Improving metabolic flux estimation via evolutionary optimization for convex solution space

Jiusheng Chen¹, Haoran Zheng¹,³,*, Haiyan Liu²,³, Junqing Niu¹, Jianping Liu², Tie Shen², Bin Rui² and Yunyu Shi²,³

¹Department of Computer Science and Technology, ²School of Life Science and ³Department of Systems Biology, University of Science and Technology of China, Hefei, China

ABSTRACT
Motivation: Flux estimation by using ¹³C-labeling pattern information of metabolites is currently the only method that can give accurate, detailed quantification of all intracellular fluxes in the central metabolism of a microorganism. In essence, it corresponds to a constrained optimization problem which minimizes a weighted distance between measured and simulated results. Characteristics, such as existence of multiple local minima, non-linear and non-differentiable make this problem a special difficulty.

Results: In the present work, we propose an evolutionary-based global optimization algorithm taking advantage of the convex feature of the problem’s solution space. Based on the characteristics of convex spaces, specialized initial population and evolutionary operators are designed to solve ¹³C-based metabolic flux estimation problem robustly and efficiently. The algorithm was applied to estimate the central metabolic fluxes in Escherichia coli and compared with conventional optimization technique. Experimental results illustrated that our algorithm is capable of achieving fast convergence to good near-optima and maintaining the robust nature of evolutionary algorithms at the same time.

Availability: Available from the authors upon request.

Supplementary information: Colour versions of the figure are available online as a part of the Supplementary data.

1 INTRODUCTION
Metabolic flux analysis (MFA) has become an indispensable tool in metabolic engineering. Its goal is to accurately quantitate the metabolite conversion rates (or fluxes) through biochemical reactions in the major pathways of a microorganism from measurement data representing the in vivo metabolic state of the cell (Wiechert, et al., 2001). Information about reaction rates is invaluable for characterizing different strains or different physiological states of a microorganism. Currently, the most accurate information about the fluxes in a complex biological system can be obtained based on the carbon labeling experiments (CLEs) where a mixture of naturally and ¹³C-labeled substrates is fed to the cell, then the ¹³C atoms are distributed all over the metabolic network and can be observed by measuring the resulting nuclear magnetic resonance (NMR) and mass spectrum (MS) of metabolic products and intermediates. The measurement data provide a large amount of additional information to quantitate the intracellular fluxes. This method has been successfully applied in many cases ranging from pathway optimization in metabolic engineering (Stephanopoulos, et al., 1998) and from characterization of physiology of an organism (Kelleher, 2001) to more efficient drug design for human diseases such as cancer (Boros, et al., 2004). Application of MFA requires a combination of ‘wet’ experiment and ‘dry’ experiment. Herein ‘wet’ experiment refers to MFA-oriented cell cultivation and measurement. When both metabolic and isotopic steady state are reached, the set of measured data may contain extracellular metabolic fluxes and ¹³C-labeling patterns. ‘Dry’ experiment means the process of flux calculation, which generally consists of automatic generation of CLE mathematical model, CLE simulation based on the model constructed and flux distribution estimation by fitting measured data iteratively (Wiechert, et al., 2001). Among these steps, the most challenging part is flux distribution estimation, which in essence corresponds to a constrained non-linear optimization problem with high-dimensional solution space, non-differentiable objective function and existence of multiple local optima. Several optimization methods had been developed to handle it, such as a simplex method, a simulated annealing algorithm and a gradient based Newton-like algorithm (Wiechert, 2001), an evolutionary strategy (Christensen, and Nielsen, 2000; Gombert, et al., 2001), a genetic algorithm (Zhao, and Shimizu, 2003), or an optimization technique relying on spatial branch and bound search (Riascos, et al., 2005). Each of the currently available algorithms for metabolic flux estimation presents some advantages and drawbacks with respect to robustness and performance. Compared with mathematical programming-based methods, evolutionary algorithms (EAs) are especially suitable for solving the problem because of their robust adaptive high-dimensional search ability, but they generally suffer from high computation effort and slow convergence to a good near optimum (Yao et al., 1999). Moreover, the metabolic flux estimation problem itself is especially sensitive to the size of

*To whom correspondence should be addressed.
metabolic network, for example, the complexity of isotopomer-based mathematical model increases exponentially with the number of metabolite’s carbon atoms in the network (Wiechert et al., 1999), usually large metabolic model is out of reach of CLE-based metabolic flux estimation. So far, the effective solution to this problem is still open for study, it is of great importance to design a powerful algorithm that can estimate metabolic fluxes both robustly and efficiently for the further development of CLE-based MFA.

