Optimization of biochemical systems through mathematical programming: Methods and applications

Julio Vera\textsuperscript{a}, Carlos González-Alcón\textsuperscript{b,∗}, Alberto Marín-Sanguinio\textsuperscript{c}, Néstor Torres\textsuperscript{d}

\textsuperscript{a}Systems Biology and Bioinformatics Group, Department of Computer Science, University of Rostock, Rostock, Germany
\textsuperscript{b}Grupo de Tecnología Bioquímica, Departamento de Estadística, Investigación Operativa y Computación, Universidad de La Laguna, La Laguna, Tenerife, Islas Canarias, Spain
\textsuperscript{c}Max Planck Institute of Biochemistry, Department of Membrane Biochemistry, Am Klopferspitz 18, D-82152 Martinsried, Germany
\textsuperscript{d}Grupo de Tecnología Bioquímica, Departamento de Bioquímica y Biología Molecular, Universidad de La Laguna, La Laguna, Islas Canarias, Spain

\textbf{A R T I C L E I N F O}

Available online 10 March 2009

Keywords:
Biochemical systems
Power-law formalism
S-systems
Generalized mass action
Linear programming
Multiobjective optimization
Geometric programming

\textbf{A B S T R A C T}

In this work we present a general (mono and multiobjective) optimization framework for the technological improvement of biochemical systems. The starting point of the method is a mathematical model in ordinary differential equations (ODEs) of the investigated system, based on qualitative biological knowledge and quantitative experimental data. In the method we take advantage of the special structural features of a family of ODEs called power-law models to reduce the computational complexity of the optimization program. In this way, the genetic manipulation of a biochemical system to meet a certain biotechnological goal can be expressed as an optimization program with some desired properties such as linearity or convexity.

The general method of optimization is presented and discussed in its linear and geometric programming versions. We furthermore illustrate the use of the method by several real case studies. We conclude that the technological improvement of microorganisms can be afforded using the combination of mathematical modelling and optimization. The systematic nature of this approach facilitates the redesign of biochemical systems and makes this a predictive exercise rather than a trial-and-error procedure.

© 2009 Elsevier Ltd. All rights reserved.

1. Introduction

In the last decade the advancement in the experimental characterisation of the biological systems at the cellular and intracellular level has revealed that a new integrative biology is necessary to investigate them (Voit [65], Kitano [22], Dollery et al. [15], Wolkenhauer [67]). In this new paradigm, the structure and dynamics of cellular systems are investigated as a whole, rather than as isolated parts, in order to understand their essential features. In this regard, the mathematical modelling of biochemical systems supported with quantitative experimental data has emerged as a new approach to investigate the dynamics of biochemical pathways. This approach is a suitable tool to support the biologists in the process of hypotheses generation and the design of experiments (Kitano [23], Wolkenhauer [67]), decreasing the experimental effort necessary to characterise a biological system. In addition, there are specific properties of the biochemical systems (feedback loops, multistability, ultrasensitivity, etc.) that escape the purely experimental approach used in the traditional molecular biology and can be understood only in terms of a dynamical mathematical description of the systems.

One of the usual approaches to model biochemical pathways is the use of mathematical models in ordinary differential equations (ODEs) (Heinrich and Schuster [20]). The power-law models (Savageau [38], Voit [65], Sorribas et al. [49], Vera et al. [59]) are a special kind of ODE models in which the processes that integrate biochemical networks are modelled using power-law expansions in the variables of the system (rate dependent and m independent) and then included in non-linear ODEs with the following structure:

\[ \frac{dX_i}{dt} = \sum_j c_{ij} \prod_{k=1}^{n} X_k^{e_{ik}} \quad \text{for} \quad i = 1, \ldots, n, \]  

(1)

where $X_i$ are the variables of the model (concentrations of biochemical compounds of the investigated network: metabolites, proteins, messenger RNAs, etc.), and $c_{ij}$ (rate constants) and $e_{ik}$ (kinetic orders) are different kinds of parameters defining the dynamics of the system. The specific numerical values for the parameters are determined using prior biological knowledge, information about the basal steady states of the system (Voit [65]), or experimental data from dynamical perturbations (Vera et al. [60,57]). Unlike in conventional
kinetic models, where the kinetic orders are always integer numbers, power-law models admit non-integer values. The most remarkable property of power-law equations is that two such equations with the same structure can describe totally different dynamics (from inhibitory process to the description of cooperativity) by just modifying the numerical values of the kinetic orders involved (Vera et al. [59]).

A special simplified sub-category of power-law models are the S-system models, in which rate aggregation is applied to the differential equations describing the dynamics of every time-dependent compound (Torres and Voit [54], Voit [65]). In S-system models, every positive (or production) fluxes are aggregated into a unique input and every negative (or degradation) fluxes are aggregated into a unique negative flux. Every differential equation in an S-system model has the same characteristic structure no matter the complexity or the nature of the biochemical system investigated:

\[
\frac{dX_i}{dt} = \sum_{j=1}^{n+m} g_{ij} X_j^{m_i} - \sum_{j=1}^{n+m} h_{ij} X_j^{n_i} \quad \text{for} \quad i = 1, \ldots, n. \tag{2}
\]

S-system models have remarkable mathematical properties related to their structural simplicity. One of their most interesting features is the possibility of computing their steady states using linear algebra just by applying a simple logarithmic transformation to the steady-state equations (Voit [65]).

From a biotechnological and biomedical perspective, mathematical models of biochemical systems can be a very valuable tool in the coming times. The use of suitable mathematical models may allow to investigate the effect of pathway modifications that are not accessible with the current experimental techniques. In addition they can be used to predict in advance the effect of costly complex interventions over biochemical systems with technological or biomedical purposes. In this vein, mathematical modelling can be combined with operations research in order to support microbiologists in the biotechnological improvement of microorganisms with industrial applications (Torres and Voit [54]). “What is the minimum set of enzymes in a given microorganism that should be modified in order to maximize the industrial production of a desired end product? “How to make a proper decision when genetically modifying a microorganism if we find that more than one biological response should be optimized?” These are technologically relevant questions that can be rationally answered using mathematical modelling and operations research. Moreover, the combination of mathematical modelling, data mining and operations research is a promising approach in drug discovery (Vera et al. [60]). In that case, some biomedical dilemma concerning the design of new therapies can also be presented as optimization programs: “What is the minimum set of biochemical interactions in a network that must be inhibited via specifically designed drugs in order to subvert the pathological condition associated to a certain disease?”

