Computers in Biology and Medicine 113 (2019) 103390

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

A non-dominated sorting Differential Search Algorithm Flux Balance Analysis (ndsDSAFBA) for *in silico* multiobjective optimization in identifying reactions knockout

Kauthar Mohd Daud[a], Mohd Saberi Mohamad[b,c,*], Zalmiyah Zakaria[a], Rohayanti Hassan[d], Zuraini Ali Shah[a], Safaai Deris[b,c], Zuwairie Ibrahim[e], Suhaimi Napis[f], Richard O. Sinnott[g]

[a] Artificial Intelligence and Bioinformatics Research Group, School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, 81310, Skudai, Johor, Malaysia
[b] Institute for Artificial Intelligence and Big Data, Universiti Malaysia Kelantan, City Campus, Pengkalan Chepa, 16100, Kota Bharu, Kelantan, Malaysia
[c] Faculty of Bioengineering and Technology, Universiti Malaysia Kelantan, Jeli Campus, Lock Bag 100, 17600, Jeli, Kelantan, Malaysia
[d] Software Engineering Research Group, School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, 81310, Johor Bahru, Malaysia
[e] Faculty of Electrical and Electronic Engineering, Universiti Malaysia Pahang, Pekan, Pahang, Malaysia
[f] Department of Cell and Molecular Biology, Faculty of Biotechnology and Biomolecular Sciences, Universiti Putra Malaysia, Universiti Putra Malaysia, 43400, Serdang, Selangor, Malaysia
[g] School of Computing and Information Systems, Melbourne School of Engineering, University of Melbourne, Victoria, 3010, Australia

ARTICLE INFO

Keywords:
Multi-objective evolutionary algorithms
Metabolic engineering
Flux balance analysis
Reaction knockout
Pareto dominance
Artificial intelligence
Bioinformatics

ABSTRACT

Metabolic engineering is defined as improving the cellular activities of an organism by manipulating the metabolic, signal or regulatory network. *In silico* reaction knockout simulation is one of the techniques applied to analyse the effects of genetic perturbations on metabolite production. Many methods consider growth coupling as the objective function, whereby it searches for mutants that maximise the growth and production rate. However, the final goal is to increase the production rate. Furthermore, they produce one single solution, though in reality, cells do not focus on one objective and they need to consider various different competing objectives. In this work, a method, termed ndsDSAFBA (non-dominated sorting Differential Search Algorithm and Flux Balance Analysis), has been developed to find the reaction knockouts involved in maximising the production rate and growth rate of the mutant, by incorporating Pareto dominance concepts. The proposed ndsDSAFBA method was validated using three genome-scale metabolic models. We obtained a set of non-dominated solutions, with each solution representing a different mutant strain. The results obtained were compared with the single objective optimisation (SOO) and multi-objective optimisation (MOO) methods. The results demonstrate that ndsDSAFBA is better than the other methods in terms of production rate and growth rate.

1. Introduction

In silico metabolic engineering is a famous method that offers several advantages, including reducing the costs and time taken, providing prior knowledge for wet lab experiments, and less laborious methodology. These advantages have enabled researchers to simulate and re-engineer organisms with the aim of improving and exploiting their capabilities, for example, optimising the production of high-demand industrial metabolites [1,2]. Constraints-based modelling (CBM) methods have been used to predict the behaviour of organisms, contributing important knowledge for predicting the effects of phenotypic and genotypic perturbations on the organisms themselves. In general, CBM imposes constraints on the metabolic network by assuming the organism is at a steady state, and the behaviour of microbial strains is predicted by maximising or minimising a particular objective function, such as growth rate or production rate [1,3].

Flux Balance Analysis (FBA) is one of the CBM methods that analyses the effect of genetic manipulations of large data by predicting the final higher steady-state of biological objective functions such as growth rate and production rate. Although the solution provided by FBA is non-unique, as it does not consider regulatory effects and metabolic concentrations. However, the existing genome-scale metabolic models (GSMM) are still incomplete, with a lack of regulatory and kinetic parameters [1,4]. Regardless of these imperfections, FBA is widely

* Corresponding author. Institute For Artificial Intelligence and Big Data, Universiti Malaysia Kelantan, City Campus, Pengkalan Chepa, 16100, Kota Bharu, Kelantan, Malaysia.
E-mail address: saberi@umk.edu.my (M.S. Mohamad).

https://doi.org/10.1016/j.compbiomed.2019.103390
Received 20 March 2019; Received in revised form 15 August 2019; Accepted 15 August 2019
Available online 16 August 2019
0010-4825/

used because it is able to determine the steady-state fluxes of the organisms as it does not require the above-mentioned parameters.

There have been several major advances in *in silico* metabolic engineering that take different approaches [5–10]. The advances in the development and refurbishment of genome-scale metabolic models of different organisms have resulted in the development of more robust methods. One of the developments is the design of a growth-coupled (GC) mutant strain. The GC mutant strain coupled the production of desired metabolites with growth, which makes the desired metabolites an essential by-product of growth [11]. The assumption is that an organism will evolve to maximise growth subsequent to the mutation [12,13]. Therefore, with coupling, the GC mutant strain will produce the desired metabolites at a higher rate as growth becomes the driving force for its production. Furthermore, the GC mutant strain is easy to implement in *in vivo* experiments due to selection pressure, whereby optimally growing mutants will surpass the non-optimally growing mutants [11]. Von Klamp and Klamt (2017) has shown that almost all metabolites are feasible for growth-coupled overproduction in five organisms; however, in many cases, this involves the deletion of an infeasible number of reactions/genes [11,14].

One of the earliest computational methods is OptKnock that predicts the reaction knockout strategies for improving the production of metabolites [7]. OptKnock is a bi-level optimisation algorithm that designs mutants with a higher production rate, below the optimal growth rate [15]. However, the simulation results are considered over-optimistic because OptKnock chooses a solution with the highest product yield [5]. Therefore, a number of computational strain optimisation methods have been developed to solve this problem. One of them is OptStrain, which includes non-native heterologous enzymes into the host [16]. OptReg allows the tuning of gene expression together with the reaction knockout [17]. FastPros identifies reaction knockout strategies by iteratively screening the shadow prices of the target products [12]. The more recent algorithm, GridProd has been developed, extending the idea of IdealKnock and pFBA [14]. GridProd conducts linear programming twice in order to design synthetic DNA, as a significant amount of time and energy is needed for knocking out several genes.

The aforementioned methods are formulated as a bi-level optimisation problem that outlines two objective functions to be optimised; namely a biological objective for inner optimisation and an engineering objective as its outer optimisation [7]. However, a method such as OptKnock transferred the bi-level optimisation into a single-level mixed-integer linear programme that can exponentially increase the computation time with the increase in problem dimensions [18]. Furthermore, the predicted flux distributions do not represent the long-term flux distributions that tend to optimise the growth rate only [19,20]. In addition, these methods produced one single solution at a time, whereby in multiple objective optimisations, a set of non-dominated solutions is required. A tremendous improvement has been made to FBA; however, FBA is only able to optimise one objective function, which is the same as the aforementioned methods. Nevertheless, the organism's systems perform numerous functions, such as protein secretion, detoxification and energy production. Thus there is a need to consider multiple and different objectives for more accurate mutant strains [21]. As an example, in *Escherichia coli*, the succinic acid production rate will be at its minimum level when the growth rate is 0, and vice versa.

Therefore, in this work, we investigate the trade-off relationship between the two objectives, namely the production rate of desired metabolites and its growth rate, by identifying a combination of knockout reactions that optimise both objectives. An approach based on Multi-objective Evolutionary Algorithms (MOEAs) is proposed to solve the optimisation of competing objectives. This method will improve a previous optimisation algorithm, DSAFBA, by incorporating the Pareto concept into the said algorithm to study the relationship between the production rate of ethanol, succinic acid and acetic acids towards the growth rate of *Escherichia coli* and *Zymomonas mobilis*. The results obtained are compared with previously developed methods including single objective optimisation (SOO), DSAFBA and other related methods. Also, we biologically validate the suggested reactions by cross-checking with related databases and biological journals.

This paper is organised as follows: the related multi-objective methods in the area of *in silico* metabolic engineering are overviewed in Section 2; the definition and concept of multi-objective optimisation are defined in Section 3. In section 4, we describe the proposed multi-objective method. We then report the results regarding the overproduction of succinic acid, ethanol and acetic acid in the two organisms, mentioned earlier, in Section 5. Meanwhile, the relationships, trade-offs and biological validation of suggested reactions towards the overproduction of desired metabolites are discussed in Section 6. Finally, we conclude the paper and give closing annotations as well as proposed possible future work in Section 7.

2. Previous multi-objective methods in *in silico* metabolic engineering

A multi-objective method in i*n silico* metabolic engineering of GSMM for optimising metabolite production was first modelled by Maia (2008), whereby the authors applied Strength Pareto Evolutionary Algorithm 2 (SPEA2) and Non-Dominated Sorting Genetic Algorithm II (NSGA-II) in identifying knockout reactions in *E.coli* to optimise the production rate and growth rate. Their findings have paved the way for the development of other multi-objective methods, including Linear Physical Programming-based Flux Balance Analysis (LPPFBA), Noninferior Set Estimation (NISE) with FBA, Genetic Design through Multi-objective Optimisation (GDMO) and Multi-objective Metabolic Engineering (MOME).

NISE is applied together with FBA to improve the production of poly-3 hydroxybutyrate in *E.coli.* According to Oh (2009), NISE is used to estimate the non-dominated, near-optimal solutions. Although this method is able to give a good approximation of conflicting objectives, it does not consider enzymatic information. Meanwhile, LPPFBA is developed with the aim of finding optimal solutions for mutually competing objectives. In a case study, LPPFBA was applied to hepatocyte function in a bioartificial liver system, specifically for optimising urea secretion and albumin, NADPH and glutathione synthesis [21]. Although this method is applicable for optimising more than 2 objectives, user supervision is still needed to define the degrees of significance for each conflicting objective.

Another related method, GDMO, was developed to solve the issues of high computational efforts and long time duration as well as data complexity. This method identifies optimum genetic manipulation designs that are able to optimise multiple cellular functions [22]. Most of the developed multi-objective methods mentioned above use other means to generate non-dominated solutions. As an example, the aim of NISE with FBA is to generate solutions for the Pareto curve [23]. However, the obtained solutions are not able to generate a non-convex Pareto graph because it is based on a weighting scheme. Thus, the subsequent methods apply the concept of the Pareto optimality ranking approach.

Pareto optimality and ϵ-dominance are used in synthetic biology to optimise the production of desired metabolites in *E.coli*. The use of these two methods has made it possible to generate different trade-offs between engineering and biological objectives, thus enabling researchers to acquire deeper biological information to perform genetic manipulations for industrial purposes [24]. These approaches were applied for the overproduction of 1,4-butanediol, myristoyl-CoA, malonyl-CoA, acetate and succinate. The study successfully identified different trade-offs between conflicting objectives, thus improving the results obtained by the single objective optimisation method [24]. Recently, a novel algorithm, namely multi-objective metabolic engineering (MOME), was developed to optimise ethanol production. Apart from identifying knockout genes, this method takes into

consideration the up and down-regulation of enzymes by using the Redirector framework [25].