In this work, we present an efficient evolutionary-based optimization algorithm for metabolic flux estimation. By taking advantage of the convex feature of the target minimization problem’s solution space, our algorithm’s convergence to optima is speed up greatly, therefore good performance can be achieved while still maintaining the robust nature of EAs. This methodology is then applied to the central metabolism of *Escherichia coli* (E. coli) and compared with a conventional constraint handling technique presented by Runarsson and Yao (2000).

The structure of the article is as follows. In the next section, the CLE-based metabolic flux estimation problem is introduced and modeled as a constrained least-squares minimization problem. Section 3 gives an analysis of characteristics of the flux space. The evolutionary-based metabolic flux estimation algorithm exploiting these characteristics is presented in Section 4. In Section 5, a case study is presented and Section 6 gives the results and the detailed performance analysis of the proposed algorithm. Finally, conclusions are presented in Section 7.

## 2 MODELING CLE-BASED METABOLIC FLUX ESTIMATION

The interconnectivity of metabolites within a network of biochemical reactions is given by reaction equations defining the stoichiometric conversion of substrates into products for every reaction. Since system is at steady state and the absolute metabolic pool sizes do not change over time, the sums of incoming and outgoing fluxes must be equal for every reaction. Since system is at steady state and the absolute metabolic pool sizes do not change over time, the sums of all reaction denoted by $i$. The vector $v$ refers to the relative activity of each flux. Each reaction step $i$ is represented by a pair of ‘natural fluxes’ $v_{net}^i$ (forward flux) and $v_{net}^{-i}$ (backward flux), and all ‘natural fluxes’ are additionally required to be non-negative. For a more intuitive representation, they are transformed into ‘application fluxes’ (Wiechert and de Graaf 1997) $v_{net}^i$ (net flux) and $v_{ch}^i$ (exchange flux).

$$v_{net}^i = v_{net}^+ - v_{net}^{-}$$

$$v_{ch}^i = \min(v_{net}^i, v_{net}^{-i})$$

In order to overcome the numerical difficulties (Schmidt et al., 1999) and to use the properties of linearized statistical analysis (Wiechert et al., 1997), the rescaled exchange fluxes (or exchange coefficients) $v_{ch}^{i[0,1]}$ were introduced by (Wiechert et al., 1997):

$$v_{ch}^{i[0,1]} = \frac{v_{ch}^i}{\beta} + v_{ch}^i$$

with $\beta$ being a constant of the order of magnitude of $v_{net}^i$.

From a computational standpoint, it is more convenient to deal with independent flux variables, also called free fluxes (Schmidt et al., 1997; Wiechert et al., 1997), rather than with all individual fluxes. The number of free fluxes is usually much smaller than the number of individual fluxes, which significantly reduces the computational time of simulation. Based on Equation (1), individual fluxes can be expressed in terms of free fluxes:

$$v = N \cdot v_{free}$$

Here, $N$ is the null space of matrix S, and $v_{free}$ is the vector of free fluxes. Generally, there is not a unique null space matrix for any given stoichiometric matrix. The size of the null space matrix and the number of free flux variables, however, are fully determined by the rank of the stoichiometric matrix. With $r = \text{rank}(S) \leq n$, the null space matrix is a $n \times (n - r)$ matrix and the number of free fluxes equals $n - r$.