Several groups have worked in the past decade using this combination of mathematical modelling and optimization in biotechnology and biomedicine (Voit [64], Torres et al. [53], Mendes and Kell [30], Hatzimanikatis et al. [19], Banga et al. [4]). In those works, the tools and routines developed by the operations research have been used, these ranging from linear programming techniques to the most sophisticated, up to date optimization algorithms.

In this paper we review and discuss the basic concepts of the mathematical optimization of biochemical systems and illustrate them using several case studies of technological and biomedical relevance.

2. Linear programming and the IOM method

The indirect optimization method (IOM) (see Fig. 1) is based on the fact that, although S-system models are non-linear, their steady-state equations are linear when the variables are expressed in logarithmic coordinates. Since a function and its logarithm assume their maxima for the same argument, yields or fluxes can thus be optimized with linear programs expressed in terms of the logarithms of the original variables. Also, some additional constraints frequently found in biological systems, such as ratios or products, become linear in the logarithmic coordinate system. Thus, transporting the steady-state system and the constraints into a logarithmic space reduces the non-linear optimization problem to a problem of straightforward linear programming (Voit [64], Vera et al. [58]). Since then, the method has been extended and applied to different types of systems (Torres et al. [53], Alvarez-Vasquez et al. [1,2], Marín-Sanguino and Torres [28], Vera et al. [58,60,62,63], Xu et al. [68], Sendín et al. [47]). The method is basically a procedure that systematizes the steps to follow for optimum results, or at least efficient S-system models. Fig. 1 illustrates the stages of the method in an outline for the most general case.

Step 1: Getting the model. The first step is the development of a model of the system. This can be modelled directly on the S-system formalism or in the form of kinetic model or GMA, and then transferred to S-system.

Step 2: Analysis of quality of the model. The S-system modelling formalism provides us with tools to analyse its stability, robustness and dynamic response. It is important to ensure the validity of the model, i.e., that it properly describes the known behaviour of the
system. Otherwise, it is necessary to refine the model prior to its optimization.

Step 3: Construction and resolution program optimization. This is the core of the procedure. All information gathered on the system and the limitations of biological or technological constraints on the solutions are converted into the mathematical equations that make up the program optimization. After ensuring the validity of the model and the adequacy of the components of the program optimization, the stated objectives and the conditions imposed on the system are moved into logarithmic space. In the logarithmic space, the optimization problem becomes a linear program and once the optimal solution is obtained it is transferred to the area of the original variables.

In the more general case (multiobjective optimization) once we reach this point we can choose among some options. The first one is the pure multicriteria problem, the multiple objective linear program, a direct extension of the linear programming case. An alternative is the weighted sum approach. Here we assign a weight to each objective or function to be optimized. In the present case, these weights are chosen between zero and one such that they sum up to one. Finally, the third option is goal programming, where a goal level of achievement is established for each objective. These goal levels are soft constraints that are included in the definitions of the objective functions. Moreover the optimized system must be at a steady state and fulfil some constraints. These constraints, as well as the objective functions, are readily translated into linear functions and inequalities, and the optimization task becomes a linear program. Some software packages are available that produces the efficient solution set for multiobjective linear programming tasks. This is the case of ADBASE (Steuer [52]) used here, besides the Matlab optimization toolbox for the goal programming and weighted sum approach.

Step 4: Analysis and refinement of the solution. At this stage the solution is analysed, that is, the system’s properties at the optimal state (stability, robustness and dynamics). If the optimal solution is sound, it is chosen as a solution to the problem. If the predicted steady state is unstable or not robust enough, the imposed restrictions or the objective function are refined and the optimization process repeated until a satisfactory solution is reached. In the case where an S-system model was built from a previous kinetic model, the S-system solution must be transferred to the original model.

In the following we will illustrate the application of this approach to three case studies.

2.1. Case study 1. Optimization of the Escherichia coli L(-)-carnitine production

L(-)-carnitine is a chiral compound widely distributed in nature and with a wide range of medical applications. A great deal of the L-carnitine is produced by chemical synthesis, with the disadvantage of producing a racemic mixture that it is necessary to separate in a costly process.

Fig. 2 shows a schematic diagram of one experimental set-up for the biotransformation of crotonobetaine into L-carnitine by an overproducing E. coli strain in a cell recycle bioreactor (Cánovas et al. [9], Alvarez-Vasquez et al. [2]). The mathematical model of the system was presented by Cánovas et al. [9]. In their original presentation, the authors included the metabolic transformation of crotonobetaine into γ-butyrobetaine, through the activity of a previously described crotonobetaine reductase (Roth et al. [36], Preussner et al. [35]). In Fig. 2 appear the process variables represented in the model. In the optimization procedure the dependent variables are X1–X4, while X5–X9 are the parameters. We started from the original model set-up and carried out its translation to the S-system representation...
predictions, both in terms of stationary states experimentally observed as well as to the production speeds of L-carnitine.

What can be seen in Table 1 is that there is close agreement between the experimental results and the model-based optimization predictions. In both cases assayed, the experimental steady state was the same predicted by the model and, more importantly, the increase in the rate of L-carnitine production was almost coincident with the predicted increase.

