Journal of Modern Energy Research
Vol. 1, No. 1, Dec 2023, pp. 1-6
ISSN: 3078-3534, DOI: 10.56947/jmer.v1i1.4

Data Mining Approach to Select Parameters of Swarm Intelligence Algorithms for Optimal Placement Reactive Power Compensation

Pavel Matrenin[1], Elena Tretiakova[2], Artem Tronin[3]
[1]Department of Power Supply Systems, Novosibirsk State Technical University, Russia
[2]Power Supply of the Industrial Enterprises department, Angarsk State Technical University, Russia
[3]Ural Power Engineering Institute, Ural Federal University, Russia

Article Info

Article history:

Received Oct 25, 2023
Revised Dec 2, 2023
Accepted Dec 13, 2023

Keywords:

Data mining
Particle swarm optimization
Parameters selection
Power system optimization
Reactive power compensation

ABSTRACT

This study considers the problem of the parameter's selection of the Particle swarm optimization algorithm. To find effective values of the algorithm parameters, the regression analysis, and rule-based classification have been applied. The regression analysis has allowed creating the model of dependency between values of the algorithm parameters and efficiency of solutions achieved. The classifier has approved allocating areas of the most effective values of the parameters. The areas obtained were used as the areas of possible values when searching extremum of regression curves per each of parameters. The effectiveness of parameters obtained has been compared with the parameters recommended by other researchers. It was found out that using the regression analysis and rule-based classifier allowed to select values of Particle swarm optimization algorithm parameters successfully. An example of using the method to solve the problem of optimal placement of reactive power compensation units in the electrical network is given.

Corresponding Author:

Pavel Matrenin
Department of Power Supply Systems, Novosibirsk State Technical University
Karl Marx Ave, 20, Novosibirsk, Russia
Email: matrenin.2012@corp.nstu.ru

1. INTRODUCTION

The engineering systems consist of thousands difference units. Therefore, a lot of hard optimization problems arise. There are not deterministic methods are allowing to get an optimal solution of the NP-hard problem within a reasonable time. So, various approximate heuristic methods are used. The drawback of the heuristic methods is the difficulty of determining their parameters, which could ensure high efficiency since for different classes of problems. Also, even for various problems of one type, the effective values of parameters are differing. The aim of this study is to investigate the applicability of the data analysis methods, such as regression analysis and classification, to find the relationship between the parameters of heuristic algorithms and obtained solutions of optimization problems. The Particle Swarm Optimization (PSO) algorithm was selected as the example. At this stage, it is planned only an initial analysis of the proposed approach on the general test problem.

The selection of parameters of Genetic Algorithms, Swarm Intelligence algorithms, and other heuristic algorithms are considered in many studies [1]–[3]. When using such methods, the experimental selections of different sets of parameters are carried out, and then the authors stopped on parameters giving the average best results for the problems under consideration. For example, M. Dorigo, the author of the Ant Colony Optimization algorithm, specifies 3-5 variants of values to be used in experiments, per each of coefficients to be selected the best ones [4]. The other method is selection manually by experiments. Such "manual" selection methods are found in many works on heuristic algorithms. M. Pedersen specifies this method as traditional and the most straightforward one [2].

A more sophisticated approach is the automatic adjustment of algorithm parameters using different heuristic rules [3], [5]. To improve the efficiency of Swarm Intelligence algorithms, the evolutionary adaptation mechanisms can be used. For example, the Genetic algorithm was used for the Swarm Intelligence algorithms, with which the selection of parameters was performed [5]–[7]. This method is called a meta-optimization, since the parameters of the heuristic algorithm have been selected using another optimization algorithm. In papers of M. Pedersen, the simple and fast algorithm Local Unimodal Sampling is used as the meta-optimization algorithm [2].

Two approaches considered use simple means of identifying dependencies between the parameters of algorithms and their efficiency. In the first case, the statistics are collected, and the parameters that are better most times are proved as the most effective. Their disadvantage is the justification based on the possibility of selection a good set of the parameter for all problems. It is contrary to the NFL-theorem [8]. And the best parameters' set is found out after solving the problem. For other problems, this set of parameters can be far from optimal.

The second approach does not have this disadvantage since the selection of the parameters is carried out dynamically, in particular, for the problem, which is being solved at the moment. But in this case, only one rule is used. It is the better the solution, the more efficiently the used set of parameters and the higher the probability that this set, or a set derived from it, will be used in subsequent iterations.

