PLOS ONE

RESEARCH ARTICLE

Optimal experimental conditions for Welan gum production by support vector regression and adaptive genetic algorithm

Zhongwei Li[1], Xiang Yuan[1], Xuerong Cui[1], Xin Liu[1], Leiquan Wang[1], Weishan Zhang[1], Qinghua Lu[1], Hu Zhu[2]*

1 College of Computer and Communication Engineering, China University of Petroleum, Qingdao 266580, Shandong, China, **2** College of Chemistry and Materials, Fujian Normal University, Fuzhou 350007, China

* zhuhu@fjnu.edu.cn

Citation: Li Z, Yuan X, Cui X, Liu X, Wang L, Zhang W, et al. (2017) Optimal experimental conditions for Welan gum production by support vector regression and adaptive genetic algorithm. PLoS ONE 12(10): e0185942. https://doi.org/10.1371/journal.pone.0185942

Editor: Xiangxiang Zeng, Xiamen University, CHINA

Received: June 30, 2017

Accepted: September 21, 2017

Published: October 9, 2017

Data Availability Statement: All relevant data are within the paper.

Funding: This work was supported by 863 program (2015AA020925), National Natural Science Foundation of China (61402187, 61502535, 61572522, 61572523, 61672033 and 61672248), Key Research and Development Program of Shandong Province (No. 2017GGX10147), China Postdoctoral Science Foundation funded project (2016M592267), PetroChina Innovation Foundation (2016D-5007-0305), and Fundamental Research Funds for the

Abstract

Welan gum is a kind of novel microbial polysaccharide, which is widely produced during the process of microbial growth and metabolism in different external conditions. Welan gum can be used as the thickener, suspending agent, emulsifier, stabilizer, lubricant, film-forming agent and adhesive usage in agriculture. In recent years, finding optimal experimental conditions to maximize the production is paid growing attentions. In this work, a hybrid computational method is proposed to optimize experimental conditions for producing Welan gum with data collected from experiments records. Support Vector Regression (SVR) is used to model the relationship between Welan gum production and experimental conditions, and then adaptive Genetic Algorithm (AGA, for short) is applied to search optimized experimental conditions. As results, a mathematic model of predicting production of Welan gum from experimental conditions is obtained, which achieves accuracy rate 88.36%. As well, a class of optimized experimental conditions is predicted for producing Welan gum 31.65g/L. Comparing the best result in chemical experiment 30.63g/L, the predicted production improves it by 3.3%. The results provide potential optimal experimental conditions to improve the production of Welan gum.

Introduction

Welan gum is a kind of polysaccharide, which is one of the secretions of Alcaligenes sp.NX-3 strain. It has good stability, ideal thickening property, unique shear thinning property, good suspension and emulsification, and assured safety, and can be used in oil drilling with its unique shear-thinning properties. Finding optimal experimental conditions to maximize the production of Welan gum is paid growing attentions. This can process the production of Welan gum industrially. In 2014, producing Welan gum fermentation in laboratory is achieved in [1], where cyperus beans are used as raw materials, protein and hydrolysis as substrate. After that, Bacillus foecalis alkaligenes are designed as starting bacterial strain, to optimize the yield process of Welan gum by response surface method [2].

Central Universities (R1607005A). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

It is found that many factors affecting the production of Welan gum, such as glucose, yeast, liquid volume, PH vale, temperature, which contribute the experimental conditions of producing Welan gum. To find the optimal experimental conditions, we need to consider the following aspects:

1. function of each factor;
2. interaction between each pair of factors;
3. relationship among all the factors.

In 2010, Li et al used the batch fermentation experiment data of Welan gum's starting bacterial strain Alcaligenessp.CGMCC2428 to carry out the dynamic model research, implemented fermentation process of Welan gum optimization control [3]. In 2016, JMP statistical analysis software was used to optimize the fermentation medium of Welan gum by Alcaligenes sp.Y5. With the optimized experimental conditions, the production of Welan gum was increased from 15.72 g/L to 26.58 g/L, with an increment of 69.08% [4].

Recently, many significant artificial intelligent algorithms and data processing strategies has been applied on data mining, such as a self-adaptive artificial bee colony algorithm based on global best for global optimization [5], the public auditing protocol with novel dynamic structure for cloud data [6], privacy-preserving smart semantic search method for conceptual graphs over encrypted outsourced data [7], a privacy-preserving and copy-deterrence content for image data processing with retrieval scheme in cloud computing [8], strategy solving NP problems such as subset sum problem based on SN P systems [9], Apriori algorithm based on tissue-like P systems [10], split clustering algorithm based on P systems on simplices [11], spatial clustering algorithm based on DNA model [12], PSO algorithm based on dynamic niche technology [13] and machine learning method have been applied for experimental condition design, see. e.g. a secure and dynamic multi-keyword ranked search scheme over encrypted cloud data [14]. In this work, we presents a hybrid computational method to optimize experimental conditions for producing Welan gum with data collected from experiments records. Specifically, Support Vector Regression (SVR) is used to model relationship between Welan gum production and experimental conditions, and then adaptive Genetic Algorithm (AGA) is used to search optimized experimental conditions. As results, a mathematic model of predicting production of Welan gum from experimental conditions with accuracy rate 88.36% is obtained, a class of optimized experimental conditions is designed to produce Welan gum 31.65g/L. Comparing the best results in chemical lab 30.63g/L, the predicted production can be improved by 3.3%. The result provides a potential experimental conditions by data mining to improve the production of Welan gum in the lab.

Related technologies

In this section, the two main methods used, Support Vector Regression (SVR) and adaptive Genetic Algorithm (AGA), are briefly recalled.

Here, we choose the SVR method mainly because of our limited samples. First of all, as for the regression of a small amount of samples, SVR has many advantages, such as a few adjusted parameters and fast arithmetic speed, etc. Secondly, the final decision function of SVR is determined by only a small number of support vectors. Finally, the computational complexity depends on the number of support vectors, not the dimension of the sample space, which also reflects that the robustness of the SVR method is better.

Genetic algorithm is a global search algorithm, which have a good reference for our problems. However, the traditional genetic algorithm still needs to be improved in terms of global

search ability and convergence speed. The adaptive Genetic Algorithm we adopt can improve these two aspects to a certain extent. In the case of crossover probability, the AGA method can enable the crossover probability to vary with the evolution process and give the same crossover ability to the individuals of the same generation population, so as to realize the global search ability better. In the case of mutation probability, according to the fitness value of each individual to be mutated, the AGA method can make the mutation probability adaptively change with the evolutionary process.