2.2. Case study 2. Application to biomedicine: design of new therapeutic approaches to human hyperuricemia (gout)

In classical hyperuricemia, a functional defect in enzyme that controls the synthesis of de novo purines increases its activity and leads to an increase in the activity of degradative metabolic fluxes yielding uric acid (see Fig. 3). A complete model, in power-law formalism was presented and used for optimization purposes (Curto et al. [12,13]). The symptoms of this disease include acute episodes of arthritic pain and inflammation, and various kinds of nephropathy. Currently, gout treatments usually include a symptomatic treatment for joint pain, and inflammation, and various kinds of nephropathy. In this case, the aim is to alter the values of critical metabolite concentrations and fluxes in relation to the symptoms. We model the functional origin of the disease by imposing a characteristic value to the enzyme activities that underlie the PS profile:

\[ X_i = X_i^{PS}. \]

In this equation, \( X_i \) represents the activity value for the deficient enzyme and \( X_i^{PS} \) is its characteristic value in the PS.

The solution near to the HS would be a stable steady state of the considered metabolic network, which is reflected by the following set of equations:

\[ \frac{dX_i}{dt} = 0, \quad i = 1, \ldots, n. \]

Moreover, we have to assume additional conditions to guarantee physiologically acceptable solutions. These conditions can be modelled as bounds in the concentrations:

\[ X_i^L \leq X_i \leq X_i^U, \]

where \( X_i^L \) and \( X_i^U \) establish the physiologically admissible lower and upper bounds for each metabolite and modifiable enzyme activity \( X_i \). Metabolites were allowed to vary from a 50% to 1000% their basal values:

\[ 0.5 X_i^{PS} \leq X_i \leq 10 X_i^{PS}, \quad i = 1, \ldots, 16, \]

meanwhile \( X_{17} \) only can vary a 50% up and down:

\[ 0.5 X_{17}^{PS} \leq X_{17} \leq 1.5 X_{17}^{PS}. \]

We consider an optimization problem for each modifiable enzyme. So, for each target enzyme \( X_i \) \((i = 20, \ldots, 46)\) we solve a problem considering a unique pair of bounds on enzymes:

\[ 0.1 X_i^{PS} \leq X_i \leq X_i^{PS}, \]

taking the other enzymes their HS values.

The outlined approach integrates modelling of the system and identification of target enzymes with the analysis of the available information and the choice of the best solutions. Each solution consists of a set of predicted values for metabolites, initial substrates and enzyme activities that describes a biologically acceptable steady state of the system that shifts the PS towards the HS. The computation of programs was carried out using the Matlab toolbox MetMAP (Vera and Torres [63]). Additional selection criteria were imposed in order to select the most feasible solutions within the generated set (Vera et al. [60]). Firstly, we imposed an upper limit on the concentration of uric acid (\( X_{16} \)) equal to 105% of the HS value: \( X_{16} \leq 1.05 X_{16}^{PS} \).

The solutions that verify this equation were called satisfactory solutions. A second criterion was defined using the Cartesian distance in the space of the metabolite concentrations from the considered solution (\( X \)) to the HS (\( X^{HS} \)):

\[ D(X, X^{HS}) = \sqrt{\frac{1}{16} \sum_{i=1}^{16} \left( \frac{X_i}{X_i^{HS}} - 1 \right)^2}. \]
We called this the metabolic distance, which measures the deviation of the metabolite concentrations from the HS values in the considered solution. Solutions with a high value of $D(X,X^{HS})$ would cause metabolic disequilibrium and undesirable physiological side effects. Finally, solutions with the highest values for the activity of the modified target enzymes were chosen. We took into account the fact that higher enzyme activities imply the use of a lower dose of the specific drug, and reduced drug doses minimise adverse side effects. Six satisfactory solutions were obtained: Inhibition of 1. Adenine phosphoribosyltransferase ($V_{\text{den}}X_{22}$); 2. AMP deaminase ($V_{\text{ampd}}X_{28}$), 3. RNases to AMP and GMP ($V_{\text{dgnuc}}V_{\text{dnaa}}X_{36}$); 4. $5'(3')$-nucleotidase ($V_{\text{dgnuc}}X_{37}$); 5. guanine hydrolase ($V_{\text{gua}}X_{40}$) and 6. xanthine oxidase ($V_{\text{hd}}V_{\text{hxd}}X_{44}$). Further details on the selected solutions can be found in Table 2. Examination of the table also shows that all the solutions have significant values of $D(X,X^{HS})$ and require substantial inhibition of the targeted enzyme (low values in $X_i$).

In Fig. 4, this group of solutions is shown in the space of $D(X,X^{HS})$ and $X_i/X^{HS}$. The figure also shows the so-called utopian solution, the potential solution which has the lowest computed distance and the highest value of $X_i$. Solution 1, that inhibit $X_{22}$, has the
lowest metabolic distance (1.56), but the highest enzyme activity value corresponds to solution 3, \(X_{36} = 0.36\). Examination of Fig. 2 shows that the best treatments with prescription of diet are through inhibition of \(X_{36}\) (solution 3) or \(X_{28}\) (solution 2), the closest solutions to the utopian point. An important observation relates to the solution 6, that involving inhibition of \(X_{44}\). This solution coincides with the most common current clinical treatment for hyperuricemia, which is based on the use of a combination of allopurinol (an inhibitor of \(X_{44}\)) and a diet low in purine precursors (\(X_{17}\)) (see Table 2). Moreover, our method predicts the well-known metabolic side effects of treatment with allopurinol, namely a strong increase in the concentration of xanthine and hypoxanthine (\(X_{13}\) and \(X_{14}\)). This represents an \textit{a posteriori} pragmatic verification of our model predictions. IOM has been efficient not only to find the current clinical treatment of the disease but also several other possible target enzymes with potentially lower side effects.

