COMPUTER SCIENCE | RESEARCH ARTICLE

Development of modified cooperative particle swarm optimization with inertia weight for feature selection

G. Samigulina1 and Zh. Massimkanova2*

Abstract: The article presents a modified Cooperative Particle Swarm Optimization with Inertia Weight (CPSOIW) for Smart-technology of forecasting and control of complex objects. The software “CPSOIW (Cooperative Particle Swarm Optimization with Inertia Weight)” based on a modified CPSOIW algorithm has been developed in Python programming language and is used to process a multidimensional data and to create an optimal set of descriptors. The proposed algorithm combines the advantages of inertia weight particle swarm optimization (IWPSO) algorithm and cooperative particle swarm optimization (CPSO) algorithm. IWPSO algorithm allows to avoid an early convergence and to prevent particles from trapping into local optima due to update an inertia weight at each iteration. CPSO algorithm explores a search space efficiency and more detailed in a real time by parallel computing of subswarms. The modelling results and comparative analysis of CPSOIW and IWPSO algorithms have been performed based on benchmark datasets and a real production data from Installation 300 of Tengizchevroil oil and gas company.

Subjects: Artificial Intelligence; Computer Engineering; Computer Science; General

ABOUT THE AUTHOR

G. Samigulina is a doctor of technical sciences, Academician MAIN, head of the lab. “Intellectual control systems and forecasting” of the Institute of Information and Computational Technologies, Kazakhstan. She developed the immune-network technology for developing intelligent systems of prediction and control of complex objects under conditions of parameter uncertainty. Her research interests are intellectual systems of industrial automation, Smart-distance learning systems for people with visual disabilities, molecular design of drugs with desired properties, forecasting the risks of complex investment projects, etc.

Massimkanova Zhazira is a second year PhD student at Al-Farabi Kazakh National University by specialty Information Systems and junior researcher at Institute of Information and Computational Technologies, Kazakhstan. Her research interests are swarm intelligence algorithms.

PUBLIC INTEREST STATEMENT

Nowadays, in leading oil and gas companies intelligent systems are successfully used to control oil production processes, optimize production, diagnose the technical state of industrial equipment and alert about emergency situations. One of the main problems of intelligent systems are a processing and analysis of large amounts of data in real time. Currently swarm intelligence algorithms are actively developing and have the ability to quickly and detailed search for optimal solutions. The proposed cooperative particle swarm optimization with weight algorithm combines the advantages of inertia weight particle swarm optimization algorithm, which allows to avoid early convergence and to prevent particles from trapping into local optima due to update an inertia weight at each iteration and cooperative particle swarm optimization algorithm, which explores the search space in a real time efficiently and more detailed by parallel computing of subswarms.
Keywords: smart-technology for forecasting and control of complex objects; feature
selection; cooperative particle swarm optimization with inertia weight; benchmark
datasets

1. Introduction
Recently the application of innovative Smart technologies for forecasting and control of complex non-
linear dynamic objects with parameter uncertainties based on intelligent methods is relevant task. These
technologies allow to detect emergencies of industrial equipment, to forecast and control a production
processes, to increase an efficiency and performance of equipment that promote to reduce a production
cost (Andreadisa et al., 2014). Honeywell Experion PKS (Process Knowledge System) distributed control
systems are widely used in modern industrial automation systems. The application of modern artificial
intelligence methods for analysis and forecasting the states of complex industrial objects on the basis of
a real production data is relevant and provides high efficiency control in case of emergency situations.
For example, in work by (Sayda, 2011) intelligent methods are successfully used to forecast the states of
complex industrial objects and to provide a timely notification about emergencies. In leading oil
companies these methods are successfully used to control oil production facilities. The research of
(2014) considers the issues of ensuring continuous data collection, a real-time monitoring assets and
environmental conditions using thousands of sensors in subsurface wells and surface facilities, weather
services, and others. Processing and analyzing the obtained data, mapping changes in the reservoir over
time allow to increase the oil and gas production, to optimize expenses, to reduce the impact of
environmental risks, and to ensure safe oil production process.

