

# Determination of an effective implementation of the differential evolution method to power shortage minimization

1<sup>st</sup> Dmitrii Iakubovskii

Department Of Energy Security  
Melentiev Energy Systems Institute of  
SB RAS  
Irkutsk, Russia  
ORCID 0000-0001-8331-6200

2<sup>nd</sup> Dmitry Krupenev

Department Of Energy Security  
Melentiev Energy Systems Institute of  
SB RAS  
Irkutsk, Russia  
ORCID 0000-0002-3093-4483

3<sup>rd</sup> Denis Boyarkin

Department Of Energy Security  
Melentiev Energy Systems Institute of  
SB RAS  
Irkutsk, Russia  
ORCID 0000-0002-7048-2848

4<sup>th</sup> Nadejda Komendantova

Advancing Systems Analysis  
International Institute for Applied  
Systems Analysis  
Laxenburg, Austria  
ORCID 0000-0003-2568-6179

**Abstract**—Constant development of electric power systems leads to their constant enlargement and complication; new ways of their control appear. In this regard, the existing models and software for adequacy assessment may work defective and ineffectively from the point of view of the adequacy of the obtained results, as well as the speed and accuracy of calculations. The key role in adequacy assessment of electric power systems (EPS) are played by optimization methods that allow to correctly determine the minimum of power shortage that occurs in various states of the EPS. A review of modern computational systems for adequacy assessment showed that the general concept of mathematical models is the same and can be described within the framework of the flow distribution problem. Despite this, each mathematical model is unique in its own way and requires an individual approach to its optimization. The purpose of this work is to analyze the efficiency of calculations in terms of accuracy and speed of various versions of the differential evolution (DE) method for the specified mathematical models within the framework of adequacy assessment of EPS. To achieve this goal, we solved several problems: two mathematical models were identified - a nonlinear model for minimizing the power shortage with the quadratic losses in flows and its modification with the controlled sections; differential evolution methods, including standard DE, composite DE, JDE, chaotic DE, adaptive DE; mutation strategies: DE/rand/1, DE/best/1, DE/rand/2, DE/best/2, DE/rand/3, DE/best/3, DE/current-to-rand/1, and DE/current-to-best/1. In this paper we tested the effectiveness of differential evolution methods, with different mutation strategies and different scales of EPSs. The experimental part was carried out using a software that was independent development by authors using C++. This complex includes the implementation of mathematical models and methods. The methods were tested on two systems with different numbers of adequacy zones, including those with three and seven adequacy zones. According to the research results, self-adaptive methods of DE are of the greatest interest for the further use and development of methods for this problem, due to automatic adjustment of the method parameters for each of the considered models and systems.

**Keywords**—heuristic algorithms, power system simulation, power system reliability

## I. INTRODUCTION

Modern electric power systems (EPS) must always meet the requirements for a high level of reliability, which is the one of the most pressing challenges of power engineering. This need stems from the fact that any disturbance in the system, subsequently provoke economic costs for both

consumers and for energy companies. Continuous development of the EPS requires monitoring, timely correction and early planning of the systems. The totality of all applied measures directly affects to the reduction of damage due to failures of power equipment and, as a result, restriction of electricity supply to consumers. The solution of the problem presented is realized by qualified adequacy assessment of prospected EPS schemes. As a result of such assessment should be a set of adequacy metrics (AM) of EPS. That metrics also have an economic interpretation and are further analyzed. In international practice, the following AM are used: Expected Unserved Energy, EUE; Loss of Load Probability, LOLP; Loss of Load Expectation, LOLE; Loss of Load Hours, LOLH.

The method of statistical tests (Monte Carlo method) is the most frequently used method for adequacy assessment [1-7]. The Monte Carlo method consists of three main stages. The first stage is to define a set of pseudorandom states of the system. Then, at the second stage, for each defined pseudorandom state of the system, each such state is modeled and power shortages at the consumer's side are calculate. Also events which affect the power shortage, such as the maximum load of generating sources in adequacy zones (AZ) and power transmission lines included in the inter-zone links, are recorded. Random events occurring at adjacent points of time can be either dependent or independent. At the third stage, the results of statistical processing determine the adequacy and other characteristics that are relevant for further analysis.