2.3. Case study 3. The multiobjective version of the IOM: application to the ethanol production by \textit{Saccharomyces cerevisiae}

We have used a mathematical model of the ethanol production by \textit{S. cerevisiae} presented by Schlosser et al. \cite{44} as a case study to apply the multiobjective version of the IOM procedure. Fig. 5 is a schematic representation of the investigated metabolic system. As we can see, the glucose provided to the microorganism is converted in ethanol after five consecutive biochemical processes. In addition, the biochemical pathway has two branches and part of the glucose is transformed into the side-products polysaccharides and glycerol. The model refers to the ethanol production under anaerobic, no growing conditions, with glucose as the sole carbon source for the microorganism and absence of nitrogen. Under these conditions, only the metabolic pathways to ethanol, glycerol and to polysaccharide storage were operative. The preliminary model by Schlosser et al. \cite{44} was converted into an \textit{S}-system model in Vera et al. \cite{58} with the following structure:

\[
\frac{dX_1}{d\tau} = 1.0006X_2^{-0.0492}X_6 - 1.6497X_1^{0.5582}X_5^{0.0465}X_7,
\]

\[
\frac{dX_3}{d\tau} = 1.6497X_1^{0.5582}X_5^{0.0465}X_7 - 0.5793X_2^{0.5907}X_5^{-0.2218}X_8^{0.8322}X_{11}^{0.1678},
\]

\[
\frac{dX_3}{d\tau} = 0.4536X_2^{0.4407}X_5^{0.2665}X_6 - 0.2456X_3^{0.4505}X_4^{0.0441}X_5^{0.092}X_9^{0.8547}X_1^{0.1453},
\]

\[
\frac{dX_4}{d\tau} = 0.2365X_5^{0.5285}X_3^{0.0994}X_9 - 2.0892X_3^{0.0075}X_5^{0.0434}X_9^{0.0484}X_{10},
\]

\[
\frac{dX_5}{d\tau} = 1.406X_2^{0.2605}X_3^{0.152}X_5^{0.0739}X_6^{0.05}X_5 - 2.9437X_5^{0.1962}X_3^{0.1791}X_4^{0.2354}X_3^{0.3514}X_7^{0.2925}X_8^{0.0589}X_9^{0.297}X_{11}^{0.113},
\]
In our investigation, we have considered three biotechnologically relevant optimization objectives (Vera et al. [58,61]): (i) the maximization of the ethanol production, \( F_{\text{prod}} \); (ii) the minimization of the total amount of internal metabolite concentrations, \( F_{\text{int}} \); and (iii) the minimization of the total amount of enzyme activities, \( F_{\text{enz}} \). The first objective directly relates to the desired enhancement of productivity in the microorganism, measured in terms of ethanol production. The other two objectives relate to the minimization of the amount of glucose transformed into other products different from ethanol (objective (ii)), but also to the cell viability by reducing the osmolarity (objective (iii)), but also to the cell viability by reducing the osmolarity stress (Atkinson [3]) and the metabolic burden [Snoep et al. [48]] (objective (iii)). Being \( Y = \ln(X) \), they are defined as follows:

\[
\min Z(Y), \quad Z(Y) = (-F_{\text{prod}}(Y), F_{\text{int}}(Y), F_{\text{enz}}(Y))
\]

where the maximization objective becomes a minimization objective by changing the sign of the objective function, which has no repercussions in the position of its extrema. The first objective function take the following form in the logarithmic transformed space:

\[
F_{\text{prod}}(Y) = -0.0075Y_3 + 0.3049Y_4 + 0.0483Y_5 + Y_9,
\]

and we consider the other two objectives as

\[
F_{\text{int}}(Y) = \sum_{i=1}^{5} Y_i \quad \text{and} \quad F_{\text{enz}}(Y) = \sum_{i=6}^{13} Y_i.
\]

In our optimization program we considered three types of constraints (Vera et al. [58,61]): (a) those that guarantee that solutions correspond to a steady-state solution of the system; (b) those setting the physiologically feasible lower and upper boundaries for the model variables and (c) a set of additional constraints related to special features of the considered biosystem. The constraints that set up the boundaries for the model variables (metabolite or enzyme concentrations) are especially important and configure the properties of the solutions; their values are dictated by the experience and are specifically formulated for each variable. In our case, previous studies with similar metabolic systems (Alvarez-Vasquez et al. [1], Marín-Sanguino and Torres [27]) led us to consider for the lower boundary 50% of the original basal steady-state value and for the upper boundary five times this basal value for most of the variables (0.5X_0 \leq X_i \leq 5X_0). The exception to this rule were the ATPase activity (X_5 < 1.5X_0), limited by additional physiological constraints, and the side-products polysaccharides and glycerol, which were maintained at their basal values (X_13 = X_14 = X_15 = 0) (see [58,61]). Moreover, an additional constraint was introduced based on our experience analysing the dynamics of the metabolic system, limiting the value of the flux ratio through the enzymes X_9 and X_10 to prevent dynamical instability in the solutions (X_9/X_10 < 1.75). We furthermore imposed a last constraint in order to ensure that any computed solution doubles at least the ethanol production in the original unmodified microorganism: \( V_{\text{eth}}(X) \geq 2V_{\text{eth}}(X^0) \).

After computing the multobjective program using a version of ADBASE [Steuer [52]] integrated in the software package MetMAP (Vera and Torres [63]), 22 efficient solutions were obtained. In our investigation, we considered only the efficient vertexes of the problem, which already contain complex enough information about the properties of the biochemical system. The initial analysis of the results suggested that solutions with a high ethanol production are only possible when the maximum of ATP \((X_5 = 1.5)\) and ATPase \((X_3 = 5)\) are provided to the system (Vera et al. [58]). From a biological perspective, this information indicates the extreme importance of the ATP turnover to increase the productivity of the system. We furthermore faced task of analysing and classifying the generated solutions. In order to facilitate this, we defined some auxiliary biologically relevant parameters that integrate the information provided by the solutions (Vera et al. [58,61]). Thus, three magnitudes were formulated, \( \rho, \sigma, \) and \( \theta \), with the following structure:

\[
\rho(X) = \frac{V_{\text{eth}}(X)}{V_{\text{eth}}(X^0)}, \quad \sigma(X) = \frac{\sum_{i=1}^{5} X_i}{\sum_{i=5}^{13} X_i^0}, \quad \theta(X) = \frac{\sum_{i=5}^{13} X_i^0}{\sum_{i=5}^{13} X_i^0}
\]

\( \sigma \) is defined as the ratio between the total intermediate concentration of the current solution and the one in the original basal solution of the system. In a similar way, we defined \( \theta \) as the ratio between the total enzyme activities at the optimum solution and the ones at the basal steady state, while \( \rho \) describes the same ratio for the ethanol production of the system.

