GEOMATICS AND ENVIRONMENTAL ENGINEERING • Volume 17 • Number 2 • 2023

https://doi.org/10.7494/geom.2023.17.2.31

Salah Arif[1], Adel Djellal[2], Nawel Djebbari[3], Saber Belhaoues[4], Hassen Touati[5], Fatma Zohra Guellati[6], Mourad Bensouilah[7]

Modelling *Microcystis* Cell Density in a Mediterranean Shallow Lake of Northeast Algeria (Oubeira Lake), Using Evolutionary and Classic Programming

Abstract: Caused by excess levels of nutrients and increased temperatures, freshwater cyanobacterial blooms have become a serious global issue. However, with the development of artificial intelligence and extreme learning machine methods, the forecasting of cyanobacteria blooms has become more feasible. We explored the use of multiple techniques, including both statistical [Multiple Regression Model (MLR) and Support Vector Machine (SVM)] and evolutionary [Particle Swarm Optimization (PSO), Genetic Algorithm (GA), and Bird Swarm Algorithm (BSA)], to approximate models for the prediction of *Microcystis* density. The data set was collected from Oubeira Lake, a natural shallow Mediterranean lake in the northeast of Algeria. From the correlation analysis of ten water variables monitored, six potential factors including temperature, ammonium, nitrate, and ortho-phosphate were selected. The performance indices showed; MLR and PSO provided the best results. PSO gave the best fitness but all techniques performed well. BSA had better fitness but was very slow across generations. PSO was faster than the other techniques and at generation 20 it passed BSA. GA passed BSA a little further, at generation 50. The major contributions of our work not only focus on the modelling process itself, but also take into consideration the main factors affecting *Microcystis* blooms, by incorporating them in all applied models.

Keywords: *Microcystis* cell density, Multiple Linear Regression, Support Vector Machine, Particle Swarm Optimization, Genetic Algorithm, Bird Swarm Algorithm

Received: 17 August 2021; accepted: 22 November 2022

[1] Badji Mokhtar University, Ecobiology Laboratory for Marine Environments and Coastal Areas, Annaba, Algeria, email: arifsalah23@hotmail.com (corresponding author), https://orcid.org/0000-0002-8341-9688

[2] Higher School for Industrial Technology, Energy Systems Technology Research Laboratory, Annaba, Algeria, email: a.djellal@esti-annaba.dz, https://orcid.org/0000-0002-5723-0099

[3] Chadli Bendjedid University, Department of Marine Biology, El Tarf, Algeria, email: djebari_nawel@yahoo.fr, https://orcid.org/0000-0002-4851-9093

[4] Badji Mokhtar University, Ecobiology Laboratory for Marine Environments and Coastal Areas, Annaba, Algeria, email: belhaoues.saber@yahoo.fr, https://orcid.org/0000-0001-6899-7431

[5] University of 8 Mai 1945 Guelma, Faculty of Natural and Life Sciences and Earth and Universe Sciences, Department of Biology, Guelma, Algeria, email: touati-hassen@hotmail.com, https://orcid.org/0000-0002-4185-5449

[6] Badji Mokhtar University, Ecobiology Laboratory for Marine Environments and Coastal Areas, Annaba, Algeria, email: guellati.fatma@yahoo.fr, https://orcid.org/0000-0001-7237-4341

[7] Badji Mokhtar University, Ecobiology Laboratory for Marine Environments and Coastal Areas, Annaba, Algeria, email: bensouilah_mourad@yahoo.fr, https://orcid.org/0000-0002-9574-5915

1. Introduction

Freshwater cyanobacterial blooms have become a serious global issue and are caused by excess levels of nutrients and increased temperature [1]. Exposure to cyanotoxins may affect public health and thus the reliable detection and quantification of cyanobacteria has become a priority in water quality management [2, 3]. However, the highly complex nonlinearity of water variables and their interactions make blooms difficult to model [4]. The bloom process is a complex dynamic system of multi-dimensional coordination associated with multiple factors, with a high, intrinsic non-linear dissipative structuring [5]. They are highly variable, and the parameters involved in their occurrence are unstable [6]. *Microcystis* predominates in several large lakes around the world, such as Lake Erie in North America and Lake Taihu in China [7–9]. It is also the most widespread species in the Mediterranean area, and its monitoring is a water quality priority [10–16]. With the development of artificial intelligence and extreme learning machine methods, the forecasting of cyanobacteria blooms has become more feasible [17]. Phenomena such as algal blooms inspired researchers to improve and develop evolutionary optimization algorithms within the frameworks of swarm intelligence, and natural selection like Particles Swarm Optimization (PSO), Bird Swarm Algorithms (BSA), and Genetic Algorithms (GA), to enhance convergence capabilities. Machine learning-based approaches have been used for a wide variety of applications in environmental sciences [18–20]. Wang [21] compared Support Vector Machine (SVM) and linear regression model for estimating phycocyanin pigment using band ratios as inputs. BSA [22], PSO [20], and GA [21] are recent meta-heuristic algorithms, which are global optimization algorithms that use a strong formulation strategy to achieve optimal or semi-optimal problem solutions.

This research aims to obtain the dependency relationship of the cyanobacteria *Microcystis* (output), as a function of the ten physical and chemical input parameters described further, reduced to six after a correlation analysis and combined according to [23]. Comparisons are made among the different models, showing that these advanced modelling techniques are effective new ways that can be used for monitoring *Microcystis* in water bodies.

2. Materials and Methods

2.1. Study Area

With 2200 ha of surface area, Oubeira Lake is the largest shallow freshwater lake in Algeria (Fig. 1). It is located in the far northeast of Algeria, in El-Kala National Park. It has four major tributaries: The Demet Rihana River in the north, Messida River in the south, Dey-Graa River in the east, and Boumerchen River in the northeast. Oubeira Lake is distinguished by its high biodiversity, as it is home to

numerous animal species, such as migratory and sedentary birds, fish, and mussels. It is also home to vegetal species dominated by floating macrophytes, such as the water chestnut (*Trapa natans*), the white water-lily (*Nymphaea alba*), and the yellow water-lily (*Nuphar lutea*). Several studies conducted in this endorheic lake revealed the existence of cyanobacteria and their cyanotoxins, with the prevalence of *Microcystis* and its toxin, microcystin [24–26].

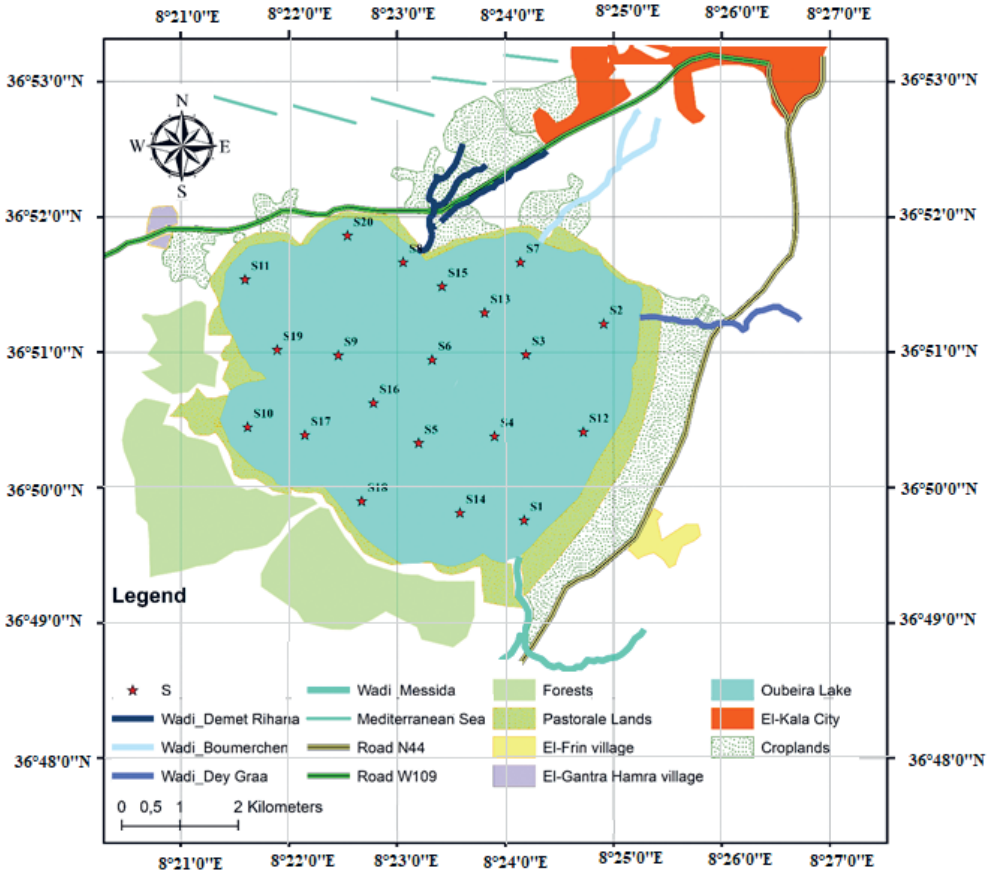


Fig. 1. Location of the different sampling stations (monthly and seasonally)

2.2. Experimental Dataset

The data sets used for the analysis were collected monthly over 12 months at 11 stations from April 2015 to March 2016 and seasonally over four seasons at 20 stations, from spring 2015 to winter 2016. The stations are designated as follows (S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}, and S_{11}) sampled monthly. While the seasonal sampling consisted of adding nine additional stations (S_{12}, S_{13}, S_{14}, S_{15}, S_{16}, S_{17}, S_{18}, S_{19}, and S_{20})

to the 11 of the monthly sampling to reach a total of 20 stations. The total number of data processed monthly was 132 values and 80 values seasonally for each parameter. The temporal distribution of this dataset was one year of monitoring ten environmental factors (inputs) and *Microcystis* densities (output). The biotic parameter information is expressed in cells per millilitre of *Microcystis*. It should be pointed out that the measurements were carried out between 8.00 a.m. and 2.00 p.m. The sampled stations were subjected to a pre-sampling survey. A strict field protocol was established to collect all of the information related to our stations, ensuring that all remarks and observations were recorded in the sampling day, as well as daily monitoring of meteorological phenomena and exceptional events occurring on our site. The choice of the location of the sampling stations was greatly influenced by: the hydrographic characteristics of Oubeira Lake, direction of prevailing winds; areas occupied by macrophytes; the presence of forests around the lake; areas frequented by fishermen; road layout; the existence of urban agglomerations; agricultural land; and pastures on its shores.

We considered the only dominant genus in the cyanobacteria community *Microcystis*. We thus obtained the dependency relationship of its density as a function of six maintained physical-chemical variables.

2.3. Description of *Microcystis* the Biotic Parameter (Output)

Microcystis [Cells · L^{-1}] is a cyanobacteria genus, unicellular – colonial, its colonies take a variety of shapes, with a colourless and homogeneous mucilage. Its cells are spherical, discoid, or irregular. The colonies are gelatinous, floating on the surface or fixed to the substrate, amorphous, irregular, even net-shaped, with vacuoles. Species are differentiated by cell size, distribution, colony structure, and mucilage texture. The cells are spherical or hemispherical after their division, with a homogeneous, greyish, yellowish, or blue-green filling. Several toxic strains can produce the hepatotoxin "microcystin", and have a global distribution. Among the most frequent are: *M. aeruginosa*, *M. wesenbergii*, *M. viridis*, *M. flos-aquae*.

2.4. Description of the Physicochemical Parameters (Abiotic Inputs Parameters)

Water temperature (*T*) [°C], dissolved oxygen (O_2) [mg · L^{-1}], conductivity (*Cond*) [μS · cm^{-1}], and pH were determined *in situ* using a multi-parameter (3420 IDS, WTW, Germany).

Transparency (*Trans*) [cm] was measured with the Secchi disc, the depth at which the patch on the disc becomes invisible is taken as a measurement of water clarity. Suspended solids (*SS*) [mg · L^{-1}] was determined by the differential weighing method after filtering the sample through a Whatman glass fibre filter (GF/C 47 mm).

Nutrients [μmol · L^{-1}]: nitrate (NO_3^-), nitrite (NO_2^-), ammonium (NH_4^+), and orthophosphate (PO_3^{3-})samples are filtered through a Whatman GF/C filter (0.45 μm). The measurements were carried out on the filtrate according to the colorimetric methods described by [27].