There are many complexes for adequacy assessing EPS in the world practice [3-11], including ANTARES, Transmission Reliability Evaluation of Large-Scale Systems (TRELSS), Transmission Contingency Analysis Reliability Evaluation (TransCARE), Siemens PTI PSS / E TPLAN, CORAL, OPTGEN / NETPLAN, Grid Reliability and Adequacy Risk Evaluator (GRARE), PLEXOS, Multi-Area Reliability Simulation (MARS), Multi-area Power System (MAPS), ORION-M, CORAL, POTOK-3, YANTAR, and a software package developed by the authors "Nadezhnost". In most complexes, the linear or linearized in the process of solving models are used. Usually that models based on the transport problem of flow distribution. However, this approach introduces a significant error in calculating the power shortage. In [1], the problem was solved where the losses in power transmission lines had a quadratic dependence

on the transmitted power. As a result, there was shown that the most adequate formulation is in a nonlinear form.

The presented complexes use various methods for solving minimization problems, including analytical ones. The Sauer algorithm, various simplex methods, interior point methods, gradient methods, and the software complexes such as GAMS, CPLEX and others. Earlier, we presented an article [12] where we analyzed the application of the standard method of Differential Evolution (DE) to nonlinear models for power shortage minimization (PSM). Studies have shown fast convergence and the ability to accurately find a global solution for one of the models under consideration. Although, the standard DE method is often inferior to its modifications with various improvements, the authors decided to conduct additional research to determine the most appropriate methods for models of PSM.

This work touches on the issue of the efficiency of calculating nonlinear models with quadratic losses using various implementations of the Differential Evolution method, including different mutation strategies [13-20]. This study will allow us to determine the most appropriate method of variation of the DE, including the specific mutation strategy. The experimental part will also identify the dependence of the method with mutation strategy and the modeled EPS.

The article is divided into three sections. In the first section, we formulate and determine the models. Next in second section we present DE methods with modification and various mutation strategies. In third section we provide the experimental part of the research.

## II. PROBLEM DESCRIPTION

The second stage of the methodology for adequacy assessment of EPS by the Monte Carlo method is consist in simulation process of random states of the EPS. Usually a transport problem is solved in terms of adequacy assessment. The main task is finding the flow distribution for given network parameters, generating capacities and power consumption levels in the AZ. There are many different formulations of the PSM in various software systems are used. It should be noted that in terms of the calculation of AM in the first place should be considered to minimize the problem of power shortage from the optimal and physically adequate generation distribution between sources and targets. However, almost all software considers the problem of minimizing various costs as the main one, i.e. the basic task uses the economic criteria, which also affects the subsequent flow distribution

### A. PSM with squared loss Problem

The "Nadezhnost" software package [2], which is currently being developed at the Melentiev ESI SB RAS, uses linear and nonlinear models of power shortage minimization. The nonlinear problem of power shortage minimization is stated as following:

for known values of operable generating power, required levels of power consumption, constraints on power transmission through inter-zone links, and power loss factors in power transmission lines, to determine the minimum value of the power shortage in the EPS.