In order to support the biotechnologists making the proper decision, we have introduced an additional criterion. Firstly, we ruled out all solutions with the lowest admissible ethanol production \( (\rho_{\text{min}} = 2.0) \). We furthermore ranked the remaining solutions according with their parametric distance to the so-called utopian point, a fictitious solution constructed with the optimal values of ethanol production \( (\rho_{\text{utp}}) \), total intermediate concentration \( (\sigma_{\text{utp}}) \) and total enzyme concentration \( (\theta_{\text{utp}}) \) from the computation of separated mono-objective programs (Vera et al. [58,61]):

\[
\rho_{\text{utp}} = 4.986, \quad \sigma_{\text{utp}} = 0.5, \quad \theta_{\text{utp}} = 1.0.
\]

This solution is actually a unfeasible solution because it does not satisfy all the physiological constraints imposed. In a similar way, we defined the anti-utopian solution, extracting the worst value from every column in the payment matrix for the separated mono-objective programs (Vera et al. [61]):

\[
\rho_{\text{min}} = 2.0, \quad \sigma_{\text{min}} = 4.76, \quad \theta_{\text{min}} = 3.30.
\]

With this information we defined a weighted parametric distance from every solution to the utopian point as follows:

\[
D(X,X_{\text{utp}}) = \left( \left( \frac{\rho_{\text{utp}} - \rho(X)}{\rho_{\text{utp}} - \rho_{\text{min}}} \right)^2 + \left( \frac{\sigma_{\text{utp}} - \sigma(X)}{\sigma_{\text{utp}} - \sigma_{\text{min}}} \right)^2 + \left( \frac{\theta_{\text{utp}} - \theta(X)}{\theta_{\text{utp}} - \theta_{\text{min}}} \right)^2 \right)^{\frac{1}{3}}.
\]

Fig. 6 shows the solutions grouped in three sets according with their \( D(X,X_{\text{utp}}) \) value. Set I corresponds to those solutions with the largest \( D(X,X_{\text{utp}}) \) value. This group is characterized by a very high productivity \( (\rho) \), a positive feature, but also by high total intermediate concentration \( (\sigma) \), an undesirable property (both with values bigger than 4). Set II includes the closer solutions to the utopian point \( D(X,X_{\text{utp}}) \), characterised by productivity and enzyme levels bigger than 3, but low metabolite concentrations \((0.8-1.85)\). Finally, Set III represents those with intermediate values of \( D(X,X_{\text{utp}}) \), high productivity and enzyme levels (bigger than 4), but medium values of total metabolites concentration \((1.25-1.75)\). Some other solutions are not classified in any of the previous sets. When the parametric distance is considered the criterion to choose the appropriate solution for biotechnological purposes, we found two solutions which are clearly better (highlighted in Fig. 6 with a five-pointed star) and are shown in Table 3.

These two solutions represent totally opposite alternatives from a biotechnological perspective. Solution I in Table 3 prefers the increase in the ethanol production at the price of high cell resources consumption, while solution II establishes a biotechnological compromise between satisfactory increasing in the ethanol production and non-excessive cell resource consumption.
Geometric programs are a class of optimization problems that present a special structure in all the functions involved (objective and constraints). As linear programming (which is included in GP) it falls in the broader category of convex optimization. Convex problems are among the few non-linear tasks where, thanks to powerful methods, the efficient determination of global optima is feasible even for large scale systems. For example, a geometric program of 1000 variables and 10,000 constraints can be solved in less than a minute on a desktop computer (Boyd and Vandenberghe [5]); the solution is even faster for sparse problems as they are found in metabolic engineering. Furthermore, easy to use solvers are starting to become available (Grant and Boyd [17]; Koh et al. [25]). The special structure that shows the objective function and the constraints of a geometric program is a sum of monomials, i.e., power-law terms as each term of the right-hand side of Eq. (1):

\[
M(X) = cX_1^{q_1}X_2^{q_2} \cdots X_n^{q_n},
\]

where \( c \) is positive and \( q_i \) are any real number. (Do not confuse with the standard “monomial” in algebra, where \( c \) can be any real and all the exponents are integers.) Sums of monomials, all with positive sign, are called posynomials. If some of the monomials enter the sum with negative signs, the collection is called a signomial. The peculiarities of convexity and GP methods render the difference between posynomials and signomials crucial. A GP problem in standard form shows as

\[
\begin{align*}
\text{min} & \quad P_0(X) \\
\text{s.t.} & \quad P_j(X) \leq 1, \quad j = 1, \ldots, p, \\
& \quad M_k(X) = 1, \quad k = 1, \ldots, q, \\
& \quad X_i > 0, \quad i = 1, \ldots, n
\end{align*}
\]

being \( M_k(X) \) monomials, while the objective function \( P_0(X) \) and the functions \( P_j(X) \) involved in inequalities must be posynomials. Signomials are not allowed, and optimization problems involving them require additional effort. Under these conditions (minimization of a posynomial subject to posomials smaller or equal to 1, monomials equal to 1) it can be guaranteed the conversion of the problem to a convex program and so the existence of a global solution, and the possibility to solve it in a very efficient manner. The equivalence between monomials and power laws immediately suggests the potential use of GP for optimization problems formulated within the framework of biochemical systems theory (BST). In the next section two different ways to afford the optimization of GMA models by a sequence of GP problems will be proposed.