Equation (1) defines an under-determined system with $n - r$ free degrees. In order to quantitate all intracellular fluxes in the central metabolism of a microorganism, two kinds of method have been developed. The first method solely relies on the known stoichiometry of a given biochemical reaction network with only measured extracellular fluxes as input data. However, this method cannot deal with complex biological systems containing reversible reactions, parallel pathways and internal cycles (Schmidt et al., 1998). Recognizing the shortcomings of the pure stoichiometric method, it becomes clear that more information is required to complement the extracellular flux data. This meant the advent of the second method which is based on CLEs. CLEs can provide a large amount of additional information to quantitate the intracellular fluxes from the measurements of isotope in the intracellular metabolic pools by techniques such as GC-MS and NMR. The mathematical model, used for simulating CLE, comprises the complete set of isotopomer balances describing the dependency between the intracellular fluxes and the stationary isotopomer distribution.

$$\mathbf{f}(v, x) = 0$$

$x$ is the vector of all isotopomer fractions in the system. By Equation (5) the model above can be transformed into:

$$x = x(v_{free})$$

It shows that the stationary isotopomer distribution can be expressed as a function of the free fluxes. Intracellular metabolic fluxes are determined from these models by minimizing the difference between the observed and the simulated measurements. Flux estimation is in essence a constrained least-squares minimization problem:

$$\min_{v_{free}} (m(v_{free}) - m_{obs})^T \cdot \Sigma^{-1} \cdot (m(v_{free}) - m_{obs})$$

s.t. $W \cdot v_{free} \geq w$

$$v = N \cdot v_{free}$$

Here, $N$ is the null space of matrix S, and $v_{free}$ is the vector of free fluxes. Generally, there is not a unique null space matrix for any given stoichiometric matrix. The size of the null space matrix and the number of free flux variables, however, are fully determined by the rank of the stoichiometric matrix. With $r = \text{rank}(S) \leq n$, the null space matrix is a $n \times (n - r)$ matrix and the number of free fluxes equals $n - r$.
The objective function is the covariance-weighted sum of squared residuals, \( m(v_{\text{free}}) \) is the vector of simulated measurements coming from Equations (5) and (7), \( m_{\text{obs}} \) is the vector of experimental data containing both labeling pattern measurements and extracellular rate measurements, and \( \Sigma_m \) is the measurement covariance matrix with measurement variances located on the diagonal. Equation (9) defines the feasible interval for every flux values. For example, the valid values for every rescaled exchange fluxes are in \([0,0.99]\), and net flux values for every irreversible reactions must be greater than or equal to 0.

The constrained minimization problem corresponding to CLE-based metabolic flux estimation features non-linear optimization, existence of multiple local minima, linear inequality constraints and non-differentiable objective function.

3 ANALYSIS OF FLUX SPACE

The steady-state flux space is a finite polytope containing all possible steady-state flux distributions as defined by stoichiometric balance Equation (1) and flux bound limits. The edges of the flux cone are defined by the network’s extreme pathways. These extreme pathways are ‘capped off’ by the upper limit values \( F_{\text{max}} \) for the reactions and the flux cone becomes a closed polytope. This polytope is referred as the feasible space or solution space, while reference space is the hypercube surrounding the polytope whose edges are defined by each reaction’s \( V_{\text{max}} \) (Wiback, et al., 2004). On condition of higher dimensions, the feasible solution space is so small, compared with the reference hypercube. If optimization algorithm conducts its search process within the reference space rather than the feasible space inside, it will be extremely difficult to find an initial feasible solution, let alone the optimal flux distributions. So it is important for the algorithm to search only the feasible space for the sake of efficiency. First, it guarantees that all solutions produced are feasible. Second, it shrinks the region, which the algorithm should explore, greatly, and spares the algorithm from the need to search the vast infeasible space. Finally, it handles the problem’s constraints implicitly, the original constrained minimization problem becomes unconstrained, so complicated constraint handling techniques are not needed.

Naturally, the next question is how to ensure that the algorithm always conducts its exploration within the feasible space. We have designed two special evolutionary operators (crossover operator and mutation operator) to accomplish it. Because the feasible space is a convex space due to the linearity of the constraints. There are two important characteristics of convex spaces, which play an essential role in the definition of these operators:

1. Let \( F \) be a convex set, for all \( v_1 \) and \( v_2 \) in \( F \) and all \( \alpha \) in the interval \([0, 1]\], the point \( \alpha \cdot v_1 + (1 - \alpha) \cdot v_2 \) is in \( F \).

2. For every point \( v_0 \) in \( F \) and any line \( p \) such that \( v_0 \) in \( p \), \( p \) intersects the boundaries of \( F \) at precisely two points, say \( l_p^0 \) and \( u_p^0 \), and \( l_p^0 \geq u_p^0 \).