It may be appropriate to use data analysis tools to identify patterns between parameters of the Swarm Intelligence algorithms and the solutions obtained. This hypothesis is considered in this paper. As an example of Swarm Intelligence algorithms was selected the Particle Swarm Optimization algorithm (PSO) because it is the most commonly used Swarm Intelligence algorithm [9], especially in the power industry [10]–[13].

The paper is organized as follows. Section 2 describes the experiments carried out: generation training dataset, applying regressive analysis and rule-based classification to select good PSO parameters. Section 3 shows comparison results achieved by this approach with results obtained by using recommendation of other researchers; and contains an example of the application the PSO algorithm for optimal placement of reactive power compensation units in the electrical network. Section 4 summarizes this paper.

2. METHODOLOGY

2.1. Particle Swarm Optimization

Particle Swarm Optimization algorithm became known as a universal and efficient algorithm for solving optimization problems thanks to the papers of Kennedy and Eberhart (then it was improved by Kennedy, Eberhart, and Shi) [14], [15]. PSO based on a bird flock' behavior. A flock acts coordinated according to several simple rules. Every bird (called particle) coordinates own movements with the movements of entire flocks (called swarm). In the PSO algorithm, every particle is denoted by a position vector, a velocity vector, and a value of the criterion. The vectors of position and velocity of all particles are updated according to a number of rules considering the best position of a particle, and the best position of the entire swarm. Also, the algorithm uses inertia weights of the particles, velocities restriction and the stochastic deviations.

The algorithm is widely described in the literature [14], [15], so in this article, the description of the algorithm is omitted. The vector of PSO parameters can be described as $P = \{\alpha_1, \alpha_2, \omega, \beta\}$[7].

2.2. Data generation

Experiments on identification of dependencies between the values of the PSO parameters efficiency of the solutions obtained requires a dataset of parameter values and solutions obtained for some optimization tasks. At this stage of the study, it was limited the experiments with a single problem.

In this paper, the well-known Rosenbrock's problem [16] has been selected, which has been modified for a multi-dimensional space of solutions. Test problem is given by the following formula:

$$f(X) = \sum_{i=1}^{N-1\Sigma}((1 - x_i) + 100(x_{i+1} - x_i^2)^2) \to min \quad (1)$$

$$-5 < x_i < 5 | i = 1, \dots, NN = 10. \quad (2)$$

The ranges of coefficient values were determined based on the experience of studies of the PSO algorithm [2], [7], [11]:

$$-3 < \alpha_1 < 3 \quad (3)$$

$$-3 < \alpha_2 < 3 \quad (4)$$

$$-1 < \omega < 1 \quad (5)$$

$$0 < \beta < 1 \quad (6)$$

100 particles and 20,000 iterations of the algorithm were used. 1,200 sets of random coefficients were generated, which values were evenly distributed in the ranges specified above. Since the PSO algorithm is stochastic, the results obtained depend on random factors (the sequence of pseudo-random numbers). Therefore, the single run of the algorithm does not correct assessment of the efficiency of coefficients used. To minimize the influence of random factors, it is necessary to perform the procedure of repeated start and the selection of the best solution several times. Then the average results will show the efficiency of the problem solution using the certain parameters set. In this experiment, the test problem was solved ten times for each set, and then the average criterion value was determined $f(X)$.

As a result, 1,200 tuples $<P, \varphi> = <\alpha_1, \alpha_2, \omega, \beta, \varphi>$ were formed, where φ is obtained by averaging the quality indicator of the corresponding set of parameters.

2.3. Regression Analysis

The dependency of the problem criterion value (1) φ from the parameters of the PSO algorithm *P*, can be described as a polynomial regression of 4th order:

$$\varphi^*(P) = \sum_{i=1}^{k}\sum_{j=1}^{k} k_{ij}(p_i)^j + C \quad (7)$$

where $\varphi^*(P)$ – assessment of the quality indicator φ, when using the vector of parameters *P*, p_i – parameter of the PSO algorithm (i = 1, 2, 3, 4 for α_1, α_2, ω, β, respectively), k_{ij} – polynomial coefficient of the parameter p_i raised to the power of j (see Table 1), *C* – a free polynomial member, which equal in this case 18449.5.

The polynomial order is selected equal four because increasing order wasn't led to significant improvements in the model's accuracy. To find the coefficients of the polynomial, the KNIME development environment of data mining systems was used.