The optimization of an S-system is included in Eqs. (3) and (5) for the particular case of a monomial objective function. This can easily be seen by undoing the logarithmic transformation. Therefore, GP can replace linear programming in the general scheme of the IOM method. The immediate advantage of this would be the possibility of including upper bounds to posynomial functions through Eq. (4). Such functions appear frequently in biochemical systems as sums of variables, i.e., the total protein content of the cell should not increase too much. By contrast, steady-state GMA equations (the right-hand side of Eq. (2) equalled to zero) do not automatically fall within the GP structure, because GMA systems usually include negative terms, thus making them signomials.

Two ways are proposed to adapt GP solvers for the treatment of GMA systems, both relying on condensation (Floudas [16]).

---

**Table 3**

<table>
<thead>
<tr>
<th>Sol.</th>
<th>( p )</th>
<th>( q )</th>
<th>( \theta )</th>
<th>( D(X,X^{\text{opt}}) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>4.987</td>
<td>1.812</td>
<td>3.301</td>
<td>0.604</td>
</tr>
<tr>
<td>II</td>
<td>2.791</td>
<td>0.596</td>
<td>2.798</td>
<td>0.013</td>
</tr>
</tbody>
</table>

**Fig. 6.** Efficient solutions for the multicriteria optimization of ethanol production by S. cerevisiae. (●) Set I, solutions with the largest \( D(X,X^{\text{opt}}) \) value. (♦) Set II, solutions closer to the utopian point (star). (■) Set III, solutions with intermediate values of \( D(X,X^{\text{opt}}) \). (×) represents the anti-ideal point.
Condensation is a standard procedure in GP of which flux aggregation is a particular case. Namely, a posynomial is approximated by a single monomial. In the terminology of GP, the condensation of a posynomial \( P(x) \) is generically denoted as \( \tilde{C}[P(x)] = C[M_1(x) + M_2(x) + \cdots + M_k(x)] = M_0(x) \), and it is done at a certain operating point \( X^0 \):

\[
\tilde{C}[M_1(x) + M_2(x) + \cdots + M_k(x)] = \prod_{i=1}^{k} \left( \frac{M_i(x)}{w_i} \right)^{w_i},
\]

being \( w_i = M_i(x^0)/P(x^0) \). In the terminology of Eq. (2):

\[
C \left[ \sum_{j=1}^{m} \prod_{k=1}^{m} x_k^{g_{jk}} \right] = x \prod_{j=1}^{m} x_k^{g_k}.
\]

where \( x \) and \( g_k \) are chosen such that equality holds at the chosen operating point; thus, the result is equivalent to the Taylor series linearization that is fundamental in BST [Savageau [38,39], Voit [65]]. As in the Taylor series, the condensed form is equal to the original equation at the operating point. It can be shown that the left and right-hand side of Eq. (7) are equivalent to those of the arithmetic–geometric (A–G) inequality,

\[
\sum_{i=1}^{n} a_i \geq \prod_{i=1}^{n} \left( \frac{a_i}{w_i} \right)^{w_i}.
\]

Therefore, at any point the condensed form is an underestimation of the original.

Objective functions can only be minimized in GP. This is seldom a problem given that the functions to maximize are often monomials that can be inverted: a variable, a reaction rate or a flux ratio. Posynomial objectives are usually entitled for minimization, like the creation of an ancilliary variable \( t \):

\[
P(X) - Q(X) = 0.
\]

If both posynomials are condensed, the equation will be reduced to the standard form for monomial equality restriction:

\[
1 = \frac{P(X)}{Q(X)} \leq \frac{\tilde{C}[P(X)]}{\tilde{C}[Q(X)]}
\]

as the quotient of two monomials is a monomial. Since the condensed steady state equations of the GMA are equivalent to those of an S-system, this method could be regarded as a direct generalization of classical IOM methods. In other words, GP can include the equations of IOM as monomial equalities but also enables us to include posynomial inequalities in the problem. The first consequence of this is that S-systems can be optimized globally together with a constraint establishing an upper level for the total enzyme in the cell. Such a constraint or any other posynomial expression cannot be directly treated in IOM since it is non-linear in the logarithmic space. There is an additional advantage coming from the possibility of using posynomial constraints. When a posynomial is approximated by condensation, the A–G inequality, Eq. (8), guarantees that the monomial is an underestimation of the original posynomial. Furthermore, the posynomial structure is not altered when divided by a monomial so the quotient between a posynomial and its condensed form, always greater than or equal to 1, provides the exact condensation error as a posynomial function that can be added to the geometric problem as an extra constraint:

\[
P(X) - Q(X) \leq 1 + tol.
\]

The importance of controlling the aggregation error has been pointed out and led to the proposal of variations in IOM [16] that included linear approximations to the error as constraints added to the problem. With the above discussed constraints, the exact error is taken into account. The constraints on the error can even go beyond providing a safer flux aggregation. When the GMA system has been obtained by recasting (and is therefore an exact representation of the original model) the approximated S-system form can be obtained through condensation instead of using a Taylor series. Although both methods yield the same result, the condensation approach can be subject to error controlling constraints to limit every possible source of error, as shown in the example below.

This method is intended to be used iteratively so that the original problem is solved as a series of GPs. An initial point \( X^0 \) is used to set up a GP taking \( X^0 \) as the operating point for the necessary condensations. The solution \( X^1 \) is used for the next problem. The extra set of constraints ensures that every iteration will only explore the neighbourhood of the feasible region in which the error due to condensation remains below an arbitrary tolerance set by the user. When two successive solutions coincide the parameter \( tol \) can be diminished.