Support vector regression

Support Vector Machine (SVM) is known as a kind of machine learning method for classification proposed in 1995 [15], has been widely used in biological data processing [16–18] and bioinformatics [19–23]. It focuses on doing classification with seeking structured minimum risk to improve the generalization ability of learning machine and minimizing empirical risk and confidence limit [24, 25], thus achieving good statistical law under the condition of the less statistical sample size. In general, it is a kind of two-category model, the basic model is defined as the feature space interval on the maximum linear classifier. The learning strategy of SVM is to maximize the interval, which finally can be converted into a convex quadratic programming problem.

Support Vector Regression (SVR) is developed based on SVM for dealing with regression forecasting problems [26, 27]. Some basic concepts of SVR are briefly recalled.

Given a set of training data $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_l, y_l)\}$, $R^n \times R$, where $\mathbf{x}_i$ denotes the input samples, y_i is the target value and l is the total number of input samples. In SVR, the goal is to find a function $f(\mathbf{x})$, i.e., an optimal hyperplane, which has at most ε deviation from the actually obtained target y_i for all the training data as flat as possible. The form of functions is denoted as

$$f(\boldsymbol{x}) = (\boldsymbol{\omega}, \Phi(\boldsymbol{x})) + b \qquad with\ \Phi : R^n \to F, \boldsymbol{\omega} \in F \tag{1}$$

where $\Phi(\cdot)$ is a nonlinear mapping by which the input data $\mathbf{x}$ is mapped into a high dimensional space F, $(\cdot, \cdot)$ denotes the dot product in space F. Eq (1) can be transformed into the following convex constrained optimization problem by introducing the non-negative slack variables ξ_i and ξ_i^* to cope with the otherwise infeasible constraints

$$\begin{aligned} min\Gamma(\boldsymbol{\omega}, \xi, \xi^*) &= \frac{1}{2} \| \boldsymbol{\omega}\|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) \\ s.t.(\boldsymbol{\omega}, \Phi(\boldsymbol{x}_i)) + b - y_i &\leqslant \varepsilon + \xi_i \\ y_i - (\boldsymbol{\omega}, \Phi(\boldsymbol{x}_i)) - b &\leqslant \varepsilon + \xi_i \\ \xi_i, \xi_i^* &\geqslant 0, \qquad i = 1, 2, \ldots, l \end{aligned} \tag{2}$$

thereinto, $C > 0$, with C being the penalty parameter. ξ_i, ξ_i^* are slack variables introduced in order to allow a certain error [28–32]. ξ is also a parameter of the ε-insensitive loss function, where ε is called the tube size [33]. The greater the value of C is, the greater the penalty for data points beyond the ε deviation, which determines the balance between the degree of smoothness of the function and the number of sample points beyond ε deviation. To find the

upper bound of a convex quadratic programming problem, Lagrangian function is applied:

$$\begin{aligned} l(\boldsymbol{\omega}, \xi_i, \xi_i^*) = \frac{1}{2} \| \boldsymbol{\omega} \|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) \\ -\sum_{i=1}^{l}\alpha_i(\boldsymbol{\varepsilon} + \xi_i - y_i + (\boldsymbol{\omega}\cdot\boldsymbol{x}_i) + b) \\ -\sum_{i=1}^{l}\alpha_i^*(\boldsymbol{\varepsilon} + \xi_i + y_i - (\boldsymbol{\omega}\cdot\boldsymbol{x}_i) - b) - \sum_{i=1}^{l}(\eta_i\xi_i + \eta_i^*\xi_i^*) \end{aligned} \tag{3}$$

thereinto, α_i, α_i^*, η_i, η_i^* are the Lagrange multiplier. The optimization problem can be obtained as follows:

$$\begin{aligned} min_{\alpha,\alpha^*} \ \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\langle\Phi(\boldsymbol{x}_i), \Phi(\boldsymbol{x}_j)\rangle + \boldsymbol{\varepsilon}\sum_{i=1}^{l}(\alpha_i + \alpha_i^*) \\ -\sum_{i=1}^{l}y_i(\alpha_i - \alpha_i^*) \\ s.t.\sum_{i=1}^{l}(\alpha_i - \alpha_i^*) = 0 \\ 0 \leqslant \alpha_i \leqslant C \\ 0 \leqslant \alpha_i^* \leqslant C \end{aligned} \tag{4}$$

where α_i^* is the nonnegative Lagrange multiplier that can be obtained by solving the convex quadratic programming problem. By exploiting the Karush-Kuhn-Tucker (KKT) conditions of the primal optimization problem [34–36], we can get the equation $\alpha_i^*\alpha_j^* = 0$, which means that both of the multipliers α_i^* and α_j^* equal to zero, or one of multipliers is zero and $(\alpha_i^* - \alpha_i^*)$ is nonzero. The data samples with non-vanishing Lagrange multipliers are called the support vectors inside or outside the $\boldsymbol{\varepsilon}$-insensitive tube [33].

The regression estimation function can be obtained by learning as follows:

$$f(\boldsymbol{x}) = \sum_{\boldsymbol{x}_i \in SV}(\alpha_i - \alpha_i^*)K(\boldsymbol{x}_i, \boldsymbol{x}) + b \tag{5}$$

thereinto,

$$\begin{aligned} b = \frac{1}{N_{NSV}}\{\sum_{0<\alpha_i<C}[y_i - \sum_{x_j \in SV}(\alpha_j - \alpha_j^*)K(\boldsymbol{x}_j, \boldsymbol{x}_i) - \boldsymbol{\varepsilon}] + \\ \sum_{0<\alpha_i^*<C}[y_i - \sum_{x_j \in SV}(\alpha_j - \alpha_j^*)K(\boldsymbol{x}_j, \boldsymbol{x}_i) + \boldsymbol{\varepsilon}]\} \end{aligned} \tag{6}$$

where N_{NSV} represents the number of standard support vectors. $K(\mathbf{x}_i, \mathbf{x}_j)$ is defined as the kernel function. According to Hilbert-Schmidt principle, when kernel function matches Mercer conditions, that is, for any given function $g(x)$, if $\int_a^b g^2(x)dx$ is limited, the value of the kernel is equal to the dot product of two vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ in the feature space $\Phi(\mathbf{x}_i)$ and $\Phi(\mathbf{x}_j)$, i.e., $K(\mathbf{x}_i, \mathbf{x}_j) = \langle\Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j)\rangle$ [33].

We choose here the Gauss radial basis function as kernel function.