Using the Eq. 7, it is possible to find a global extremum of the value $\varphi^*(P)$ in the parameter value area (3)–(6). However, the regression equation built is not sufficiently accurate in predicting the quality index φ. The deviation of the estimation values $\varphi^*(P)$ from actual values φ is shown in Figure 1, where a line illustrates the dependency of the value $\varphi^*(P)$ from α_2, and markers show values φ at given values of α_2 parameter in the generated set of tuples.

Table 1. Regression equation coefficients

j		1	2	3	4
P	α_1	-1366.9	-1061.8	2.55	80.37
	α_2	-8021.9	438.0	693.3	-8.76
	ω	-10.57	-3938.9	-770.6	5394.5
	β	-2513.7	82258.0	-104806	47376.6

2.4. Classifiers creation

Besides the regression analysis, the study of parameters was conducted using classifiers. Among all tuples, set of 5%, which are the best on the values of the criterion φ, were placed in the class A. Everything remained tuples were put in class B. It was hypothesized that it is possible to build a classifier that can create, according to the training samples, the rules that distinguish the *A* class tuples from tuples of class *B* using the values of the parameters. With confirmation of the hypothesis, it is possible to apply the rules for a more proper selection of parameters, which are the most efficient for the optimization task solved.

In the training set, 75% of tuples of class 1 and 10% of tuples of class *B* were selected randomly. So, the testing set was 12.75% of the whole set. The classifier was created as a rule-based model in disjunctive normal form. As a result, such classifier was obtained:

IF (α_2 > 1.4192 AND α_1 > -1.10315 AND α_1 < 1.92441)
THEN class =A
ELSE class = B

The accuracy of the classifier is 90.3% on the testing set. The model obtained shows that effective solutions with high probability can be received using the value α_2 over 1.4192 and α_1 in the range from -1.10315 to 1.92441.

The next step is the combination of models obtained with the classification and with the regression analysis. For this, the model built by the rule-based classifier is used as the set of restrictions when determining the extremum of the regression dependency. As a result, the combination of the regression analysis and a classifier gives the following parameters of the PSO algorithm for the test optimization problems: $\alpha_1 = 1.925$, $\alpha_2 = 1.89$, $\omega = 0.661$, $\beta = 0.291$.