2.5. *Microcystis* Colonies Identification and Cells Enumeration

The raw water samples were filtered through a 20-μm mesh size plankton net and preserved in 5% formaldehyde. An additional filtration through polycarbonate membranes (47 mm diameter, Whatman, Germany) with a nominal porosity of 5 μm was conducted preceding the identification process and performed using an optical microscope (Carl Zeiss, Axiostar plus, Germany) equipped with a UI-1240 SE camera (IDS, Germany). The latter was used to take measurements of *Microcystis* colonies and cells. As proposed in the updated literature by [28, 29] the identification was based on the microscopic observation of morpho-anatomical criteria using the classical method. The count of cells was carried out by injecting a volume of the sample in the wells of the Nageotte counting cell (ISOLAB, 0.5 mm deep). This latter is a special slide with a grid of 40 strips, equal to a given surface area and a volume of 50 μL. The number of cells observed on a certain number of strips, therefore, corresponds to a certain volume, which allows an estimation per millilitre [30]. Cell densities in a colony were determined as a function of colony surface area and mean cell surface area. The number of cells was then obtained from the following formulas:

$$N_{cells} = \frac{S_c}{S_{cm}} - A \tag{1}$$

where:

N_{cells} – number of cells,
S_c – colonial surface area,
S_{cm} – cell mean surface,
A – visual estimation of the proportion $x/100$ of the colony void:

$$A = \frac{S_c}{S_{cm}} \cdot \frac{x}{100} \tag{2}$$

The obtained total number of colonial cells was then introduced into the formula below to determine the cellular density per litre of raw water:

$$N_{cells} / 50\ \mu L = \left(\frac{\sum n_{cells}}{b} \right) \frac{40}{50} \tag{3}$$

$$N_{cells} / \text{mL} = \frac{\frac{n_{cells}}{50\ \mu\text{L}}}{v} \tag{4}$$

where:

N_{cells} – the total of the cells counted from the different colonies
$\sum n_{cells}$ – the sum of the cells counted from the different colonies,
b – the number of strips on which we counted 30 individuals,
v – volume of the filtered sample [mL],
40 – total number of strips in the swimmer cell,
50 – volume of Nageotte's cell [μL].

To obtain a satisfactory estimation of cyanobacteria abundance, the counting process was replicated for each sample (3 to 5 observations).

3. Theory

3.1. *Microcystis* Model

To ensure some non-linearity in the inputs, the data set was adapted and a linear prediction model inspired by the work of [23] was modified:

$$Mictis_{predicted}(t) = b_0 + \sum_{i=1}^{m}\left(b_i X_i + \sum_{j=1}^{m}\left(b_{ij} X_i(t) X_j(t) \right) \right) \tag{5}$$

$$Mictis_{predicted}(t) = f(X_1, X_1^2, X_1X_2, \ldots, X_1X_m, X_2, X_2^2, X_2X_3, \ldots, X_m, X_m^2) \tag{6}$$

where:

$Mictis_{predicted}(t)$ – the predicted *Microcystis* density,
$X_i(t)$, $X_j(t)$ – physicochemical parameters,
b_i, b_{ij} – coefficients computed by later modelling algorithms.

This parameter combination will increase the input variables from only 6 to 28 parameters. To adapt Equation (17), in Section 3, to the generated variables in Equation (5), variable changing is required. This implies new variables $X_{ij} = X_i \cdot X_{ij} = X_i \cdot X_j$ and in this case Equation (5) becomes:

$$Mictis_{predicted} = b_0 + \sum_{i=1}^{m}\left(b_i X_i + \sum_{j=1}^{m}(b_{ij} X_{ij}) \right) \tag{7}$$

Equation (7) will be used in this work as the main *Microcystis* model and can be extended to *n* combinations as follows:

$$Mictis_{predicted} = b_0 + \sum_{i_2=1}^{m}\left(b_{i_1} X_{i_1} + \sum_{i_2=i_1}^{m}\left(b_{i_1 i_2} X_{i_1 i_2} + \ldots + \sum_{i_n=i_{n-1}}^{m}\left(b_{i_1,\ldots,i_n} X_{i_1,\ldots,i_n} \right)\right)\right) \tag{8}$$

where $X_{i_1,\ldots,i_n} = \prod_{j=i_1}^{i_n} X_j$ is the n^{th} variable combination. n = 2 is used and the main reason, as seen in Table 1, is due to the number of observations collected in our work, since we have only 132 observations for training and 80 for testing. Therefore, if we used a higher number of combinations, the number of coefficients would exceed the number of equations.

Table 1 presents the number of variables after combination using Equation (8) and for six maintained variables.

Table 1. New number of variables after combination

Number of variables	1	2	3	4	5	6
Number of combined variables	7	28	84	210	462	924

3.2. Model Performance Indices

We computed six performance indices and compared the developed models. The six indices, inspired by [19], are the coefficient of correlation (*R*) (9), the Willmott index of agreement (*d*) (10), the Nash–Sutcliffe efficiency (NSE) (11), the root mean squared error (RMSE) (12), the mean absolute error (MAE) (13), and the mean squared relative error (MSRE) (14). RMSE and *R* are used as the main comparison components, the other indices are computed as a reference for further work.

$$R = \frac{\frac{1}{N}\sum_{i=1}^{N}(O_i - O_m)(P_i - P_m)}{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(O_i - O_m)^2}\sqrt{\frac{1}{N}\sum_{i=1}^{N}(P_i - P_m)^2}} \tag{9}$$

$$d = 1 - \frac{\sum_{i=1}^{N}(P_i - O_i)^2}{\sum_{i=1}^{N}\left(\left|P_i - O_m\right| + \left|O_i - O_m\right|\right)^2} \tag{10}$$

$$\mathrm{NSE} = 1 - \frac{\sum_{i=1}^{N}(O_i - P_i)^2}{\sum_{i=1}^{N}(O_i - O_m)^2} \tag{11}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(O_i - P_i)^2} \tag{12}$$

$$\mathrm{MAE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left|O_i - P_i\right|} \tag{13}$$

$$\mathrm{MSRE} = \frac{\sum_{i=1}^{N}(O_i - P_i)^2}{\sum_{i=1}^{N}O_i^2} \tag{14}$$

where:

N – number of data points,
O_i – some measured value,
P_i – corresponding model prediction,
O_m, P_m – the average values of O_i and P_i respectively.

3.3. Multiple Linear Regression (MLR)

To understand how MLR works, assume we have n pairs of observation data set $\{x_i, y_i\}_{i=1,\cdots,n}$ as shown in Figures 2 and 3.

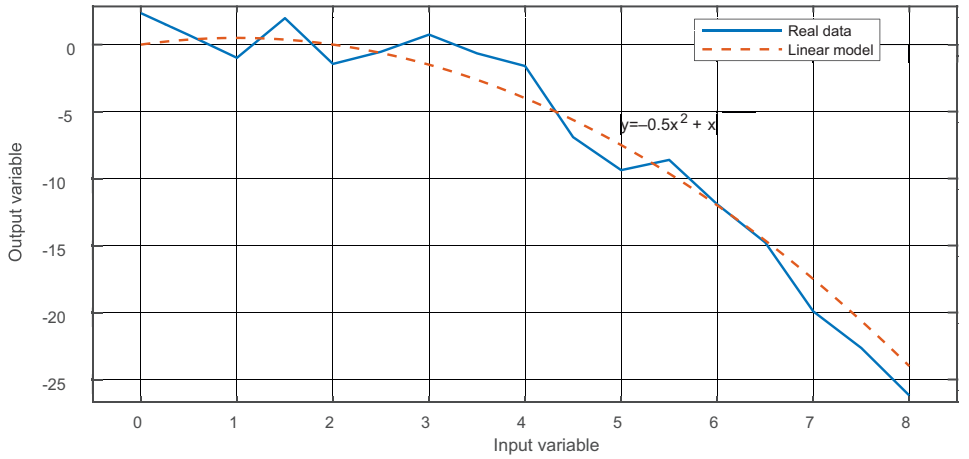


Fig. 2. Example of a linear model with $y = -0.5x^2 + x$

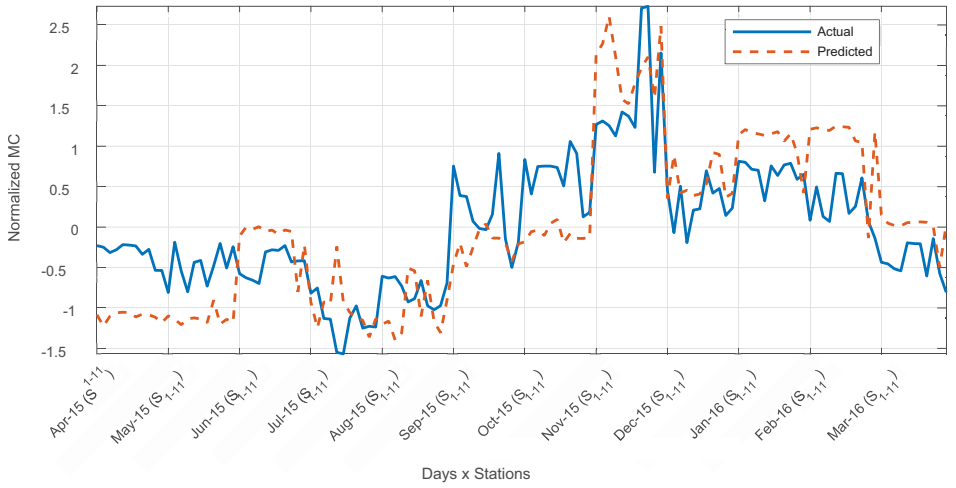


Fig. 3. *Microcystis* linear model modelled as

$$Mictis_{predicted}(t) = 0.58265 \cdot T + 0.3325 \cdot Condu - 0.37068 \cdot Trans + 0.093884 \cdot NH_4 + 0.19691 \cdot NO_3 + 0.012184 \cdot PO_4$$

The main modelling objective was to develop a simple relationship between the two variables x (i.e. input variable) and y (i.e output variable) so that we could develop a linear Equation (15):

$$y = a + bx \tag{15}$$

where a is a constant (i.e. bias) and b is the slope of the line.

Generally, the straight line will never pass by all the points in the graph. Thus, Equation (15) should be rewritten as follows:

$$y = a + bx + \epsilon \tag{16}$$

where ϵ represents the error difference between the values of x_i and y_i at any sample i. Thus, to formulate the most accurate line to approximate x and y, we have to formulate the problem as an optimization problem such that we can search and find the best values of the parameters (i.e. $\hat{a}$ and $\hat{b}$). In this case, we need to minimize the sum of the error over the whole data set. The simple linear model (16) can be expanded to a multivariate system of equations as follows:

$$y = b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_i x_i \tag{17}$$

where x_j is the j^{th} independent variable. In this case, we need to use LS estimation to compute the optimal values for the parameters b_0, b_1, b_2, ⋯, b_j. Equation (17) will be

used as the form of Equation (7). Thus, we have to minimize the optimization function *L*, which in this case can be presented as:

$$L = \sum_{i=1}^{N} \epsilon_i^2 = \sum_{i=1}^{N} \left(y_i - \hat{b}_0 - \hat{b}_1 x_{i1} - \hat{b}_2 x_{i2} - \ldots - \hat{b}_j x_{ij} \right) \tag{18}$$

To obtain the optimal values of the parameters $\hat{b}_0, \ldots, \hat{b}_j$, we have to compute the differentiation for the functions:

$$\frac{\partial L}{\partial \hat{b}_0} = \frac{\partial L}{\partial \hat{b}_1} = \frac{\partial L}{\partial \hat{b}_2} = \ldots = \frac{\partial L}{\partial \hat{b}_j} = 0 \tag{19}$$

By solving the set of Equations (19), we can produce the optimal values of the model parameters and solve the multiple regression problem. This solution is more likely to be biased by the available measurements. If there is a large number of observations, the computed estimate of the parameters would be more robust [31]. This technique provides poor results when the number of observations is small.

Algorithm 1 shows the MLR algorithm used in this work. Inspired from [32, 33], this algorithm computes model coefficients based on orthogonal-triangular decomposition [34]. Matlab function regress [33] was used to produce model coefficients.

Algorithm 1: Regress

```
Result:
Model coefficients for the data set X;
[Q,R,E]=qr (X);
% Orthogonal-Triangular Decomposition (produces an "economy size"
decomposition in which E is a permutation vector, so that A (:,E)  = Q × R);
b = (Q' * y)/R;
% b is the coefficient of the MLR model;
```

3.4. Support Vector Machines (SVM)

SVM is a powerful supervised learning model for prediction and classification [31]. SVM was firstly introduced by Vladimir Vapnik and his co-workers at AT&T Bell Laboratory [35]. The main idea of SVM is to approximate the training data set with higher dimensional space using a nonlinear mapping function and then perform linear regression in higher dimensional space to separate the data [36]. Data mapping was done using a predetermined kernel function. Data separation was done by finding the optimal hyperplane (called support vector with the maximum margin from the separated classes). Figures 4 and 5 show an example of optimal hyperplane. Figure 4 shows different lines separating data but with small margins, Figure 5 shows the optimal line separating data sets with maximum margins.

The kernel trick avoids the explicit mapping, instead of learning a nonlinear function or decision boundary, this trick implies getting linear learning algorithms. For all x and x^0 in the input space X, certain functions $K(x, x^0)$ can be expressed as

an inner product in another space V. The function K: $X \times X \to \mathbb{R}$ is often referred to as a kernel or a kernel function. Figures 4 and 5 show an application of kernel transformation. In this example, the data set can be modelled as a nonlinear function $y(x) = x + 2x^2$ with single variable x, but if a variable substitution is done a $x_1 = x$ and $x_2 = x^2$ y becomes the linear function $y(x_1, x_2) = x_1 + 2x_2$.

Training for a SVM has two phases [37]: the first is to transform input data to a high-dimensional feature space using the Kernel function. The second is to solve a quadratic optimization problem to fit an optimal hyperplane.

