived in this document. Histograms are presented showing the distribution of the parameters. For the best-performing optimization algorithms, the fitness is plotted depending on the evaluations of the objective function.
Click here for file
[http://www.biomedcentral.com/content/supplementary/1752-0509-3-5-S2.pdf]

Acknowledgements

References

1. Kitano H: **Computational systems biology.** *Nature* 2002, **420 (6912)**:206–210.
2. Albert R: **Network Inference, Analysis, and Modeling in Systems Biology.** *Plant Cell* 2007, **19(11)**:3327–3338.
3. Gombert AK and Nielsen J: **Mathematical modelling of metabolism.** *Current Opinion in Biotechnology* 2000, **11(2)**:180–186.
4. Covert MW, Schilling CH, Famili I, Edwards JS, Goryanin II, Selkov E and Palsson BO: **Metabolic modeling of microbial strains *in silico*.** *Trends in Biochemical Sciences* 2001, **26(3)**:179–186.
5. Heinrich R and Schuster S: *The Regulation of Cellular Systems. 115 Fifth Avenue New York, NY 10003* Chapman and Hall; 1996.
6. Kanehisa M, Goto S, Hattori M, Aoki-Kinoshita KF, Itoh M, Kawashima S, Katayama T, Araki M and Hirakawa M: **From genomics to chemical genomics: new developments in Kᴇɢɢ.** *Nucl Acids Res* 2006, **34**:D354–357.
7. Ogata H, Goto S, Sato K, Fujibuchi W, Bono H and Kanehisa M: **KEGG: Kyoto Encyclopedia of Genes and Genomes.** *Nucleic Acids Research* 2000, **27**:29–34 http://nar.oxfordjournals.org/cgi/content/abstract/27/1/29.
8. Caspi R, Foerster H, Fulcher CA, Hopkinson R, Ingraham J, Kaipa P, Krummenacker M, Paley S, Pick J, Rhee SY, Tissier C, Zhang P and Karp PD: **MetaCyc: a multiorganism database of metabolic pathways and enzymes.** *Nucleic Acids Research* 2006, 34 Database: D511–D516.
9. Bulik S, Grimbs S, Selbig J and Holzhütter HG: **Combining mechanistic and simplified enzymatic rate equations: A promising approach for speeding up the kinetic modeling of complex metabolic networks.** *FEBS Journal* 2009, **276**:410–524.
10. Gillespie DT: **The chemical Langevin equation.** *Journal of Chemical Physics* 2000, **113**:297–306 http://link.aip.org/link/?JCPSA6/113/297/1.
11. Gillespie DT: **A General Method for Numerically Simulating the Stochastic Time Evolution of Coupled Chemical Reactions.** *Journal of Computational Physics* 1976, **22(4)**:403–434 http://www.sciencedirect.com/science/article/B6WHY-4DD1NC9-CP/2/43ade5f11fb949602b3a2abdbbb29f0e.
12. Visser D and Heijnen J: **The Mathematics of Metabolic Control Analysis Revisited.** *Metabolic Engineering* 2002, **4**:114–123 http://www.sciencedirect.com/science/article/B6WN3-45V802C-3/2/d624a20d0e70ca2a1058359d7fd00cb0.
13. Hatzimanikatis V, Floudas CA and Bailey JE: **Analysis and design of metabolic reaction networks via mixed-integer linear optimization.** *AIChE* 1996, **42(5)**:1277–1292 http://cat.inist.fr/?aModele=afficheN&cpsidt=3105838.
14. Hatzimanikatis V and Bailey JE: **Effects of spatiotemporal variations on metabolic control: Approximate analysis using (log)linear kinetic models.** *Biotechnology and Bioengineering* 1997, **54(2)**:91–104 http://www3.interscience.wiley.com/journal/71003853/abstract.
15. Hatzimanikatis V, Emmerling M, Sauer U and Bailey JE: **Application of mathematical tools for metabolic design of microbial ethanol production.** *Biotechnology and Bioengineering* 1998, **58 (2)**:154–161 http://www3.interscience.wiley.com/journal/71002326/abstract.
16. Liebermeister W and Klipp E: **Bringing metabolic networks to life: convenience rate law and thermodynamic constraints.** *Theor Biol Med Model* 2006, **3**:41.
17. Segel IH: *Enzyme Kinetics – Behavior and Analysis of Rapid Equilibrium and Steady-State Enzyme Systems* Wiley Classics Library Edition; 1993.
18. Bisswanger H: *Enzymkinetik – Theorie und Methoden* Weinheim, Germany: Wiley-VCH; 32000.
19. Cornish-Bowden A: *Fundamentals of Enzyme Kinetics* 59 Portland Place, London: Portland Press Ltd; 32004.
20. Gillespie DT: **Stochastic simulation of chemical kinetics.** *Annu Rev Phys Chem* 2007, **58**:35–55.
21. Savageau MA: **Biochemical systems analysis. 3. Dynamic solutions using a power-law approximation.** *J Theor Biol* 1970, **26(2)**:215–226.
22. Visser D and Heijnen JJ: **Dynamic simulation and metabolic redesign of a branched pathway using linlog kinetics.** *Metab Eng* 2003, **5(3)**:164–176.