4 EVOLUTIONARY-BASED OPTIMIZATION ALGORITHM

4.1 Algorithm procedure

EAs utilize principles of natural selection and are robust adaptive search schemes suitable for searching non-linear and high-dimensional spaces. This class of algorithms is being increasingly applied to obtain optimal or near-optimal solutions to many complex real-world optimization problems.

The procedure of the proposed algorithm (see Algorithm 1) is similar to other EAs, except that the algorithm’s components are tailored towards metabolic flux estimation problem based on the characteristics of convex space.

4.2 Initial population

In general, EA performs a multi-directional search by maintaining a population of potential solutions. Algorithm generates an initial population of \( \lambda \) individuals, each individual, \( v_{\text{free}} \), \( i = 1, \ldots, \lambda \), is a real-valued vector containing \( n \) elements, denoting a vector of \( n \) free flux values. Since \( v_{\text{free}} \) determines a flux distribution in metabolic network uniquely, each individual in population means a possible flux distribution.

In order to exploit the convex feature of the optimization problem’s feasible solution space better and to explore the feasible region more effectively, the initial population is endowed with two features. First, all individuals belong to the initial population are feasible, namely, all of them satisfy Equation (9). Second, all individuals are located in the boundary of feasible region randomly, intending to make the initial population spread the region widely, although it is not always the case.

To obtain a solution of Equation (9), consider the following linear program with \( n + 1 \) variables at first:

\[
\begin{align*}
\max & \quad v_{\text{free}} \cdot x_0 - x_0 \\
\text{s.t.} & \quad W \cdot v_{\text{free}} + \begin{pmatrix} x_0 \\ x_0 \\ \vdots \\ x_0 \end{pmatrix} \geq w \\
& \quad x_0 \geq 0
\end{align*}
\]

\[
(10)
\]

Algorithm 1 Evolutionary-based metabolic fluxes estimation algorithm

```plaintext
procedure EVOLUTIONARY_ALGORITHM

\( t \leftarrow 1 \) \( \triangleright \) \( t \) records generation number

Initialize population \( P(t) \) \( \triangleright \) see Section 4.2

Evaluate \( P(t) \)

repeat

\( P_{\text{cro}}(t) \leftarrow \text{Cross} \ P(t) \) \( \triangleright \) see Section 4.3

\( P_{\text{mut}}(t) \leftarrow \text{Mutate} \ P(t) \) \( \triangleright \) see Section 4.4

Evaluate \( P_{\text{cro}}(t) \) and \( P_{\text{mut}}(t) \)

Select \( P(t + 1) \) from \( P(t) \), \( P_{\text{cro}}(t) \) and \( P_{\text{mut}}(t) \) \( \triangleright \) see Section 4.5

\( t \leftarrow t + 1 \) \( \triangleright \) Increase generation number by 1

until termination-condition

end procedure
```

1117
Then we claim that program, given as in Equation (9), is feasible if and only if the optimal objective value of Equation (10) is 0.

Suppose \( \nu_{\text{free}} \) is a feasible solution. Then Equation (10) is equivalent to program

\[
\max_{x_0} -x_0 \text{ s.t. } x_0 \geq 0,
\]

clearly, its optimal objective value is 0. So, \( x_0 = 0 \) combined with \( \nu_{\text{free}} \) is the optimal solution to Equation (10).

Conversely, suppose that the optimal objective value of Equation (10) is 0. Then \( x_0 = 0 \), and the values of the remaining variables satisfy the constraints of Equation (9).

So, the initial feasible flux distribution of CLE-based MFA can be obtained by solving the linear programming problem defined as Equation (10). Methods such as simplex and interior point can be employed to perform this task.

To encourage the exploration of the whole feasible space, the initial population is constructed by having each individual located in the boundary of the space. If there exists a solution to Equation (9) with one or more inequality constraints are active, it is said this solution is located in the boundary.