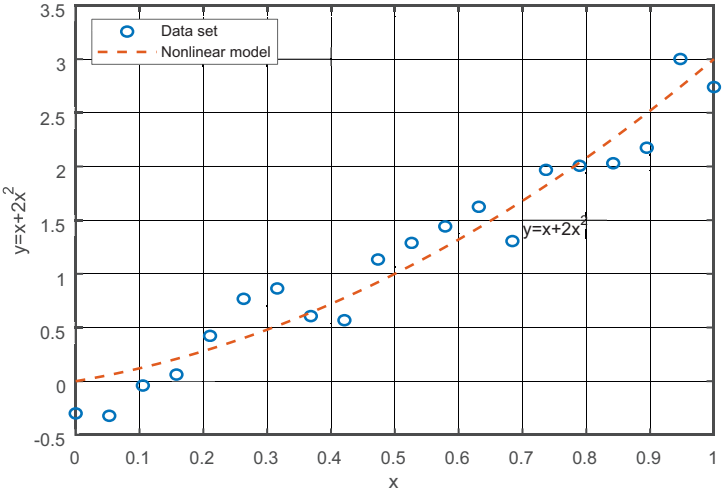


Fig. 4. Nonlinear model with a single variable

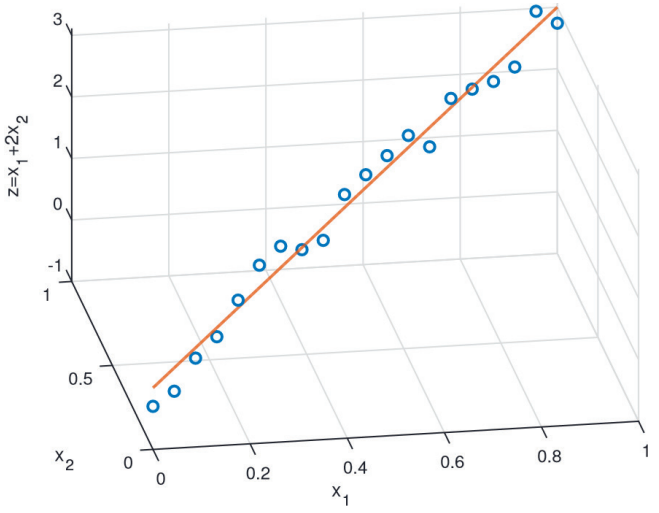


Fig. 5. Optimal line with two variables $K(x) = (x, x^2)$

Table 2 shows popular kernels used with SVM.

Table 2. New number of variables after combination

Type of SVM	Kernel function	Description
Gaussian	$K(x_1, x_2) = \exp\frac{\left\|x_1 - x_2\right\|^2}{2\sigma^2}$	σ is the width of the kernel
Linear	$K(x_1, x_2 = x_1^T x_2)$	–
Polynomial	$K(x_1, x_2) = (x_1^T x_2 + 1)^\rho$	ρ is the order of the polynomial
Sigmoid	$K(x_1, x_2) = \tanh(\beta_0 x_1^T x_2 + \beta_2)$	–

3.5. Particle Swarm Optimization (PSO)

PSO is an interesting evolutionary computation algorithm first introduced by Kennedy and Eberhart [38]. This algorithm is inspired by the swarm behaviour of organisms such as the flocking of birds and fish schools. This algorithm consists of a swarm of particles that search for the best position, including the best personal and global position, based on its best solution [31]. Equation (20) shows the moving process of each particle:

$$\begin{aligned} V_{new} &= w \cdot V + c_1 \cdot r_1 (p_{best} - X) + c_2 \cdot r_2 (g_{best} - X) \\ X_{new} &= X + V_{new} \end{aligned} \tag{20}$$

where:

c_1, c_2 – learning factors,
V, X – current particle velocity and position respectively,
V_{new}, X_{new} – new velocity and position of particles, respectively,
w – the inertial weight,
r_1, r_2 – the random numbers between 0 and 1.

Figure 6 shows particle inertia behaviour in function of its own and global inertia [39].

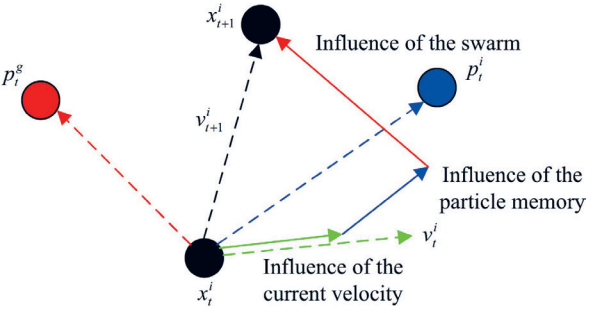


Fig. 6. Iteration scheme of the particles

Wang [39] summarized the PSO algorithm as follows: It is a swarm-based searching process, in which each individual, also known as a particle, is defined as a potential solution of the problem defined in D-dimensional search space, each particle can memorize the optimal position of the swarm and that of its own, as well as its velocity. In each generation, each particle adjusts its velocity according to the swarm information; the new velocity is used to compute the new position of the particle. According to [39], the PSO algorithm can be summarized as in Algorithm 2.

Algorithm 2: Particle Swarm Optimization algorithm

```
Result: Particle with best solution
Swarm initialization;
    while Ending condition is not satisfied do
        Particle fitness evaluating;
        Calculating the individual historical optimal position;
        Calculating the swarm historical optimal position;
        Updating particle velocity and position according to the velocity
        and position updating equation;
    end
```

3.6. Genetic Algorithms (GA)

Unlike the standard search techniques, GA search among a population of points, work with the coding of the parameter set and use probabilistic transition rules [21]. Initially, a population of m points is chosen randomly in the search space. The fitness function values are calculated at all points and compared.

Given a function $f = f(x_1, x_2, \cdots, x_n)$ subject to $a_i \le x_i \le b_i$, $i = 1, 2, \cdots, n$, the main objective is to find the set of parameters which leads to a minimum value of f. Genetic Algorithms work with the coding of the parameters. The most frequently employed parameter is binary coding [21]. An l-bit binary variable is used to represent one parameter x_i. The integer of the decoded binary variable ranges from 0 to $2^l - 1$ and can be mapped linearly to the parameter range (a_i, b_i). Connecting the coding of all parameters forms the coding for each point in the space to be searched, for example:

$$\begin{array}{ccccc} 1010010 & 0100111 & \ldots & 1001010 & 0000100 \\ x_1 & x_2 & \ldots & x_{n-1} & x_n \end{array} \tag{21}$$

It is important to note that the search range for each parameter must be specified. One of the most often used GA follows the following steps [22, 34]:

Step 1: Locate m points randomly in the search space.

Step 2: Find the fitness function value for each point.

Step 3: Rank the points so that their function values are in descending order.

Step 4: Assign a probability value p_j, using fitness proportionate selection or roulette wheel selection, to each point giving the higher probability to point of the lower (better) function value.

Step 5: Select two points A and B from the m points at random according to the probability distribution[8], p_j, $j = 1, 2, \cdots, m$.

Step 6: Select two-bit positions, k_1 and k_2, along with the overall coding of the parameter set at random (Fig. 7), giving each bit position the same chance.

Step 7: Form a new point by taking the values of the bits from k_1 to $k_2 - 1$ of the A point coding and values of the bits from k_2 to the end and from 1 to $k_j - 1$ of the B point coding (crossover).

Step 8: Randomly at a probability $p_{mutation}$, change some of the bits of the newly formed point (mutation).

Step 9: Repeat steps five and eight m^0 times (with $m^0 \leq m$) so that m^0 new points are produced. m^0 points are then replaced by the new ones keeping $m - m^0$ best points, forming a new generation for further search.

Step 10: Repeat steps 2–9 n times[9]. The point with best fitness value is recorded, and in step 2 if the newly generated m points are all inferior to the best point, the latter is re-inserted into the population by replacing one of the m points randomly.

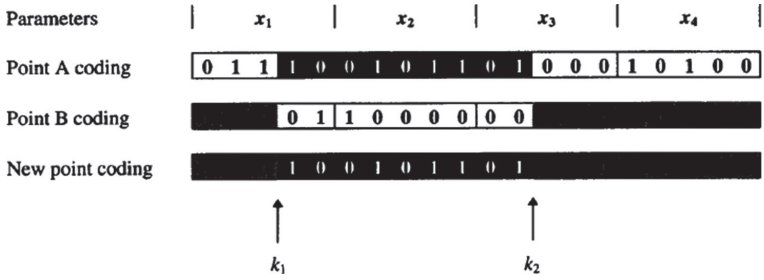


Fig. 7. A schematic diagram showing how a new point is generated from two existing points (steps 6 and 7)

3.7. Bird Swarm Algorithm (BSA)

BSA is a new swarm intelligent and global optimization algorithm inspired by the behaviour of the social iteration of birds in nature [22, 40]. BSA is based on three main behaviours of birds, namely foraging, vigilance, and flight. The algorithm can be summarized in the following five rules:

Rule 1: Each bird can be in one of two states: vigilance or foraging.

Rule 2: In the foraging status, each bird saves and keeps track of its own best experience and the best experience among the swarm about food positions. This information will help it look for food.

[8] Having assigned higher probabilities to better points in the last step, the better points have better chances to be selected.

[9] Variable n is the maximum number of generations. Having assigned higher probabilities to better points in the last step, the better points have better chances to be selected.

Rule 3: In the vigilance status, each bird tries competitively to move toward the centre.

Rule 4: Birds keep moving from one site to another and they iteratively keep switching between producing and scrounging. It is assumed that birds with the highest reserves are producers and the lowest are scroungers. While the birds with neither highest nor lowest reserve are randomly assumed to be producers or scroungers.

Rule 5: Producing birds lead the search for food while the scrounging ones randomly follow a producing bird.

4. Results

4.1. Data Collection

Training and validation were done using Matlab R2018b installed in a Windows 7 64-bit computer equipped with an Intel core i3-3210 CPU @ 3.20 GHz and 6 GB of RAM. The *Microcystis* density data set consists of 10 features and 12 months of data, covering a one-year period (April 2015 to March 2016). For the training data set, we sampled the data on a monthly basis for each of 11 stations so that only 132 samples were used in our experiment. For testing, we acquired data from 20 stations during four seasons, giving 80 samples.

The statistical parameters of *Mictis* and the ten water quality variables data that are the mean, maximum, minimum, standard deviation, coefficient of variation values, and the coefficient of correlation with *Mictis* (i.e., X_{mean}, X_{max}, X_{min}, *Sx*, *Cv*, and *CC* respectively) are given in Tables 3 and 4 for training and testing data sets.

Table 3. The statistical parameters of the variables for training

Variable	Description	Unit	X_{mean}	X_{max}	X_{min}	*Sx*	*Cv*	*CC*
T	water temperature	°C	20.393	31.400	8.450	7.385	54.538	0.402
pH	potential for hydrogen	–	8.888	11.680	7.870	0.858	0.737	0.080
O_2	dissolved oxygen	$mg \cdot L^{-1}$	8.352	12.050	3.980	2.041	4.164	−0.275
Condu	water conductivity	$\mu S \cdot cm^{-1}$	497.174	551.000	388.000	45.830	2100.405	0.232
Trans	water transparency	cm	12.803	25.000	5.000	6.124	37.503	−0.426
NH_4	ammonium	$\mu mol \cdot L^{-1}$	4.257	25.846	0.205	4.322	18.675	0.449
NO_2	nitrogen dioxide	$\mu mol \cdot L^{-1}$	1.259	4.725	0.156	0.670	0.449	0.323
NO_3	nitrate	$\mu mol \cdot L^{-1}$	4.623	11.493	1.888	1.455	2.118	0.451
PO_4^{3-}	ortho-phosphate	$\mu mol \cdot L^{-1}$	3.226	32.982	0.573	4.630	21.433	0.465
SS	suspended solids	$mg \cdot L^{-1}$	99.462	362.000	10.000	59.584	3550.220	0.189
Mictis	*Microcystis* density	$\cdot 10^3$ cell $\cdot mL^{-1}$	171,052	431,311	31,533	99,819	9963,940	1

Table 4. The statistical parameters of the variables for testing

Variable	Description	Unit	X_{mean}	X_{max}	X_{min}	*Sx*	*Cv*	*CC*
T	water temperature	°C	21.890	31.400	11.000	7.378	54.438	0.643
pH	potential for hydrogen	–	8.629	9.081	8.210	0.254	0.064	–0.029
O_2	dissolved oxygen	$mg \cdot L^{-1}$	7.893	11.480	4.300	1.969	3.878	–0.688
Condu	water conductivity	$\mu S \cdot cm^{-1}$	513.050	551.000	477.000	27.476	754.909	–0.395
Trans	water transparency	cm	11.688	20.000	5.000	5.213	27.180	–0.522
NH_4	ammonium	$\mu mol \cdot L^{-1}$	3.747	25.846	0.291	4.753	22.594	0.500
NO_2	nitrogen dioxide	$\mu mol \cdot L^{-1}$	1.161	4.725	0.168	0.644	0.415	0.612
NO_3	nitrate	$\mu mol \cdot L^{-1}$	4.439	11.493	1.601	2.178	4.745	0.521
PO_4	phosphate	$\mu mol \cdot L^{-1}$	4.371	32.982	0.692	6.741	45.443	0.539
SS	suspended solids	$mg \cdot L^{-1}$	115.050	362.000	15.000	68.549	4698.909	0.092
Mictis	*Microcystis* density	$\cdot 10^3$ cell $\cdot mL^{-1}$	197,252	431,311	9,418	123,588	15,274,000	1

All input and output variables were standardized according to the Z-score method, also known as autoscaling transformation. According to [41], the data was normalized such that the inputs and output had a mean of zero and a standard deviation of one. This guarantees that the measurement scales were removed. Z-score was calculated as follows:

$$x_{ni,k} = \frac{x_{i,k-m_k}}{S_{dk}} \tag{22}$$

where:

$x_{ni,k}$ – the normalized value of k^{th} variable (input or output) for each i^{th} sample,

$x_{i,k}$ – the original value,

m_k, S_{dk} – the mean and standard deviation of the variable k respectively. Normalization is an important process which significantly increases the performance of the models [19].