Big data processing and analysis in a real time are the main problems of intelligent systems.
Optimization methods are getting more attention to solve these problems, especially swarm
intelligence algorithms, which have an ability in fast and detailed search of optimal solutions.
Swarm intelligence algorithms are Ant Colony Optimization (ACO) (Yaralidaranli & Shahverdi, 2016),
Particle Swarm Optimization (PSO) (Huang et al., 2016), Artificial Bee Colony (ABC) (Chen et al.,
2019), Cuckoo Search (CS) (Rajabioun, 2011), Firefly Algorithm (FA) (Yang, 2010a), Bat Algorithm
(BA) (Yang, 2010b), Grey Wolf Optimizer (GWO) (Kohli & Arora, 2018), Dolphin Echolocation (DE)
(Kaveh & Farhoudi, 2013), Whale Optimization Algorithm (WOA) (Kaur & Arora, 2018), Fruitfly
Optimization Algorithm (FOA) (Pan, 2012), and others.

In many developed applications, prospects of using swarm intelligence algorithms for prelimi-
mary data processing and feature selection are presented. The existing listed above swarm intelli-
gence algorithms are not universal. Depending on a type and a volume of used data, the
development of efficiency-modified algorithms, which provide parallel processing in real time, is
an actual task. For example, paper by (Shumugapriya & Kanmani, 2017) proposes a novel hybrid
algorithm based on ant colony and bee colony algorithms for feature selection and classification.
The evaluation of proposed algorithm using 13 benchmark datasets shows the efficiency of its
application. Article by (Gholi et al., 2014) presents an efficiency ACO algorithm to optimize of gas
allocation to a group of wells in a gas lift. In article by (Abramov, 2016) bee colony algorithm is
proposed to place objects in oil production. The algorithm describes a decentralized behavior of
intelligent agents, which represent a self-organizing system. Paper by (Pradana et al., 2014) is
devoted to a joint use of Binary Particle Swarm Optimization algorithm and decision tree for
multidimensional data processing and feature selection. The experimental results based on 11
datasets show that the proposed method is more efficiency than naive Bayesian classifier and
support vector machine. In paper by (Teng et al., 2017) v-shaped binary particle swarm optimiza-
tion is used for adaptive feature selection. The modelling outputs show an optimization of feature
subsets using the proposed algorithm. In study by (Najafzadeh et al., 2017) neuro-fuzzy group
method of data handling (NF-GMDH) is implemented using PSO, gravitational search algorithm
(GSA) and genetic algorithm (GA) to evaluate the pier scour depth. The testing results show the
efficiency of NF-GMDH-PSO model in comparison with NF-GMDH-GA and NF-GMDH-GSA models.
Article by (Gholamian & Meybodi, 2014) describes an enhanced comprehensive learning
cooperative particle swarm optimization with fuzzy inertia weight (ELCIFPSO-IW) to solve a problem of early convergence and to avoid a local optimum. Paper by (Umaphathy et al., 2010) presents PSO algorithm with three different inertia weights for optimal power flow solution. Algorithms with a constant inertia weight, a time-varying inertia weight and a global-local best inertia weight are considered to analyze an impact of inertia weight on convergence of PSO algorithm. The paper by (Saleem et al., 2019) presents the niche based bat algorithm (NBBA) to find the relevant subsets. In this algorithm a random walks of bats allow to avoid a local minima and enhances diversity of solutions in a search space. In research by (Tatwani & Kumar, 2019) a master slave parallel genetic algorithm is used for feature selection in high dimensional datasets. The experiments are performed on three high dimensional gene expression data. The modelling outputs show that a dimensionality and an execution time are significantly reduced.