Such a solution can be obtained simply by changing one or more inequality constraints in Equation (9) to corresponding equality constraints randomly. Then the linear programming method introduced above can still be used so long as an equality constraint is treated as two inequalities. For example, \( W \cdot \nu_{\text{free}} = w \) is equivalent to \( W \cdot \nu_{\text{free}} \geq w \) and \(-1 \cdot W \cdot \nu_{\text{free}} \geq -1 \cdot w\). In case of contradictory constraints, linear programming cannot produce optimal result, more than one tries may be needed.

4.3 Crossover operator

To exploit the convex flux space, arithmetical crossover is adopted in our algorithm. Let crossover ratio \( p_{\text{cr}} \) be the proportion of crossover individuals to the population. For two parent individuals \( \nu_{\text{free}}^i \) and \( \nu_{\text{free}}^j \), they are crossed resulting in two offspring:

\[
\begin{align*}
\nu_{\text{free}}^i &= \alpha \nu_{\text{free}}^i + (1 - \alpha) \nu_{\text{free}}^j \\
\nu_{\text{free}}^j &= (1 - \alpha) \nu_{\text{free}}^i + \alpha \nu_{\text{free}}^j
\end{align*}
\]

where \( \alpha \in [0,1] \). From the first characteristic of convex set, it can be guaranteed that the two offspring individuals \( \nu_{\text{free}}^i \) and \( \nu_{\text{free}}^j \) are feasible as long as their parents \( \nu_{\text{free}}^i \) and \( \nu_{\text{free}}^j \) are feasible. So it spares us the efforts to validate the feasibility of offspring generated after crossover operation.

A further analysis of this crossover operator explains the need of the specialized initial population (see Section 4.2) and mutation operator (see Section 4.4). Take Figure 1 for example, considering the areas \( S^i \) and \( S^j \), which are covered by the first population (A, B, C, D and E) and the second population (A, F, G and H) respectively, we claim that \( S^i \subseteq S^j \) because the new individual generated through crossover operation cannot lie outside the area covered by its parent population according to characteristics of convex space. And the following conclusion can be drawn further:

\[
\forall i < j: S^i \subseteq S^j
\]

where \( i \) and \( j \) are evolutionary generation number, \( S^i(S^j) \) denotes the area covered by the \( i(\text{or } j) \)-th generation population.

Equation (12) shows that the area explored by latter population is only a subset of the area explored by the previous population. If the proposed crossover is the only evolutionary operator adopted, the larger the algorithm progresses, the smaller the space that the algorithm explored becomes. The algorithm tends to do local area search, and search converges to a point in the space finally. This search strategy undermines the algorithm’s global exploration ability, so the algorithm must employ another mechanisms to complement the crossover operation.

First, the impact of initial population on the algorithm’s performance cannot be neglected. Suppose point \( I \) is an optimal solution in our example (see Fig. 1). Nevertheless, the optimum will never be reached. \( I \notin S^i \), the space searched latter is \( S^j \), and \( S^j \subseteq S^i \) according to Equation (12), so \( I \notin S^j \). \( I \) is beyond the area the algorithm explores. It is definitely impossible that this point can be found. To avoid this shortcoming, the area covered by the initial population should be as wide as possible in the hope that the area contains the optimal solution(s). This is the reason for the construction of special initial population in Section 4.2.

Second, mutation operation is vital for the algorithm. The combination of the crossover and the mutation operation can cause the relationship defined in Equation (12) no longer established. To a certain extent, mutation strengthens algorithm’s global exploration ability. For example, even the point \( I \) does not belong to the space covered by the initial population, \( I \notin S^i \), it is still possible that it can be found after some period of evolutionary process.
4.4 Mutation operator

The first feature of the mutation operation (see Algorithm 2) is that the children generated through mutation of feasible parents are also feasible. It is called a closure operator. Suppose \( v_{\text{free}} \) is a feasible solution. According to the second characteristic of convex set, there certainly exists an interval in which the \( j \)-th element of vector \( v_{\text{free}} \) can vary while still keeping itself feasible. This interval is denoted as the slack interval of the \( j \)-th element of vector \( v_{\text{free}} \) (see Fig. 2). When mutating the \( j \)-th element of individual \( v_{\text{free}} \), it is sure for certain that the after-mutated individual is also feasible as long as \( v_{\text{free}} \) is feasible and the after-mutated value of the \( j \)-th element belongs to its slack interval. When mutating multiple elements of a vector, these elements are mutated in turn as if only one element needs to be mutated one time. From the definition of crossover operation above, crossover is a closure operator as well. Both crossover and mutation are specially designed closure operators which are able to keep the search of the algorithm always inside the feasible region, spare the efforts to explore the vast infeasible space. So, these operators help the algorithm enhance search efficiency.