All the above-mentioned methods, approaches, and algorithms aim to explore and identify a set of trade-offs between two or more objective functions for genetic manipulations and exploitations of organisms. Most of them revolve around *in silico* simulation whereby the results can be used as prior knowledge for a wet lab experiment. Apart from finding a combination of knockout reactions using multi-objective concepts, they also validated their results using a wet lab experiment [26]. Here, the production of target organic acids, including acetic acid, lactic and succinic acid can be maximised while the formation of by-products is minimised.

3. The concept of multi-objective optimisation (MOO)

In the real world, most problems require the optimisation of multiple objectives. The goal in multi-objective optimisation is to obtain a set of non-dominated solutions that are close to each other and well-distributed along the true Pareto front. Non-dominated solutions are solutions with *"win and lose"* situations among competing and conflicting objectives. The Non-dominated Sorting Genetic Algorithm (NSGA), introduced by Srinivas and Deb (1994) and improved by Deb (2002), is the first method that applies a non-dominated sorting strategy to optimise multiple objectives. Commonly, this approach is applied to find a near-optimal Pareto-set which consists of non-dominated solutions.

a) The multi-objective approach can be expressed as follows:

$$\max/\min\, y = F(x) = [f_1(x),\ f_2(x), f_3(x),\ ...f_k(x)]$$

$$\text{subject to: } e(x) = [e_1(x),\ e_2(x),\ e_3(x),\ ...e_n(x)],\ x \in \mathrm{P}$$

$$\text{where } x = \{x_1,\ x_2,\ x_3, ..., x_h\};\ x\ \in P$$

Where P is the solutions space, $f(x) \in F$ is the function to be optimised, k is the number of objectives involved, $e(x)$ is the constraints being imposed on the system, h is the number of variable parameters and n is the number of constraints. From the expressions and Fig. 1, there are h-dimensional decision variable parameters ($x_1,\ x_2,\ ...,x_h$) initialised in the solution space, P; and the task is to find a vector of x that optimises the set of k objective functions $F(x)$. The solution space is restricted by the vector of constraints, $e(x)$.

b) Dominance

In multi-objective optimisation (MOO), several solutions are generated simultaneously. Thus, it is crucial to determine acceptable and feasible solutions that satisfy the conflicting objectives without being dominated by other solutions. As it is difficult to conclude which solution is better than another solution, dominance has been applied in order to determine the goodness of a solution. Dominance is defined as the superiority of one solution compared with another solution. However, considering that there are vectors of solutions, trade-offs among the solutions are taken into consideration. The rule of dominance is as follows:

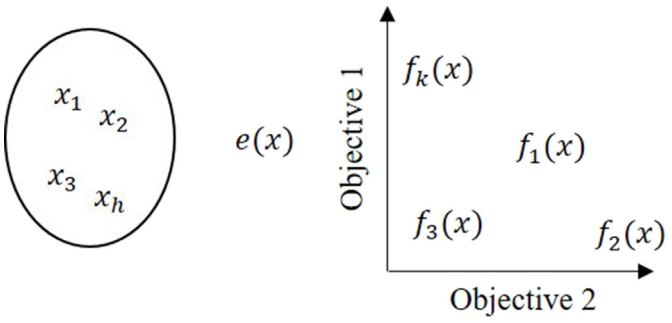

Fig. 1. Illustration of multi-objective optimisation.

$$\forall\ i \in \{1,2,...,k_{obj}\}: f_i(u_1) \le f_i(u_2) \quad (1a)$$

$$\exists\ i\ \in \{1,2,...,k_{obj}\}: f_i(u_1)\ < f_i(u_2) \quad (2a)$$

As shown in Equations (1a) and (2a), in the case of minimisation, solution u_1 is said to dominate solution u_2 when: (1) solution u_1 is no worse than the solution u_2 in all objectives, and (2) solution u_1 is better than the solution u_2 in at least one objective.

c) Non-dominated solution

A solution u is said to be a non-dominated solution which will be included in the Pareto set of $S = [u_i]$ where $i = 1,2, ...h$ if there is no solution u' that dominates u with condition that u and $u' \in$ P.

d) Pareto optimal solution

Suppose that the non-dominated solutions of the Pareto optimal solution, S is defined as $PF = \{f_1(u_1), f_2(u_2), ..., f_k(u_h)\},\ u \in$ P. These solutions are known as Pareto front solutions and when mapped to a graph, it is known as the Pareto graph.

There are two goals in determining the non-dominated solutions in MOO. First, the obtained non-dominated solutions should be as close as possible to the true Pareto solution and second, the solutions must be uniformly distributed along the Pareto graph. Based on the obtained non-dominated solutions, the decision makers may decide on the final solution according to their preference and the problem they are facing. There are several articles that have reviewed different MOO problems, including transportation asset management, bioinformatics and computational biology, data mining and others [23,27,28].

4. Materials and proposed method

Considering that this work involves biological information and computational methods, in this section we introduce the proposed multi-objective method for simulating the genetic manipulations of metabolic network models of the abovementioned organisms. We also illustrate the problem of finding suitable reactions for knockout using the proposed algorithm. Previously, we have developed an algorithm, DSAFBA, to enhance the production of the desired metabolites [29]. However, the algorithm is only limited to one objective function and the exploration and exploitation processes of the Differential Search Algorithm (DSA) does not take into account the trade-offs between two objectives. Therefore, ndsDSAFBA is proposed to improve DSAFBA by incorporating the concept of Pareto ranking.

4.1. Non-dominated sorting strategy

Previously, the standard method of solving the multi-objective problems was to treat them as a single objective problem, with classical optimisation algorithms being used to solve the problems [30,31]. As MOO generates a set of solutions, a good trade-off solution is needed for multiple and conflicting objectives. The goal of MOO is to obtain a list of non-dominated solutions that have a similar degree of importance among the objectives involved. As mentioned before, a solution is said to be non-dominated if there is no solution in the search space that dominates it. According to Refs. [32–34], there are several approaches to determine non-dominated solutions based on the selected category of the non-dominated solutions, including weighted sum, epsilon-constraint method and Pareto ranking.

Among them, the weighted sum is the most superior in terms of implementation as it does not require many parameters. However, it is only applicable to small-scale problems as it will require more computational costs for larger problems. Furthermore, in this approach, the assignment of weight is arbitrary as users can assign favourable weights to the solutions that are needed. In the case of the epsilon-constraint method, the non-dominated solutions that make up the Pareto curve are obtained from the iterative calculation of one objective value, while the other objectives were reduced from the maximum to the minimum [23]. The disadvantages of this approach are that it is susceptible to the

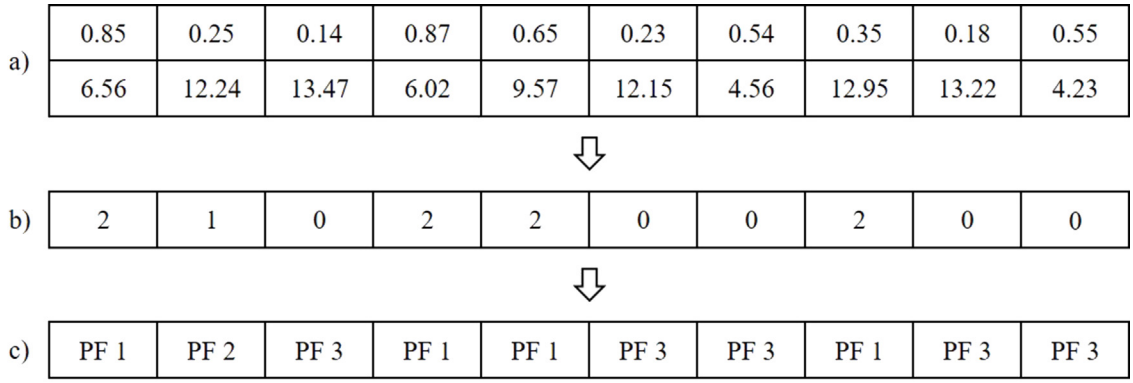

Fig. 2. The illustration of Pareto dominance and how to assign the Pareto front.

shape of the Pareto curve and will be biased towards one objective.

Therefore, in the proposed method, we consider Pareto dominance ranking for determining the non-dominated solutions, as it is the most popular approach due to its low computational complexity [33]. This method utilises the concept of dominance in assessing the goodness of a solution. Overall, Pareto dominance ranking works by assigning a rank to a solution depending on its dominance rule, which is the total number of solutions dominated by other solutions (as reviewed in Section 3). The lower the rank, the better the solution is. Fig. 2 indicates the flow of Pareto dominance. As indicated in this figure, we illustrate the fitness of 10 artificial-organisms having 2 objectives to be optimised (Fig. 2(a)). For each artificial-organism, dominance relation is stipulated to determine the number of times a solution dominates others (Fig. 2(b)). Based on the value of occurrences, then, each artificial-organism is assigned to its respective Pareto front (Fig. 2(c)).

4.2. Flux Balance Analysis

FBA is an approach that is widely used for studying metabolic networks of a GSMM by modelling and simulating the biological processes of cellular function [35,36]. A metabolic network model consists of reactions and metabolites that can be represented mathematically in the form of a stoichiometric matrix of reactions and metabolites. Analysis based on fluxes only requires stoichiometric information, which is readily available in the metabolic networks, thus providing an informative description of cell physiology as they calculate the contributions of different pathways towards a specific cellular function. FBA evaluates the optimality of a cell by optimising the cell's metabolic function, for instance, growth rate or energy expenditure.

4.3. Non-dominated sorting Differential Search Algorithm and Flux Balance Analysis (ndsDSAFBA)

Previously, the developed DSAFBA method was only able to optimise a single objective. As shown in Fig. 3, DSA is used to find several combinations of reactions randomly, while FBA is used to solve and analyse the metabolic model quantitatively by solving the stoichiometric model using linear programming (LP) with a preferred objective. Usually, the objective can be either the maximisation of production rate, the minimisation of by-product production, the minimisation of nutrient uptake or redox production or others. However, FBA can only be used to optimise a single objective and cannot be used to optimise multiple objectives. Thus, the multi-objective optimisation problems produce more advantages in the decision-making process as there are other alternative solutions being made.

In this work, Differential Search Algorithm and Flux Balance Analysis (DSAFBA) is improvised by adopting a non-dominated sorting strategy in the exploration process to optimise multiple objectives. The non-dominated sorting strategy is used to explore and identify Pareto solutions whereby the improvement of one objective can only be done at the loss of other objectives. The concept of multi-objectives is described in Section 3, whereas the concept of the non-dominating strategy is described in Subsection 4.1. Furthermore, this strategy is used to select non-dominated solutions among the solutions generated by DSA that are optimised by analysing the stoichiometric model using FBA.