Article by (Han & Bian, 2018) describes a combination of PSO algorithm and support vector machine (SVM) to determine oil recovery factor in a low-permeability reservoir. The accuracy and efficiency of proposed model are evaluated using 34 datasets. Modelling outputs show a low average percentage of absolute error. Paper by (Li et al., 2016) proposes a hybrid system based on PSO algorithm and neural network for power transformer fault diagnosis. The modelling results show a good performance of the proposed algorithm. The paper by (Wu et al., 2015) describes Multi-Agent Particle Swarm Optimization (MAPSO) for a power system economic load dispatch. The experimental results demonstrate that MAPSO algorithm exceeds an accuracy and a convergence speed of Interactive Genetic Algorithm and PSO algorithm regardless of a size of data. In article by (Wang et al., 2012) an integrated model based on PSO algorithm and a multiagent approach is used for optimal resource allocation. Paper by (Lindegaard Mikkelsen et al., 2013) presents Multi-layered Multi-Agent System (MMAS) based on PSO algorithm for allocation a production resources. Paper by (Shangxiong, 2008) proposes a parallel PSO algorithm based on multiagent cooperative algorithm. The system structure consists of several compute units, which are standard PSO algorithms in parallel processing. At each iteration, local best values of subswarms are transferred to a main swarm. The simulation results show that a distributed structure can increase an efficiency of system and a calculation speed.

Therefore, literature review proves a relevance of using modified swarm intelligence algorithms to solve problems of feature selection and to create an optimal dataset. The most important advantage of PSO algorithm, compared to other swarm intelligence algorithms, is its speedy convergence to global optimum (Amoshahy et al., 2016). The proposed modified Cooperative Particle Swarm Optimization with Inertia Weight (CPSOIW) combines the advantages of Inertia Weight Particle Swarm Optimization (IWPSO) algorithm (Maca & Pech, 2015) and Cooperative Particle Swarm Optimization (CPSO) algorithm (Xia et al., 2018). Standard PSO algorithm initially uses a constant value of inertia weight, which leads to premature convergence. In order to preserve a diversity of solutions and to prevent particles from trapping into local optima, in paper by (Technological regulations, 2016) IWPSO algorithm was proposed, in which an inertia weight varies from iteration to iteration. When developing systems for forecasting and control of complex objects, an analysis a large amount of data in real time and a calculation time of algorithm are relevant. CPSO algorithm is aimed at reducing a calculation time of algorithm due to parallel processing of subswarms, it is advisable to combine the advantages of two algorithms in the modified CPSOIW algorithm.

The following structure of article is proposed: Section 2 presents the problem statement of research. Section 3 is devoted to the description of solution methods and algorithms. Section 4 presents the modelling results of developed algorithm and comparative analysis. Conclusion and references are given at the end of article.

2. Problem statement
The problem statement is formulated as following: it is necessary to develop modified Cooperative Particle Swarm Optimization with Inertia Weight (CPSOIW) algorithm for feature selection based
on Smart-technology of forecasting and control of complex objects using a real production data from Tengizchevroil oil and gas company.

The research has been performed using a real production data of Tengizchevroil oil and gas company, the largest industrial enterprise in Kazakhstan. The enterprise consists of several complex technological lines and installations for various purposes. As a complex object, we consider Installation 300, which designed for cleaning petroleum gases from acidic components (Samigulina & Samigulina, 2019; Technological regulations, 2016).

3. Solution methods and algorithms

The modified CPSOIW algorithm based on Inertia Weight Particle Swarm Optimization (IWPSO) and Cooperative Particle Swarm Optimization (CPSO) algorithms has been developed. In CPSO algorithm (Luo & Gao, 2019), the population of agents is divided into one main swarm and several subswarms. The agents of subswarms are focused on an intensive local search and acceleration of convergence speed. The agents of main swarm are focused on a global search and maintaining a solutions variety. Information sharing and rapid adjustment of strategies based on their own experience allow agents to converge quickly to a local solution. In CPSO algorithm, an inertia weight is a constant during iterations, which leads to early convergence. In order to prevent particles from trapping into local optima and to accelerate the particles’ velocity, an inertia weight is varies from iteration to iteration in IWPSO algorithm. Figure 1 shows an agent cooperation, in which each subswarm is IWPSO algorithm.