Another feature of the mutation operation is controllability. There are two parameters relating to mutation operation. The first is \( P_{\text{mut}} \), which is the ratio of mutation individuals to the whole population. The individuals to be mutated are chosen randomly from the parent population according to \( P_{\text{mut}} \). It can be implemented from mutating the whole population to mutating only a part of it by tuning \( P_{\text{mut}} \). For every individual vector, it can be mutated by varying one of its element (called single-point mutation) or some of its element (called multiple-point mutation), or even all of its element (called full-point mutation). We did not limit which mutation method should be used in our algorithm beforehand. Instead, a more flexible strategy was adopted. That is the core of the second parameter \( P_{\text{ene}} \) by tuning which we can implement all these three mutations. The parameters \( P_{\text{mut}} \) and \( P_{\text{ene}} \) together contribute to the controllability of the mutation operation.

Algorithm 2 Mutation operation

```plaintext
procedure MutateSingleElement(\( v_{\text{free}}, j \))
    \( \Rightarrow \) Mutate the \( j \)-th element of vector \( v_{\text{free}} \)

    \[ \text{slack interval for the} \quad \text{\( j \)-th element of} \quad \text{\( v_{\text{free}} \)} \]

    repeat
        move \( \leftarrow \) \( N(0, \min(\text{slack interval}) - \text{lb, ub}) - v_{\text{free}}(j)) \)
        \( \Rightarrow \) normally distributed one-dimensional random number
        newValue \( \leftarrow v_{\text{free}}(j) + \text{move} \)
    until newValue \( \in [\text{lb, ub}] \)
    \( v_{\text{free}}(j) \leftarrow \text{newValue} \)
    return \( v_{\text{free}} \)
end procedure
```

4.5 Selection

Population of children is formed through crossover and mutation on parent population. While selecting individuals from the population of children and the parent population to make up of the next-generation population, we took into account the fitness (defined as the reciprocal of Equaiton (8)) of each individual and its distance (Hamming distance was used) to the center of the population together, instead of using the fitness factor as the single criterion as most EAs do.

\[
\text{rank}(v_i) = \omega \times \text{rank}^{\text{fit}}(v_i) + (1 - \omega) \times \text{rank}^{\text{dis}}(v_i) \tag{13}
\]

\( \text{rank}^{\text{fit}} \) and \( \text{rank}^{\text{dis}} \) are individual evaluation results according to criteria of fitness and distance, respectively. The final evaluation result is rank. \( \omega \in (0.5, 1] \), which is called fitness weight value, can be used to adjust the effect between fitness and distance. To accelerate convergence rate, \( \omega \) should take some big values approximating 1, but it may reduce individuals’ diversity, algorithm becomes greedy, therefore risks of getting stuck in local optima. To encourage global exploration in the whole feasible space, \( \omega \) should take some small values approximating 0.5, but any value <0.5 would cause the algorithm unable to converge, therefore should be avoid. In general, the higher an individual’s fitness is and the more marginal its location is, the more likely it will survive, vice versa. This selection mechanism helps the algorithm avoid getting trapped by local optima, enlarge its search scope, strengthen diversity among individuals in population and enhance the global exploration ability.

5 CASE STUDY

For the labeling experiment, the wild type \( E. \text{coli JM101} \) was cultivated using 1.31 L modified M9 medium (3.00 g/L glucose, 42 mM Na₂HPO₄, 22 mM KH₂PO₄, 19 mM NH₄Cl, 4.5 Selection

4.4 Mutation operator

4.3 Selection

4.2 Selection

4.1 Selection

4.0 Selection
9 mM NaCl, 1 mM MgCl2, 0.1 mM CaCl2, 100 mg/L VB1) at 37°C in a 2.5 L chemostat (BIOSTAT B, B.Braun biotech), which was equipped with pH, pO2, level, antifoam and temperature sensor at a dilution rate 0.15 h⁻¹. pH was kept to 7.0 ± 0.5 by addition of 4M NaOH, and pO2 was maintained at (40 ± 5)% throughout the experiment. After about six volumes, the culture system reached steady state, indicated by constant OD600, pO2 and NaOH consumption rate with glucose specific uptake rate about 0.50 g h⁻¹ (g DW)⁻¹ (DW means dry weight). The same medium, except 10% [U-13C]-labeled and 90% naturally labeled glucose, was then used for labeling for two volumes. 