23. Hatzimanikatis V and Bailey J: **MCA Has More to Say.** *Journal of theoretical Biology* 1996, 233–342.
24. Barthelmes J, Ebeling C, Chang A, Schomburg I and Schomburg D: **BRENDA, AMENDA and FRENDA: the enzyme information system in 2007.** *Nucl Acids Res* 2007, **35(suppl_1)**:D511–514 http://nar.oxfordjournals.org/cgi/content/abstract/35/suppl_1/D511.
25. Schomburg I, Chang A and Schomburg D: **BRENDA, enzyme data and metabolic information.** *Nucl Acids Res* 2002, **30**:47–49.
26. Schomburg I, Chang A, Ebeling C, Gremse M, Heldt C, Huhn G and Schomburg D: **BRENDA, the enzyme database: updates and major new developments.** *Nucleic Acids Research* 2004, **32 (Database Issue)**:D431–433.
27. Metzler DE: *Biochemistry* Harcourt/Academic Press; 2001.
28. Banga JR: **Optimization in computational systems biology.** *BMC Systems Biology* 2008, **2**:47.
29. Rodriguez-Fernandez MR, Mendes P and Banga JR: **A hybrid approach for efficient and robust parameter estimation in biochemical pathways.** *Biosystems* 2006, **83**:248–265.
30. Rodriguez-Fernandez M, Egea JA and Banga JR: **Novel metaheuristic for parameter estimation in nonlinear dynamic biological systems.** *BMC Bioinformatics* 2006, **7**:483.
31. Balsa-Canto E, Peifer M, Banga JR, Timmer J and Fleck C: **Hybrid optimization method with general switching strategy for parameter estimation.** *BMC Systems Biology* 2008, **2**:26.
32. Spieth C, Streichert F, Speer N and Zell A: **Optimizing Topology and Parameters of Gene Regulatory Network Models from Time-Series Experiments.** *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2004), (Part I) of LNCS* 2004, **3102**:461–470 http://www.springerlink.com/content/cx9mmtkl2ca0fcx4/.
33. Spieth C, Worzischek R, Streichert F, Supper J, Speer N and Zell A: **Comparing Evolutionary Algorithms on the Problem of Network Inference.** *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2006)* 2006 http://portal.acm.org/citation.cfm?id=1143997.1144052.
34. Spieth C, Hassis N, Streichert F, Supper J, Beyreuther K and Zell A: **Comparing Mathematical Models on the Problem of Network Inference.** *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2006)* 2006 http://portal.acm.org/citation.cfm?id=1143997.1144045.
35. Weise T: *Global Optimization Algorithms – Theory and Application* Thomas Weise; 22008 http://www.it-weise.de/.
36. Charbonneau P and Knapp B: *A User's Guide to PIKAIA 1.0* 1995 http://download.hao.ucar.edu/archive/pikaia/userguide.ps.
37. Streichert F and Ulmer H: **JavaEvA – A Java Framework for Evolutionary Algorithms.** *Technical Report WSI-2005–06, Center for Bioinformatics Tübingen* University of Tübingen; 2005 http://w210.ub.uni-tuebingen.de/dbt/volltexte/2005/1702/.
38. Kronfeld M: *EvA2 Short Documentation* University of Tübingen, Dept. of Computer Architecture, Sand 1, 72076 Tübingen; 2008 http://www.ra.cs.uni-tuebingen.de/software/EvA2.
39. Mathtools.net : MATLAB/Optimization. http://www.mathtools.net/MATLAB/Optimization.
40. Holland JH: *Adaptation in Natural and Artificial Systems* The University of Michigan Press; 1975.
41. Rechenberg I: *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution* Fromman-Holzboog, Stuttgart; 1973.
42. Storn R: **On the Usage of Differential Evolution for Function Optimization.** *1996 Biennial Conference of the North American Fuzzy Information Processing Society* Berkeley: IEEE, New York, USA; 1996, 519–523 http://www.icsi.berkeley.edu/~storn/bisc1.ps.gz.
43. Tovey CA: **Hill climbing with multiple local optima.** *Alg Disc Meth* 1985, **6(3)**:384–393 http://link.aip.org/link/?SML/6/384/1.
44. Kirkpatrick S, Gelatt CD and Vecchi MP: **Optimization by Simulated Annealing.** *Science* 1983, **220(4598)**:671–680 http://www.sciencemag.org/cgi/content/abstract/220/4598/671.
45. Clerc M and Kennedy J: **The Particle Swarm – Explosion, Stability, and Convergence in a Multidimensional Complex Space.** *IEEE Transactions on Evolutionary Computation* 2002, **6**:58–73 http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=985692.
46. Clerc M: *Particle Swarm Optimization* ISTE Ltd; 2005.
47. Dräger A, Supper J, Planatscher H, Magnus JB, Oldiges M and Zell A: **Comparing Various Evolutionary Algorithms on the Para-**