DSAFBA was developed to identify knockout reactions for optimising product rates of ethanol, succinic acid and acetic acid [29]. DSAFBA is able to outperform other strong and reliable methods, including OptKnock, IdealKnock and ReacKnock, with a better production rate and growth rate. This is because the search strategy of DSA uses multiple particles in reaching the near-optimal solution without the inclination to go directly towards the best possible solution. Meanwhile, FBA is used to assess the phenotypic disturbances towards the metabolic systems. However, the developed algorithm can only be used to optimise a single objective. Also, DSAFBA is not able to identify and determine the relationship between the production rate and growth rate.

We adopt a non-dominated sorting strategy used in NSGA II [37] into DSAFBA to rank and determine the non-dominated solutions. The problem that we want to solve is how to select a subset of reactions from GSMM for maximising the production rate and its growth rate by turning off the activities of respective reactions. The flowchart of the proposed ndsDSAFBA method for solving knockout reactions in maximising the production rate and growth rate is shown in Fig. 4.

Fundamentally, the parameters for this algorithm are (1) n, the size of artificial-organisms; (2) *maxIter*, the maximum number of iterations to be performed that acts as the stopping criterion; (3) *maxKOs*, the number of knockouts allowed; (4) p_1 and p_2 are control parameters for exploration; and (5) d, the number of candidate reactions. This research used a binary variable representation that corresponds to the knockout of reactions in the model. The value 1 corresponds to the reaction being knocked out, while it is 0 otherwise. The aforementioned parameters are initialised beforehand.

There are seven steps involved in Fig. 4 and the description for each step is summarised as follow:

Step 1: Firstly, ndsDSAFBA initialises a superorganism, *SO*, of a matrix of size $n \times d$ with a binary number of 0 and 1, where n is the size of artificial-organisms while d is the number of candidate reactions for the knockout. The assignment of 0 and 1 are carried out randomly, and a total of 1 does not exceed the number of *maxKOs*. Hence, each artificial-organism represents as a mutant with respective knockout reactions. Fig. 5 shows the metabolic genotype representation of artificial-organisms. As shown in the figure, artificial-organism 1 or solution 1 suggests that reaction 1 and 3 as a knockout reaction, while the artificial-organism 2 suggests knockout reactions 2 and 4. After the initialisation, the fitness of each artificial-organism is evaluated. Step 2 explains the process of fitness evaluation.

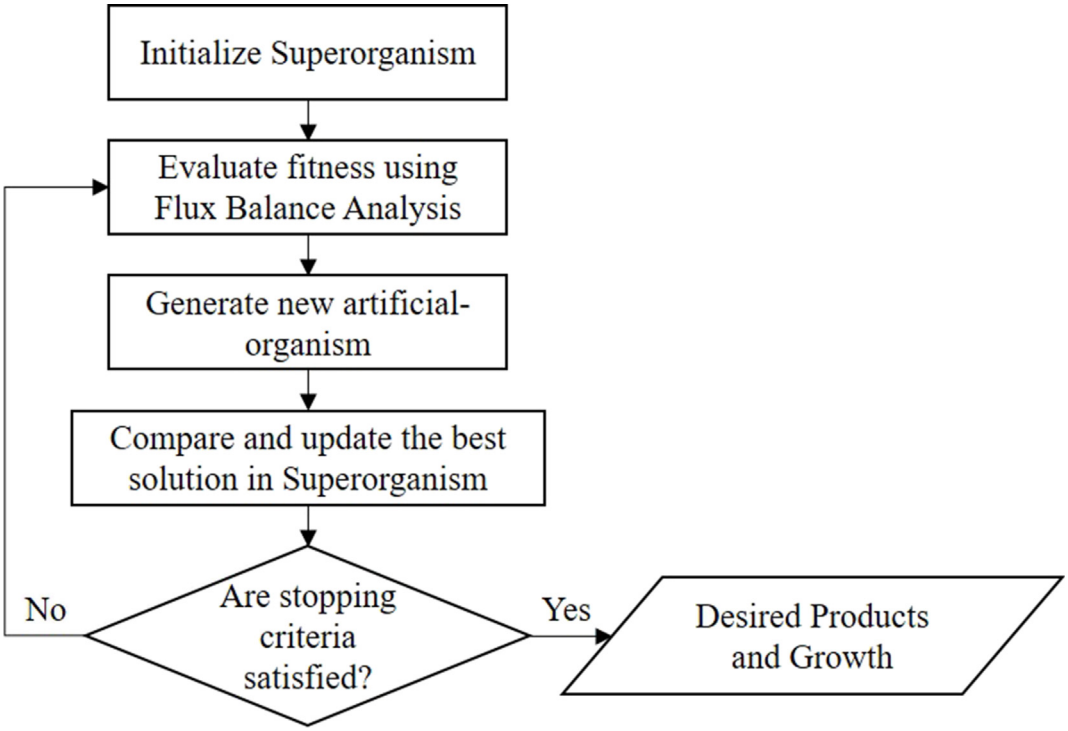

Fig. 3. Flowchart of DSAFBA. Flux Balance Analysis is hybridised into conventional DSA as an objective function to predict the effect of reactions knockout.

Step 2: We then determine which artificial-organism is the best by using FBA as the fitness function in ndsDSAFBA. The problem to identify near-optimal reaction knockout strategies from the metabolic networks can be formulated as follows: Suppose a stoichiometric matrix of m reactions and n metabolites that defines the linear relationship between flux vector (v) and concentration of metabolite (x) of a metabolic network. The flux distributions are evaluated by constraining the underdetermined system with lower and upper bound constraints of reactions, considering that the flux vector has infinite values. The formulation of FBA is represented below:

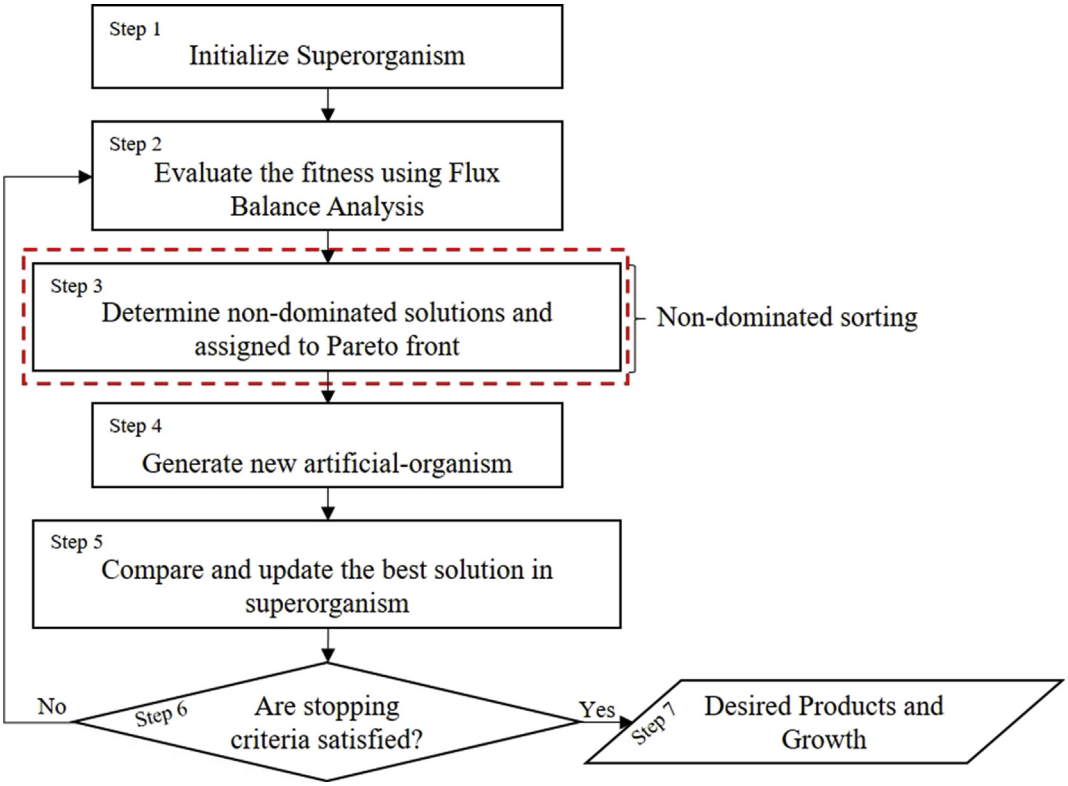

Fig. 4. Flowchart of ndsDSAFBA. Note: The red-dotted box is non-dominated sorting that is implemented into DSAFBA in order to determine the non-dominated solutions among multiple objectives.

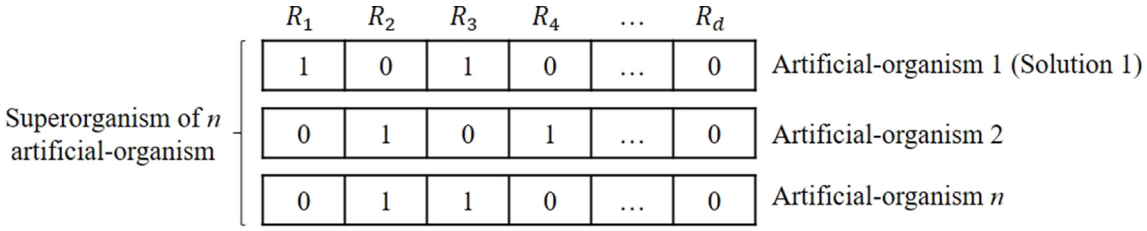

Fig. 5. Representation of artificial-organisms in ndsDSAFBA.

Flux vector $v = (v_1, v_2, ..., v_m)^T$ (1b)

Concentration vector $x = (x_1, x_2, ..., x_n)^T$ (2b)

Dynamic mass balance equation, $\frac{dx}{dt} = S \times v$ (3)

where T means transposed. The optimal knockout reactions that can improve the production rate of desired metabolites is then determined by using linear programming as follows:

$maximise\ Z = c^T x$ (4)

subject to:

$S \times v = 0\ lowerbound \leq x \leq upperbound$

where v is the flux vector, S is the stoichiometric matrix and c is the weight vector, corresponding to the amount of v contributes to the objective. The expression $(c^T x)$ is the objective function to be maximised or minimised. In this work, we optimise both the production rate (PR) and growth rate (GR) as stated:

$$Z^1 = GR = \frac{B}{t}\left(\frac{1}{hr}\right) \quad (5)$$

$$Z^2 = PR = \frac{P}{t}\left(\frac{mmol}{gDW \cdot hr}\right) \quad (6)$$

where Z^1 is the growth rate (GR) and B is the grams of biomass produced in hr^{-1}. Z^2 is the production rate (PR), P is the amount of production in mmol gDW^{-1} hr^{-1}. Z^1 and Z^2 can be depicted as a multi-objective optimisation problem that we need to maximise by identifying knockout reactions. Both objective functions are evaluated with respect to t, which is the time in hours. In this paper, the desired metabolites are ethanol, succinic acid and acetic acid (EX_etoh, EX_succ, EX_ac). Fig. 6 shows the flow of fitness calculation. A mutant with a growth rate greater than 0.1 is accepted to be a modified model (mutant). This is to ensure the survivability of the modified model. The lower bound growth rate for the mutant is set to be 10% from the value obtained from FBA, as suggested by Refs. [38,39]. This is because the production rate of desired metabolites is at 0 when the growth rate is at maximum.