4.2. Input Extraction

After computing the MLR model of *Microcystis* concentration as output and each parameter as input, and computing *R* coefficient for training and testing data set (Fig. 8), we can see that some parameters change coefficient severely between training and testing. From the correlation analysis of ten water variables that were monitored, six variables were selected (*T*, *Condu*, *Trans*, NH_4, NO_3, and PO_4). This can be explained by the weak effect of the unselected parameters (pH, O_2, NO_2 and *SS*) on *Mictis*.

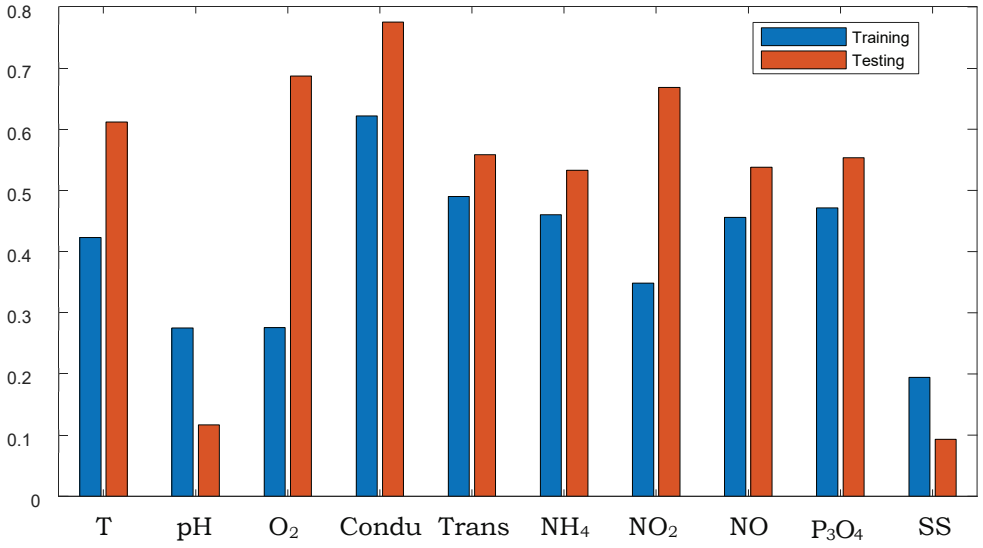


Fig. 8. Values of *R* for each parameter

To confirm the selected combination, Table 5 shows MLR results for different parameter combinations: all parameters, nutrients only[10], parameters with high correlation *CC*[11] and selected parameters[12]. RMSE was used as reference. It can be seen that the modelling with all parameters gave better RMSE for training, but the used parameters gave higher RMSE for training and better results for the testing data set.

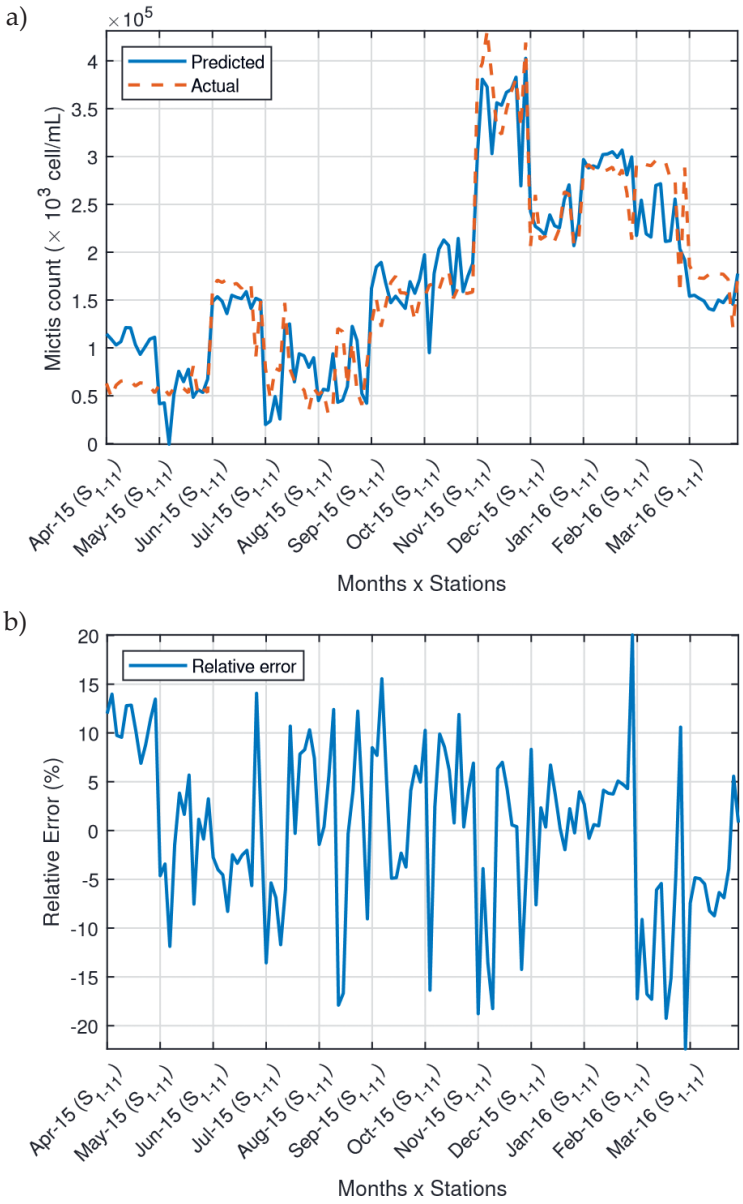
Table 5. Modelling results for different parameter combinations

Parameters	Case	RMSE	MAE	MSRE	NSE	*R*	*d*
All parameters	training	0.260	0.190	0.068	0.931	0.965	0.969
	testing	0.762	0.587	0.588	0.411	0.814	0.743
Nutrients only	training	0.749	0.604	0.565	0.434	0.659	0.658
	testing	0.638	0.551	0.412	0.587	0.768	0.744
Parameters with high correlation *CC*	training	0.559	0.445	0.315	0.684	0.827	0.852
	testing	0.549	0.424	0.305	0.694	0.859	0.882
Selected parameters	training	0.374	0.301	0.141	0.858	0.926	0.934
	testing	0.609	0.487	0.376	0.623	0.862	0.760

[10] NH_4, NO_2, NO_3, PO_4.
[11] *T*, *Trans*, NH_4, NO_3, PO_4.
[12] *T*, *Condu*, *Trans*, NH_4, NO_3, PO_4.

4.3. Modelling Results

Modelling using Multiple Regression Model, the coefficients b_{ij} were estimated using MLR to produce the optimal values $\hat{b}_{ij}$. The actual and estimated *Microcystis* density based on MLR in training and testing cases are shown in Figure 9. The scattered plots of the actual and predicted densities are shown in Figure 10. Equation (23) shows model computed using MLR.



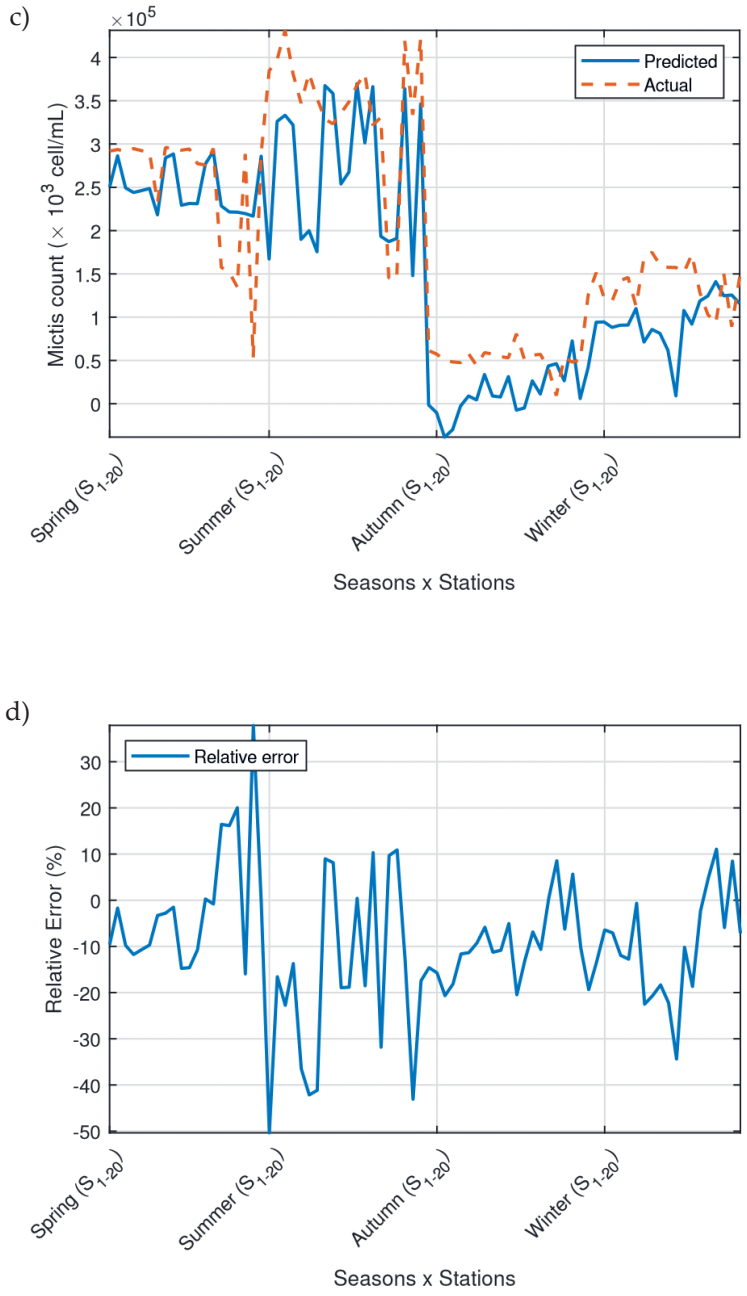


Fig. 9. Regression – actual and estimated *Microcystis* values in the training and testing cases:
a) actual and predicted *Microcystis* regression for 11 stations (monthly);
b) error difference – training case model – training case;
c) actual and predicted *Microcystis* regression for 20 stations (seasonally);
d) error difference – testing case model – testing case

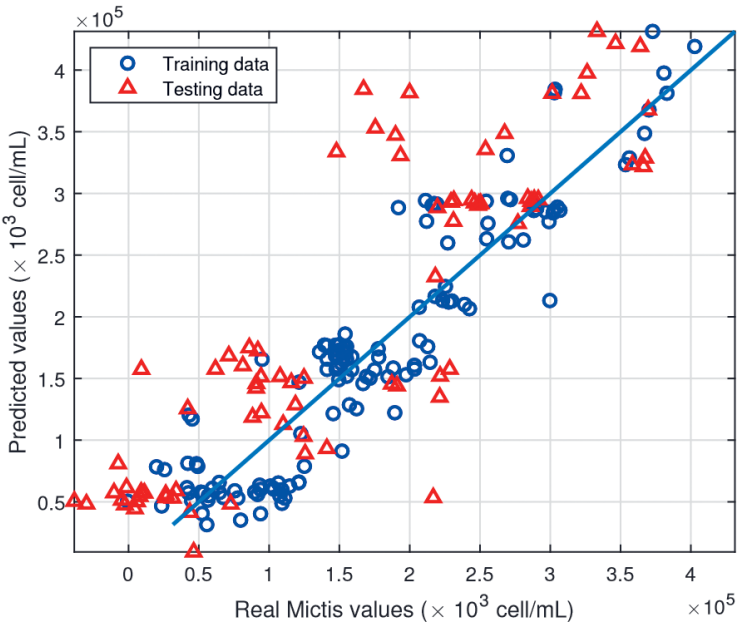


Fig. 10. Regression scattered plot

$$\begin{aligned} Mictis_{predicted} &= 0.6485 + 0.02598 \cdot T + 0.03416 \cdot T^2 + 1.051 \cdot T \cdot Condu - \\ &- 0.2741 \cdot T \cdot Trans - 0.01246 \cdot T \cdot NH_4 + 0.02714 \cdot T \cdot NO_3 + \\ &+ 0.102 \cdot T \cdot PO_4 + 0.1447 \cdot Condu - 0.152 \cdot Condu^2 - \\ &- 0.02303 \cdot Condu \cdot Trans + 0.0836 \cdot Condu \cdot NH_4 + \\ &+ 0.1384 \cdot Condu \cdot NO_3 - 0.1198 \cdot Condu \cdot PO_4 - 0.2118 \cdot Trans + \\ &+ 0.2036 \cdot Trans^2 + 0.06361 \cdot Trans \cdot NH_4 - 0.03032 \cdot Trans \cdot NO_3 + \\ &+ 0.001786 \cdot Trans \cdot PO_4 + 0.07373 \cdot NH_4 - 0.01882 \cdot NH_4^2 - \\ &- 0.1801 \cdot NH_4 \cdot NO_3 + 0.1765 \cdot NH_4 \cdot PO_4 - 0.01545 \cdot NO_3 + \\ &+ 0.001577 \cdot NO_3^2 + 0.08957 \cdot NO_3 \cdot PO_4 + 0.3644 \cdot PO_4 - 0.1699 \cdot PO_4^2 \end{aligned} \tag{23}$$

Modelling using SVM is based on SVM Matlab toolbox [35]. Table 6 shows SVM parameters, and Figure 11 – scattered plot of training and testing using SVM.