In mathematical terms, the linear problem is stated as follows:

$$\sum_{i=1}^n (\bar{y}_i - y_i) \rightarrow \min_{y, x, z}, \quad (1)$$

given that the following balance constraints with quadratic losses on power lines are respected:

$$x_i - y_i + \sum_{j=1}^n (1 - a_{ji} z_{ji}) z_{ji} - \sum_{j=1}^n z_{ij} = 0, \quad i=1, \dots, n. \quad (2)$$

As well as the constraints on optimized variables:

$$0 \leq y_i \leq \bar{y}_i, \quad i=1, \dots, n, \quad (3)$$

$$0 \leq x_i \leq \bar{x}_i, \quad i=1, \dots, n, \quad (4)$$

$$0 \leq z_{ij} \leq \bar{z}_{ij}, \quad i=1, \dots, n, \quad j=1, \dots, n, \quad i \neq j. \quad (5)$$

$$z_{ji} * z_{ij} = 0, \quad i=1, \dots, n, \quad j=1, \dots, n. \quad (6)$$

where:  $x_i$  - utilized power in adequacy zone  $i$  (MW);  $\bar{x}_i$  - available power in adequacy zone  $i$  (MW);  $y_i$  - actually consumed power in adequacy zone  $i$  (MW);  $\bar{y}_i$  - maximum consumed power in adequacy zone  $i$  (MW);  $z_{ij}$  - power flow from adequacy zone  $i$  to  $j$  (MW);  $\bar{z}_{ij}$  - total transmission capacity of power transmission lines between AZ  $i$  and  $j$  (MW);  $z_{ji}$  - power flow from adequacy zone  $j$  to  $i$  (MW);  $\bar{z}_{ji}$  - total transmission capacity of power transmission lines between AZ  $j$  and  $i$  (MW);  $a_{ji}$  - specific power loss ratios during its transmission from adequacy zone  $j$  to  $i$ ,  $j \neq i$ ,  $i = 1, \dots, n$ ,  $j = 1, \dots, n$ .

The presented model (1-6) is a common flow distribution model in the field of adequacy assessment (next  $M_1$ ). This model is a transport problem. The main part of the model is described by the balance equations, equalities, which make the main contribution to modeling the active power distribution. Also, for the correct modeling of counter-flow power flows, we used the additional constraint (6), this makes possible to modeling only one-way direction for power flow in each operating mode of the system.

### B. PSM with squared loss and controlled sections Problem

Under the current conditions of operation and development of the EPS, data on the capacity of inter-zone connections are often completely or partially unavailable. At the same time, to control the power transmission between EPS adequacy zones, controlled sections (CS) with a given maximum allowable active power flows (MAF) characteristic are used. Such CSs include up to several branches (power transmission lines) with a designated direction of power flow. Thus, it becomes impossible to use the models presented above without introducing additional changes, and therefore it is proposed to consider the necessary adjustments and additions for stating the mathematical problem with quadratic losses.

First of all, it is necessary to designate matrix  $S$  of the controlled sections, in which the presence of branches in the CS is denoted with their directions taken into account. The dimension of matrix  $l \times m$ , where  $l$  - the number of CSs, and  $m$  - the number of branches, the elements of the matrix are denoted as  $CS_{kf}$ :

$$cs_{kf} = \begin{cases} 1, & \text{if a branch is present in current CS} \\ 0, & \text{if a branch is absent in current CS} \end{cases} \quad (7)$$

Each CS has its MAF in the forward and reverse directions, to store them one needs matrix  $M$  of dimension  $l \times 2$ , where the first element ( $md_{k1}$ ) in the row contains the values of the forward MAF, the second element ( $md_{k2}$ ) contains the values of the reverse MAF. For this model to operate one has also to introduce CS constraints, for forward and reverse MAFs.

$$\sum_{f=1}^m cs_{kf} \cdot z_f^{fwd} \leq md_{k1}; k=1, \dots, l, \quad (8)$$

$$\sum_{f=1}^m cs_{kf} \cdot z_f^{bwd} \leq md_{k2}; k=1, \dots, l, \quad (9)$$

where  $z_f^{fwd}$  - forward power flow  $z_{ij}$  (MW), and  $z_f^{bwd}$  - reverse power flow (MW), i.e.,  $z_{ji}$  that are determined when forming matrix  $S$ . Thus, the existing model  $M_1$  (1-6) should be transformed into the model (1-9), to eliminate the problem of bidirectional loading of CS. This problem was solved by introducing additional constraints on the CS:

$$\left( \sum_{f=1}^m cs_{kf} \cdot z_f^{fwd} \right) \cdot \left( \sum_{f=1}^m cs_{kf} \cdot z_f^{bwd} \right) = 0, i, j=1, \dots, n \quad (10)$$