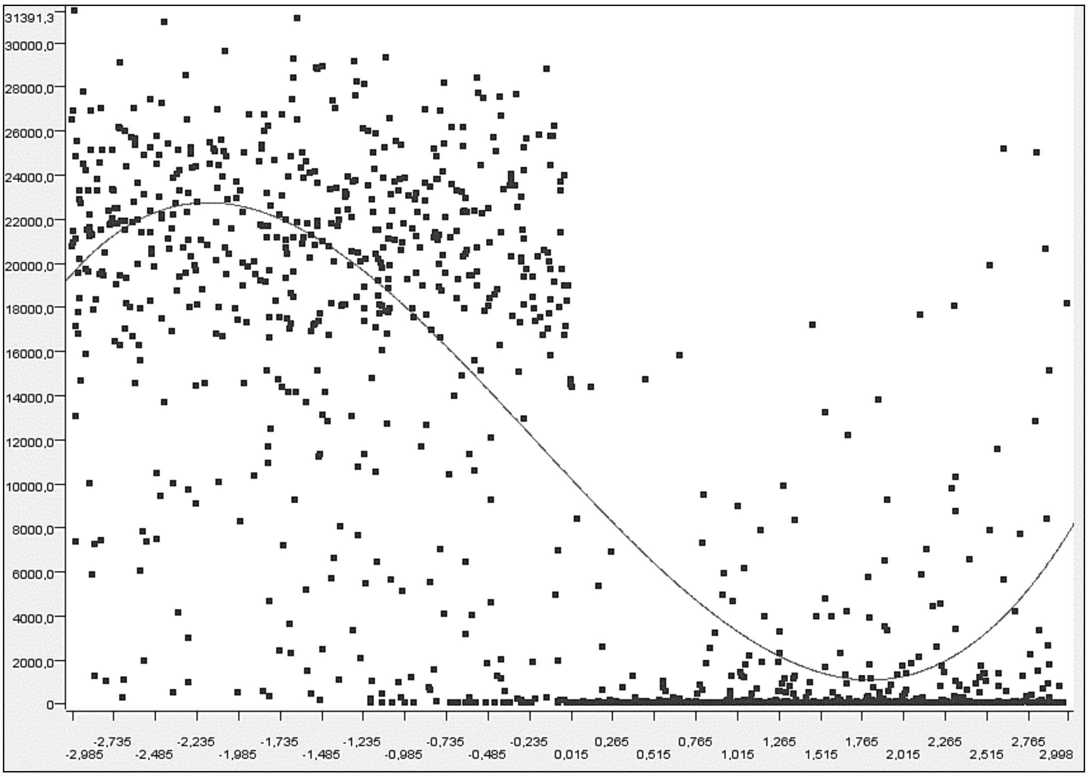

Figure 1. Regression curve for the parameter α_2

3. RESULTS AND DISCUSSION

3.1. The method evaluation

To evaluate the effectiveness of the parameters obtained ($\alpha_1 = 1.925$, $\alpha_2 = 1.89$, $\omega = 0.661$, $\beta = 0.291$), 200 runs of the test problem solution (1) were performed by the PSO algorithm with the given parameter's values. After that, the PSO parameters were selected, taking into account recommendations of the other researchers. The well-known researchers in the field of the Swarm Intelligence of Eberhart and Shi present the following values of the parameters as effective for many continuous optimization problems: $\alpha_1 = 1.49445$, $\alpha_2 = 1.49445$, $\omega = 0.729$, $\beta = 1.0$ [15]. In this case, β is not considered as the changeable parameter as in the standard PSO algorithm, the maximum values of the particle velocity are limited only by the size of the search space.

In the paper of M. Pedersen [17], several sets of parameters are given for various numbers of iterations and different dimensions of optimization tasks. For the situation considered (the dimension of the task is 10, and number of iterations - 20,000), it is recommended to use the following parameters: $\alpha_1 = -0.2746$, $\alpha_2 = 4.8976$, $\omega = -0.3488$, $\beta = 1.0$. In this work, the particle velocity limits are not considered either as the changeable algorithm parameter.

As specified above, for all three sets of parameters, the test problem (1) was solved 200 times. Results obtained by using these sets are given in Table 2.

Thus, the regression analysis and rule-based classifier gave the average results of the solution of the test problem much better than results of using the recommended values in the literature. On the one hand, this was to be expected, since the parameters ($\alpha_1 = 1.49445$, $\alpha_2 = 1.49445$, $\omega = 0.729$, $\beta = 1.0$) and ($\alpha_1 = -0.2746$, $\alpha_2 = 4.8976$, $\omega = -0.3488$, $\beta = 1.0$) were selected for a broad class of problems, and parameters for the one particular issue. On the other hand, most of the optimization problems emerging in practice are non-standard, so the use of the recommended fixed values of parameters may not result in high-performance solutions for the majority of cases. It is necessary to note that the results of the parameters ($\alpha_1 = 1.49445$, $\alpha_2 = 1.49445$, $\omega = 0.729$, $\beta = 1.0$) can be defined as excellent ones. It is much better than solutions obtained at the stage of the data generation, i.e., with the arbitrary values of parameters. Among 1,200 tuples received at this juncture, the average value of the criterion makes up 955.4, and the worst results exceed 3,000.

Table 2. Comparison of sets of parameters

α_1	α_2	ω	β	Average value of criterion	Min. value of criterion
1.925	1.89	0.661	0.290		
1.49445	1.49445	0.729	1.0	8.87	2.90E-11
-0.2746	4.8976	-0.3488	1.0	879.24	588.92

3.2. Optimal placement of reactive power sources

The problem of decreasing energy losses in power supply system up to 10 kV by installing reactive power compensation units (RPCU) is important optimization problem in the power industry [18]–[20]. The power supply system under consideration represents a substation comprising two sections. For electric energy distribution among consumers, power distribution points fed from substation section bus bars with radial arrangement are applied. A considerable part of power distribution points is geographically located to the utmost close to consumers but relatively far from the substation section. The design model of one section contains 10 power distribution points, two pumps with electric motors, two main supply lines of boost compressors. Electric motors of pumps and of compressors in the main supply line are powered (supplied) directly from the bus bars of substation sections.