**Modified CPSOIW algorithm:**

1. Initialization of parameters $c_1$, $c_2$, $w_{\text{max}}, w_{\text{min}}, t_{\text{max}}$, where $c_1$ and $c_2$ – acceleration coefficients, $w_{\text{max}}$ – the start of inertia weight, $w_{\text{min}}$ – the end of inertia weight, $t_{\text{max}}$ – the maximum number of iterations.

2. Initialization of search space. The generation of agents population in D-dimensional space. A random generation of initial positions ($x_i$) and velocities ($v_i$) of particles of main swarm $M$ and two subswarms $S_i$ where $i = 1, 2, \ldots, n$.

3. Fitness-function evaluation of $S$ subswarms $f(x_i(t + 1))$ and determination the best values of fitness-function $f(p_i^b)$. As a fitness-function, CFS (correlation-based feature selector) algorithm (Hall, 1999) is used, which presented as following Eq.1:

$$f_s = \frac{kF_i}{\sqrt{k + k(k - 1)F_i}}$$

(1)

**Figure 1. Agent cooperation in the modified CPSOIW algorithm.**
where $s$—subset, which consist of $k$ descriptors, $\bar{r}_{df}$ — the mean correlation “descriptor-class” ($f \in s$),$\bar{r}_{df}$ — the mean correlation between “descriptor-descriptor”.

(1) The comparison of current value of fitness-function $f(x^f_i(t + 1))$ and the local best values of fitness-function $f(p^g_i)$ of subswarms.

(2) The comparison the local best values of fitness-function of $S$ subswarms $f(p^g_i)$ and the global best value of $f(p^M_g)$ main swarm, which calculated based on the best values of $S$ subswarms.

(3) The update of velocity of $M$ main swarm $v^M_i$ by the following Eq. 2:

$$v^M_i(t + 1) = \omega v^M_i(t) + r_1c_1 + r_2c_2(p^M_g - x^M_i(t)) + r_3c_3(p^g_i - x^M_i(t))$$  (2)

where $t$ — iteration, $w$ — inertia weight, $r_1, r_2, r_3$ — random variables distributed on [0,1], $p^M_g$ — the best value of main swarm, $p^g_i$ — the best value of subswarms, $x^M_i$ — the position of main swarm.

(1) Update the position of particles of $M$ main swarm $x^M_i$ by the Eq. 3:

$$x^M_i(t + 1) = x^M_i(t) + v^M_i(t), i = 1, 2, \ldots n.$$  (3)

(1) The calculation of inertia weight $w$ for the whole swarm. The inertia weight varies from iteration to iteration and it is determined by Eq. 4:

$$w = w_{max} - \frac{w_{max} - w_{min}}{k_{max}}t$$  (4)

9. Checking of stop condition $t < t_{max}$

10. Execution 3–9 steps until the end of iterations.

11. The creation an optimal set of descriptors based on the best values of particles of $M$ main swarm for further use at forecasting and control of complex objects.

The flow chart of modified CPSOIW algorithm is presented in Figure 2.

4. Modelling results

The software “CPSOIW (Cooperative Particle Swarm Optimization with Inertia Weight)” has been developed in Python programming language for feature selection.

The efficiency of proposed modified CPSOIW algorithm has been tested using benchmark datasets UCI (UCI Machine Learning Repository). In the repository, the datasets are stored in arff format. The datasets contain features and instances. The following datasets are used:

1. Dataset “Sonar” contains 208 instances and 61 descriptors in range from 0,0 to 1,0. Data is classified into 2 classes, such as “R” (rock) and “M” (metal cylinder).

2. Dataset “Vehicle” consists of 846 instances and 19 descriptors of various types. Data is classified into 19 classes, such as cost, condition, etc.

3. Dataset “Letter” contains 20,000 instances and 17 descriptors in range from 0 to 15. Data is classified into 16 classes.
4. Dataset “Segment” consists of 2310 instances and 19 descriptors, which is classified into 7 classes.
The proposed CPSONW algorithm is compared with IWPSO algorithm. The parameters, which used at comparing, are shown in Table 1.