Table 6. SVM parameters

Parameter	Value
Kernel function	linear
Kernel scale (auto)	1.6992
Solver	sequential minimal optimization

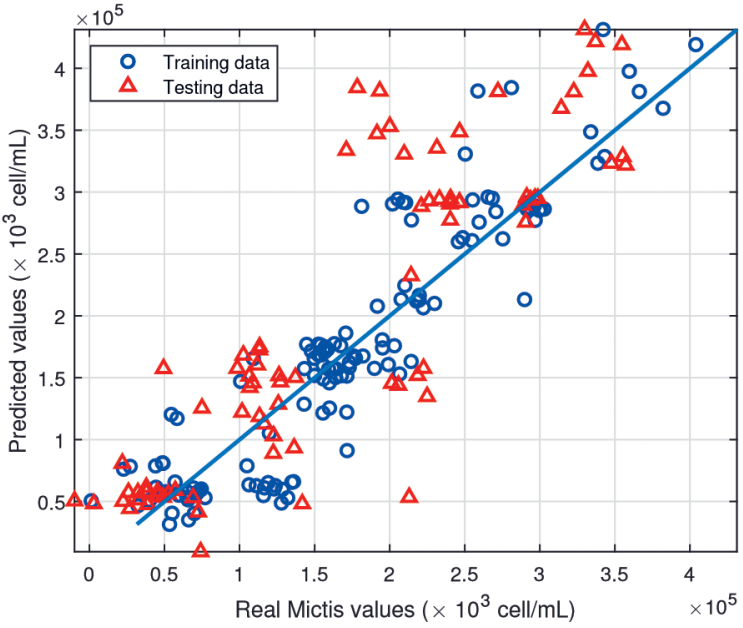


Fig. 11. SVM scattered plot

Modelling using PSO is based on PSO toolbox for MATLAB developed by Mostapha Kalami Heris [42]. After several tests[13], presented in Figure 12, PSO parameters are presented in Table 7, except for c_1 and c_2 that were inspired from the work of [43].

Table 7. PSO parameters

Parameter	Value
Maximum number of epochs	500
Number of particles	200
Lower bound of variables	−1
Upper bound of variables	1
Inertia weight	0.5
Inertia weight damping ratio *wdamp*	1
Personal learning (acceleration) coefficient c_1	1.49445
Global learning (acceleration) coefficient c_2	1.49445
Fitness function	$J_{\text{PSO}} = \text{RMSE}(Mictis_{real}, Mictis_{predicted})$

[13] The tests are done using 200 epochs with a population of 50 particles.

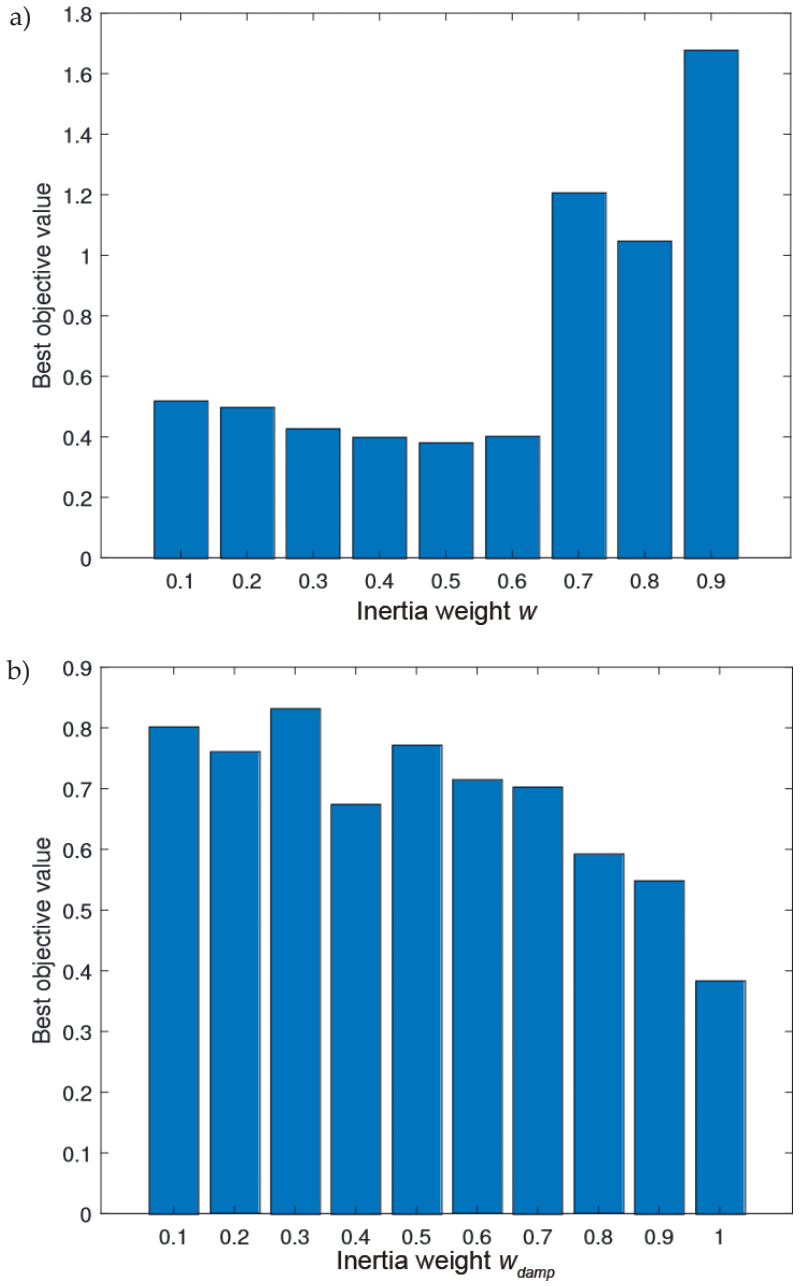


Fig. 12. PSO parameters: a) inertia weight w; b) inertia weight damping ratio w_{damp}

Figure 13 shows the scattered plot of training and testing using PSO. Equation (24) shows the optimal model.

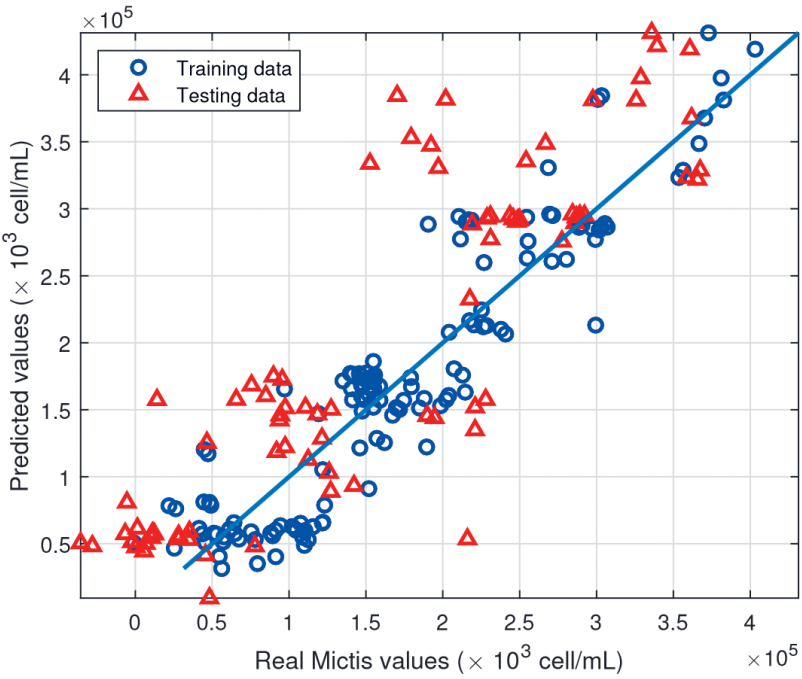


Fig. 13. PSO scattered plot

$$
\begin{aligned}
Mictis_{predicted} = {} & 0.6402 + 0.0343 \cdot T + 0.02064 \cdot T^2 + 1 \cdot T \cdot Condu - \\
& - 0.2758 \cdot T \cdot Trans - 0.009521 \cdot T \cdot \mathrm{NH_4} + 0.01783 \cdot T \cdot \mathrm{NO_3} + \\
& + 0.1176 \cdot T \cdot PO_4 + 0.1463 \cdot Condu - 0.1602 \cdot Condu^2 - \\
& - 0.01552 \cdot Condu \cdot Trans + 0.08692 \cdot Condu \cdot \mathrm{NH_4} + \\
& + 0.134 \cdot Condu \cdot \mathrm{NO_3} - 0.08251 \cdot Condu \cdot \mathrm{PO_4} - 0.2151 \cdot Trans + \\
& + 0.2096 \cdot Trans^2 + 0.06627 \cdot Trans \cdot \mathrm{NH_4} - 0.02803 \cdot Trans \cdot \mathrm{NO_3} + \\
& + 0.001645 \cdot Trans \cdot \mathrm{PO_4} + 0.7555 \cdot \mathrm{NH_4} - 0.01874 \cdot \mathrm{NH_4^2} - \\
& - 0.1817 \cdot \mathrm{NH_4} \cdot \mathrm{NO_3} + 0.1764 \cdot \mathrm{NH_4} \cdot \mathrm{PO_4} - 0.008773 \cdot \mathrm{NO_3} + \\
& + 0.0008725 \cdot \mathrm{NO_3^2} + 0.09401 \cdot \mathrm{NO_3} \cdot \mathrm{PO_4} + 0.3504 \cdot \mathrm{PO_4} - 0.172 \cdot \mathrm{PO_4^2}
\end{aligned}
\tag{24}
$$

After several tests[14], presented in Figure 14, GA parameters are shown in Table 8. GA MATLAB Toolbox [42] was used to compute model coefficients. Figure 15 shows scattered plot of training and testing using GA.

[14] The tests are done using 200 epochs with a population of 50 individuals.

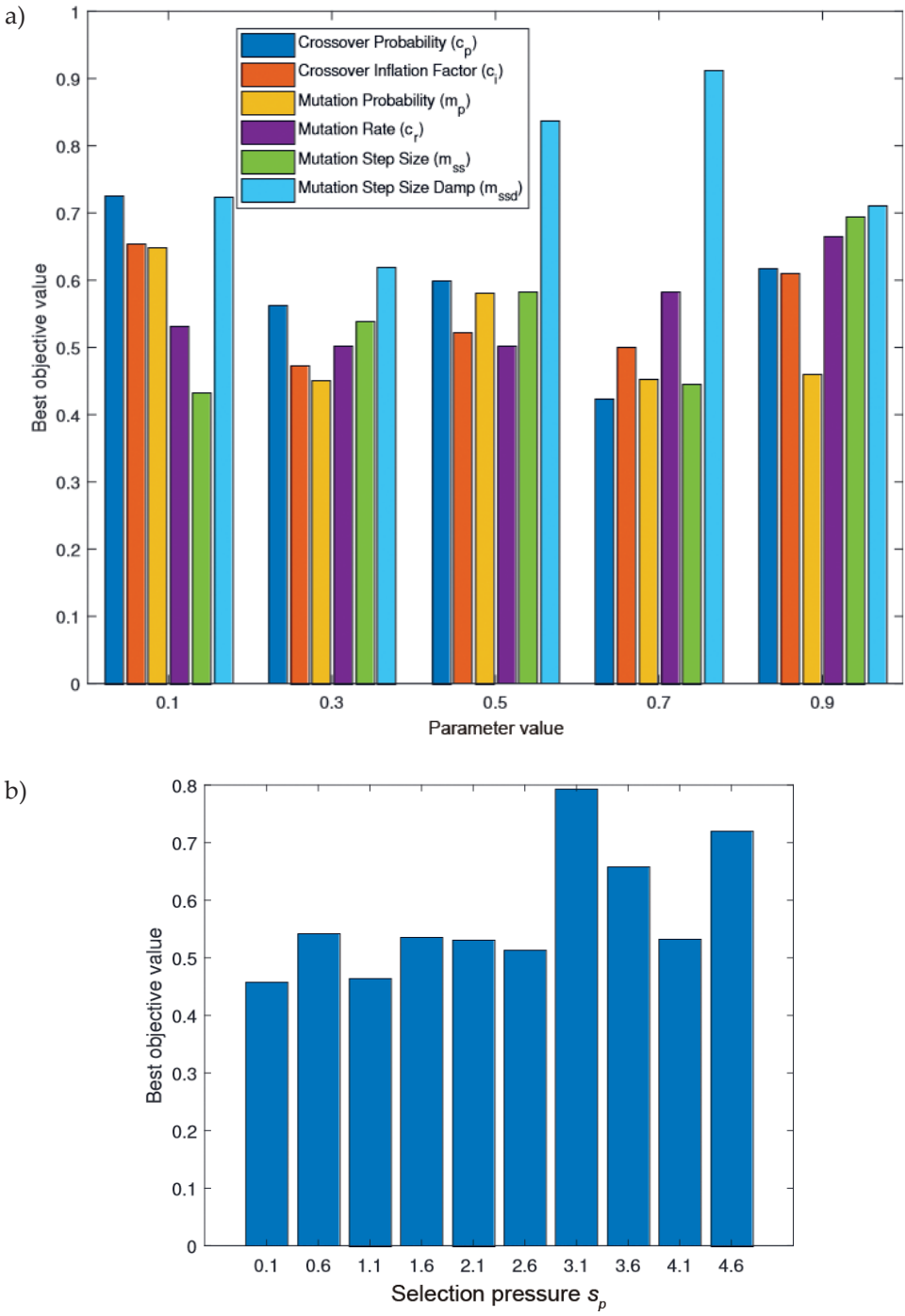


Fig. 14. GA parameters: a) best objective value; b) inertia weight damping ratio w_{damp}