Minimization of active power losses requires either the installation of the necessary number of RPCU at bas bars of the service consumption station, or optimization of RPCU arrangement near consumers, i.e. deep compensation of reactive power. The last option has been considered in this paper. The optimization objectives are to minimize active power losses, minimize RPCU costs and ensure the power factor (the ratio of reactive power to active power) tgφ < 0.35. This is a multicriteria problem, and the simplest way of its reducing to a monocriterion condition is a linear convolution. Active power losses and costs of RPCU may be combined in a single criterion of financial losses, if we consider the cost of a power loss unit (1 kW) and a RPCU power unit (1 kVar). Costs of power losses are defined by the tariff of the facility considered, and costs of RPCU can be calculated as a mean RPCU cost power ratio for the facility considered. The tg (φ) value was transferred from the criteria class to the limitation class, besides, the low limit was applied to the value tg (φ), as too low value could cause the system instability. As a result, the optimization problem is formulated as follows:

$$W(Q_{RPCU}) = Z_{\Delta P}(Q_{RPCU}) + Z_{RPCU}(Q_{RPCU}) \rightarrow min$$
$$Q_{RPCU} = \{Q_1, Q_2, \ldots, Q_n\}$$
$$Z_{\Delta P}\ (Q_{RPCU}) = c_p \tau \Delta P(Q_{RPCU})$$
$$Z_{RPCU}(Q_{RPCU}) = c_{cu}\ (Q_1 + Q_2 + \ldots + Q_n)$$
$$0 < Q_i < Q_{max\ i}\ ,\ i = 1,\ldots n\ ,\ 0{,}1 < tg(\varphi) < 0{,}35$$

where $Z_{\Delta P}$ is financial losses of active power; Z_{RPCU} is financial losses of installing the RPCU; Q_{RPCU} is the RPCU power vector; Q_i is RPCU power in the *i*-th junction (if $Q_{\max i} = 0$, then RPCU is not installed in the *i*-th junction); *n* is a number of junctions where a RPCU could be installed; it is shown that there are 14 such junctions for the block considered; τ is a considered operation interval expressed in hours at maximum load; ΔP is total losses of active power within a network; $Q_{\max i}$ is the maximum allowable RPCU power level in the *i*-th junction.

During experiments the optimization task was first resolved with the PSO without adaptation, i.e. with arbitrary sets of parameters, and after that it was resolved with the parameter's selection. Given that task, the dimensionality is low; the number of algorithm iterations was limited to a thousand, and the number of agents to a hundred. The obtained results for the solution of the optimization problem are given it Table 3. The experiments showed that the use of deep reactive power compensation allowed reducing active power losses by 30–32%, which, in turn, allowed saving 16.9–17.65 thousand dollars on the considered facility for the settlement period of 4 years.

Table 3. RPCU optimization results

Algorithm	$W(Q_{RPCU})$, $	Active power losses, kW	RPCU total power, kVar	tg(φ)
without optimization	55806	40.55	0	0.55
PSO	42226	27.79	16.61	0.1
Adaptive PSO	42129	27.73	16.57	0.1

The experiment confirmed that for obtaining the highest possible quality solution of the problem, it is reasonable to apply the algorithm parameters' selection. The substation comprises two similar blocks, and that facility has 4 substations, thus the difference will now amount 4 thousand dollars. Besides, a relatively short operation period was considered, 4 years only, and the economy because of deep compensation linearly depends on the period duration.

4. CONCLUSION

To find effective values of PSO algorithm parameters, the models have been built based on the regressive analysis and the rule-based classification. The models have been constructed according to the training dataset containing various values of parameters and averaged test problem solution results obtained by these values. The regression analysis has allowed creating the model of dependency between values of the parameters and efficiency of solutions achieved. The classifier has allowed approved areas of the most efficient values of the parameters. The areas obtained have been used as the areas of allowed values when searching extremum of regression curves per each of parameters.

The parameters found by this method have shown the quality of solutions much better than average quality (average result 0.59 against 955.4 for optimization task described by Eq. 1. So, the hypothesis of the possibility of finding the valid values of the PSO algorithm parameters for a particular optimization problem has been confirmed.

The comparison with the values of the parameters recommended in the literature has been conducted. The comparison has shown a significant superiority of the parameters found (0.59 vs. 8.87 and 879.24). It should be noted that the parameters recommended for a broad class of problems by R. Eberhard and Y. Shi ($\alpha_1 = 1.49445$, $\alpha_2 = 1.49445$, $\omega = 0.729$, $\beta = 1.0$) [9] have showed a high efficiency.