Step 3: Considering that we want to know the relationship between growth rate and production rate as well as finding the trade-offs between these two objectives, the non-dominated strategy is applied to address the dominant superorganism. Each of the non-dominated solutions are assigned to the Pareto front based on its strength of dominance. The most dominated artificial-organism is assigned to the first Pareto front, while the least dominated artificial-organism is assigned to the last Pareto front. Herein, we maximise two objectives as shown in Equations (5) and (6), and the way of identifying the dominant solutions among the hundreds of solutions obtained is according to the formula below:

$$D = all(x^i \geq x^{i+1})\ AND\ any(x^i > x^{i+1}) \quad (7)$$

where

$$x^i = [Z_1, Z_2] \quad (8)$$

where D represents the dominance, and x is the artificial-organism with the size of $i = 1...n$, that contains Z^1 and Z^2, while n is the size of the artificial-organism. The figure below illustrates the process used in this study to find the near-optimal solution in an organism.

Fig. 7 depicts the rule used in the proposed method to determine the dominant solution. For instance, there are 6 artificial-organisms (x^1, x^2, x^3, x^4, x^5, x^6) with their respective growth rates and production rates (Z^1 and Z^2). In order to determine the dominant solution, Equation (7) is applied to each artificial-organism. For example, when x^3 is compared with the other artificial-organisms, the objectives Z^1 and Z^2 are compared with the objectives of other artificial-organisms. If it dominates the other solution, *Dom* is assigned to 1. In this figure, we only illustrate *Dom* for the third artificial-organism as an example. This process is continued until all objectives in each artificial-organisms have been assessed and compared with. As for dominance, D, the higher the value of dominance, the better the solution is. Therefore, in this example, the third artificial-organism, x^3 dominates the objectives in the sixth artificial-organism, x^6. However, the rest of the artificial-organisms do not dominate each other. Therefore, D for the third artificial-organism has the highest value of dominance which indicates that it dominates other solutions; hence it is considered as the dominant solution.

Step 4: In identifying another set of knockout reactions (in conventional DSA known as a stopover), the binary number of individuals may change along the process in order to generate a new artificial-organism. The reason for this lies in the fact that there are probably other combinations that may produce better results. ndsDSAFBA uses *Brownian-random walk* to generate new solutions by determining the *stopover, s*:

$$s = artificialorganism_i + \gamma.(donor - artificialorganism_i)$$

where *donor* is a randomly selected artificial-organism, γ is the gamma distribution scale factor that controls the positional changes among the members.

Step 5: Again, the growth and production rates of the desired metabolites for each new artificial-organism are evaluated and compared with the previous superorganism. The best solution is updated in the superorganism:

$$SO = \begin{cases} s, \ if\ y(SO) < y(s) \\ SO, \ otherwise \end{cases}$$

where $y(SO)$ and $y(s)$ are the fitness evaluations of the *Superorganism* and *stopover*, respectively. The greedy rule is used in selection strategy in ndsDSAFBA.

Step 6: The process continues until the stopping criterion is met, as defined by the user. In our case, the stopping criterion is the maximum iterations that have been defined.

Step 7: The final output of the proposed method is non-dominated solutions of the two objectives, production rate and growth rate. From that, we can deduce what reactions are being knocked out and hence identify the relationship between production rate and growth rate using the Pareto curve. As a conclusion, the outcome of the proposed method is a list of solutions which correspond to different mutants that optimise the production rate and growth rate while satisfying the constraints imposed on the system. In the next section, a detailed description of datasets and experiments conducted, as well as the results and

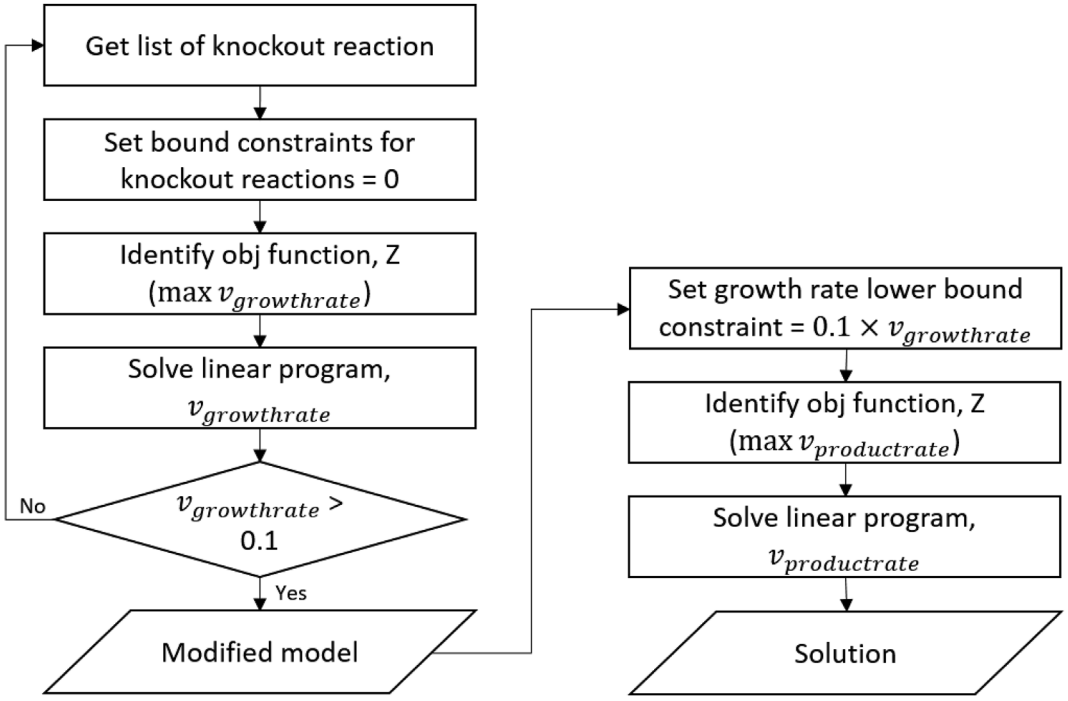

Fig. 6. Flow of fitness calculation.

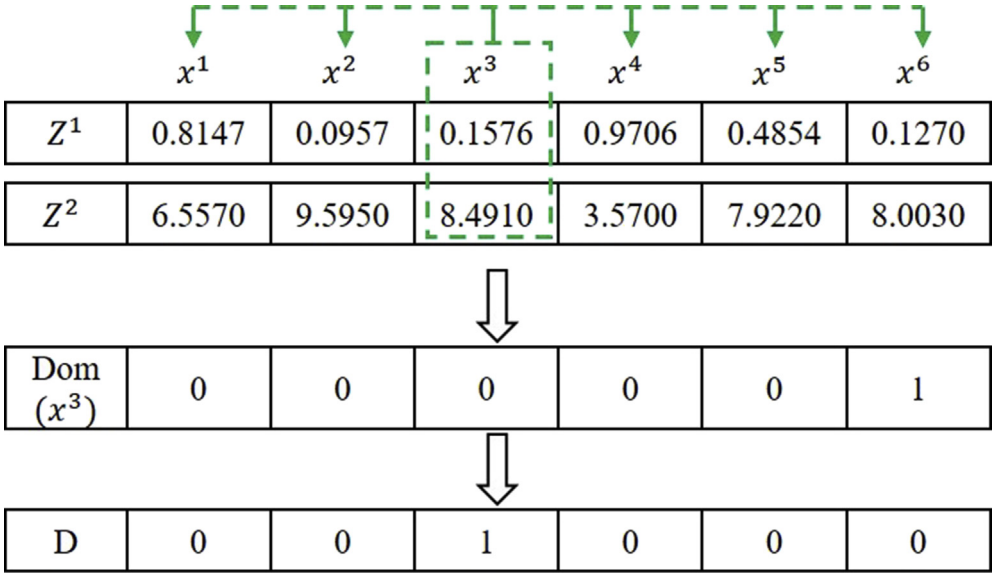

Fig. 7. Illustration of objective function evaluation and dominance used in this problem domain, where Z^1 and Z^2 represent the growth rate and production rate, respectively; *Dom* represents the count of dominance for each artificial-organism, *D* represents the dominance of a solution, while x^1 until x^6 are the artificial-organisms.

discussion, are presented.

4.4. Datasets and software requirements

Three GSMMs are used to test the proposed algorithm, which are *E.coli* core model, iAF1260 and iEM439. The first two GSMMs are from *Escherichia coli* and iEM439 is from *Zymomonas mobilis*. The reasons for this selection are *E.coli* has been extensively studied and used for product optimisation on an industrial scale, while *Zymomonas mobilis* is an ethanologenic bacteria which is in the spotlight due to its capability to produce ethanol. As mentioned earlier, GSMM consists of hundreds and thousands of reactions which make up a large searching space. This complicates the process of optimising and determining reactions that can enhance the production and growth rate. Thus, processing of the dataset is carried out beforehand to remove unnecessary reactions. The processing steps are based on computational approaches and biological assumptions. Eventually, this step may reduce the searching space and computational time. Table 1 shows the result of data pre-processing and all data were downloaded from a repository of the genome-scale metabolic model of various organisms, accessible at http://systemsbiology.ucsd.edu/InSilicoOrganisms/OtherOrganisms.

The study aims to optimise the production of ethanol, succinic acid and acetic acid and the growth rate of the *E.coli* core model, iAF1260 and iEM439. These metabolites are chosen based on their popularity in the industrial field, including food processing, drugs, medical sectors and others. MATLAB version R2013b is used in this study to implement

Table 1
Pre-processed model and candidate reactions for knockout.

Model	Raw ModelReaction	Metabolite	Candidate reactions for knockout	Target metabolite	Ref.
E.coli core	95	75	48	Succinic acid	[40]
iAF1260	2162	1461	461	Succinic acidAcetic acid	[9]
iEM439	767	705	684	Ethanol	[8]

Table 2
Parameter settings used throughout this study.

Model	Oxygen uptake rate (mmol gDW^{-1} hr^{-1})	Substrate uptake rate (mmol gDW^{-1} hr^{-1})	Size of artificial-organism	Number of reactions knockout	Ref.
E.coli core	10	10	40	8	[40]
iAF1260	18.5	10	40	6 5	[41]
iEM439	0	10	80	3	[8]

the proposed algorithm. Meanwhile, the modelling and analysis of GSMM is done by FBA algorithm in Constraints Based Reconstruction and Analysis (COBRA) toolbox using Gurobi solver. Throughout this work, the parameters used are shown in Table 2. The number of maximum knockouts was selected based on pre-experimental results conducted earlier, where the result that produced the highest production rate is selected. Meanwhile, for *maxIter,* the proposed method is allowed to run up to 400 iterations.