The modelling results of CPSONW algorithm based on dataset “Sonar” are shown in Figure 3. The interface of software consists of a database connection button, an input field for parameters, a panel for displaying data from database and a panel for outputs.

The selection of informative descriptors based on IWPSO algorithm is performed using WEKA software (Samigulina & Massimkanova, 2019), which is widely used for machine learning and data mining. The software is written in Java programming language and has GNU General Public License. The comparison of modified CPSONW algorithm and IWPSO algorithm is shown in Table 2.

| Parameters/Descriptions       | Values |
|-------------------------------|--------|
| The number of particles in swarm (P) | 20     |
| The end of inertia weight ($w_{\text{min}}$) | 0.33   |
| The start of inertia weight ($w_{\text{max}}$) | 0.9    |
| Acceleration coefficient ($c_1$) | 0.33   |
| Acceleration coefficient ($c_2$) | 0.34   |
| The maximum number of iterations ($t_{\text{max}}$) | 100    |

Samigulina & Massimkanova, Cogent Engineering (2020), 7: 1788876
https://doi.org/10.1080/23311916.2020.1788876
Table 2. The modelling results of IWPSO and CPSOIW algorithms

| Algorithms          | Database | Sonar | Vehicle | Letter | Segment |
|---------------------|----------|-------|---------|--------|---------|
|                     | Total amount of descriptors |       |         |        |         |
| CPSOIW algorithm    | 61       | 19    | 17      | 19     |         |
|                     | The number of selected descriptors | 13    | 6       | 4      | 7       |
|                     | The value of fitness-function | 0.397 | 0.813   | 0.837  | 0.533   |
| IWPSO algorithm     | The number of selected descriptors | 17    | 11      | 11     | 9       |
|                     | The value of fitness-function | 0.221 | 0.303   | 0.478  | 0.710   |

The relevance of selected descriptors is estimated by the value of fitness-function. A value, which close to one, is considered as the best value of fitness-function. In modelling the first dataset using CPSOIW algorithm 13 informative descriptors from 61 descriptors have been chosen. The modelling outputs of IWPSO algorithm show that 17 informative descriptors from 61 descriptors have been selected. The value of fitness-function of CPSOIW algorithm is more than IWPSO algorithm. The modelling results of the second, the third and the fourth datasets based on CPSOIW algorithm show that 6, 4, and 7 informative descriptors from 19, 17, and 19 descriptors have been chosen.

The modified CPSOIW algorithm has been tested on a real production data from Installation 300 of Tengizchevroil oil and gas company (Maca & Pech, 2015). Installation 300 (I300) consists of the following main units: high-pressure gas cleaning, medium-pressure gas cleaning, dietalome regeneration, amine filtration, chemical supply, etc. Normal functioning of I300 is supported by control and measuring devices such as LIC31002—buoy level gauge, FT31005—difference converter, TT31020—temperature converter, etc. As a database we will use the daily measurements from 19 sensors of Installation 300. Data from sensors is recorded every 4.5 min during the day. The dimension of database is 19×799, 15,181 instances (Table 3).

At forecasting a technical state of I300, the complex object parameters are classified to normal, boundary and emergency operation mode of I300. Table 4 shows the comparison of CPSOIW and IWPSO algorithms based on a real production data from I300.

The modelling outputs of CPSOIW algorithm show that the best value is achieved in a short time due to parallel computation. There are selected 9 informative descriptors from 19 descriptors. The computation time of CPSOIW algorithm is shorter than in IWPSO algorithm.