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\| \boldsymbol{x} - \boldsymbol{x}_i\|^2}{\sigma^2}\right), \tag{7}$$

where σ is the kernel parameter.

Adaptive genetic algorithm

Genetic Algorithm (GA) derives from the computer simulation study of biological system [37], which has been widely used function optimization, combinatorial optimization, job shop scheduling problems [38], complex network clustering, pattern mining [39–41]. However, there are still some disadvantages, the most obvious disadvantages are the low efficiency and easy to fall into local optimum [42, 43].

In 2000, adaptive Genetic Algorithm (AGA) [44] is proposed, which improves the performance of traditional GA to some extent. After that, adaptive GA is improved by involving certain intelligent strategies, including crossover to avoid inbreeding, crossover probability associated with the number of evolution and regulating adaptive mutation probability [45]. The formula which is only related to the number of evolution for cross-probabilistic computing is as follows:

$$m_{tmp} = P_{c,max} * 2^{-\frac{t}{T_{Gen}}} \tag{8}$$

$$P_c(t) = \begin{cases} m_{tmp} & , \quad m_{tmp} > P_{c,min} \\ P_{c,min} & , \quad m_{tmp} \leqslant P_{c,min} \end{cases} \tag{9}$$

In the formula, m_{tmp} is an intermediate variable for calculation, T_{Gen} is the maximum evolutionary number preset, t is the current evolutionary number ($0 \leq t \leq T_{Gen}$), $P_{c,\,max}$ is the largest crossover probability preset, $P_{c,\,min}$ is the smallest crossover probability preset, and $P_c(t)$ is the crossover probability of current population.

The formula of adaptive mutation probability related to the number of genetic evolution and individual fitness is as follows:

$$m_{tmp} = \exp\left[-\left|\frac{f_{max} - f(\boldsymbol{x}_i)}{f_{max}}\right|\right] \cdot \frac{1}{1 + \frac{t}{T_{Gen}}} \cdot P_{m,max} \tag{10}$$

$$P_m(t) = \begin{cases} m_{tmp} & , \quad m_{tmp} > P_{m,min} \\ P_{m,min} & , \quad m_{tmp} \leqslant P_{m,min} \end{cases} \tag{11}$$

In the formula, $P_{m,\,max}$ is the largest mutation probability preset, $P_{m,\,min}$ is the smallest mutation probability preset, $f(\mathbf{x}_i)$ is the fitness value of individual $\mathbf{x}_i$, f_{max} is the maximum value of fitness in current populations, $P_m(t)$ is the mutation probability of individual $\mathbf{x}_i$ in current population [45].

The mathematic model and data experiments

In this section, it starts by selecting probable elements from original data, and then the values of two important parameters of the model are determined. After that, the mathematic model

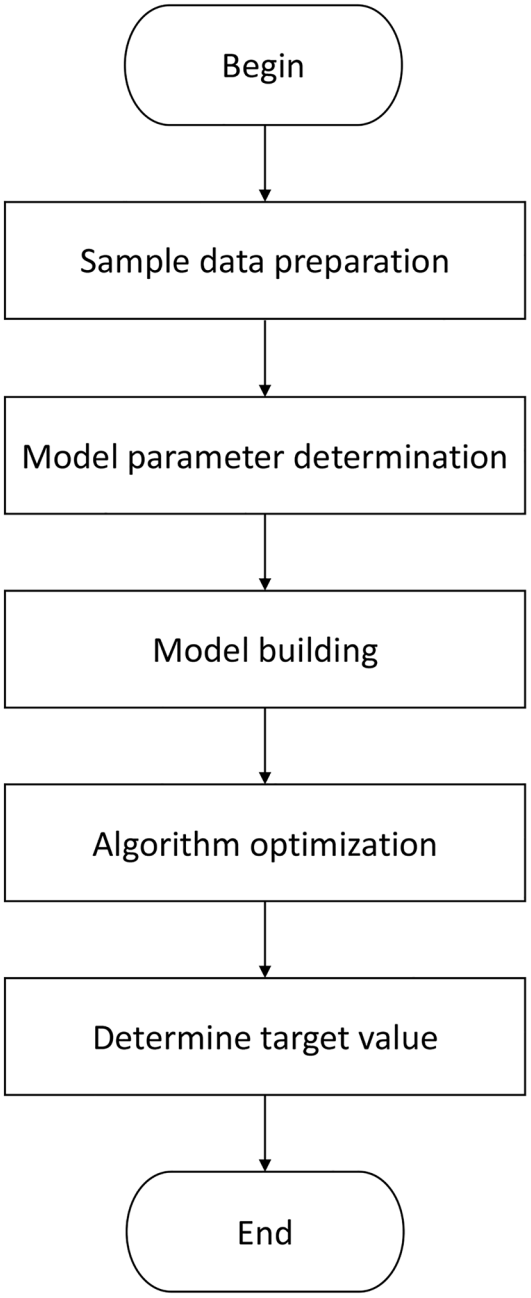

Fig 1. Main work flow chart.

https://doi.org/10.1371/journal.pone.0185942.g001

based on SVR is built to describe the relationship between Welan gum products and experimental conditions. With the model, AGA is applied to find the optimal sample point of the model, which corresponds to a class of potential optimal experimental conditions to maximize the production of Welan gum. The flowchart is shown in Fig 1.

The mathematic model

Data preparation. Before building the mathematic model for describing the relationship between Welan gum production and experimental conditions, it needs to normalize the data.

SVR mainly deals with the nonlinear problems, so the magnitude of the eigenvalues of the samples should be different greatly, the results will be greatly affected without normalizing samples. Besides, normalizing samples can avoid the small weight of the model and leading to the instability of the numerical calculation, so that the parameter optimization can converge at a faster speed and the accuracy of the model can be improved. The normalized formula used in our method is as follows:

$$y = \frac{(y_{max} - y_{min}) \cdot (x - x_{min})}{x_{max} - x_{min}} + y_{min}, \tag{12}$$

where x is the original data, y is the normalized data, x_{min} is the minimum of the original data, x_{max} is the maximum of the original data, y_{min} is the minimum of the normalized data, y_{max} is the maximum of the normalized data. The value of y_{min} is set to be 0 and the value of y_{max} to be 1. The normalized data is shown in Tables 1 and 2 below:

Without losing the generality, all 67 samples collected from Welan gum producing experiments are classified according to the production, which are divided into three types: high, middle and low level production. Specifically, productions between 0g/L and 5g/L belong to low level production data, in total 8 groups; productions between 5g/L and 20g/L are in medium level, in total 39 groups; productions more than 20g/L are in high level, in total 20 groups.