A further stage of the research is to study the approach applicability to solve various problems based on some limited class. In other words, it needs to check whether the parameters selected on the one issue will be useful for other similar problems. The main drawback of the approach described is the necessity to generate a training dataset for the optimization problems on the hands. However, if once selected parameters per a single problem will be effective on all similar problems, then this approach will allow to significantly reduce the time to solve problems, in contrast to the methods of meta-optimization and manual selection of parameters, which involve the implementation of selection for each problem to be solved separately.

REFERENCES

[1] Eberhart, R., Shi, Y., & Kennedy, J. (2001). Swarm Intelligence. Morgan Kaufmann.
[2] Pedersen, M. & Chipper, A. (2010). Simplifying Particle Swarm Optimization. Applied Soft Computing, 10(2), 618-628.
[3] Matrenin, P. et al. (2021). Generalized swarm intelligence algorithms with domain-specific heuristics. IAES International Journal of Artificial Intelligence, 10(1), 157-165.
[4] Dorigo, M., Maniezzo, V. & Colorni, A. (1996). The Ant System: Optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics, Part B, 26(1), 29-41.
[5] Matrenin, P. (2023). Improvement of Ant Colony Algorithm Performance for the Job-Shop Scheduling Problem Using Evolutionary Adaptation and Software Realization Heuristics. Algorithms, 16(1), 15.
[6] Gong, Y. -J. et al. (2016). Genetic Learning Particle Swarm Optimization. IEEE Transactions on Cybernetics, 46(10), 2277-2290.
[7] Matrenin, P.V. & Sekaev, V.G. (2015). Particle Swarm optimization with velocity restriction and evolutionary parameters selection for scheduling problem. In Control and Communications International Siberian Conference (SIBCON). IEEE.
[8] Wolpert, D.H. & Macready, W.G. (1997). No Free Lunch Theorems for Optimization. IEEE Transactions on Evolutionary Computation, 1, 67-82.
[9] Zhu, Y. & Tang, X. (2010). Overview of swarm intelligence. Computer Application and System Modeling, 9, 400-409.
[10] Vijay, A., Afzal, W., Tariq, M., Mustafa, A., & Shahzad, A. (2022). Review of Power Generation Optimization Algorithms: Challenges and its Applications. In 2022 International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), (pp. 1391-1397). IEEE.
[11] Charles, P., Mehazzem, F. & Soubdhan, T. (2020). A Review on Optimal Power Flow Problems: Conventional and Metaheuristic Solutions. In 2020 2nd International Conference on Smart Power & Internet Energy Systems (SPIES) (pp. 577-582). IEEE.
[12] Ntombela, M., Musasa K. & Leoaneka C.M. (2022). Review of Optimization Techniques for Power Network Reconfiguration. In 2022 30th Southern African Universities Power Engineering Conference (SAUPEC). IEEE.
[13] Matrenin, P., Manusov, V. & Antonenkov, D. (2020). Control of Power Prosumer Based on Swarm Intelligence Algorithms. E3S Web of Conferences, 209, 02020.
[14] Kennedy, J. & Eberhart, R. (1995). Particle Swarm Optimization. In International Conference on Neural Network (pp. 1942-1948). IEEE.
[15] Eberhart, R.C. & Shi, Y. (2001). Particle swarm optimization: developments, applications and resources. In Congress on Evolutionary Computation (pp. 81-86). IEEE.
[16] Rosenbrock, H.H. (1960). An automatic method for finding the greatest or least value of a function. The Computer Journal, 3, 175-184.
[17] Pedersen, M. (2010). Good Parameters for Particle Swarm Optimization. Hvass Laboratories Technical Report, no. HL1001.
[18] Manusov, V., Matrenin, P. & Kokin, S. (2017). Swarm intelligence algorithms for the problem of the optimal placement and operation control of reactive power sources into power grids. International Journal of Design and Nature and Ecodynamics, 12(1), 101-112.
[19] Ismail, B. et al. (2020). A Comprehensive Review on Optimal Location and Sizing of Reactive Power Compensation Using Hybrid-Based Approaches for Power Loss Reduction, Voltage Stability Improvement, Voltage Profile Enhancement and Loadability Enhancement. IEEE Access, 8, 222733-222765.
[20] Rauch, J., Brückl, O. (2023). Achieving Optimal Reactive Power Compensation in Distribution Grids by Using Industrial Compensation Systems. Electricity, 4, 78-95.