A global approach: penalty treatment. A similar yet distinct strategy that minimizes the use of condensation is an extension of the penalty treatment method (Roundtree and Rigler [37]), a classic algorithm for signomial programming. In this method, a signomial constraint such as Eq. (9) is replaced by two posynomial equalities through the creation of an ancilliary variable \( t \):

\[
P(X) = t,
\]

\[
Q(X) = t.
\]

These are not valid GP constraints, so the following relaxed version is used:

\[
P(X) \leq t,
\]

\[
Q(X) \leq t.
\]

Upon dividing by \( t \), the feasible area of the original problem is contained in the feasible area of the new relaxed version and approximation by condensation is not needed. In order to force these inequalities to be tight in the final solution, the objective function is augmented with penalty terms that grow with the slackness of the constraints, namely the inverses of the condensation of the relaxed constraints. The result of this procedure is a legal GP:

\[
\min P_0(X) + \sum_{i} \left( w_i^f \left( \frac{t}{\tilde{C}[P_0(X)]} \right) + w_i^f \left( \frac{t}{\tilde{C}[Q(X)]} \right) \right)
\]

s.t.

\[
\frac{P(X)}{t} \leq 1,\quad \frac{Q(X)}{t} \leq 1,\quad i = 1, \ldots, n,
\]

where the condensed terms are initially calculated at the basal steady state. If the obtained solution falls within the feasible area of the original problem, it is taken as a solution; if it does not (any of the relaxed inequalities is below 1) the solution is used as the next reference point: condensations are calculated again, the weights of the violated constraints are increased and the new problem is solved. This procedure is repeated until a satisfactory solution is obtained. These variations on the original method serve to prevent the penalty terms from dominating the objective function and pushing the relaxed problem towards the boundaries of the feasible region from the very beginning.
Table 4
Comparison of results obtained by IOM and GP (both penalty and error tolerance) when applied to anaerobic fermentation in S. cerevisiae.

<table>
<thead>
<tr>
<th></th>
<th>Basal</th>
<th>Predicted IOM</th>
<th>Final IOM</th>
<th>GP</th>
</tr>
</thead>
<tbody>
<tr>
<td>X_1</td>
<td>0.034</td>
<td>2.0</td>
<td>2.19</td>
<td>2.0</td>
</tr>
<tr>
<td>X_2</td>
<td>1.01</td>
<td>2.0</td>
<td>1.58</td>
<td>2.0</td>
</tr>
<tr>
<td>X_3</td>
<td>9.18</td>
<td>2.0</td>
<td>1.53</td>
<td>2.0</td>
</tr>
<tr>
<td>X_4</td>
<td>0.0095</td>
<td>2.0</td>
<td>1.19</td>
<td>2.0</td>
</tr>
<tr>
<td>X_5</td>
<td>1.127</td>
<td>0.50</td>
<td>0.28</td>
<td>0.50</td>
</tr>
<tr>
<td>X_6</td>
<td>19.7</td>
<td>7.49</td>
<td>7.33</td>
<td></td>
</tr>
<tr>
<td>X_7</td>
<td>68.5</td>
<td>3.85</td>
<td>3.78</td>
<td></td>
</tr>
<tr>
<td>X_8</td>
<td>31.7</td>
<td>2.92</td>
<td>2.86</td>
<td></td>
</tr>
<tr>
<td>X_9</td>
<td>49.9</td>
<td>6.48</td>
<td>4.72</td>
<td></td>
</tr>
<tr>
<td>X_{10}</td>
<td>3440</td>
<td>5.72</td>
<td>4.16</td>
<td></td>
</tr>
<tr>
<td>X_{11}</td>
<td>14.31</td>
<td>0.010</td>
<td>0.010</td>
<td></td>
</tr>
<tr>
<td>X_{12}</td>
<td>203</td>
<td>0.010</td>
<td>0.010</td>
<td></td>
</tr>
<tr>
<td>X_{13}</td>
<td>25.1</td>
<td>27.04</td>
<td>14.04</td>
<td></td>
</tr>
<tr>
<td>X_{14}</td>
<td>0.042</td>
<td>1.0</td>
<td>1.0</td>
<td></td>
</tr>
<tr>
<td>Flux</td>
<td>30.22</td>
<td>273.12</td>
<td>214.62</td>
<td>198.85</td>
</tr>
</tbody>
</table>

Violated constraints appear in boldface. The three last columns are given as quotient by the basal value.

Table 5
Comparison of results obtained by IOM, modified IOM and CP (both penalty and error tolerance) when applied to a model of tryptophan production in E. coli.

<table>
<thead>
<tr>
<th></th>
<th>Basal</th>
<th>Iterative IOM</th>
<th>Modified IOM</th>
<th>GP</th>
</tr>
</thead>
<tbody>
<tr>
<td>X_1</td>
<td>0.18</td>
<td>1.20</td>
<td>1.20</td>
<td>1.20</td>
</tr>
<tr>
<td>X_2</td>
<td>7.99</td>
<td>1.09</td>
<td>1.05</td>
<td>1.15</td>
</tr>
<tr>
<td>X_3</td>
<td>1418</td>
<td><strong>0.46</strong></td>
<td><strong>0.27</strong></td>
<td>0.8</td>
</tr>
<tr>
<td>X_4</td>
<td>0.0031</td>
<td>0.005</td>
<td>0.002</td>
<td>0.004</td>
</tr>
<tr>
<td>X_5</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>X_6</td>
<td>2283</td>
<td>5000</td>
<td>5000</td>
<td>5000</td>
</tr>
<tr>
<td>X_7</td>
<td>430</td>
<td>1000</td>
<td>1000</td>
<td>1000</td>
</tr>
<tr>
<td>V_{rep}</td>
<td>1.31</td>
<td>3.88</td>
<td>4.54</td>
<td>3.06</td>
</tr>
</tbody>
</table>

Violated constraints appear in boldface. Values divided by the corresponding basal.

Application to the optimization of the ethanol production by S. cerevisiae (Marin-Sanguino et al. [29]). In Table 4 we can see the results of optimizing a GMA version of the fermentation model presented above for S. cerevisiae. The imposed constraints limited the maximum and minimum values of each metabolite to twofold and half the basal level, respectively. An additional constraint was imposed to keep the total enzyme concentration below four times the basal value. The ‘Predicted IOM’ column shows the results predicted by IOM after aggregating the fluxes of the GMA and applying linear programming to the resulting S-system. The ‘Final IOM’ column shows the results of substituting the enzyme levels predicted by IOM in the original GMA model. The discrepancies between them are not too significant for most metabolites, which remain within the designated margins. An exception is the value of ATP (X_5) which drops to almost half the lower bound imposed to it in the optimization. As can be seen in the next column, the optimization through GP (both controlled error and penalty treatment) finds a solution without violating any constraints.