Table 8. GA parameters

Parameter	Value
Maximum number of epochs	500
Number of particles	200
Lower bound of variables	−1
Upper bound of variables	1
Crossover probability (c_p)	0.7
Crossover inflation factor (c_i)	0.4
Mutation probability (m_p)	0.3
Mutation rate (m_r)	0.5
Mutation step size (m_{ss})	0.7
Mutation step size damp (m_{ssd})	0.3
Selection method	'roulette wheel'
Selection pressure (s_p)	1.1
Elitism?	yes (best from current and previous generation)
Fitness function	J_{GA} = RMSE($Mictis_{real}$, $Mictis_{predicted}$)

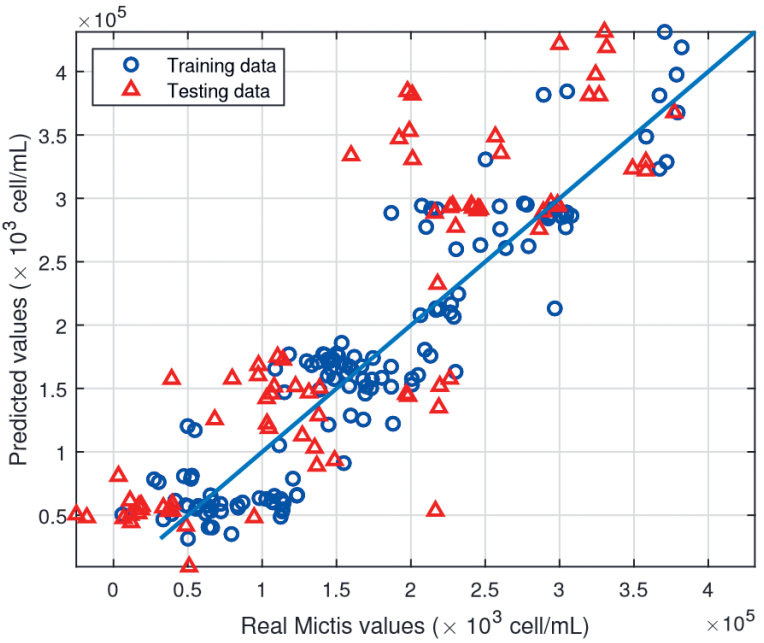


Fig. 15. GA scattered plot

$$\begin{aligned} Mictis_{predicted} = {} & 0.4889 + 0.6664 \cdot T + 0.6926 \cdot T^2 + 0.9947 \cdot T \cdot Condu - \\ & - 5363 \cdot T \cdot Trans + 0.6832 \cdot T \cdot NH_4 + 0.3657 \cdot T \cdot NO_3 - \\ & - 0.08945 \cdot T \cdot PO_4 + 0.9979 \cdot Condu + 0.1753 \cdot Condu^2 - \\ & - 0.1888 \cdot Condu \cdot Trans + 0.6646 \cdot Condu \cdot NH_4 + \\ & + 0.8178 \cdot Condu \cdot NO_3 - 0.2681 \cdot Condu \cdot PO_4 - 0.04688 \cdot Tans + \\ & + 0.2529 \cdot Trans^2 - 0.04139 \cdot Trans \cdot NH_4 + 0.1033 \cdot Trans \cdot NO_3 + \\ & + 0.6444 \cdot Trans \cdot PO_4 + 0.4344 \cdot NH_4 - 0.6137 \cdot NH_4^2 + \\ & + 0.787 \cdot NH_4 \cdot NO_3 - 0.09844 \cdot NH_4 \cdot PO_4 + 0.1674 \cdot NO_3 - \\ & - 0.04532 \cdot NO_3^2 - 0.7838 \cdot NO_3 \cdot PO_4 - 0.6969 \cdot PO_4 + 0.5697 \cdot PO_4^2 \end{aligned} \tag{25}$$

BSA MATLAB toolbox [22] was used to optimize *Mictis* model using parameter shown in Table 9. These parameters were deduced after some tests with different values[15] shown in Figure 16.

Table 9. BSA parameters

Parameter	Value
Maximum number of epochs	500
Number of particles	200
Lower bound of variables	−1
Upper bound of variables	1
The frequency of birds' flight behaviours	19
The probability of foraging for food	0.1
c_1	0.7
c_2	0.7
a_1	0.5
a_2	0.9
Fitness function	J_{BSA} = RMSE($Mictis_{real}$, $Mictis_{predicted}$)

Figure 17 shows scattered plot of training and testing using BSA. Equation (26) is the *Mictis* model computed using said toolbox and parameters.

[15] The tests are done using 200 epochs with a population of 50 birds.

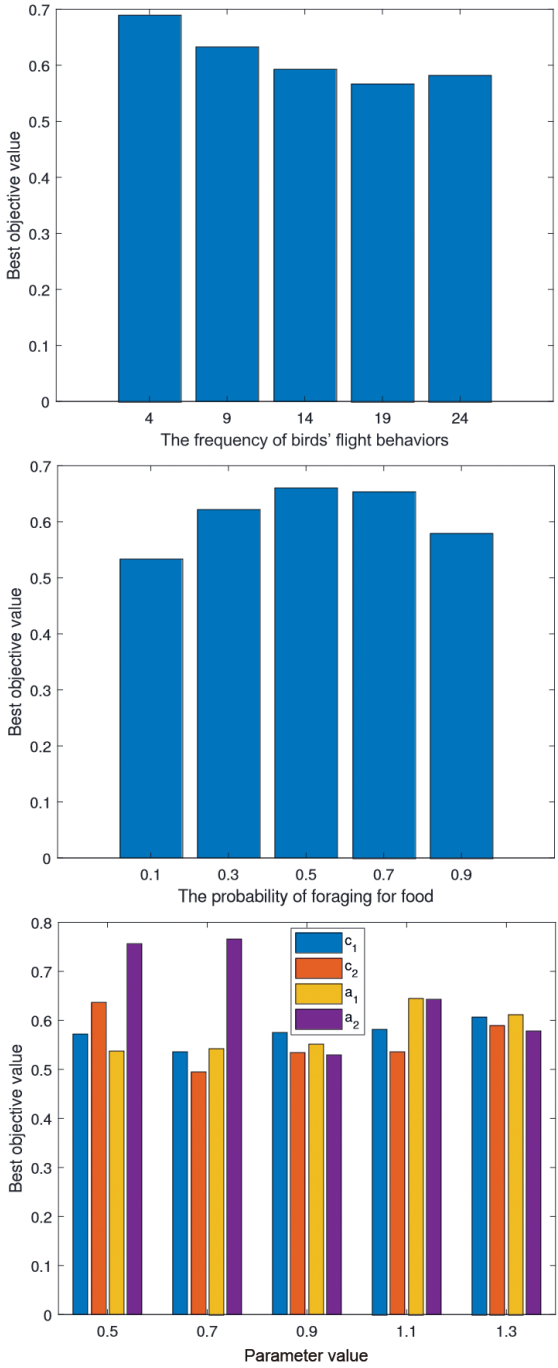


Fig. 16. BSA parameters: a) the frequency of birds' fight behaviors; b) the probability of foraging for food; c) learning parameters

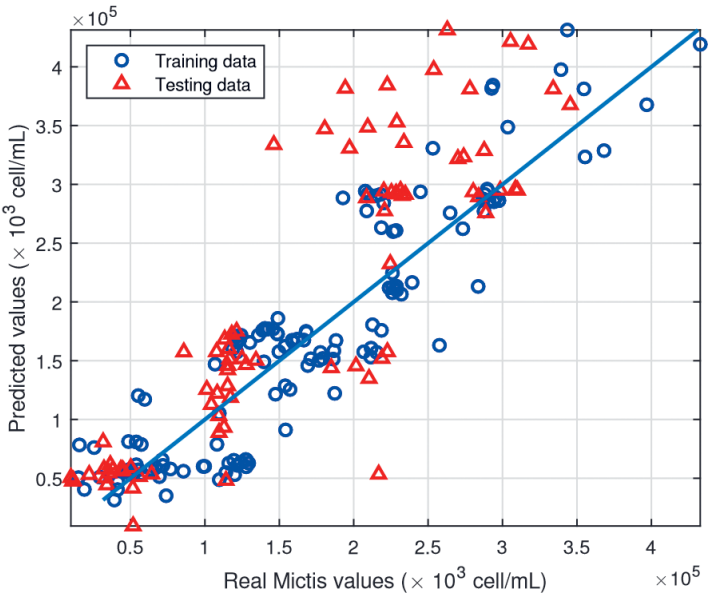


Fig. 17. BSA scattered plot

$$
\begin{aligned}
Mictis_{predicted} = {} & 0.3101 + 0.2782 \cdot T + 0.03631 \cdot T^2 + 0.8044 \cdot T \cdot Condu - \\
& - 0.1535 \cdot T \cdot Trans - 0.0009097 \cdot T \cdot NH_4 - 0.1232 \cdot T \cdot NO_3 + \\
& + 0.1407 \cdot T \cdot PO_4 + 0.4002 \cdot Condu - 0.07746 \cdot Condu^2 - \\
& - 0.04718 \cdot Condu \cdot Trans - 0.1585 \cdot Condu \cdot NH_4 - \\
& - 0.06338 \cdot Condu \cdot NO_3 + 0.2548 \cdot Condu \cdot PO_4 - 0.2279 \cdot Trans + \\
& + 0.1993 \cdot Trans^2 - 0.03408 \cdot Trans \cdot NH_4 - 0.1273 \cdot Trans \cdot NO_3 + \\
& + 0.03364 \cdot Trans \cdot PO_4 - 0.006117 \cdot NH_4 + 0.02171 \cdot NH_4^2 - \\
& - 0.1366 \cdot NH_4 \cdot NO_3 + 0.03142 \cdot NH_4 \cdot PO_4 + 0.2009 \cdot NO_3 - \\
& - 0.05032 \cdot NO_3^2 + 0.1182 \cdot NO_3 \cdot PO_4 + 0.045 \cdot PO_4 - 0.06235 \cdot PO_4^2
\end{aligned}
\tag{26}
$$

Results of the evaluation criterion for MLR, SVM, PSO, GA, and BSA models for training and testing are shown in Tables 10 and 11.

Table 10. Evaluation criteria for the developed models using computation-based techniques

Criteria	Regression		SVM	
	training	testing	training	testing
Root Mean Squared Error (RMSE)	0.374	0.609	0.395	0.561
Mean Absolute Error (MAE)	0.301	0.487	0.296	0.427

Table 10. cont.

Mean Squared Relative Error (MSRE)	0.141	0.376	0.158	0.319
Nash–Sutcliffe Efficiency (NSE)	0.858	0.623	0.842	0.680
Coefficient of correlation (R)	0.926	0.862	0.918	0.861
Willmott index of agreement (d)	0.934	0.760	0.921	0.776

Table 11. Evaluation criteria for the developed models using heuristic-based techniques

Criteria	PSO		GA		BSA	
	training	testing	training	testing	training	testing
Root Mean Squared Error (RMSE)	0.374	0.598	0.382	0.565	0.417	0.577
Mean Absolute Error (MAE)	0.301	0.478	0.305	0.447	0.337	0.429
Mean Squared Relative Error (MSRE)	0.141	0.363	0.147	0.324	0.175	0.337
Nash–Sutcliffe Efficiency (NSE)	0.858	0.637	0.852	0.675	0.824	0.662
Coefficient of correlation (R)	0.926	0.864	0.923	0.867	0.907	0.870
Willmott index of agreement (d)	0.934	0.766	0.931	0.780	0.914	0.721

Figure 18 shows training results for the different techniques, and they were all within real results.