This constraint, in addition to physically correct distribution of power flows over sections in one of the directions, also allows eliminating constraint (6) from the model thanks to the doubling of actions with respect to the direction of power flows. The result of the above additions and modifications is the following power shortage minimization model with quadratic losses and factoring in of CS: (1 – 5), (7 – 10), referred to as ( $M_2$ ) in what follows.

### III. DIFFERENTIAL EVOLUTION AND MODIFICATIONS

DE is a popular metaheuristic optimization technique used for multidimensional real-valued functions. This method uses a set of individual solutions and belongs to the class of stochastic optimization algorithms using some ideas of genetic algorithms, but does not require working with variables in a binary code. There is no need to calculate the gradient of the function to work this method, which means that the optimization problem should not be differentiable. The algorithm searching for the space with a solution, maintaining a population of possible solutions (individual vectors) and creating new ones, combining existing vectors in accordance with a mutation and crossover process. The candidates with the best target values are retained at the next iteration of the algorithm in such a way that the new objective value of the individual is improved and generating more suitable populations. The process is repeated until the specified stopping criterion is met.

One of the convenient features of the method is considering the upper and lower limits at the stage of calculating new individuals, which simplifies the work with the function without overloading it. This method is also suitable for solving conditional optimization problems with equality and inequality constraints. To solve such problems, the penalty functions are used:

$$\Phi(x, \gamma) = \frac{\gamma}{2} \left( \sum_{i=1}^I (\varphi_i(x))^2 + \sum_{j=1}^J \left( \max \left( 0, g_j(x) \right) \right)^2 \right) \quad (11)$$

where  $\gamma$  – penalty coefficient,  $\varphi_i(x)$  – equal constrains,  $g_j(x)$  nonequal constrains, with:

$$\max(0, g_j(x)) = \begin{cases} 0, & g_j(x) \leq 0 \\ g_j(x), & g_j(x) > 0 \end{cases} \quad (12)$$

The objective  $F(x, \gamma)$  is becoming to:

$$F(x, \gamma) = f(x) + \Phi(x, \gamma) \quad (13)$$

where:  $f(x)$  – the plain original objective function.

#### A. The algorithm of classical DE

The DE algorithm is an easily modifiable constructor. A classical algorithm developed by Storn and Price in 1997 [13], consists of several blocks. Block 1 includes determining the initial values of the parameters -  $N$ : number of elements in the vector (solution),  $NP$ : population size, number of individuals ( $NP = N * m$ , where  $5 \leq m \leq 15$ ,  $F$ : mutation factor [0.1, 1.0],  $CR$ : probability of crossing [0.0, 1.0],  $P_{max}$ : maximum number of generations, which is empirically determined,  $G$ : current generation number;

Block 2 is responsible for initializing the initial population; the more uniform distribution within the upper and lower constraints increase the possibility to find a global solution. In this work, the authors used the Mersenne twister pseudo-random number generator ( $C++$ ,  $std::mt19937$ ) to form the initial population. Then we should perform several operations for each vector from the current population (name target vector).