Table 1. Sample data before normalization.

	glucose (g/L)	yeast (g/L)	KH_2PO_4 (g/L)	$MgSO_4$ (g/L)	liquid volume (ml)	PH value	temperature (˚C)	rotational speed (rpm)	inoculation amount	production (g/L)
1	40	2	5	0.1	50	10	28	150	5	0.9084
2	40	2	5	0.1	50	2	28	150	5	1.1484
3	40	2	5	0.1	50	3	28	150	5	1.6588
4	40	2	5	0.1	50	9	28	150	5	1.914
5	40	2	5	0.1	50	4	28	150	5	2.9348
6	60	10	5	0.1	50	7	32.5	175	5	3.08
7	40	2	5	0.1	50	5	28	150	5	4.0832
8	40	2	5	0.1	50	5.5	28	150	5	4.5936
9	40	2	5	0.1	50	8	28	150	5	6.2496
10	60	9	5	0.1	50	7	32.5	175	5	6.29
11	10	2	5	0.1	50	7	32.5	175	5	6.75
12	40	2	5	0.1	50	6	28	150	5	8.1664
13	60	8	5	0.1	50	7	32.5	175	5	8.7
14	20	2	5	0.1	50	7	32.5	175	5	9.23
15	40	2	5	0.1	50	6.8	28	150	1	10.73
16	40	2	5	0.1	50	7.5	28	150	5	10.9084
17	40	2	5	0.1	50	6.8	28	150	10	11.52
18	40	2	5	0.1	50	6.8	28	150	8	12.05
19	40	2	5	0.1	50	6.8	28	150	7	12.28
20	40	2	5	0.1	50	6.8	28	150	3	12.68
21	60	1	5	0.1	50	7	32.5	175	5	12.8
22	40	2	5	0.1	50	7	32.5	125	5	12.982
23	40	2	5	0.1	50	6.8	28	150	6	13.45
24	40	2	5	0.1	50	6.5	28	150	5	14.036
25	60	7	5	0.1	50	7	32.5	175	5	14.31

https://doi.org/10.1371/journal.pone.0185942.t001

Table 2. Sample data after normalization.

	glucose (g/L)	yeast (g/L)	KH_2PO_4 (g/L)	$MgSO_4$ (g/L)	liquid volume (ml)	PH value	temperature (°C)	rotational speed(rpm)	inoculation amount	production (g/L)
1	0.375	0.1111	1	0	0.25	1	0.3	0.25	0.4444	0
2	0.375	0.1111	1	0	0.25	0	0.3	0.25	0.4444	0.005777
3	0.375	0.1111	1	0	0.25	0.125	0.3	0.25	0.4444	0.018064
4	0.375	0.1111	1	0	0.25	0.875	0.3	0.25	0.4444	0.024207
5	0.375	0.1111	1	0	0.25	0.25	0.3	0.25	0.4444	0.04878
6	0.625	1	1	0	0.25	0.625	0.75	0.5	0.4444	0.052275
7	0.375	0.1111	1	0	0.25	0.375	0.3	0.25	0.4444	0.076425
8	0.375	0.1111	1	0	0.25	0.4375	0.3	0.25	0.4444	0.088711
9	0.375	0.1111	1	0	0.25	0.75	0.3	0.25	0.4444	0.128575
10	0.625	0.8889	1	0	0.25	0.625	0.75	0.5	0.4444	0.129547
11	0	0.1111	1	0	0.25	0.625	0.75	0.5	0.4444	0.14062
12	0.375	0.1111	1	0	0.25	0.5	0.3	0.25	0.4444	0.174716
13	0.625	0.7778	1	0	0.25	0.625	0.75	0.5	0.4444	0.187561
14	0.125	0.1111	1	0	0.25	0.625	0.75	0.5	0.4444	0.20032
15	0.375	0.1111	1	0	0.25	0.6	0.3	0.25	0	0.236428
16	0.375	0.1111	1	0	0.25	0.6875	0.3	0.25	0.4444	0.240723
17	0.375	0.1111	1	0	0.25	0.6	0.3	0.25	1	0.255445
18	0.375	0.1111	1	0	0.25	0.6	0.3	0.25	0.7778	0.268203
19	0.375	0.1111	1	0	0.25	0.6	0.3	0.25	0.6667	0.27374
20	0.375	0.1111	1	0	0.25	0.6	0.3	0.25	0.2222	0.283369
21	0.625	0	1	0	0.25	0.625	0.75	0.5	0.4444	0.286258
22	0.375	0.1111	1	0	0.25	0.625	0.75	0	0.4444	0.290639
23	0.375	0.1111	1	0	0.25	0.6	0.3	0.25	0.5556	0.301905
24	0.375	0.1111	1	0	0.25	0.5625	0.3	0.25	0.4444	0.316011
25	0.625	0.6667	1	0	0.25	0.625	0.75	0.5	0.4444	0.322607

https://doi.org/10.1371/journal.pone.0185942.t002

Each time the model data is taken, the order of the samples within each yield is randomly arranged, For each level data groups, the first 70% of each type data is used as training data, the 30% data left are used as the testing data.

Before building the mathematic model, it is necessary to determine the values of two parameters, namely penalty factor parameters (c) and kernel function parameters (g). Here, grid search method is used to determine the optimal values of the two parameters. The result is shown in Fig 2 below:

In the above figure of contour line, two red dotted lines are represented separately the optimal values of the two parameters. The intersection of two lines, that is, the red point in the figure represents the value of the "CVmse". The CVmse means that the mean of the squares of the difference between the predicted value and the true value under the 5-fold cross validation.

After the values of the parameters are determined, the training data and testing data are determined according to the selection of the aforementioned method. The index of the accuracy of the model is reflected in the square of correlation coefficient. The diagrams in Figs 3 and 4 reflect the model's prediction of the testing data and the relative error.

Finding optimal experimental conditions by AGA

With the mathematical model constructed, an improved AGA is used to find experimental conditions for optimal production. The process has the following steps.