meter Optimization of the Valine and Leucine Biosynthesis in *Corynebacterium glutamicum*. *2007 IEEE Congress on Evolutionary Computation* IEEE Computational Intelligence Society, Singapore: IEEE Press: Srinivasan D, Wang L 2007, 620–627 http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4424528.

48. Dräger A, Kronfeld M, Supper J, Planatscher H, Magnus JB, Oldiges M and Zell A: **Benchmarking Evolutionary Algorithms on Convenience Kinetics Models of the Valine and Leucine Biosynthesis in *C. glutamicum*.** *2007 IEEE Congress on Evolutionary Computation* IEEE Computational Intelligence Society, Singapore: IEEE Press: Srinivasan D, Wang L 2007, 896–903 http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4424565.

49. Chassagnole C, Noisommit-Rizzi N, Schmid JW, Mauch K and Reuss M: **Dynamic Modeling of the Central Carbon Metabolism of *Escherichia coli*.**Wiley Periodicals, Inc; 2002, 54–73 http://www3.interscience.wiley.com/journal/93519745/abstract.

50. Klipp E, Nordlander B, Kruger R, Gennemark P and Hohmann S: **Integrative model of the response of yeast to osmotic shock.** *Nature Biotechnology* 2005, **23(8):**975–982.

51. Guthke R, Schmidt-Heck W, Pless G, Gebhardt R, Pfaff M, Gerlach JC and Zeilinger K: **Dynamic Model of Amino Acid and Carbohydrate Metabolism in Primary Human Liver Cells.** *VII International Symposium on Biological and Medical Data Analysis* 2006 http://www.springerlink.com/content/f78167p04n1426w6/.

52. Magnus JB, Hollwedel D, Oldiges M and Takors R: **Monitoring and Modeling of the Reaction Dynamics in the Valine/Leucine Synthesis Pathway in *Corynebacterium glutamicum*.** *Biotechnology Progress* 2006, **22(4):**1071–1083.

53. Guthke R, Zeilinger K, Sickinger S, Schmidt-Heck W, Buentemeyer H, Iding K, Lehmann J, Pfaff M, Pless G and Gerlach JC: **Dynamics of amino acid metabolism of primary human liver cells in 3D bioreactors.** *Bioprocess Biosystem Engineering* 2006, **28(5):**331–340.

54. Savageau MA: **Biochemical systems analysis. I. Some mathematical properties of the rate law for the component enzymatic reactions.** *J Theor Biol* 1969, **25(3):**365–369.

55. Savageau MA: **Biochemical systems analysis. II. The steady-state solutions for an *n*-pool system using a power-law approximation.** *J Theor Biol* 1969, **25(3):**370–379.

56. Liu PK and Wang FS: **Inference of Biochemical Network Models in S-System Using Multi-Objective Optimization Approach.** *Bioinformatics* 2008, **24(8):**1085–1093.

57. Vera J, Sun C, Oertel Y and Wolkenhauer O: **PLMaddon: a power-law module for the Matlab™ SBToolbox.** *Bioinformatics* 2007, **23(19):**2638–2640.

58. Shapiro BE, Levchenko A, Meyerowitz EM, Wold BJ and Mjolsness ED: **Cellerator: extending a computer algebra system to include biochemical arrows for signal transduction simulations.** *Bioinformatics* 2002, **19(5):**677–678.

59. Hoops S, Sahle S, Gauges R, Lee C, Pahle J, Simus N, Singhal M, Xu L, Mendes P and Kummer U: **COPASI-a COmplex PAthway SImulator.** *Bioinformatics* 2006, **22(24):**3067–3074.

60. Dräger A, Hassis N, Supper J, Schröder A and Zell A: **SBMLsqueezer: a CellDesigner plug-in to generate kinetic rate equations for biochemical networks.** *BMC Systems Biology* 2008, **2:**39.