Block 3 - mutation, in this case we mean the generation of a new mutant vector and the calculation of the value of its objective function within the framework of a mutation strategy using the parameter  $F$ . Let's analyze the main idea of mutation for the DE/rand/1. First we randomly selected several unique auxiliary parent vectors from that population for each selected target vector  $x_G^i, i = 1, 2, \dots, NP$ . This strategy requires 3 parent vectors, it's:  $x_G^{r1}, x_G^{r2}, x_G^{r3}$ , where  $r1, r2, r3 \in \{1, 2, \dots, NP\}$  and  $i \neq r1 \neq r2 \neq r3$ . Next, using the DE/rand/1 mutation strategy a mutant vector  $x_G^{mut}$  is generated as the:

$$x_G^{mut} = x_G^{r1} + F(x_G^{r2} - x_G^{r3}) \quad (14)$$

In practice, there are many different mutation strategies that exist and are used [16–17]. In this article, the authors have implemented and experimentally tested 8 different mutation strategies:

DE/best/1	$x_G^{mut} = x_G^{best} + F(x_G^{r1} - x_G^{r2})$
DE/current to rand/1	$x_G^{mut} = x_G^i + F(x_G^{r1} - x_G^i) + F(x_G^{r2} - x_G^{r3})$
DE/current to best/1	$x_G^{mut} = x_G^i + F(x_G^{best} - x_G^i) + F(x_G^{r1} - x_G^{r2})$
DE/rand/2	$x_G^{mut} = x_G^{r1} + F(x_G^{r2} - x_G^{r3}) + F(x_G^{r4} - x_G^{r5})$
DE/best/2	$x_G^{mut} = x_G^{best} + F(x_G^{r1} - x_G^{r2}) + F(x_G^{r3} - x_G^{r4})$
DE/rand/3	$x_G^{mut} = x_G^{r1} + F(x_G^{r2} - x_G^{r3}) + F(x_G^{r4} - x_G^{r5}) + F(x_G^{r6} - x_G^{r7})$
DE/best/3	$x_G^{mut} = x_G^{best} + F(x_G^{r1} - x_G^{r2}) + F(x_G^{r3} - x_G^{r4}) + F(x_G^{r5} - x_G^{r6})$

In block 4, is characterized by the crossing operation, where new trial vector is formed element by element, from the mutant and target vectors using the  $CR$  parameter. For each of the vector elements a random  $Ir_{and} \in \{1, 2, \dots, N\}$  and a

random number  $r_{v,j} \in \{0.0, 1.0\}$  are generated and then checked for condition:

$$u_{G,j}^i = \begin{cases} x_{G,j}^{mut}, & \text{if } r_{v,j} \leq CR \text{ or } j = \text{Irand} \\ x_{G,j}^i, & \text{otherwise} \end{cases}, j = \{1, \dots, N\} \quad (15)$$

Block 5 - the selection block completes the procedure for forming a new member of the population. In this block it is determined which of the vectors will continue to exist - the trial vector or the target vector. The determination is made by comparing the values of the target functions.

$$x_{G+1}^i = \begin{cases} u_G^i, & \text{if } f(u_G^i) \leq f(x_G^i) \\ x_G^i, & \text{otherwise} \end{cases} \quad (16)$$

Generations are updated in cycle until their number is completely exhausted, or another stopping criterion is reached. We have implemented both the standard DE algorithm and some of its known modifications, namely: composite DE (coDE), jDE, chaotic DE (chDE) and adaptive DE (aDE).

**Algorithm:** Pseudocode of DE.

Initialize:  
N = dimensionality of problem; NP = N\*10;  
F = rand (0.1, 1.0);  
CR = rand (0.0, 1.0);  
P\_max = 50000; G = 1;  
P\_G = initialize population rand;  
**while** (termination criteria not satisfied **or** G≠P\_max) **do**  
best = define best individual  
**for** i with each individual  $x_G^i$  in  $P_G$  **do**  
 $x_G^{mut}$  = Mutation (strategy, F,  $x_G^i$ , best)  
 $u_G^i$  = Crossover (CR,  $x_G^i$ ,  $x_G^{mut}$ )  
 $x_{G+1}^i$  = Selection ( $x_G^i$ ,  $u_G^i$ )  
**end for**  
G = G+1  
**end while**