Table 3. The fragment of daily measurements from industrial I300

| The number of records | LIC31002 | FT31005 | TT31020 | ... | PDT31007 |
|-----------------------|----------|---------|---------|-----|----------|
| 1                     | 65,4     | 85,257  | 108,655 | ... | 54,822   |
| 2                     | 65,7     | 85,844  | 107,803 | ... | 54,517   |
| 3                     | 65,9     | 87,093  | 109,279 | ... | 57,028   |
| ...                   | ...      | ...     | ...     | ... | ...      |
| 799                   | 64,9     | 86,189  | 107,698 | ... | 55,314   |
The selected informative descriptors are further used to forecast emergencies based on artificial immune systems (Samigulina & Samigulina, 2018, 39).

5. Conclusion
The development of modified cooperative particle swarm optimization with inertia weight allows to perform a multidimensional data preprocessing and to select an informative descriptors for Smart-technology of forecasting and control of complex objects. This developed algorithm combines the advantages of cooperative particle swarm optimization and inertia weight particle swarm optimization. The cooperative algorithm allows to explore a multidimensional space in detail and to improve a preprocessing speed due to parallel computing. The inertia weight particle swarm optimization algorithm updates an inertia weight at each iteration and improves solutions variety. Smart-technology of forecasting and control complex objects allows to determine the emergency situations of industrial equipment, quick react to changes in operating mode of installations, which significantly increases the efficiency and the performance of equipment.

Acknowledgements
The research is conducted by grant no. A05130018 CS MES RK (2018-2020) on theme: Cognitive Smart-technology design for intelligent complex objects control system based on artificial intelligence approaches.

Funding
This work was supported by Committee of Science of Ministry of Education and Science of Republic of Kazakhstan, [grant number A05130018] on theme: Cognitive Smart-technology design for intelligent complex objects control system based on artificial intelligence approaches.

Author details
G. Samigulina1
E-mail: qalinassamigulina@mail.ru
Zh. Massimkanova2
E-mail: masimkanovazh@gmail.com
1 Institute of Information and Computational Technologies, Almaty, Kazakhstan.
2 Al-Farabi Kazakh National University, Almaty, Kazakhstan.

Citation information
Cite this article as: Development of modified cooperative particle swarm optimization with inertia weight for feature selection, G. Samigulina & Zh. Massimkanova, Cogent Engineering (2020), 7: 1788876.

References
Abramov, A. S. (2016). Metody iskusstvennogo intellekta dlya resheniya zadachi optimalnogo razmesheniya kustovykh ploshadok i raspredeleniya skvajin medju nimi. Gazovaya promyshlennost, 4(736), 25–29. https://elibrary.ru/item.asp?id=27252684
Amoshay, M. J., Shamsi, M., & Sedaaghi, M. H. (2016). A novel flexible inertia weight particle swarm optimization. PLOS One, 11(8). https://doi.org/10.1371/journal.pone.0161558
Andreadis, G., Klazoglou, P., Nia, K. D. et al (2014). Classification and Review of multi-agents systems in the manufacturing section. Procedia Engineering, 69, 282-290. doi: 10.1016/j.proeng.2014.02.233
Chen, X., Tianfield, H., & Li, K. (2019). Self-adaptive differential artificial bee colony algorithm for global optimization problems. Swarm and Evolutionary Computation, 45, 70–91. doi: 10.1016/j.swevo.2019.01.003
Ghaedi, M., Ghotbi, C., & Aminshahidy, B. (2014). The optimization of gas allocation to a group of wells in a gas lift using an efficiency Ant Colony Algorithm (ACD). Journal of Energy Sources, Part A: Recovery, Utilization and Environmental Effects, 36(11), 1234–1248. https://doi.org/10.1080/15567036.2010.536829
Gholamian, M., & Meybodi, M. R. (2014). Enhanced comprehensive learning cooperative particle swarm optimization with fuzzy inertia weight (ECLCFPSO-IW). In Proceedings of 2015 AI & robotics (IRANOPEN). https://doi.org/10.1109/ROIS.2015.7270730
Holl, M. A. Correlation-based feature selection for machine learning. Thesis of doctoral dissertation (1999), The University of Waikato.
Han, B., & Bian, X. (2018). A hybrid PSO-SVM based model for determination of oil recovery factor in the low-permeability reservoir. Journal of Petroleum, 4, 43–49. https://doi.org/10.1016/j.jpetl.2017.06.001
Huang, S., Tian, N., Wang, Y., & Ji, Z. (2016). Particle swarm optimization using multi-information characteristics of all personal-best information, S(1632). Springer Plus. https://doi.org/10.1186/s40064-016-3244-8.