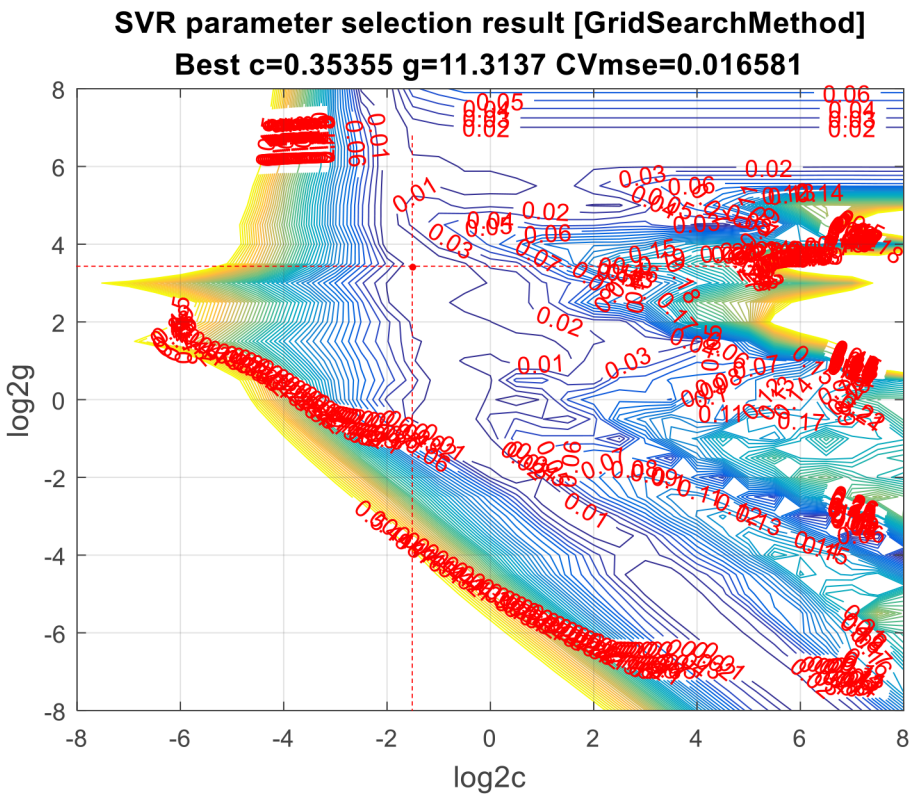

Fig 2. SVR parameter selection result[GridSearchMethod].

https://doi.org/10.1371/journal.pone.0185942.g002

Step 1: Initialize the population and encode the individuals.

Each sample is related to nine variables, so we consider the nine variables as nine genes that make up a chromosome. For example, encode [glucose, yeast, KH_2PO_4, $MgSO_4$, fluid volume, PH value, temperature, rotational speed, inoculation amount] to [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9], where $x_1 \in [5, 95]$, $x_2 \in [1, 10]$, $x_3 \in [1, 6]$, $x_4 \in [0.1, 1]$, $x_5 \in [25, 125]$, $x_6 \in [2, 12]$, $x_7 \in [25, 35]$, $x_8 \in [125, 250]$, $x_9 \in [1, 10]$.

Step 2: Select good individuals based on the fitness values.

Step 3: Perform crossover operation. From the first individual in the population, the corresponding crossover probability of the individual is calculated, denoted as cross_rate. We randomly generate a random number between 0 and 1, denoted as rand_num. If the value of rand_num is less than cross_rate, the individual is performed crossover operation. That is, two integers between 1 and 9 are randomly generated, where the smaller number is the starting position of the crossed chromosome, the larger number is the ending position, the chromosome of the individual is exchanged with the chromosome of the next adjacent individual, in the range from the starting position to the termination position. In addition, if the i-th individual did not perform the crossover operation, the above-described process is repeated for the i+1-th individual; if the i-th individual performed the crossover operation, the above-described process is repeated for the i+2-th.

Step 4: Perform mutation operation. From the first individual in the population, the corresponding mutation probability of the individual is calculated, denoted as mutate_rate. We randomly generate a random number between 0 and 1, denoted as rand_num. If the value of rand_num is less than mutate_rate, the individual is performed mutation operation. That is,

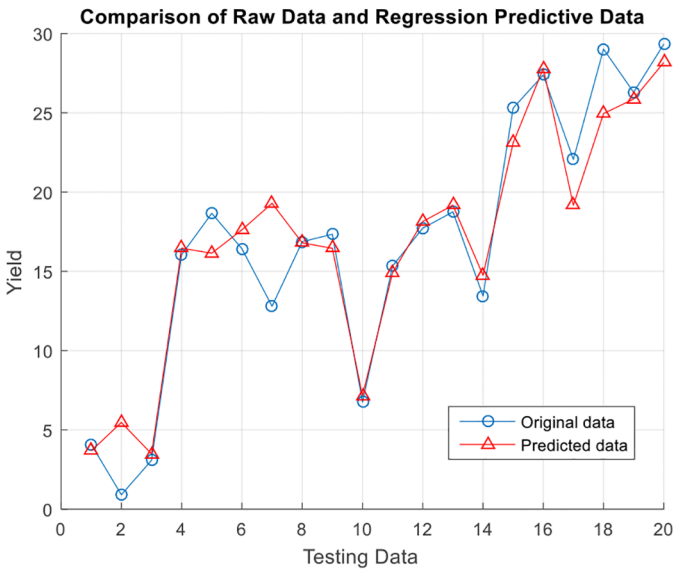

Fig 3. Comparison of raw data and regression predictive data.

https://doi.org/10.1371/journal.pone.0185942.g003

an integer between 1 and 9 is randomly generated as the location of the gene that needs to be mutated, regenerate the gene at the location.

Step 5: The new individuals generated by the above operations constitute the new population, and go to step 2.

Repeat these steps until we find the optimal individual.

The size of initial population is set to be 300, that is there are 300 individuals, the number of iterations is 500. The selection operator is roulette selection method, which is also known as the proportional selection operator. The basic idea is that the probability of each individual

Fig 4. Relative error plot.

https://doi.org/10.1371/journal.pone.0185942.g004

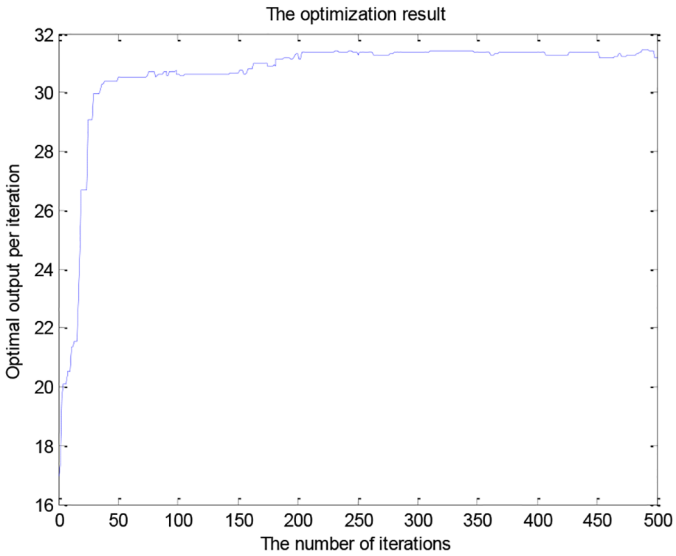

Fig 5. The optimization result.

https://doi.org/10.1371/journal.pone.0185942.g005

selected is proportional to its fitness value.

$$P(x_i) = \frac{f(x_i)}{\sum_{i=1}^{K} f(x_i)}, \tag{13}$$

where $P(x_i)$ is the selection probability of individual x_i, K is the population size. The value of parameter $P_{c,min}$ is set to be 0.6, $P_{c,max}$ to be 0.9, $P_{m,max}$ to be 0.1 and $P_{m,max}$ to be 0.001. The search results are shown in Fig 5.