5. Results

We tested the proposed algorithm to maximise production of ethanol, succinic acid and acetic acid in three different GSMMs, as modelled by the FBA. First, we selected non-dominated solutions based on knockout reactions that gave higher production and growth rates. Subsequent to that, the list of reactions suggested by ndsDSAFBA were validated with results from other studies in order to define whether the respective reactions play a major role in maximising the growth rate and production rate. Then, we correlated and discussed the relationship between production rates and growth rates in a graph, where the non-dominated solutions are represented as a Pareto curve. Lastly, we compared the obtained results with SOO methods and MOO methods.

5.1. Succinic acid production in E.coli core model

Fig. 8 shows the results obtained on the relationship of succinic acid production and its growth rate while Table 3 shows the 15 best maximisations of succinic acid production and its growth rate in the *E.coli* core model. Here, 'best' is defined as the values closer to the maximum theoretical production of a metabolite in an organism without any knockout. The highest production rate of succinic acid found is 5.57% lower than the production of wild type, which is equal to 15.47. This result is obtained by strain S1 that reproduces at nearly half the percentage rate compared to the wild type. The set of knockout reactions in this strain is: [ACKr, G6PDH2r, GLUN, ME2, PTAr, PYK, SUCDi and SUCOAS].

Aside from that, the lowest production rate of mutants identified by the proposed algorithm is 14.8034, that is 9.65% lower than the wild type, with a growth rate at 0.8636. The suggested knockout reactions for strain S15 are ACKr, G6PDH2r, GLUN, GLUSy, GND, LDH_D, ME2 and NADTRHD. Furthermore, as shown in the table, the targeted knockout reactions that can maximise most of the production and growth rate of succinic acid are succinyl-CoA synthetase (SUCOAS) and pyruvate kinase (PYK) that target the citric acid cycle and pyruvate metabolism, respectively; with nine occurrences in the different strains. The PYK reaction encoded by the *pykA* gene is responsible for the conversion of phosphoenolpyruvate into other by-products including lactate, formate, acetate and ethanol to name a few; the knockout of this reaction will redirect the flux towards the citric acid cycle [42]. Meanwhile, the SUCOAS reaction encoded by *sucC* and *sucD* genes catalysed the conversion of succinate to succinyl-CoA. Previous research has shown that these genes are important for cell viability and the removal of these genes may affect the growth of the strain [43].

As shown in Fig. 8, there is a clear linear relationship between the

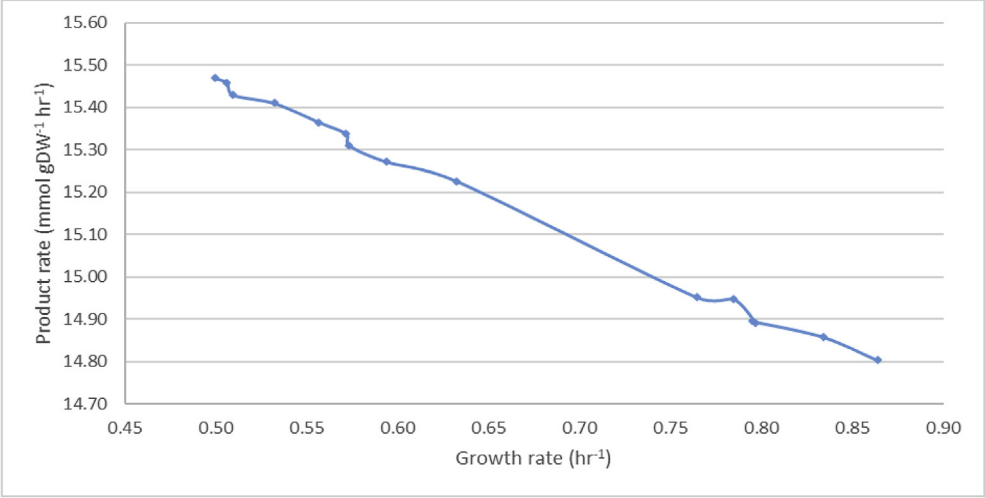

Fig. 8. Results for optimisation of succinic acid production and growth rate in E.coli core model.

Table 3
Maximisation of succinic acid production and growth rates in *E.coli* core model.

Strain	Production rate (mmol gDW^{-1} hr^{-1})	Growth rate (hr^{-1})	Knockout reactions
Wild type	16.3842	0.8739	–
S1	15.4700	0.4995	ACKr, G6PDH2r, GLUN, ME2, PTAr, PYK*, SUCDi, SUCOAS*
S2	15.4585	0.5057	ACALD, AKGDH, LDH_D, ME2, PTAr, PYK*, SUCDi, TALA
S3	15.4301	0.5092	ACKr, GLUDy, ME1, NADTRHD, PYK*, SUCDi, SUCOAS*, TKT1
S4	15.4103	0.5321	ACALD, AKGDH, GLUN, GLUSy, PYK*, SUCDi, TKT1
S5	15.3653	0.5567	ACALD, ACKr, AKGDH, ME1 ME2, PFL, SUCDi, TKT1
S6	15.3383	0.5715	FBP, GLUN, GND, LDH_D, ME1, ME2, PGL, PYK*
S7	15.3107	0.5730	ALCD2x, GLUDy, ME1, PFL, PGK, PYK*, SUCDi, SUCOAS*
S8	15.2720	0.5937	AKGDH, ALCD2x, GLUDy, GND, NADTRHD, PGL, SUCDi, SUCOAS*
S9	15.2260	0.6325	GLUN, GLUSy, LDH_D, ME1, PFL, PPCK, SUCDi, TKT2
S10	14.9514	0.7647	ACALD, ALCD2x, GLUDy, LDH_D, ME2, PYK*, SUCOAS*, TALA
S11	14.9484	0.7846	ACALD, ALCD2x, ME2, NADTRHD, PGL, PPCK, PYK*, SUCOAS*
S12	14.8915	0.7968	ACALD, ACKr, G6PDH2r, GLUDy, LDH_D, ME1, ME2, SUCOAS*
S13	14.8948	0.7951	FBP, G6PDH2r, GLUDy, GND, PGL, PTAr, PYK*, SUCOAS*
S14	14.8583	0.8338	AKGDH, G6PDH2r, GND, NADTRHD, PFL, PPCK, PTAr, SUCOAS*
S15	14.8034	0.8636	ACKr, G6PDH2r, GLUN, GLUSy, GND, LDH_D, ME2, NADTRHD

Note: * indicate the most suggested reactions found in each strain.

production of succinic acid and the growth rate of mutants, which confirms the trade-off Pareto concept whereby maximising one objective will affect the other competing objective. As shown above, maximisation of succinic acid production slightly affects the viability of the organism. Furthermore, there is a gap between strains S9 and S10. The solutions on the graph can be clustered into two groups, upper and lower groups, which might indicate low diversification of the algorithm. The production rate and growth rate difference between strains S9 and S10 are 0.2746 and 0.1322, respectively. DSA used *Brownian-random walk* for its diversification, yet it can only manage to find limited and repetitive combinations of reactions. Not only that, the intrinsic complexity of GSMM may affect the diversification process.

On the other hand, compared with the previous SOO algorithm, DSAFBA, the best succinic acid production was 15.50 with a growth rate of 0.4836. Regardless of the slight differences in rates of production, the proposed multi-objective algorithm suggested various solutions with higher growth rates, thus allowing decision makers to choose their own preferences. Furthermore, from the total of 15 strains identified by the proposed algorithm, ndsDSAFBA, the range of percentages for growth rate and production rate are 1.18%–42.82% and 5.58%–9.65%, with respect to the wild-type.

5.2. Succinic acid and acetic acid production in iAF1260 model

In Fig. 9 and Fig. 10, we analysed the overproduction of succinic acid and acetic acid in the iAF1260 model. From the Pareto curve, we deduced that both of the graphs show a linear relationship between production and growth rates. Furthermore, more solutions resided on the lower part of the graphs compared to the upper part, whereby only one solution was found. Here, we found 216 and 240 genotypically different, Pareto optimal trade-off strains for the production of succinic acid and acetic acid, respectively. Nevertheless, it appeared that the value differences between these solutions are not overly significant.

For the overproduction of succinic acid in iAF1260, the highest production rate found was 15.6504, 6.4% less than the theoretical maximum production. The specific knockouts for this strain are GLYCL, GND, GSPMDA, PPKr, PSP_L and SUCDi, resulting in −34.77% growth rate as compared to the wild-type. Considerable changes were noted in the graph (see Fig. 9) in terms of the production rate and growth rate of succinic acid. It seems that most of the knockout reactions suggested by the proposed algorithm have a lower production rate and higher growth rate. This tends to happen due to the natural characteristics of the organisms themselves, whereby regardless of mutations, genetic perturbations, or extreme environmental changes, the organisms will try their best to survive.

Additionally, the lowest production rate of succinic acid is 15.1332, with a growth rate at 0.8856. From Fig. 9 there are approximately 30 strains identified by the proposed method and the results are able to overcome the solutions suggested by DSAFBA. From these solutions found, the reaction knockout that was suggested most is succinate

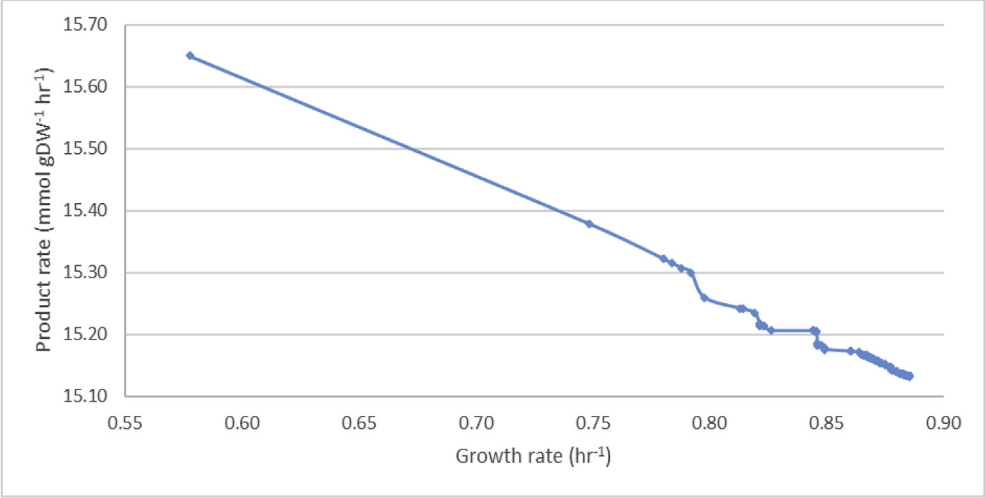

Fig. 9. Results for optimisation of succinic acid production and growth rates in iAF1260 model.