Table 4. The modelling results of IWPSO and CPSOIW algorithms based on a real production data

| Algorithms | CPSOIW algorithm | IWPSO algorithm |
|------------|------------------|-----------------|
| Total amount of descriptors/measuring devices | 19 | 19 |
| The best value of fitness-function | 0.78 | 0.61 |
| The number of selected descriptors | 9 | 10 |
| Calculation time | 2 ms | 5 ms |
Kaur, G., & Arora, S. (2018). Chaotic whale optimization algorithm. *Journal of Computational and Engineering*, 7(3), 275–284. doi:10.1016/j.jcde.2017.12.006

Kaveh, A., & Farhoudi, N. (2013). A new optimization method: Dolphin echolocation. *Advances in Engineering Software*, 59(53), 53–70. https://doi.org/10.1016/j.advengsoft.2013.03.004

Kohli, M., & Arora, S. (2018). Chaotic grey wolf optimization algorithm for constrained optimization problems. *Journal of Computational Design and Engineering*, 5(4), 458–472. https://doi.org/10.1016/j.jcde.2017.02.005

Li, H., Wang, F., & Wang, R. (2016). Transformer internal insulation fault diagnosis based on rbf neural network evolved by immune particle swarm optimization. In Lecture Notes in Electrical Engineering, Chapter 1, 89–100.

Lindegaard Mikkelsen, L., Naemmann, J. R., Demeouze, Y., & Jørgensen, B. N. (2013). Demonstrating multi-layered MAS in control of offshore oil and gas production. In Advances on practical applications of agents and multi-agent systems, PAAMS 2013. Lecture notes in computer science.

Luo, J., & Gao, Y. (2019). Cooperative particle swarm optimization algorithm with cloud mutation operator based on normal cloud model. *International Journal of Machine Learning and Computing*, 9(5), 554–556. https://doi.org/10.1817/jimlmc.2019.9.5.839

Marcia, P., & Pech, P. (2015). The inertia weight updating strategies in particle swarm optimisation based on the beta distribution. *Journal of Mathematical Problems in Engineering*, 2015, 1–9. https://doi.org/10.1155/2015/790465

Najdof-Zadeh, M., Saberi-Movahed, F., & Sarkamaryan, S. (2017). NF-GMDH-Based self-organized systems to predict bridge pier scour depth under debris flow effects. *Journal of Marine Geosources & Geotechnology*, 36(5), 589–602. https://doi.org/10.1080/10641197.2013.1355944

Pan, W. T. (2012). A new fruit fly optimization algorithm: Taking the financial distress model as an example. Knowledge-Based Systems, 26, 69–74. https://doi.org/10.1016/j.knosys.2011.07.001

Pradana, A. C., Adiwijaya, K., & Aditsania, A. (2014). Implementing binary particle swarm optimization and C4.5 decision tree for cancer detection based on microarray data classification. *Journal of Physics: Conference Series*, 1192, 1–13. https://doi.org/10.1088/1742-6596/1192/1/012014

Quoniam, L., & Baaziz, A. (2014). How to use big data technologies to optimize operations in upstream petroleum industry. https://hal.archives-ouvertes.fr/hal-00944668/file/Paper_IJl_Baaziz_Quoniam_11_1-9.Doi:10.5556/iij

Rojabioun, R. (2011). Cuckoo optimization algorithm. *Applied Soft Computing*, 11(8), 5508–5518. https://doi.org/10.1016/j.asoc.2011.05.008

Saleem, N., Zafar, K., & Sezbwari, A. F. (2019). Enhanced feature subset selection using niche based bat algorithm. *Journal of Computational and Engineering*, 7(3), 49. https://doi.org/10.3390/computation7030049