To improve the accuracy and further reduce the range of the nine gene variables. We made the following changes by observing the genetic variables of samples with productions higher than 30g/L, which is $x_1 \in [55, 60]$, $x_2 \in [2.5, 3.1]$, $x_3 \in [5, 5.5]$, $x_4 \in [0.1, 0.3]$, $x_5 \in [48, 51.5]$, $x_6 \in [6.7, 7.15]$, $x_7 \in [32, 33]$, $x_8 \in [176, 179]$, $x_9 \in [4.85, 5.15]$. The average maximum fitness value of data experiments with 500 iterations each time is shown in Fig 6.

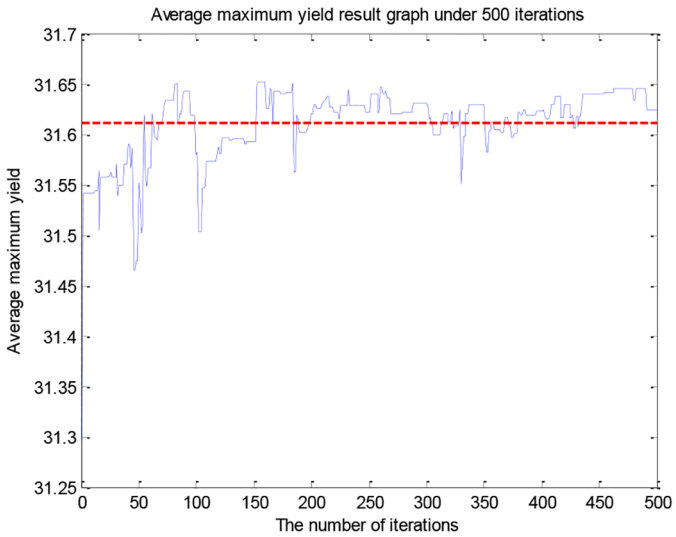

Fig 6. The average maximum yield result graph under 500 iterations.

https://doi.org/10.1371/journal.pone.0185942.g006

Table 3. The optimal medium composition ratio.

glucose (g/L)	yeast (g/L)	KH_2PO_4 (g/L)	$MgSO_4$ (g/L)	liquid volume (ml)	PH value	temperature (˚C)	rotational speed(rpm)	inoculation amount
55.26	2.89	5.23	0.1	49.8	7.01	32.53	177.51	5

https://doi.org/10.1371/journal.pone.0185942.t003

Results

The accuracy of the established mathematic model is 88.36%, the optimal medium composition ratio is shown in Table 3 below:

The maximum production of Welan gum is 31.65g/L.

This hybrid computational method, which combines with SVM and AGA, has the intelligent learning ability and can overcome the limitation of large-scale biotic experiments [46–51]. A mathematic model of predicting production of Welan gum from experimental conditions with accuracy rate 88.36% is obtained, a class of optimized experimental conditions is designed to produce Welan gum 31.65g/L. Comparing the best results in chemical experiment 30.63g/L, the predicted production can be improved by 3.3%.

Conclusion

We focused on building a mathematic model of Welan gum, the nine factors which contribute the experimental conditions of producing Welan gum as preparative optimization indicators. The nine factors include glucose, yeast, KH_2PO_4, $MgSO_4$, fluid volume, PH value, temperature, rotational speed and inoculation amount. A hybrid computational method combined with SVM and AGA is proposed. Through the training of sample data, a mathematic model of predicting production of Welan gum from experimental conditions is obtained. We find the optimal sample point in the sample space, i.e. a class of optimized experimental conditions. This hybrid computational method has a good learning ability, which can avoid the high cost problem caused by large-scale biological experiments. It also overcomes the "mature" defects of traditional Genetic Algorithm. The result provides a potential experimental conditions by data mining to improve the production of Welan gum in the lab.

For further research, neural-like computing models, e.g., spiking neural P systems [52] can be used for optimization of Welan gum production. As well, some recently developed data processing and mining methods, such as the speculative approach to spatial-temporal efficiency for multi-objective optimization in cloud data and computing [53], privacy-preserving smart similarity search methods in simhash over encrypted data in cloud computing [53], k-degree anonymity with vertex and edge modification algorithm [54], kernel quaternion principal component analysis for object recognition [55], might be used for optimizing experimental conditions of Welan gum. In the aspect of data preparation, decision tree [56] can be used to deal with the missing attribute value of some samples in dataset.

Acknowledgments

This work was supported by 863 program (2015AA020925), National Natural Science Foundation of China (61402187, 61502535, 61572522, 61572523, 61672033 and 61672248), Key Research and Development Program of Shandong Province (No. 2017GGX10147), China Postdoctoral Science Foundation funded project (2016M592267), PetroChina Innovation Foundation (2016D-5007-0305), Fundamental Research Funds for the Central Universities (R1607005A). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Author Contributions

Conceptualization: Zhongwei Li, Xin Liu, Hu Zhu.

Data curation: Xiang Yuan.

Formal analysis: Zhongwei Li.

Project administration: Zhongwei Li.

Software: Xiang Yuan, Xuerong Cui.

Supervision: Zhongwei Li, Hu Zhu.

Validation: Xiang Yuan.

Writing – original draft: Xiang Yuan.

Writing – review & editing: Xin Liu, Leiquan Wang, Weishan Zhang, Qinghua Lu, Hu Zhu.