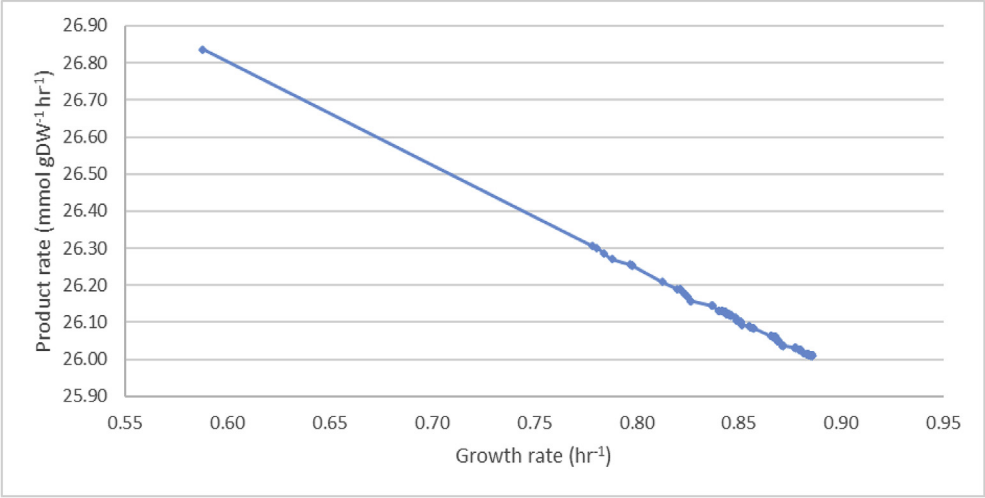

Fig. 10. Results for optimisation of acetic acid production and growth rates in the iAF1260 model.

Table 4
Maximisation of succinic acid production and growth rates in iAF1260 model.

Strain	Production rate (mmol gDW^{-1} hr^{-1})	Growth rate (hr^{-1})	Knockout reactions
Wild type	16.7221	0.8856	–
S1	15.6504	0.5777	GLYCL, GND, GSPMDA, PPKr, PSP_L, SUCDi*
S2	15.3787	0.7487	DKMPPD3, G6PDH2r, GLYCL, HCYSMT, MDH2, SUCDi*
S3	15.3218	0.7805	ACALD, G6PDH2r, ME1, SUCDi*, UACGAMPpp, XYL12
S4	15.3149	0.7841	ALLPI, DHAPT, GLYAT, SUCDi*, TALA, URDGLYCD
S5	15.3075	0.7881	AKGDH, ALCD19, G6PDA, GUI1, RPE, SUCDi*
S6	15.3000	0.7921	GLUNpp, LYSDC, MANPGH, RPE, SGDS, SUCDi*
S7	15.2599	0.7977	ALDD4, CRNCAL2, CYSSADS, PGLYCP, SUCDi*, TKT2
S8	15.2424	0.8128	ASPT, DHPPD, PSPS_L, SPMS, SUCDi*, TRE6PS
S9	15.2410	0.8142	ACCOAL, GARFT, GLTPD, PSERT, TKT2, TRE6PS
S10	15.2351	0.8193	MALDH, MCITS, PFL, PSERT, SGDS, TRPAS2

Note: * indicate the most suggested reactions found in each strain.

dehydrogenase, SUCDi. Considering that succinic acid is an intermediate product of fermentation, thus, the knockout of SUCDi may hinder the production of fumaric acid which is encoded by *sdhA, sdhB, sdhC* or *sdhD* genes [44]. The list of 10 best solutions is shown in Table 4, together with their suggested reactions for the knockout.

The Pareto front for the values of acetic acid production and growth rates are shown in Fig. 10. This figure depicts the linear relationship between production and growth rates, like the other Pareto curves for the production of different metabolites. Furthermore, there are more solutions concentrated on the lower part of the graph which correspond to a higher growth rate. This is consistent with the natural behaviour of organisms when dealing with changes, as they tend to maximise their growth in order to survive. Despite that, our proposed algorithm was able to identify the strain with the highest production rate, which is 26.8354. This improvement is 26.17% and 15.92% more than reactions suggested by DSAFBA and OptKnock, respectively. However, this

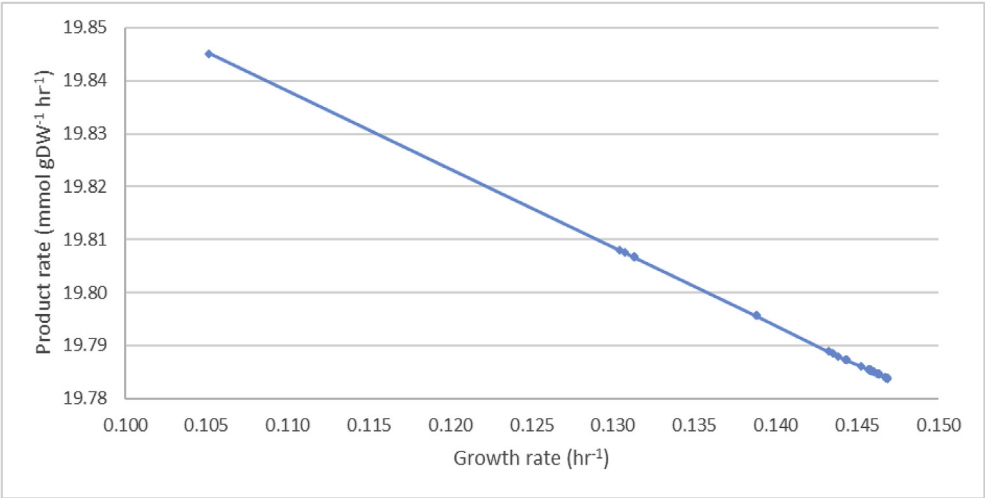

Fig. 11. Results for the optimisation of ethanol production and growth rates in iEM439 model.

Table 5
Maximisation of acetic acid production and growth rates in iAF1260 model.

Strain	Production rate (mmol gDW^{-1} hr^{-1})	Growth rate (hr^{-1})	Knockout reactions
Wild type	28.4670	0.8856	–
S1	26.8354	0.5881	ABTA, G6PDH2r, GLYOX, THRA2i, TPI
S2	26.3055	0.7785	FCLK, G5SD, GND, SUCDi*, UNK3
S3	26.3016	0.7805	ALDD4, ASCBPL, PGL, SUCDi*, TREHpp
S4	26.2862	0.7841	ABTA, CTBTCAL2, GLCATr, SUCDi*, TKT1
S5	26.2702	0.7881	DKGLCNR1, GAL1PPpp, MANAO, RPE, SUCDi*
S6	26.2560	0.7969	AKGDH, G6PDH2r, MTRI, PSERT, UACGALPpp
S7	26.2537	0.7977	BUTCT, DMSOR2, GLYCTO2, PSP_L, SUCDi*
S8	26.2537	0.7977	ACGAL1PPpp, DPRA, GSPMDS, PGCD, SUCDi*
S9	26.2537	0.7977	FTHFD, HEX7, PSP_L, SUCDi*, UNK3
S10	26.2537	0.7977	ADNUC, AMANAPEr, ARAI, PSERT, SUCDi*

Note: * indicate the most suggested reactions found in each strain.

growth rate is lower compared to the growth rates recorded by the SOO methods. Furthermore, the proposed algorithm is only able to find one solution with a higher production rate but a lower growth rate. This is believed to be the first non-dominated solution found by the algorithm, hence, the organism overcame this by maximising the growth rate instead.

Table 5 shows the 10 best non-dominated solutions obtained by our proposed algorithm, together with their respective knockout reactions. Although the algorithm was able to identify 240 non-dominated solutions, the differences in production rate are not too significant, with a difference of only 0.8255 between the highest and lowest production rates, whereas for growth rate, there is a significant difference as shown in the graph. From the table below, strains S7, S8, S9 and S10 produce the same production rate and growth rate, which is 26.2537 and 0.7977, respectively. In regard to that, each strain will have different specific knockout reactions except for succinate dehydrogenase, SUCDi. Apart from that, from the 10 best strains obtained, SUCDi appeared to occur frequently than other reactions. This is probably due to the involvement of competing reactions that hinder the production of acetic acid.

5.3. Ethanol production in iEM439 model

Fig. 11 represents the result of maximising ethanol production and growth rates in iEM439. The line of this curve is similar to the previous graphs, as it indicates a linear relationship between the production rate of ethanol and the growth rate of the non-dominated solutions. The algorithm was able to obtain 80 non-dominated solutions that are represented in the graph below. As shown, the data distribution is not widespread along the curve. If we manually cluster the solutions by the naked eye, there are 4 separate clusters of solutions. The middle cluster that consists of 14 solutions provides good ethanol production without affecting the growth rate, while the first cluster only consists of one solution with the highest production of ethanol obtainable.

Table 6 represents the 10 best non-dominated solutions for maximising ethanol production and growth rates in iEM439. Similar to the production of acetic acid in iAF1260 strains, S4 to S8 produced the same amount of ethanol and have the same growth rate despite the difference in genotypic reactions. The highest ethanol production found is S1, which is 0.77% lower than wild-type production. As shown in Table 6 and Fig. 11, although strain S1 produced modest quantities of ethanol, however, the growth rate for this mutant is 28.44% lower than the wild-type. On the contrary, solutions at the end of the graph produced a smaller amount of ethanol although the growth rate is almost similar to the growth rate of wild-type.

The proposed algorithm is able to overcome the solutions obtained by SOO methods, DSAFBA and OptKnock, with a difference of 0.0451 and 1.9451 ethanol production, respectively. Yet, the growth rate obtained by SOO methods was higher than the S1. Regardless of these differences, multi-objective optimisations allow multiple solutions to be generated in one run, while single objective optimisations only allow one solution generated at a time. Therefore, decision-makers have a variety of solutions to choose from based on their own preferences. From the results, the glutamate dehydrogenase, GLUDy, reaction is the most selected for in the knockout with 6 occurrences.

Table 6
Maximisation of ethanol production and growth rates in iEM439 model.

Strain	Production rate (mmol gDW^{-1} hr^{-1})	Growth rate (hr^{-1})	Knockout reactions
Wild type	20	0.1470	–
S1	19.8451	0.1052	G6PDH2r, HEX1, SHSL2
S2	19.8080	0.1304	GLUDy*, HEX1, ILETRS
S3	19.8076	0.1307	GLUDy*, HCYSS, PLIPA2E141pp
S4	19.8067	0.1313	GLUDy*, DUTPDP, KDOCT2
S5	19.8067	0.1313	ADK3, GLUDy*, PSSA120
S6	19.8067	0.1313	E4PD, EAR40x, GLUDy*
S7	19.8067	0.1313	GLUDy*, OMMBLHX, PYK1
S8	19.8067	0.1313	GLUDy*, GLUTRS, SOTA
S9	19.7956	0.1388	GLYTRS, PPKr, UDPGL4E
S10	19.7956	0.1388	NTD10, PPKr, RNDR2

Note: * indicate the most suggested reactions found in each strain.