At the end, labeled cells (300–400 ml culture, cell concentration was 0.78 ± 0.02 mg DW/ml) were collected by centrifugation, washed twice with deionized water, then hydrolyzed in 10 ml 6 M HCl in a sealed pyrex tube at 110°C for 24 h. After filtration through 0.22μm filter, the hydrolysate was lyophilized, and dissolved in 0.1 M HCl in D2O as NMR sample.

2D [13C, 1H]-HSQC spectra were recorded with a Bruker Avance 500 NMR spectrometer as described by Flores et al. (2002). Simulated values are according to the estimated flux distributions in Figure 3. Measurements with too low signal-to-noise ratio are marked with an asterisk (*).

### Table 1. Relative intensities of 13C resonance multiplet components, singlet (S), doublet (D1, D2) and doublet of doublets or triplets (DD) as determined by 2D [13C, 1H]-HSQC spectra of the amino acids in cell hydrolysate of E. coli

<table>
<thead>
<tr>
<th>Carbon</th>
<th>Measured</th>
<th>Simulated</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>S</td>
<td>D1</td>
</tr>
<tr>
<td>Ala 2</td>
<td>0.14</td>
<td>0.07</td>
</tr>
<tr>
<td>Ala 3</td>
<td>0.23</td>
<td>0.77</td>
</tr>
<tr>
<td>Glu 2</td>
<td>0.29</td>
<td>0.33</td>
</tr>
<tr>
<td>Glu 3</td>
<td>0.58</td>
<td>0.41</td>
</tr>
<tr>
<td>Glu 4</td>
<td>0.23</td>
<td>0.63</td>
</tr>
<tr>
<td>Glu 5</td>
<td>0.22</td>
<td>0.78</td>
</tr>
<tr>
<td>Ser 2</td>
<td>0.13</td>
<td>0.16</td>
</tr>
<tr>
<td>Ser 3</td>
<td>0.30</td>
<td>0.70</td>
</tr>
<tr>
<td>Thr 2</td>
<td>0.28</td>
<td>0.33</td>
</tr>
<tr>
<td>Thr 3</td>
<td>0.27</td>
<td>0.56</td>
</tr>
<tr>
<td>Thr 4</td>
<td>0.51</td>
<td>0.49</td>
</tr>
<tr>
<td>Ile 2</td>
<td>0.37</td>
<td>0.50</td>
</tr>
<tr>
<td>Ile 3</td>
<td>0.45</td>
<td>0.48</td>
</tr>
<tr>
<td>Ile 4</td>
<td>0.51</td>
<td>0.49</td>
</tr>
<tr>
<td>Ile 5</td>
<td>0.18</td>
<td>0.69</td>
</tr>
<tr>
<td>Leu 2</td>
<td>0.80</td>
<td>0.19</td>
</tr>
<tr>
<td>Leu 5</td>
<td>0.28</td>
<td>0.72</td>
</tr>
<tr>
<td>Lys 3</td>
<td>0.22</td>
<td>0.65</td>
</tr>
<tr>
<td>Lys 4</td>
<td>0.47</td>
<td>0.53</td>
</tr>
<tr>
<td>Lys 5</td>
<td>0.21</td>
<td>0.63</td>
</tr>
<tr>
<td>Lys 6</td>
<td>0.41</td>
<td>0.59</td>
</tr>
<tr>
<td>Val 2</td>
<td>0.19</td>
<td>0.69</td>
</tr>
<tr>
<td>Val 4</td>
<td>0.23</td>
<td>0.77</td>
</tr>
<tr>
<td>Phe 2</td>
<td>0.15</td>
<td>–</td>
</tr>
<tr>
<td>Phe 3</td>
<td>0.17</td>
<td>0.83</td>
</tr>
</tbody>
</table>

Although MS data was not used in our experiment due to implementation reasons, it should be noted that the proposed algorithm can be applied to all labeling data including both NMR and MS measurements. Because we have modeled the metabolic flux estimation problem as a uniform expression as defined in Equations (8)–(9), which is completely independent of the type of measured data.