References

1. Long K, Li X, Xie T, Zhang Y, Liu W. Welan gum production from cyperus esculentus fermented by sphingomonas sp. ATCC 31555. Chemical Engineer. 2014; 8:002.
2. Li H, Li S, Feng X, Wang F, Xu H. Production of Welan Gum by Alcaligenes sp.NX-3 with Fed-batch Fermentation. Food & Fermentation Industries. 2009; 35(1):1–4.
3. Li H, Xu H, Li S, Feng X, Xu H, Ouyang P. Effects of dissolved oxygen and shear stress on the synthesis and molecular weight of welan gum produced from Alcaligenes sp. CGMCC2428. Process Biochemistry. 2011; 46(5):1172–1178. https://doi.org/10.1016/j.procbio.2011.02.007
4. Liang J, Li Z, Chen B. Optimization of Fermentation Media for Welan Gum Using JMP. Food Research And Development. 2016; 37(18):104–108.
5. Xue Y, Jiang J, Zhao B, Ma T. A self-adaptive artificial bee colony algorithm based on global best for global optimization. Soft Computing. 2017; (8):1–18.
6. Shen J, Shen J, Chen X, Huang X, Susilo W. An Efficient Public Auditing Protocol With Novel Dynamic Structure for Cloud Data. IEEE Transactions on Information Forensics & Security. 2017; 12(10):2402–2415. https://doi.org/10.1109/TIFS.2017.2705620
7. Fu Z, Huang F, Ren K, Weng J, Wang C. Privacy-Preserving Smart Semantic Search Based on Conceptual Graphs Over Encrypted Outsourced Data. IEEE Transactions on Information Forensics & Security. 2017; 12(8):1874–1884. https://doi.org/10.1109/TIFS.2017.2692728
8. Xia Z, Wang X, Zhang L, Qin Z, Sun X, Ren K. A Privacy-Preserving and Copy-Deterrence Content-Based Image Retrieval Scheme in Cloud Computing. IEEE Transactions on Information Forensics & Security. 2016; 11(11):2594–2608. https://doi.org/10.1109/TIFS.2016.2590944
9. Zhao Y, Liu X, Wang W. Spiking Neural P Systems with Neuron Division and Dissolution. PLOS ONE. 2016; 11(9):e0162882. https://doi.org/10.1371/journal.pone.0162882 PMID: 27627104
10. Liu X, Zhao Y, Sun M. An Improved Apriori Algorithm Based on an Evolution-Communication Tissue-Like P System with Promoters and Inhibitors. Discrete Dynamics in Nature and Society. 2017; 2017: https://doi.org/10.1155/2017/6978146
11. Liu X, Xue J. A Cluster Splitting Technique by Hopfield Networks and P Systems on Simplices. Neural Processing Letters. 2017; 46(1):171–194. https://doi.org/10.1007/s11063-016-9577-z
12. Liu X, Xiang L, Wang X. Spatial Cluster Analysis by the Adleman-Lipton DNA Computing Model and Flexible Grids. Discrete Dynamics in Nature and Society. 2012; 2012(1–4):132–148.
13. Liu X, Liu H, Duan H. Particle swarm optimization based on dynamic niche technology with applications to conceptual design. Advances in Engineering Software. 2006; 38(10):668–676. https://doi.org/10.1016/j.advengsoft.2006.10.009
14. Xia Z, Wang X, Sun X, Wang Q. A Secure and Dynamic Multi-keyword Ranked Search Scheme over Encrypted Cloud Data. IEEE Transactions on Parallel & Distributed Systems. 2016; 27(2):340–352. https://doi.org/10.1109/TPDS.2015.2401003
15. Vapnik V. The nature of statistical learning theory. Springer science & business media; 2013.
16. Wang X, Miao Y, Cheng M. Finding motifs in DNA sequences using low-dispersion sequences. Journal of Computational Biology. 2014; 21(4):320–329. https://doi.org/10.1089/cmb.2013.0054 PMID: 24597706

17. Wang X, Miao Y. GAEM: a hybrid algorithm incorporating GA with EM for planted edited motif finding problem. Current Bioinformatics. 2014; 9(5):463–469. https://doi.org/10.2174/1574893609666140901222327

18. Wu T, Wang X, Zhang Z, Gong F, Song T, Chen Z, et al. NES-REBS: a novel nuclear export signal prediction method using regular expressions and biochemical properties. Journal of bioinformatics and computational biology. 2016; 14(03):1650013. https://doi.org/10.1142/S021972001650013X PMID: 27225342

19. Liu B, Zhang D, Xu R, Xu J, Wang X, Chen Q, et al. Combining evolutionary information extracted from frequency profiles with sequence-based kernels for protein remote homology detection. Bioinformatics. 2013; 30(4):472–479. https://doi.org/10.1093/bioinformatics/btt709 PMID: 24318998

20. Zeng X, Liao Y, Liu Y, Zou Q. Prediction and validation of disease genes using HeteSim Scores. IEEE/ACM Transactions on Computational Biology and Bioinformatics. 2017; 14(3):687–695. https://doi.org/10.1109/TCBB.2016.2520947 PMID: 26890920

21. Liu B, Xu J, Lan X, Xu R, Zhou J, Wang X, et al. iDNA-Prot| dis: identifying DNA-binding proteins by incorporating amino acid distance-pairs and reduced alphabet profile into the general pseudo amino acid composition. PloS one. 2014; 9(9):e106691. https://doi.org/10.1371/journal.pone.0106691 PMID: 25184541

22. Zeng X, Zhang X, Liao Y, Pan L. Prediction and validation of association between microRNAs and diseases by multipath methods. Biochimica et Biophysica Acta (BBA)-General Subjects. 2016; 1860(11):2735–2739. https://doi.org/10.1016/j.bbagen.2016.03.016

23. Wang X, Song T, Pan Z, Hao MT Shaohua. Spiking Neural P Systems with Anti-Spikes and without Annihilating Priority. Romanian Journal of Information Science and Technology. 2017; 20(1):32–41.

24. Vapnik VN, Vapnik V. Statistical learning theory. vol. 1. Wiley New York; 1998.

25. Burges CJ. A tutorial on support vector machines for pattern recognition. Data mining and knowledge discovery. 1998; 2(2):121–167. https://doi.org/10.1023/A:1009715923555

26. Wu Y, Krishnan S. Combining least-squares support vector machines for classification of biomedical signals: a case study with knee-joint vibroarthrographic signals. Journal of Experimental & Theoretical Artificial Intelligence. 2011; 23(1):63–77. https://doi.org/10.1080/0952813X.2010.506288

27. Cai S, Yang S, Zheng F, Lu M, Wu Y, Krishnan S. Knee joint vibration signal analysis with matching pursuit decomposition and dynamic weighted classifier fusion. Computational and mathematical methods in medicine. 2013; 2013. https://doi.org/10.1155/2013/904267

28. Xuegong Z. Introduction to statistical learning theory and support vector machines. Acta Automatica Sinica. 2000; 26(1):32–42.