Ethanol is produced with several other by-products during the fermentation process, including glycerol. Under anaerobic conditions, NADH is reoxidised to NAD^+ and forms glycerol which consumes glucose. One strategy to improve the formation of ethanol is to eliminate glycerol synthesis by deleting the GLUDy reaction as it will drain off the surplus formation of NADH, thus allowing the flux towards ethanol production [45].

6. Discussion

In this paper, we have identified a problem related to the previous developed algorithm, DSAFBA, and other developed algorithms such as OptGene, which is the inability to identify non-dominated solutions of competing objectives. The previous algorithms are able to identify near-optimal knockout reactions for growth-coupled design strains, yet these algorithms tend to produce mutant strains that merely maximise the production rate. Furthermore, they can only produce one single solution at a time. Therefore, to solve the aforementioned problem, we have developed non-dominated sorting DSAFBA (ndsDSAFBA) to optimise the production rate of the desired metabolites and the growth rate by identifying a set of non-dominated solutions that represent knockout reactions. ndsDSAFBA is capable of producing multiple solutions that can deduce the relationship between the production rate and the growth rate of mutants.

In ndsDSAFBA, non-dominated sorting is used to identify non-dominated solutions of conflicting objectives generated by DSA,

Table 7
Comparison of results from SOO methods and MOO methods.

Organism	Method	Production rate (mmol gDW^{-1} hr^{-1})	Growth rate (hr^{-1})	Ref.
Escherichia coli	ReacKnock	9.13	0.129	[49]
	IdealKnock	9.25	0.0973	[50]
	DSAFBA	12.45	0.575	[29]
	NSGA-II	14.70	0.005	[51]
	SPEA2	14.10	0.010	[51]
	ndsDSAFBA	15.65	0.577	This work
Zymomonas mobilis	OptKnock	17.90	0.134	[7]
	DSAFBA	19.80	0.130	[29]
	ndsDSAFBA	19.84	0.105	This work

whereby FBA is used to calculate the production rate and growth rate of mutant strains. Despite that optimisation algorithms have their own exploration and exploitation mechanisms in finding near-optimal solutions, however, DSA could not identify the trade-offs between the two objectives [46]. Thus, the concept of Pareto dominance using a non-dominated sorting strategy is incorporated in order to determine non-dominated solutions and generate Pareto curves. A non-dominated sorting strategy that uses the concept of a Pareto dominance ranking approach assigns a strength probability depending on its dominance occurrences [47].

Herein, two organisms, *Escherichia coli* and *Zymomonas mobilis* were used to optimise the production of ethanol, succinic acid and acetic acid, as these organisms are natural producers for the aforementioned metabolites. *E.coli* has been exhaustively studied and widely used by biologists, industrialists and computer scientists in exploiting and increasing the capabilities of cells. Meanwhile, Z.*mobilis* is an ethanologenic bacterium that is similar to *S.cerevisiae*, as both organisms can produce ethanol. The advantages of Z.*mobilis* are that it is able to produce high production of ethanol despite its low growth rate and is not sensitive to a higher concentration of alcohol. We applied a knockout reactions strategy to identify mutant strains that produce a high production rate of the desired metabolites and growth rate of mutants. The knockout reactions strategy is chosen due to the practicality and stability of these simulated knockout reactions that can be easily implemented in *in vivo* analysis that usually requires a long time and requires plasmid reconstruction [48].

The performance of ndsDSAFBA is compared with other single objective and multi-objective methods including ReacKnock, DSAFBA, IdealKnock, NSGA-II, SPEA2 and OptKnock. The comparison of production rate and growth rate obtained from the different methods is given in Table 7. For ethanol production in *Z.mobilis*, our proposed method was able to outperform other methods with a total production of 19.84, despite a 26% loss in growth rate. Furthermore, it can be observed from Table 7 that ndsDSAFBA was able to improve the production rate and growth rate of succinic acid in *E.coli* compared to ReacKnock, IdealKnock, DSAFBA, NSGA-II and SPEA2, with a maximum production of 15.65 achieved by ndsDSAFBA, and a growth rate that is notably higher than the other methods.

From the results represented in Table 3 - Table 6, a total of 15, 216, 241 and 80 non-dominated solutions are found for differently desired metabolites (as plotted in the graph in Figs. 8–11). Though it successfully identifies the non-dominated solutions, the distribution of non-dominated solutions along the graph is relatively weak as we can observe that the solutions are not uniformly distributed and diversely placed on the graph. This is probably because the framework of ndsDSAFBA greedily takes the best solution without considering the effect towards a non-dominated solution on the Pareto front. Interestingly, some of the non-dominated solutions produced the same phenotypic results, despite being genotypically different (for example in Table 6, where strain S3 to strain S8 produced the same production and growth rates despite different knockout reactions). This is due to the complexity of the GSMM that consists of hundreds and thousands of reactions. The nearly best trade-off solutions for each case study produced the highest production rate but a slightly lower growth rate. Nonetheless, there are other solutions available, which decision-makers can use and select. Additionally, most of the solutions in multi-objective optimisation are able to overcome the solutions found by SOO methods, including DSAFBA and OptKnock.

Overall, the paper and the proposed algorithm considered: (1) optimisation of two objective functions which are growth rate and production rate, (2) non-dominated sorting strategy to identify non-dominated solutions, producing multiple solutions that conform to the concept of multi-objective optimisation, and (3) deducing the relationship between production rate of desired metabolites and growth rate of mutants.

7. Conclusion

This study proposed a multi-objective evolutionary algorithm termed ndsDSAFBA with the aim of identifying knockout reactions for maximising the production of desired metabolites and the growth rate in *in silico* simulation. The proposed algorithm was tested with three GSMMs to maximise the production of succinic acid, acetic acid and ethanol, while at the same time maximising the growth of the cells. We have adapted a non-dominated sorting strategy in order to determine non-dominated solutions and generate Pareto curves. The non-dominated sorting strategy that uses the concept of a Pareto dominance ranking approach assigns a strength probability depending on its dominance occurrences. The nearly-best trade-off solutions for each case study produced the highest production rate but a slightly lower growth rate. Nonetheless, there are other solutions available, which decision-makers can use and select. Additionally, most of the solutions in multi-objective optimisation are able to overcome the solutions found by SOO methods, including DSAFBA and OptKnock. As a conclusion, multi-objective optimisation in *in silico* metabolic engineering is still work in progress, due to the lack of information pertaining to models, methods and frameworks. Despite that, the results that we obtained herein can be used as prior knowledge in aiding biologists in solving problems and analysing the effects of knockout reactions towards the phenotypic characteristics of the cells, specifically focusing on optimising the production of industrially desired metabolites. Relevant improvements that can be made to the proposed algorithm include inserting regulatory and kinetic information for more accurate analysis and results, considering a more robust diversification process for generating distributed non-dominated solutions along the Pareto curve, and including automated decision-making processes in selecting the preferred solutions from non-dominated solutions.

Conflicts of interest

None Declared.

Acknowledgement

This research was funded by Fundamental Research Grant Scheme - Malaysia's Research Star Award (FRGS-MRSA) from Ministry of Education Malaysia. We also would like to thank the Ministry of Education Malaysia for supporting this research by the Fundamental Research Grant Schemes (grant number: RDU190113 and R.J130000.7828.4F720).

References

[1] P. Maia, M. Rocha, I. Rocha, In Silico constraint-based strain optimization methods: the quest for optimal cell factories, Microbiol. Mol. Biol. Rev. 80 (2016) 45–67, https://doi.org/10.1128/MMBR.00014-15 (Address).

[2] Z. Rejc, L. Magdevska, T. Trselic, T. Osolin, R. Vodopivec, J. Mraz, E. Pavliha, N. Zimic, T. Cvitanovi, D. Rozman, M. Moskon, M. Mraz, Computational modelling of genome-scale metabolic networks and its application to CHO cell cultures Ziva, Comput. Biol. Med. 88 (2017) 150–160, https://doi.org/10.1016/j.compbiomed.2017.07.005.

[3] M. Budinich, J. Bourdon, A. Larhlimi, D. Eveillard, A multi-objective constraint-based approach for modeling genome-scale microbial ecosystems, PLoS One, 2017, pp. 1–22, , https://doi.org/10.1371/journal.pone.0171744.

[4] K.D. Rawls, B.V. Dougherty, E.M. Blais, E. Stancliffe, G.L. Kolling, K. Vinnakota, V.R. Pannala, A. Wallqvist, J.A. Papin, K. Vinnokata, V.R. Pannala, A. Wallqvist, J.A. Papin, A simplified metabolic network reconstruction to promote understanding and development of flux balance analysis tools, Comput. Biol. Med. 105 (2019) 64–71 S0010482518304086.

[5] G. Nair, C. Jungreuthmayer, J. Zanghellini, Optimal knockout strategies in genome-scale metabolic networks using particle swarm optimization, BMC Bioinf. 18 (2017) 1–9, https://doi.org/10.1186/s12859-017-1483-5.

[6] S. Mutturi, FOCuS: a metaheuristic algorithm for computing knockouts from genome-scale models for strain optimization, Mol. Biosyst. 13 (2017) 1355–1363, https://doi.org/10.1039/C7MB00204A.

[7] A.P. Burgard, P. Pharkya, C.D. Maranas, OptKnock: a bilevel programming framework for identifying gene knockout strategies for microbial strain optimization, Biotechnol. Bioeng. 84 (2003) 647–657, https://doi.org/10.1002/bit.10803.

[8] E. Motamedian, M. Saeidi, S.A. Shojaosadati, Reconstruction of a charge balanced genome-scale metabolic model to study energy-uncoupled growth of Zymomonas mobilis ZM1, Mol. Biosyst. 12 (2016) 1241–1249, https://doi.org/10.1039/C5MB00588D.

[9] A.M. Feist, C.S. Henry, J.L. Reed, M. Krummenacker, A.R. Joyce, P.D. Karp, L.J. Broadbelt, V. Hatzimanikatis, B.Ø. Palsson, A genome-scale metabolic reconstruction for Escherichia coli K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information, Mol. Syst. Biol. 3 (2007) 121, https://doi.org/10.1038/msb4100155.

[10] M.A. Arif, M.S. Mohamad, M.S. Abd Latif, S. Deris, M.A. Remli, K. Mohd Daud, Z. Ibrahim, S. Omatu, J.M. Corchado, A hybrid of Cuckoo Search and Minimization of Metabolic Adjustment to optimize metabolites production in genome-scale models, Comput. Biol. Med. 102 (2018) 112–119, https://doi.org/10.1016/j.compbiomed.2018.09.015.

[11] A. Von Kamp, S. Klamt, Growth-coupled overproduction is feasible for almost all metabolites in five major production organisms, Nat. Commun. 8 (2017) 1–10, https://doi.org/10.1038/ncomms15956.