The metabolic network considered includes: the Embden–Meyerhoff–Parnas (EMP) pathways, the pentose-phosphate (PP) pathways, the tricarboxylic acid (TCA) cycle as well as glyoxylate shunt (see Fig. 3). Anapleurotic pathways and some biosynthetic reactions leading to the formation of some amino acids are also included. The complete CLE model is available from the authors on request.

### 6 RESULTS AND DISCUSSION

The CLE model described above was used for a rigorous, quantitative analysis of E. coli. central metabolism using complete isotopomer models. During the computation, the exchange coefficients defined in Equation (4) were limited to unsteady state using the same methods as reported by van Winden et al. (2001). And the SD for all the measurements was assumed to be 0.05.

Although MS data was not used in our experiment due to implementation reasons, it should be noted that the proposed algorithm can be applied to all labeling data including both NMR and MS measurements. Because we have modeled the metabolic flux estimation problem as a uniform expression as defined in Equations (8)–(9), which is completely independent of the type of measured data.

The metabolic network considered includes: the Embden–Meyerhoff–Parnas (EMP) pathways, the pentose-phosphate (PP) pathways, the tricarboxylic acid (TCA) cycle as well as glyoxylate shunt (see Fig. 3). Anapleurotic pathways and some biosynthetic reactions leading to the formation of some amino acids are also included. The complete CLE model is available from the authors on request.
values between 0.00 and 0.99 to avoid extremely large exchange flux and save computation efforts (Schmidt et al., 1999). Comparative flux estimation experiments were performed between our algorithm and the constrained handling algorithm presented by Runarsson and Yao’s algorithm. StochRank, which is a well-known effective constrained optimization problem solving technique in the field of evolutionary optimization, searches the full reference space, spares the algorithm from the need to explore the vast infeasible space. Consequently, all solutions produced are infeasible, all constraints are handled implicitly, the algorithm conducts the search within the reference space, it is very likely that the solution found is infeasible due to the huge size difference between the feasible space and the reference space.

At the initial phase of evolution, it is extremely difficult for StochRank to locate the feasible region, there are few feasible individuals in the population. Then, StochRank gradually discovers the approximate location of the feasible region, the feasible solutions in population become more and more. Finally, evolutionary search converges to the feasible region, the whole population becomes feasible. The analysis above shows that the efficiency of StochRank is enhanced gradually, while FEA2 always maintains high search efficiency.

Table 2 compares StochRank and FEA2 during different evolutionary phase. The ‘Mean Best’ results once again indicate FEA2 is always capable of finding a better solution. The SD results show that the good performance of FEA2 is stable. So it can be concluded that FEA2 owns faster convergence speed, better optimization ability and more stable good performance.

7 CONCLUSION

In order to improve metabolic flux estimation in CLE-based MFA, this article presents an efficient evolutionary global optimization algorithm, which takes advantage of the characteristics of the target problem’s solution space. By designing special initial population and evolutionary operators, the algorithm conducts its search only within the feasible solution space, spares the algorithm from the need to explore the vast infeasible space. Consequently, all solutions produced are feasible, all constraints are handled implicitly, the algorithm has a much smaller search region. Naturally, this robust EA can achieve a good performance and the metabolic fluxes can
be estimated both robustly and efficiently. We applied the algorithm to estimate the central metabolic fluxes in *E. coli* and compared it with a conventional optimization algorithm. As a result, it is shown that the proposed algorithm is capable of solving CLE-based metabolic flux estimation problem better, faster and more stable.
And the proposed algorithm can also be applied to other problems with convex solution spaces.

ACKNOWLEDGEMENT

This work has been supported by the Knowledge Innovation Program of Chinese Academy of Science (No. KSCX2-SW-329).

Conflict of Interest statement. None declared.

REFERENCES