61. Yang CR, Shapiro BE, Mjolsness ED and Hatfield GW: **An enzyme mechanism language for the mathematical modeling of metabolic pathways.** *Bioinformatics* 2004, **21(6):**774–780.

62. Borger S, Liebermeister W, Uhlendorf J and Klipp E: **Automatically generated model of a metabolic network.** *International Conference on Genome Informatics* 2007, **18:**215–224 http://eproceedings.worldscinet.com/9781860949920/9781860949920_0021.html.

63. Jamshidi N and Palsson BO: **Formulating genome-scale kinetic models in the post-genome era.** *Molecular Systems Biology* 2008, **4:**.

64. Voss HU, Timmer J and Kurths J: **Nonlinear Dynamical System Identification from Uncertain and Indirect Measurements.** *International Journal of Bifurcation and Chaos* 2004, **14(6):**1905–1933 http://www.worldscinet.com/cgi-bin/details.cgi?id=pii:S0218127404010345&type=html.

65. Eggeling L and Bott M: *Handbook of Corynebacterium glutamicum* Boca Raton: Taylor & Francis; 2005.

66. Hansen N and Ostermeier A: **Completely Derandomized Self-Adaptation in Evolution Strategies.** *Evolutionary Computation* 2001, **9(2):**159–195.

67. Gillespie D: **A rigorous derivation of the chemical master equation.** *Physica A* 1992, **188:**404–425 http://www.comp.nus.edu.sg/~cs6280/Materials/06-gillespie92.pdf.

68. Kitano H, Funahashi A, Matsuoka Y and Oda K: **Using process diagrams for the graphical representation of biological networks.** *Nature Biotechnology* 2005, **23(8):**961–966.

69. Funahashi A, Tanimura N, Morohashi M and Kitano H: **CellDesigner: a process diagram editor for gene-regulatory and biochemical networks.** *BioSilico* 2003, **1(5):**159–162 http://www.sciencedirect.com/science/article/B75GS-4BS08JD-5/2/5531c80ca62a425f55d224b8a0d3f702.

70. Spieth C, Supper J, Streichert F, Speer N and Zell A: **JCell – a Java-based framework for inferring regulatory networks from time series data.** *Bioinformatics* 2006, **22(16):**2051–2052 http://bioinformatics.oxfordjournals.org/cgi/content/abstract/22/16/2051.

71. Shampine LF and Reichelt MW: **The MATLAB ODE Suite.** *Tech rep* 2007 http://www.mathworks.com/access/helpdesk/help/pdf_doc/otherdocs/ode_suite.pdf.

72. Shampine LF, Reichelt MW and Kierzenka JA: **Solving Index-1 DAEs in MATLAB and Simulink.** *SIAM Rev* 1999, **41(3):**538–552 http://www.mathworks.com/support/solutions/files/s8314/dae.ps.

73. Schauer M and Heinrich R: **Quasi-steady-state approximation in the mathematical modeling of biochemical reaction networks.** *Mathematical Bioscience* 1983, **65:**155–171 http://cat.inist.fr/?aModele=afficheN&cpsidt=9308909.

74. Kloeden PE and Platen E: *Numerical Solution of Stochastic Differential Equations* Applications of Mathematics, Berlin: Springer-Verlag; 1992.

75. Bentele M: **Stochastic Simulation and System Identification of large Signal Transduction Networks in Cells.** *PhD thesis* Combined Faculties for the Natural Sciences and for Mathematics of the Ruperto-Carola University Heidelberg, Germany; 2004 http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=488968.

76. Kennedy J and Eberhart R: **Particle Swarm Optimization.** *IEEE Int Conf on Neural Networks, Perth, Australia* 1995.

77. Carlisle A and Dozier G: **An off-the-shelf PSO.** *Proceedings of the Workshop on Particle Swarm Optimization* Indianapolis: Purdue School of Engineering and Technology, Indianapolis; 2001.

78. Kaufman L and Rousseeuv PJ: *Finding Groups in Data: An Introduction to Cluster Analysis* Probability and Mathematical Statistics, New York: John Wiley and Sons, Inc; 1990.

79. Maechler M, Rousseeuw P, Struyf A and Hubert M: **Cluster Analysis Basics and Extensions.**2005.

80. R Development Core Team: *R: A Language and Environment for Statistical Computing* R Foundation for Statistical Computing, Vienna, Austria; 2007 http://www.R-project.org, .

81. Hartung J: *Statistik* München: Oldenburg Wissenschaftsverlag GmbH; 2002.