[12] S. Ohno, H. Shimizu, C. Furusawa, FastPros: screening of reaction knockout strategies for metabolic engineering, Bioinformatics 30 (2014) 981–987, https://doi.org/10.1093/bioinformatics/btt672.

[13] T.B. Alter, L.M. Blank, B.E. Ebert, Determination of growth-coupling strategies and their underlying principles, BioRxiv (2018) 258996, , https://doi.org/10.1101/258996.

[14] T. Tamura, Grid-based computational methods for the design of constraint-based parsimonious chemical reaction networks to simulate metabolite production: GridProd, BMC Bioinf. 19 (2018) 1–9, https://doi.org/10.1186/s12859-018-2352-6.

[15] K. Jensen, V. Broeken, A.S.L. Hansen, N. Sonnenschein, M.J. Herrgård, OptCouple: joint simulation of gene knockouts, insertions and medium modifications for prediction of growth-coupled strain designs, Metab. Eng. Commun. 8 (2019), https://doi.org/10.1016/j.mec.2019.e00087.

[16] P. Pharkya, A.P. Burgard, C.D. Maranas, OptStrain: a computational framework for redesign of microbial production systems, Genome Res. 14 (2004) 2367–2376, https://doi.org/10.1101/gr.2872004.

[17] P. Pharkya, C.D. Maranas, An optimization framework for identifying reaction activation/inhibition or elimination candidates for overproduction in microbial systems, Metab. Eng. 8 (2006) 1–13, https://doi.org/10.1016/j.ymben.2005.08.003.

[18] F.-S. Wang, W.-H. Wu, Optimal design of growth-coupled production strains using nested hybrid differential evolution, J. Taiwan Inst. Chem. Eng. 54 (2015) 57–63, https://doi.org/10.1016/j.jtice.2015.03.015.

[19] S.S. Fong, B. Palsson, Metabolic gene-deletion strains of Escherichia coli evolve to computationally predicted growth phenotypes, Nat. Genet. 36 (2004) 1056–1058, https://doi.org/10.1038/ng1432.

[20] K. Shabestary, E.P. Hudson, Computational metabolic engineering strategies for growth-coupled biofuel production by synechocystis, Metab. Eng. Commun. 3 (2016) 216–226, https://doi.org/10.1016/j.meteno.2016.07.003.

[21] D. Nagrath, M.M. Avila-Elchiver, F.F.F. Berthiaume, A.W. Tilles, A. Messac, M.L. Yarmush Martin, M.L. Yarmush, Soft constraints-based multiobjective framework for flux balance analysis, Metab. Eng. 12 (2010) 429–445, https://doi.org/10.1016/j.ymben.2010.05.003.

[22] J. Costanza, G. Carapezza, C. Angione, P. Liò, G. Nicosia, Multi-objective optimisation, sensitivity and robustness analysis in FBA modelling, Comput. Methods Syst. Biol., Springer Berlin Heidelberg, 2012, pp. 127–147, , https://doi.org/10.1007/978-3-642-33636-2_9.

[23] Y.-G. Oh, D.-Y. Lee, S.Y. Lee, S. Park, Multiobjective flux balancing using the NISE method for metabolic network analysis, Biotechnol. Prog. 25 (2009) 999–1008, https://doi.org/10.1002/btpr.193.

[24] A. Patané, A. Santoro, J. Costanza, G. Carapezza, G. Nicosia, I. Member, Pareto optimal design for synthetic biology, IEEE Trans. Biomed. Circuits Syst. 9 (2015) 555–571.

[25] A. Patané, G. Jansen, P. Conca, G. Carapezza, J. Costanza, G. Nicosia, Multi-objective optimization of genome-scale metabolic models: the case of ethanol production, Ann. Oper. Res. 276 (2018) 1–17, https://doi.org/10.1007/s10479-018-2865-4.

[26] T.Y. Kim, J.M. Park, H.U. Kim, K.M. Cho, S.Y. Lee, Design of homo-organic acid producing strains using multi-objective optimization, Metab. Eng. 28 (2015) 63–73, https://doi.org/10.1016/j.ymben.2014.11.012.

[27] Q. Bai, S. Labi, K.C. Sinha, Trade-off analysis for multiobjective optimization in transportation asset management by generating Pareto frontiers using extreme points nondominated sorting genetic algorithm II, J. Transport. Eng. 138 (2012) 798–808, https://doi.org/10.1061/(ASCE)TE.1943-5436.0000369.

[28] A. Mukhopadhyay, U. Maulik, S. Bandyopadhyay, C.A.C. Coello, Survey of multiobjective evolutionary algorithms for data mining: part II, IEEE Trans. Evol. Comput. 18 (2014) 25–35, https://doi.org/10.1109/TEVC.2013.2290082.

[29] K.M. Daud, Z. Zakaria, Z.A. Shah, M.S. Mohamad, S. Deris, S. Omatu, J.M. Corchado, A hybrid of Differential Search Algorithm and Flux Balance Analysis to identify knockout strategies for in silico optimization of metabolites production, Int. J. Adv. Soft Comput. Its Appl. 2018, pp. 84–107.

[30] D. Lee, Y.-G. Oh, H. Yoon, S.Y. Lee, S. Park, Exploring flux distribution profiles for switching pathways using multiobjective flux balance analysis, Genome Inf. 13 (2002) 363–364.

[31] D. Nagrath, M. Avila-Elchiver, F. Berthiaume, A.W. Tilles, A. Messac, M.L. Yarmush, Integrated energy and flux balance based multiobjective framework for large-scale metabolic networks, Ann. Biomed. Eng. 35 (2007) 863–885, https://doi.org/10.1007/s10439-007-9283-0.

[32] A. Konak, D.W. Coit, A.E. Smith, Multi-objective optimization using genetic algorithms: a tutorial, Reliab. Eng. Syst. Saf. 91 (2006) 992–1007, https://doi.org/10.1016/j.ress.2005.11.018.

[33] Y. Cui, Z. Geng, Q. Zhu, Y. Han, Review: multi-objective optimization methods and application in energy saving, Energy 125 (2017) 681–704, https://doi.org/10.1016/j.energy.2017.02.174.

[34] H.R. Cheshmehgaz, H. Haron, A. Sharifi, The review of multiple evolutionary searches and multi-objective evolutionary algorithms, Artif. Intell. Rev. 43 (2013) 311–343, https://doi.org/10.1007/s10462-012-9378-3.

[35] K. Raman, N. Chandra, Flux balance analysis of biological systems: applications and challenges, Briefings Bioinf. 10 (2009) 435–449, https://doi.org/10.1093/bib/bbp011.

[36] A. Chowdhury, A.R. Zomorrodi, C.D. Maranas, Bilevel optimization techniques in computational strain design, Comput. Chem. Eng. 72 (2015) 363–372, https://doi.org/10.1016/j.compchemeng.2014.06.007.

[37] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2002) 182–197, https://doi.org/10.1109/4235.996017.

[38] K.R. Patil, I. Rocha, J. Forster, J. Nielsen, Evolutionary programming as a platform for in silico metabolic engineering, BMC Bioinf. 6 (2005) 308, https://doi.org/10.1186/1471-2105-6-308.

[39] C.E.G. Sánchez, C.A.V. García, R.G.T. Sáez, Predictive potential of flux balance analysis of Saccharomyces cerevisiae using as optimization function combinations of cell compartmental objectives, PLoS One 7 (2012), https://doi.org/10.1371/journal.pone.0043006.

[40] J.S. Edwards, M. Covert, B. Palsson, Metabolic modelling of microbes: the flux-balance approach, Environ. Microbiol. 4 (2002) 133–140, https://doi.org/10.1046/j.1462-2920.2002.00282.x.

[41] M. Rocha, P. Maia, R. Mendes, J.P. Pinto, E.C. Ferreira, J. Nielsen, K.R. Patil, I. Rocha, Natural computation meta-heuristics for the in silico optimization of microbial strains, BMC Bioinf. 9 (2008) 499, https://doi.org/10.1186/1471-2105-9-499.

[42] S. Noda, T. Shirai, S. Oyama, A. Kondo, Metabolic design of a platform Escherichia coli strain producing various chorismate derivatives, Metab. Eng. 33 (2016) 119–129, https://doi.org/10.1016/j.ymben.2015.11.007.

[43] B.J. Yu, B.H. Sung, J.Y. Lee, S.H. Son, M.S. Kim, S.C. Kim, sucAB and sucCD are mutually essential genes in Escherichia coli, FEMS Microbiol. Lett. 254 (2006) 245–250, https://doi.org/10.1111/j.1574-6968.2005.00026.x.

[44] S. Ren, B. Zeng, X. Qian, Adaptive bi-level programming for optimal gene knockouts for targeted overproduction under phenotypic constraints, BMC Bioinf. 14 (Suppl 2) (2013) S17, , https://doi.org/10.1186/1471-2105-14-S2-S17.

[45] T.L. Nissen, M.C. Kielland-Brandt, J. Nielsen, J. Villadsen, Optimization of ethanol production in Saccharomyces cerevisiae by metabolic engineering of the ammonium assimilation, Metab. Eng. 2 (2000) 69–77, https://doi.org/10.1006/mben.1999.0140.

[46] J. Liu, C. Wu, J. Cao, X. Wang, K.T. Lay, A Binary differential search algorithm for the 0 – 1 multidimensional knapsack problem, Appl. Math. Model. 40 (2016) 9788–9805, https://doi.org/10.1016/j.apm.2016.06.002.

[47] P. Gustavsson, A. Syberfeldt, A new algorithm using the non-dominated tree to improve non-dominated sorting, Evol. Comput. (2017) 1–28.

[48] A.M. Feist, D.C. Zielinski, J.D. Orth, J. Schellenberger, M.J. Herrgard, B.O. Palsson, Model-driven evaluation of the production potential for growth-coupled products of Escherichia coli, Metab. Eng. 12 (2010) 173–186, https://doi.org/10.1016/j.ymben.2009.10.003.

[49] Z. Xu, P. Zheng, J. Sun, Y. Ma, ReacKnock: identifying reaction deletion strategies for microbial strain optimization based on genome-scale metabolic network, PLoS One 8 (2013), https://doi.org/10.1371/journal.pone.0072150.

[50] D. Gu, C. Zhang, S. Zhou, L. Wei, Q. Hua, IdealKnock: A framework for efficiently identifying knockout strategies leading to targeted overproduction, Comput. Biol. Chem. 61 (2016) 229–237, https://doi.org/10.1016/j.compbiolchem.2016.02.014.

[51] P. Maia, I. Rocha, E.C. Ferreira, M. Rocha, Evaluating evolutionary multiobjective algorithms for the in silico optimization of mutant strains, 8th IEEE Int. Conf. Bioinforma. Bioeng. BIBE, vol. 2008, 2008, https://doi.org/10.1109/BIBE.2008.4696733.