### B. The algorithm of composite DE

For the first let us focus on the coDE developed in 2011 [14]. The coDE algorithm repeats the basic idea of the classical DE, but has a slightly different approach. Initially, in the block 1 three paired values are formed for the parameters F and CR,  $[F_1 = 1.0, CR_1 = 0.1]$ ,  $[F_2 = 1.0, CR_2 = 0.9]$ ,  $[F_3 = 0.8, CR_3 = 0.2]$ . Further, within the framework of the formation of new vectors using block 3, three mutant vectors are calculated. On this step method uses different mutation strategies: DE/rand/1, DE/Rand/2, DE/current to rand/1, while the values of F and CR are randomly selected from the available ones. The resulting mutant vectors are crossed with the target vector in block 4 to fill the three trial vectors. Next, it should be selected the best of the three trial vectors. Best vector falls into block 5 for the operation of selecting and updating the target vector.

### C. The algorithm of self-adaptive jDE

The DE method is a constructor that lends itself well to changes within blocks, but in general its structure does not change. jDE [14-15] is the modification of DE. For the first step of jDE it is necessary to randomly determine the parameters F and CR for each vector in the population, ie. create vectors  $F_G$  and  $CR_G$  in block 1. The generation of new trial vectors is the same procedure is used as in the classical DE algorithm. However, after block 5, new values

$F_{G+1}^i$  и  $CR_{G+1}^i$  are generated for each member of the population, which will be used in the next generation. To generate new tuning parameters, the following are used:

$$F_{G+1}^i = \begin{cases} F_1 + \text{rand}_1 \cdot F_u, & \text{if } \text{rand}_2 < \tau_1 \\ F_G^i, & \text{otherwise} \end{cases} \quad (17)$$

$$CR_{G+1}^i = \begin{cases} \text{rand}_3, & \text{if } \text{rand}_4 < \tau_2 \\ CR_G^i, & \text{otherwise} \end{cases} \quad (18)$$

where:  $\text{rand}_j$  with  $j \in \{1, 2, 3, 4\}$  are uniform random values in  $[0.0, 1.0]$  and  $\tau_1, \tau_2$  represent probabilities to adjust factors F and CR with suggested values  $\tau_1 = \tau_2 = 0.10$  while  $F_l = 0.10$  and  $F_u = 0.9$  and as a result the new  $F_{G+1}^i$  takes values in the range  $[0.1, 1.0]$ . The new  $CR_{G+1}^i$  takes values in  $[0.0, 1.0]$ . The main idea of this method is self-adaptation of the F and CR parameters for each population. That adaptive system simplifies the handling of the method and does not require the selection of parameters from the user.

### D. The algorithm of chaotic DE

There are several different schemes for applying chaos theory to the DE method. This work uses the same approach to storing individual FG and CRG values for each member of the population, as described earlier for jDE. The first initialization for the parameters of the chDE method [15] [19-20] was carried out by generating random values in the region  $[0.1, 1.0]$  for F and  $[0.0, 1.0]$  for CR in block 1. The generation condition for F and CR  $\notin \{0.00, 0.25, 0.50, 0.75, 1.0\}$  must be satisfied. After setting the initial parameters, the population is initialized in accordance with block 2, then the procedure of mutation, crossing and selection is carried out (block 3, block 4, block 5). Just as in jDE, after each iteration, the operation of determining new values of F and CR is performed, for this it is necessary to touch upon the definition of the logistic equation:

$$y(k) = \mu \cdot y(k-1) \cdot [1 - y(k-1)] \quad (19)$$

where: k is the number of the vector in the generation,  $\mu$  is the controlling parameter, varying within  $0 \leq \mu \leq 4$ . Thus, substituting instead of y (k) the values of F (k) and CR (k), new parameters will be determined to control the process of mutation and crossing.