29. Van Gestel T, Suykens JA, Baesens B, Viaene S, Vanthienen J, Dedene G, et al. Benchmarking least squares support vector machine classifiers. Machine Learning. 2004; 54(1):5–32. https://doi.org/10.1023/B:MACH.0000008082.80494.e0

30. Amari Si, Wu S. Improving support vector machine classifiers by modifying kernel functions. Neural Networks. 1999; 12(6):783–789. https://doi.org/10.1016/S0893-6080(99)00032-5 PMID: 12662656

31. Chen W, Xing P, Zou Q. Detecting N6-methyladenosine sites from RNA transcriptomes using ensemble Support Vector Machines. Scientific reports. 2017; 7:40242. https://doi.org/10.1038/srep40242 PMID: 28079126

32. Wu Y, Luo X, Zheng F, Yang S, Cai S, Ng SC. Adaptive linear and normalized combination of radial basis function networks for function approximation and regression. Mathematical Problems in Engineering. 2014; 2014.

33. Wei G, Yu X, Long X. Novel approach for identifying Z-axis drift of RLG based on GA-SVR model. Journal of Systems Engineering and Electronics. 2014; 25(1):115–121. https://doi.org/10.1109/JSEE.2014.00013

34. Burges CJ. Geometry and invariance in kernel based methods. Advances in kernel methodssupport vector learning. 1999; p. 89–116.

35. Schölkopf B, Burges CJ. Advances in kernel methods: support vector learning. MIT press; 1999.

36. Shevade SK, Keerthi SS, Bhattacharyya C, Murthy KRK. Improvements to the SMO algorithm for SVM regression. IEEE transactions on neural networks. 2000; 11(5):1188–1193. https://doi.org/10.1109/72.870050 PMID: 18249845

37. Goldberg DE, Holland JH. Genetic algorithms and machine learning. Machine learning. 1988; 3(2):95–99. https://doi.org/10.1023/A:1022602019183

38. Zhang L, Pan H, Su Y, Zhang X, Niu Y. A Mixed Representation-Based Multiobjective Evolutionary Algorithm for Overlapping Community Detection. IEEE Transactions on Cybernetics. 2017;. https://doi.org/10.1109/TCYB.2017.2711038

39. Ju Y, Zhang S, Ding N, Zeng X, Zhang X. Complex network clustering by a multi-objective evolutionary algorithm based on decomposition and membrane structure. Scientific reports. 2016; 6. https://doi.org/10.1038/srep33870

40. Zhang X, Duan F, Zhang L, Cheng F, Jin Y, Tang K. Pattern Recommendation in Task-oriented Applications: A Multi-Objective Perspective;.

41. Song T, Gong F, Liu X, Zhao Y, Zhang X. Spiking neural P systems with white hole neurons. IEEE transactions on nanobioscience. 2016; 15(7):666–673. https://doi.org/10.1109/TNB.2016.2598879 PMID: 28029614

42. Zeng X, Yuan S, Huang X, Zou Q. Identification of cytokine via an improved genetic algorithm. Frontiers of Computer Science: Selected Publications from Chinese Universities. 2015; 9(4):643–651. https://doi.org/10.1007/s11704-014-4089-3

43. Song T, Pan L. Spiking neural P systems with request rules. Neurocomputing. 2016; 193:193–200. https://doi.org/10.1016/j.neucom.2016.02.023

44. Srinivas M, Patnaik LM. Adaptive probabilities of crossover and mutation in genetic algorithms. IEEE Transactions on Systems, Man, and Cybernetics. 1994; 24(4):656–667. https://doi.org/10.1109/21.286385

45. Ouyang S. A New Improved Genetic Algorithm. Computer Engineering & Applications. 2003;.

46. Li Z, Sun B, Xin Y, Wang X, Zhu H. A Computational Method for Optimizing Experimental Environments for Phellinus igniarius via Genetic Algorithm and BP Neural Network. BioMed Research International. 2016; 2016.

47. Wang X, Song T, Gong F, Zheng P. On the computational power of spiking neural P systems with self-organization. Scientific reports. 2016; 6:27624. https://doi.org/10.1038/srep27624 PMID: 27283843

48. Zhang X, Tian Y, Cheng R, Jin Y. A decision variable clustering-based evolutionary algorithm for large-scale many-objective optimization. IEEE Transactions on Evolutionary Computation. 2016;. https://doi.org/10.1109/TEVC.2016.2600642

49. Zhang X, Tian Y, Jin Y. A knee point-driven evolutionary algorithm for many-objective optimization. IEEE Transactions on Evolutionary Computation. 2015; 19(6):761–776. https://doi.org/10.1109/TEVC.2014.2378512

50. Song T, Wang X, Zhang Z, Chen Z. Homogenous spiking neural P systems with anti-spikes. Neural Computing & Applications. 2014; 24.

51. Song T, Zheng P, Wong MD, Wang X. Design of logic gates using spiking neural P systems with homogeneous neurons and astrocytes-like control. Information Sciences. 2016; 372:380–391. https://doi.org/10.1016/j.ins.2016.08.055

52. Song T, Xu J, Pan L. On the universality and non-universality of spiking neural P systems with rules on synapses. IEEE Transactions on NanoBioscience. 2015; 14(8):960–966. https://doi.org/10.1109/TNB.2015.2503603 PMID: 26625420

53. Liu Q, Cai W, Shen J, Fu Z, Liu X, Linge N. A speculative approach to spatial-temporal efficiency with multi-objective optimization in a heterogeneous cloud environment. Security & Communication Networks. 2016; 9(17):4002–4012. https://doi.org/10.1002/sec.1582

54. Ma T, Zhang Y, Cao J, Shen J, Tang M, Tian Y, et al. KDVEM: a (k)-degree anonymity with vertex and edge modification algorithm. Computing. 2015; 97(12):1165–1184. https://doi.org/10.1007/s00607-015-0453-x

55. Chen B, Yang J, Jeon B, Zhang X. Kernel quaternion principal component analysis and its application in RGB-D object recognition. Neurocomputing. 2017;. https://doi.org/10.1016/j.neucom.2017.05.047

56. Wang R, Kwong S, Wang XZ, Jiang Q. Segment Based Decision Tree Induction With Continuous Valued Attributes. IEEE Transactions on Cybernetics. 2015; 45(7):1262. https://doi.org/10.1109/TCYB.2014.2348012 PMID: 25291806