Application to the optimization of the tryptophan production by E. coli. Table 5 compares several optimization methods applied to a model of tryptophan production in E. coli. (Xiu et al. [69]). The original model was not a power law, so it was transformed into a GMA through recasting (Marin-Sanguino et al. [29]). The recasting process generated some extra posynomial and signomial constraints that were condensed in each iteration together with the steady-state constraints. The first column shows the results predicted by the IOM in S-system form. The second column shows the result of adjusting the parameters of the original system according to the results of IOM and letting it reach a new steady state (Marin-Sanguino and Torres [27]). As can be seen, there is a discrepancy between the prediction of the S-system and the real steady state. Such discrepancy is especially significant in the level of Tryptophan (represented by X_5). The next column shows the results of optimizing with an extra set of constraints that approximate the condensation error linearly (Xu et al. [68]). Finally, the results obtained by both GP methods are shown in the last column.

4. Discussion

Although several groups have published a number of applications during the last decade, the integrated approach for the improvement of biotechnological systems optimization consisting in the combination of mathematical modelling and optimization of biochemical systems remains underexploited. Two main reasons explain this situation. Firstly, the importance of mathematical modelling in biochemical systems has been acknowledged only in very recent times, but also the experimental techniques necessary for a proper characterisation of such models have been accessible only in the last years. Secondly, the optimization of heavily non-linear systems presents inherent difficulties, which are evident when one confronts (structurally and dynamically) complex biological systems. Thus, most of the recent practical and experimental applications of mathematical optimization in biochemical systems deal almost exclusively with stoichiometric models, a special fully linear class of ODE models for biochemical systems (Palsson [33]).

In this paper we have shown how to overcome these difficulties through the use of a special kind of ODE models called power-law models and linear programming (Torres and Voit [54]). The homogeneous structure of the power-law models is the most relevant feature of this formalism because it allows the standardization of the modelling-based analysis of biochemical systems (composed by model design and calibration, steady-state analysis, dynamical analysis and simulation), which in many cases becomes a fuzzy, non-systematic procedure within other modelling formalisms. This makes power-law models very convenient for the integrated optimization of biochemical systems.

In addition, when the logarithmic transformation proposed in the IOM is applied to biotechnological optimization programs (Voit [64], Torres et al. [53]), they become linear and a plethora of refined computationally efficient algorithms for linear programming can be applied. Moreover, the method scales very well with problem size: when using the IOM the computational requirements are negligible in most of the state-of-the-art biochemical problems analysed and therefore they can be solved in almost real time. This means that the IOM procedure allows an interactive optimization of biochemical models, in which the researcher can define the program, compute and analyse the solutions and refine the program in a fast iterative manner. This is a very valuable feature when biochemical systems are investigated, which very often requires the use of experience and intuition over several iterations. In this vein, simulation software packages like MetMAP [63] enables this iterative procedure because the user can define the optimization program, compute the solutions and analyse their biological features via computational simulation using a unique integrated software tool.

However, the IOM has some drawbacks. While the computation of the solution via Simplex method is fast, every time the optimization program is modified the logarithmic transformation of the equations must be performed. This increases the total computational time required for the method. In addition, the heuristic nature of the last step in the IOM method may provoke a loss of efficiency in the solutions when the behaviour of the original model departs from their S-system expansion. This may result in the violation of some of the restrictions imposed. However, our experience shows that these violations are often negligible and the steady-state solutions usually stay in the vicinity of the real efficient solution. Moreover, the method...
incorporates tools to estimate solutions closer to the optimum in the non-linear domain. This latter step consumes considerable computational effort, but even more important, the solution obtained ceases to be optimal. The practical application of the method indicates however that in most cases it remains a high quality solution.

The original IOM method is based on a local linearization in logarithmic coordinates. Replacing LP by GP in the error controlled method provides an important improvement since it provides the optimization process with exact knowledge on the error introduced by such linearization. Furthermore, the penalty method offers a completely different approach that aims to converge on a global optimum. GP seems to offer interesting possibilities in the optimization of biotechnological systems, both through local and global methods. It remains yet to be seen how both approaches perform comparatively. On one hand, the penalty treatment is most promising since it offers a global approach. On the other hand, its convergence capabilities are yet to be tested on large scale problems.

The special nature of the biological systems makes the interpretation of the solutions arising from the IOM or other methods rather peculiar. From a biotechnological point of view, we are actually not interested in the numerical values for the model variables encoded by the optimal solution (this precision is actually not reachable with genetic engineering). Our interest is to focus on the information that such solution contains concerning what are the critical cellular compounds which concentration must be modified to improve the technological features of the investigated biosystem. The ability of the IOM to allow real time iterative analysis rather than the accuracy of its solutions makes this method a suitable tool for prototyping optimal genetic modifications of microorganisms or suggesting critical biochemical interactions in drug discovery.

Finally, we think that in the near future the development of the DNA recombinant technologies (Porro et al. [34]), together with the emergence of the new paradigms of the metabolic engineering (Stephanopoulos [50], Bro and Nielsen [81]) and synthetic biology (Meyer et al. [31], Greber and Fussenegger [181]) will enhance the importance of the model-based biochemical optimization in biotechnology, making virtually possible any change in the cell proteome.

Acknowledgements

This work was supported by the Spanish Ministry of Education and Science, research Grant no. BEO2005-08898-C02-02. The contribution of Julio Vera is supported by the German Federal Ministry of Education and Research (BMBF) as a part of the project CALSYS-FORSYS under contract 0315264 (www.sbi.uni-rostock.de/calsys). The authors acknowledge the support of José Hormiga in the figure design.

References