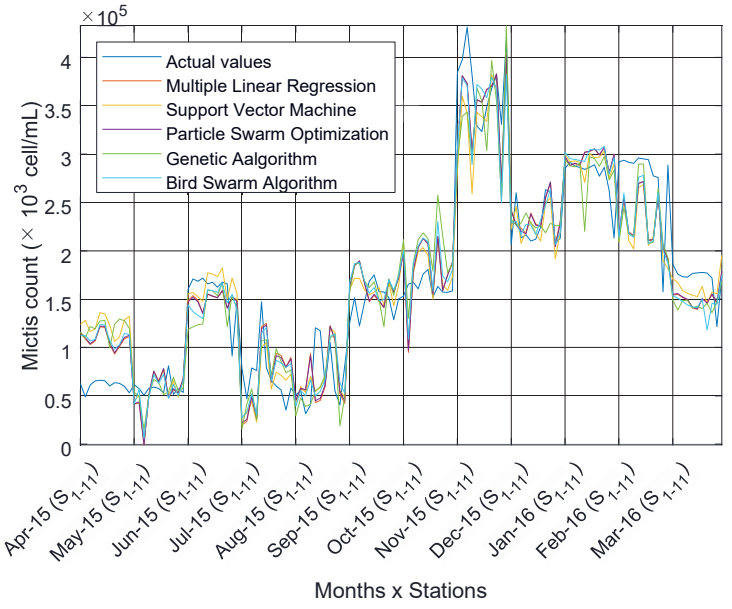


Fig. 18. Training results for different techniques

Figure 19 shows the performance indices of each technique for training and testing. RMSE which must be as low as possible, and MLR and PSO gave the best results (for training and testing). Letter *d* is the Willmott index of agreement which carries a value between 0 and 1 and a value of 1 means a perfect match [19], as before MLR and PSO were near 1 for training phase, but for testing, PSO gave better results, which means a better agreement with real values.

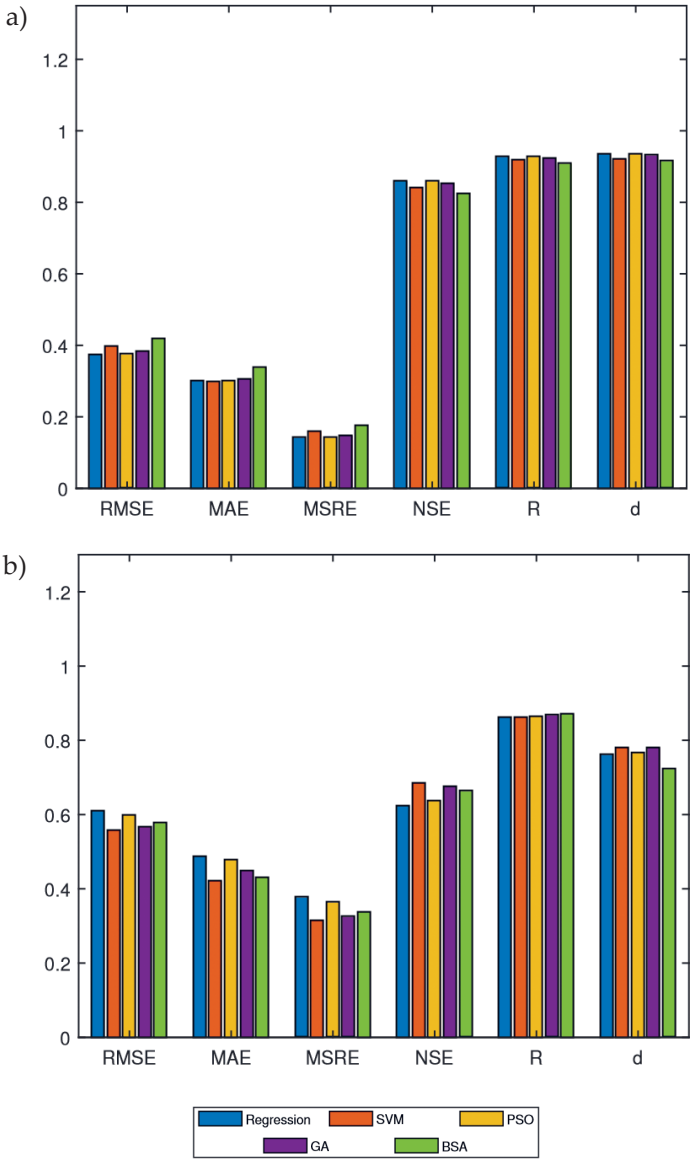


Fig. 19. Performance indices of the different techniques: a) training; b) testing

Figure 20 shows best cost in the function of iterations for the heuristic techniques, PSO gave the best fitness but all the techniques gave good results. Initially, BSA had the better fitness, but was very slow in terms of the function of generations. PSO was faster than the other techniques and had passed BSA by generation 20. GA passed BSA a little further on, at generation 50. PSO gave the best result than the other ones and in a faster time; this can make the difference if the data set were larger and training was meant to take several hours. Our results are clearly impacted by the time scale adopted since the scatterplots show a visible bias at high concentration values for the testing but not the training dataset. This is due to the time interval between samples. For the training phase, data is about one month for each of the two successive sampling periods, so they are quite close compared to the testing phase. For this latter, the interval between two successive sampling rounds is three months, which means that on the time scale of the two sampling rounds, the time between two successive data used for testing is three times longer (3 months) than that of two successive data used for training (1 month).

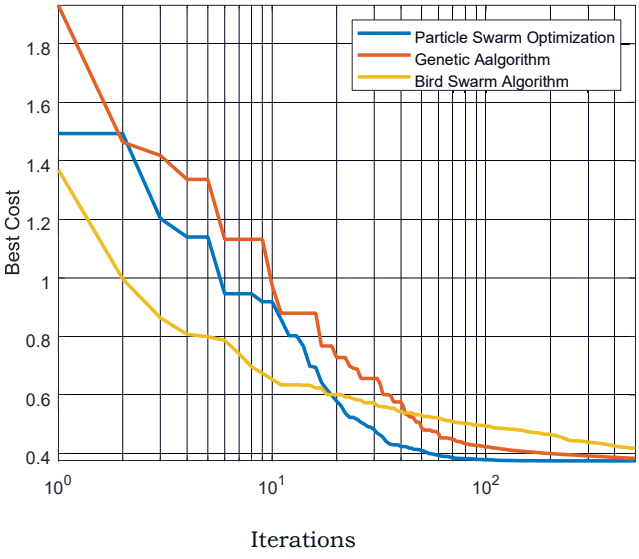


Fig. 20. Training best cost in function of the iterations

5. Discussion

As a first observation, it can be noted that the results of our study revealed that the applied approaches provide approximately the same level of precision. The PSO has revealed its overall potential for research through its performance in some optimization problems, particularly in function minimization. In PSO, a population of arbitrary solutions is initialized and optimizations are sought

by updating generations. Despite this fact, the PSO is not equipped with evolutionary operators such as crossover and mutation. The potential solutions in the PSO, also known as particles, progress through the problem field by tracing the existing optimal solutions [44, 45]. The fact that *Microcystis* has a natural collective behaviour of decentralized, self-organized system, therefore, and that PSO uses information sharing in the framework of social partnership [46]; this appears to be an excellent technique to apply for modelling and forecasting *Microcystis* densities. The PSO has several features in common with evolutionary calculation techniques [45].

SVM is a new technique taken from the statistical learning model, is designed to classify elements by assigning them to one of two separated (disjoint) half-spaces [47]. Therefore, SVM performs as a highly sophisticated learning machine to perform classifications, and temporal simulations [48]. It is an effective method for overcoming problems of low sampling, non-linearity and large dimension. However, the choice of the different parameters of the SVM has a significant impact on the reliability of the SVM classification. Nevertheless, it is very complicated to choose the most adequate SVM parameters [49] because of the complexity of the cyanobacteria bloom process [5]. This may not only explain the discrepancies observed between SVM and PSO but also between the applied techniques, which were certainly caused by several factors influencing *Microcystis* densities that are not included in current models. Bobbin and Recknagel [50] studied the application for rule-based modelling and concluded that GA provide rules for extracting and developing models using temporal water quality data and confirm their effectiveness in predicting and determining the time and severity of algal blooms. However, the GA results in our work may be justified assuming that other factors controlling the proliferation of *Microcystis* are not taken into account in the model developed in our work. Despite the existence of several algorithms to handle optimization applications, it is recognized that there is no universal algorithm.

Swarm intelligence algorithms often show premature convergence and risk hitting with local optima. As a result, research remains in progress to develop more efficient algorithms [22, 51]. For that, BSA as a new bio-inspired algorithm remains in need of improvement since it presents similar problems in some situations, even though it has demonstrated its superiority over PSO in the work of [22]. Altay and Alatas [51] have concluded from their work on the BSA that it presents a premature convergence and may stumble in local optima of specific problem types, based on the limited amount of work carried out on the BSA in order to eliminate its defects by increasing its performance. Linear regression is one of the classical statistical models used to establish the link between dependent and independent variables. It is applied to show the dependence of one variable on many independent ones [52]. MLR has dominated several areas of time series forecasting [53]. It revealed good results even it assumes that the dependent and independent variables are linearly related, and they are normally distributed.

6. Conclusions

We explored the use of multiple techniques, both statistical and evolutionary, to approximate models for the prediction of *Microcystis* density. From the correlation analysis of the ten water variables that were monitored, six potential parameters were selected (*T*, NH_4, NO_3, PO_4, conductivity *Cond*, and transparency *Trans*), which were then combined to generate 28 variables. The combination of variables was useful and successful in increasing the number of factors needed for modelling. The performance indices showed that MLR and PSO provided the best results among all applied techniques. PSO gave the best fitness, but all techniques performed well. BSA had better fitness, but was very slow across generations. PSO was faster than the other techniques and at generation 20 it passed BSA. GA passed BSA a little further, at generation 50. The major contributions of our work not only focus on the modelling process itself, but also take into full consideration the main factors affecting *Microcystis* blooms by incorporating them in all applied models.

Acknowledgements

We are grateful to the staff of Lake Oubeira for the excellent assistance provided during the study period.

We are very grateful to the Directorate General of Scientific Research and Technological Development, Algeria, for funding this research, under the grant (D00L03UN230120150002).

Author Contributions

Author 1: conceptualization, methodology, data curation, formal analysis, writing – original draft preparation, writing – review & editing.

Author 2: formal analysis, writing – original draft preparation, writing – review & editing.

Author 3: writing – original draft preparation, writing – review & editing, methodology, investigation.

Author 4: software, formal analysis, writing – review & editing.

Author 5: methodology, investigation, writing – review & editing.

Author 6: methodology, investigation, writing – review & editing.

Author 7: conceptualization, validation, formal analysis, resources, writing – review & editing, supervision, project administration.

References

[1] Al-Sammak M.A., Hoagland K.D., Snow D.D., Cassada D.: *Methods for simultaneous detection of the cyanotoxins BMAA, DABA, and anatoxin-a in environmental samples*. Toxicon, vol. 76, 2013, pp. 316–325. https://doi.org/10.1016/j.toxicon.2013.10.015.

[2] Merel S., Walker D., Chicana R., Snyder S., Baures E., Thomas O.: *State of knowledge and concerns on cyanobacterial blooms and cyanotoxins*. Environment International, vol. 59, 2013, pp. 303–327. https://doi.org/10.1016/j.envint.2013.06.013.

[3] Nieto P.J.G., Fernández J.R.A., Suárez V.M.G., Muñiz C.D., Gonzalo E.G., Bayón R.M.: *A hybrid PSO optimized SVM-based method for predicting of the cyanotoxin content from experimental cyanobacteria concentrations in the Trasona reservoir: A case study in Northern Spain*. Applied Mathematics and Computation, vol. 260, 2015, pp. 170–187. https://doi.org/10.1016/j.amc.2015.03.075.

[4] Paerl H.W., Otten T.G.: *Harmful cyanobacterial blooms: causes, consequences, and controls*. Microbial Ecology, vol. 65(4), 2013, pp. 995–1010. https://doi.org/10.1007/s00248-012-0159-y.

[5] Bai X.Z., Zhang H.Y., Wang X.Y., Wang L., Xu J.P., Yu J.N.: *The adaptive-clustering and error-correction method for forecasting cyanobacteria blooms in lakes and reservoirs*. Advances in Mathematical Physics, vol. 7, 2017, 9037358. https://doi.org/10.1155/2017/9037358.

[6] Qin B.Q., Yang G.J., Ma J.R., Deng J.M., Li W., Wu T.F., Liu L.Z., Gao G., Zhu G.G.W., Zhang Y.L.: *Dynamics of variability and mechanism of harmful cyanobacteria bloom in Lake Taihu, China*. Chinese Science Bulletin, vol. 61(7), 2016, pp. 759–770. https://doi.org/10.1360/N972015-00400.

[7] Lehman P.W., Kurobe T., Lesmeister S., Baxa D., Tung A., Teh S.J.: *Impacts of the 2014 severe drought on the Microcystis bloom in San Francisco Estuary*. Harmful Algae, vol. 63, 2017, pp. 94–108. https://doi.org/10.1016/j.hal.2017.01.011.