Samigulina, G. A., & Massimkanova, Z. A. (2019). Development of smart-technology for forecasting technical state of equipment based on modified particle swarm algorithms and immune-network modelling. In *Proceedings of ICCES2019*. Computational and experimental simulations in engineering, Springer, 283–293. DOI: 10.1007/978-3-030-27053-7_26

Samigulina, G. A., & Samigulina, Z. I. (2018). Modified immune network algorithm based on the Random Forest approach for the complex objects control. *Artificial Intelligence Review*, 1-17. doi:10.1007/s10462-018-9621-7

Samigulina, G. A., & Samigulina, Z. I. (2019). Development of smart technology for complex objects prediction and control on the basis of a distributed control system and artificial immune systems approach. *Advances in Science, Technology and Engineering Systems Journal*, 4(3), 75–87. https://doi.org/10.25046/oj40312

Sadya, A. F. (2013). Multi-agent systems for industrial applications: design, development, and challenges. In book: multi-agent systems - modelling, control, programming, simulations and applications. Edited by Dr. Faisal Alkhateeb, InTech 469-494. https://doi.org/10.5772/15696

Shangxiang, S. (2009). A particle swarm optimization (PSO) algorithm based on multi-agent system. In *2018 international conference on intelligent computation technology and automation (ICICTA)*. https://doi.org/10.1109/ICICTA.2008.367

Shunmugapriya, P., & Kannani, S. (2017). A hybrid algorithm using ant and bee colony optimization for feature selection and classification (AC-ABC Hybrid). *Journal of Swarm and Evolutionary Computation*, 36, 27–36. https://doi.org/10.1016/j.swevo.2017.04.002

Tatwani, S., & Kumar, E. (2019). A master slave parallel genetic algorithm for feature selection in high dimensional datasets. *International Journal of Recent Technology and Engineering*, 8(3), 379–384. DOI:10.35940/ijrte.C4184.098319

Technological regulations for the technological process of hydrocarbon gases cleaning of Installation 300. (2016). Tengizchevron LLP.

Teng, X., Dong, H., & Zhou, X. (2017). Adaptive feature selection using x-shaped binary particle swarm optimization. *PloS One*, 12(3). https://doi.org/10.1371/journal.pone.0173907

Umamathy, P., Venkateseshiah, C., & Senthil Arumugam, M. (2016). Particle swarm optimization with various inertia weight variants for optimal power flow solution. *Journal of Discrete Dynamics in Nature and Society*, 3, 1–15. https://doi.org/10.1155/2010/462145

Wang, L., Wang, Z., & Yang, R. (2012). Intelligent multi-agent control system for energy and comfort management of smart and sustainable buildings. *IEEE Transactions on Smart Grid*, 3(2), 605–617. https://doi.org/10.1109/TSG.2011.2178044

Wu, C., Li, H., Wu, L., & Wu, Z. (2015). Agent particle swarm optimization for power system economic load dispatch. *Journal of Computer and Communications*, 3(9), 83–89. https://doi.org/10.4236/jcc.2015.39009

Xia, X., Gui, L., & Zhan, Z. (2018). A multi-swarm particle swarm optimization algorithm based on dynamical topology and purposeful detecting. *Journal of Applied Soft Computing*, 67(126), 126–140. https://doi.org/10.1016/j.asoc.2018.02.042

Yang, X. S. (2010a). Firefly algorithm. *Engineering Optimization*, 42(1), 211–230. https://doi.org/10.1080/03052150701859037

Yang, X. S. (2010b). A new metaheuristic bat-inspired algorithm. In Nature inspired cooperative strategies for optimization (NICSO 2010).

Yaralidarani, M., & Shahverdi, H. (2018). An improved Ant Colony Optimization (ACO) technique for estimation of flow functions from core-flood experiments. *Journal of Natural Gas Science and Engineering*, 33 (624), 624–633. https://doi.org/10.1016/j.jngse.2016.05.067