[8] Levy S.: *Microcystis rising: why phosphorus reduction isn't enough to stop cyano-HABs*. Environmental Health Perspectives, vol. 125(2), 2017, pp. A34–A39. https://doi.org/10.1289/ehp.125-A34.

[9] Zhu W., Zhou X., Chen H., Gao L., Xiao M., Li M.: *High nutrient concentration and temperature alleviated formation of large colonies of Microcystis: evidence from field investigations and laboratory experiments.* Water Research, vol. 101, 2016, pp. 167–175. https://doi.org/10.1016/j.watres.2016.05.080.

[10] Cook C.M., Vardaka E., Lanaras T.: *Toxic cyanobacteria in Greek freshwaters, 1987 2000: occurrence, toxicity, and impacts in the mediterranean region*. Acta Hydrochimica et Hydrobiologica, vol. 32(2), 2004, pp. 107–124. https://doi.org/10.1002/aheh.200300523.

[11] Mariani M.A., Padedda B.M., Kashirtovsky J., Buscarinu P., Sechi N., Virdis T., Luglie A.: *Effects of trophic status on microcystin production and the dominance of cyanobacteria in the phytoplankton assemblage of Mediterranean reservoirs*. Scientific Reports, vol. 5(1), 2015, 17964. https://doi.org/10.1038/srep17964.

[12] Saoudi A., Barour C., Brient L., Ouzrout R., Bensouilah M.: *Environmental parameters and spatio-temporal dynamics of cyanobacteria in the reservoir of Mexa (Extreme North-East of Algeria).* Advances in Environmental Biology, vol. 9(11), 2015, pp. 109–121.

[13] Bouhaddada R., Nélieu S., Nasri H., Delarue G., Bouaïcha N.: *High diversity of microcystins in a Microcystis bloom from an Algerian lake*. Environmental Pollution, vol. 216, 2016, pp. 836–844. https://doi.org/10.1016/j.envpol.2016.06.055.

[14] Bidi-Akli S., Hacene H., Arab A.: *Impact of abiotic factors on the spatio-temporal distribution of cyanobacteria in the Zeralda's dam (Algeria)*. Revue d'Écologie, vol. 72(2), 2017, pp. 159–167.

[15] Guellati F.Z., Touati H., Tambosco K., Quiblier C., Humbert J.-F., Bensouilah M.: *Unusual cohabitation and competition between Planktothrix rubescens and Microcystis sp. (cyanobacteria) in a subtropical reservoir (Hammam Debagh) located in Algeria*. PloS One, vol. 12(8), 2017, e0183540. https://doi.org/10.1371/journal.pone.0183540.

[16] Touati H., Guellati F.Z., Arif S., Bensouilah M.: *Cyanobacteria dynamics in a Mediterranean reservoir of the north east of Algeria: vertical and seasonal variability*. Journal of Ecological Engineering, vol. 20(1), 2019, pp. 93–107. https://doi.org/10.12911/22998993/94606.

[17] Lou I., Xie Z., Ung W.K., Mok K.M.: *Freshwater algal bloom prediction by extreme learning machine in Macau Storage Reservoirs*. [in:] Sun F., Toh K.-A., Romay M.G., Mao K. (eds.), *Extreme Learning Machines 2013: Algorithms and Applications*, Adaptation, Learning, and Optimization, vol. 16, Springer, Cham 2014, pp. 95–111. https://doi.org/10.1007/978-3-319-04741-6_8.

[18] Belourghi B., Houichi L., Heddam S.: *Réseaux de Neurones Arti ciels pour la Modelisation du Dosage du Coagulant dans les Stations de Traitements des Eaux de Surface a Faible Turbidite*. Conference paper at: ATGRSR 2012. II. Séminaire International Euro-Meditérraneen Aménagement du Térritoire, Gestion des Risques et Sécurité Routière Batna, Algerie, 2012.

[19] Heddam S.: *Multilayer perceptron neural network-based approach for modeling phycocyanin pigment concentrations: case study from lower Charles River buoy, USA*. Environmental Science and Pollution Research, vol. 23, 2016, pp. 17210–17225. https://doi.org/10.1007/s11356-016-6905-9.

[20] Hasanipanah M., Naderi R., Kashir J., Noorani S.A., Qaleh A.Z.A.: *Prediction of blast-produced ground vibration using particle swarm optimization*. Engineering with Computers, vol. 33(2), 2017, pp. 173–179. https://doi.org/10.1007/s00366-016-0462-1.

[21] Wang Q.J.: *Using genetic algorithms to optimise model parameters*. Environmental Modelling & Software, vol. 12(1) 1997, pp. 27–34. https://doi.org/10.1016/S1364-8152(96)00030-8.

[22] Meng X.-B., Gao X.Z., Lu L., Liu Y., Zhang H.: *A new bio-inspired optimisation algorithm: Bird Swarm Algorithm*. Journal of Experimental & Theoretical Artificial Intelligence, vol. 28(4), 2016, pp. 673–687. https://doi.org/10.1080/0952813X.2015.1042530.

[23] Daghighi A.: *Harmful Algae Bloom Prediction Model for Western Lake Erie Using Stepwise Multiple Regression and Genetic Programming*. Cleveland State University, Cleveland 2017 [M.Sc. thesis].

[24] Nasri H., El Herry S., Bouaicha N.: *First reported case of turtle deaths during a toxic Microcystis spp. bloom in Lake Oubeira, Algeria*. Ecotoxicology and Environmental Safety, vol. 71(2), 2008, pp. 535–544. https://doi.org/10.1016/j.ecoenv.2007.12.009.

[25] Amrani A., Nasri H., Azzouz A., Kadi Y., Bouaicha N.: *Variation in cyanobacterial hepatotoxin (microcystin) content of water samples and two species of fishes collected from a shallow lake in Algeria*. Archives of Environmental Contamination and Toxicology, vol. 66(3), 2014, pp. 379–389. https://doi.org/10.1007/s00244-013-9993-2.

[26] Boussadia M.I., Sehli N., Bousbia A., Ouzrout R., Bensouilah M.: *The effect of environmental factors on Cyanobacteria abundance in Oubeira lake (Northeast Algeria)*. Research Journal of Fisheries and Hydrobiology, vol. 107(1), 1985, pp. 33–35. https://doi.org/10.1007/s00244-013-9993-2.

[27] Aminot A., Kérouel R.: *Hydrologie des écosystèmes marins: paramètres et analyses.* Infremer, Brest 2004.

[28] Komárek J., Kaštovský J., Mareš J., Johansen J.R.: *Taxonomic classification of cyanoprokaryotes (cyanobacterial genera) 2014, using a polyphasic approach*. Preslia, vol. 86(4), 2014, pp. 295–335.

[29] Komárek J.: *A polyphasic approach for the taxonomy of cyanobacteria: principles and applications*. European Journal of Phycology, vol. 51(3), 2016, pp. 346–353. https://doi.org/10.1080/09670262.2016.1163738.

[30] Luc B., Lengronne M., Bertrand E., Rolland D., Sipel A., Steinmann D., Baudin I. et al.: *A phycocyanin probe as a tool for monitoring cyanobacteria in freshwater bodies*. Journal of Environmental Monitoring, vol. 10(2), 2008, pp. 248–255. https://doi.org/10.1039/b714238b.

[31] Sheta A.F., Ahmed S.E.M., Faris H.: *A comparison between regression, artificial neural networks and support vector machines for predicting stock market index.* International Journal of Advanced Research in Artificial Intelligence, vol. 4(7), 2015, pp. 55–63. https://doi.org/10.14569/IJARAI.2015.040710.

[32] Chatterjee S., Hadi A.S.: *Influential observations, high leverage points, and outliers in linear regression*. Statistical Science, vol. 1(3), 1986, pp. 379–393. https://doi.org/10.1214/ss/1177013622.

[33] *Regress* (*Multiple linear regression*). MathWorks. https://www.mathworks.com/help/stats/regress.html [access: 7.06.2020].

[34] Khanmohammadi M., Azqhandi M.A.: *Introducing an orthogonal-triangular decomposition algorithm and its application in multivariate calibration*. Analytical Methods, vol. 3(12), 2011, pp. 2721–2725.

[35] *Support Vector Machine (SVM)*. MathWorks. https://uk.mathworks.com/discovery/support-vector-machine.html [access: 12.06.2020].

[36] Vapnik V.N.: *The Nature of Statistical Learning Theory*. Information Science and Statistics, Springer, New York 1995. https://doi.org/10.1007/978-1-4757-3264-1.

[37] Bacue R.J.: *An analytic overview of Estes' statistical learning theory*. Ohio University, 1999 [Ph.D. thesis].

[38] Noori R., Abdoli M., Ghasrodashti A.A., Ghazizade M.J.: *Prediction of municipal solid waste generation with combination of support vector machine and principal component analysis: A case study of Mashhad*. Environmental Progress & Sustainable Energy, vol. 28(2), 2009, pp. 249–258. https://doi.org/10.1002/ep.10317.

[39] Wang D., Tan D., Liu L.: *Particle swarm optimization algorithm: an overview*. Soft Computing, vol. 22(2), 2018, pp. 387–408. https://doi.org/10.1007/s00500-016-2474-6.

[40] Aljarah I., Faris H., Al-Madi N., Sheta A., Mafarja M.: *Evolving neural networks using bird swarm algorithm for data classification and regression applications*. Journal of Cluster Computing, vol. 22(3), 2019, pp. 1317–1345. https://doi.org/10.1007/s10586-019-02913-5.

[41] Kingston G.B., Maier H.R., Lambert M.F.: *Calibration and validation of neural networks to ensure physically plausible hydrological modelling*. Journal of Hydrology, vol. 314(1–4), 2006, pp. 158–176. https://doi.org/10.1016/j.jhydrol.2005.03.013

[42] Heris S.M.K.: *YPEA: Yarpiz Evolutionary Algorithms*. Yarpiz, 2019. https://yarpiz.com/477/ypea-yarpiz-evolutionary-algorithms [access: 16.06.2020].

[43] Clerc M., Kennedy J.: *The particle swarm-explosion, stability and convergence in a multi dimensional complex space*. IEEE Transactions on Evolutionary Computation, vol. 6(1), 2002, pp. 58–73. https://doi.org/10.1109/4235.985692.

[44] Lin S.-W., Ying K.-C., Chen S.-C., Lee Z.-J.: *Particle swarm optimization for parameter determination and feature selection of support vector machines*. Expert Systems with Applications, vol. 35(4), 2008, pp. 1817–1824. https://doi.org/10.1016/j.eswa.2007.08.088.

[45] Lou I., Xie Z., Ung W.K., Mok K.M.: *Integrating support vector regression with particle swarm optimization for numerical modeling for algal blooms of freshwater*. [in:] Lou I., Han B., Zhang W. (eds.), *Advances in Monitoring and Modelling Algal Blooms in Freshwater Reservoirs: General Principles and a Case study of Macau*, Springer, Dordrecht 2017, pp. 125–141. https://doi.org/10.1007/978-94-024-0933-8_8.

[46] Olsson A.E. (ed.): *Particle Swarm Optimization: Theory, Techniques and Applications*. Nova Science Publishers, Hauppauge, New York 2010.

[47] Alba E., Garcia-Nieto J., Jourdan L., Talbi E.G.: *Gene selection in cancer classification using PSO/SVM and GA/SVM hybrid algorithms*. [in:] *CEC 2007: 2007 IEEE Congress on Evolutionary Computation: 25–28 September, 2007 Singapore*, IEEE, Piscataway 2007, pp. 284–290. https://doi.org/10.1109/CEC.2007.4424483.

[48] Shen J., Qin Q., Wang Y., Sisson M.: *A data-driven modeling approach for simulating algal blooms in the tidal freshwater of James River in response to riverine nutrient loading*. Ecological Modelling, vol. 398, 2019, pp. 44–54. https://doi.org/10.1016/j.ecolmodel.2019.02.005.

[49] Li H., Zhang Y.: *An algorithm of soft fault diagnosis for analog circuit based on the optimized SVM by GA*. [in:] *2009 9th International Conference on Electronic Measurement & Instruments, Beijing 2009*, IEEE, Piscataway, pp. 4-1023–4-1027. https://doi.org/10.1109/ICEMI.2009.5274151.

[50] Bobbin J., Recknagel F.: *Mining water quality time series for predictive rules of algal blooms by genetic algorithms*. [in:] Oxley L. (ed.), *MODSIM 1999 International Congress on Modelling and Simulation*, Modelling and Simulation Society of Australia and New Zealand, Perth 1999, pp. 691–696.

[51] Altay E.V., Alatas B.: *Bird swarm algorithms with chaotic mapping*. Artificial Intelligence Review, vol. 53(2), 2020, pp. 1373–1414. https://doi.org/10.1007/s10462-019-09704-9.

[52] Brown S.H.: *Multiple linear regression analysis: a matrix approach with MATLAB*. Alabama Journal of Mathematics, vol. 34 (Spring/Fall), 2009, pp. 1–3.

[53] Rajaee T., Boroumand A.: *Forecasting of chlorophyll-a concentrations in South San Francisco Bay using five different models*. Expert Systems with Applications, vol. 35(4), 2008, pp. 1817–1824. https://doi.org/10.1016/j.apor.2015.09.001.