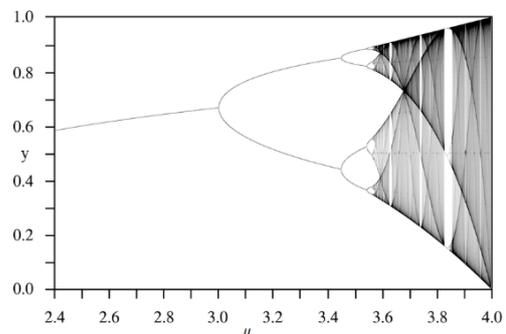


Fig. 1. Bifurcation diagram for the logistic mapping  $x \rightarrow \mu x (1 - x)$

According to the diagram presented, the greatest chaos is achieved at the value of the parameter  $\mu = 4$ . We used this value for further calculations. The selection of the parameters F and CR was carried out using the next equations:

$$F_{G+1}^i = \mu \cdot F_G^i \cdot [1 - F_G^i] \quad (20)$$

$$CR_{G+1}^i = \mu \cdot CR_G^i \cdot [1 - CR_G^i] \quad (21)$$

#### E. The algorithm of adaptive DE

The idea of the considered algorithm aDE [14] is very similar to the two previous algorithms, the parameters are initialized according to the same scheme as in jDE in block 1, a distinctive feature is the way of generating new parameters F and CR. The main control parameter for self-adaptation of F and CR is the average function value of generation ( $f_{avg}$ ) in current moment of generating new parameters. Next equations shown how the parameters generated:

$$F_{G+1}^i = \begin{cases} F_{G+1}^i & \text{if } f(x_G^i) < f_{avg} \\ \text{rand}(0.1, 1.0), & \text{otherwise} \end{cases} \quad (22)$$

$$CR_{G+1}^i = \begin{cases} CR_{G+1}^i & \text{if } f(x_G^i) < f_{avg} \\ \text{rand}(0.0, 1.0), & \text{otherwise} \end{cases} \quad (23)$$

Thus, the parameters depend on how much the function values are improved, and are adapted due to this.

#### IV. EXPERIMENTAL PART

We carried out experimental calculations for systems of various configurations with different initial parameters. The systems with 3 ( $S_1$ ) and 7 ( $S_2$ ) AZ were used as tested systems. Each of the systems was modeled in two formats - using the  $M_1$  – PSM with squared loss model and its modified version with CS  $M_2$ . In total, two stages of calculations were carried out:

- The first stage included systems  $S_1$ ,  $S_2$  modeled using  $M_1$ ;
- The second stage included systems  $S_1$ ,  $S_2$ , while the systems were modeled using  $M_2$ .

The search for optimal solutions at two stages was executed by DE, coDE, jDE, chDE and aDE, we used each of the mutation strategies for each of the methods DE/best/1, DE/current to rand/1, DE/current to best/1, DE/rand/2, DE/best/2, DE/rand/3, DE/best/3. The accuracy of the calculations was checked using the General Algebraic Modeling System (GAMS).

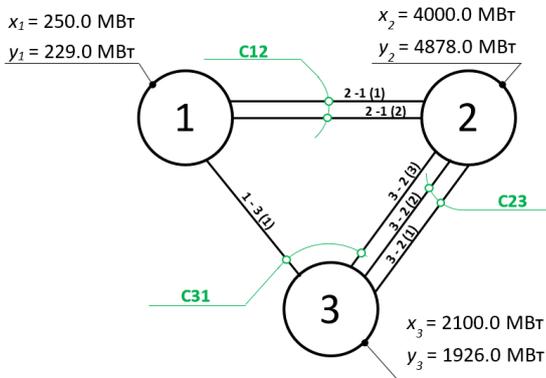


Fig. 2. Systems with 3 adequacy zones.

The program, which includes procedures for generating models and working methods, was realized and built by the MinGW w64 6.0 [x86\_64-8.1.0-posix-seh-rt\_v6-rev0] build

environment using the GCC compiler [version 8.1.0] (C ++), compiler flags "-O2 -march = corei7 -mfpmath = sse -ftrivializer-verbose = 1". All stages of experimental research were carried out on a PC with the following technical parameters: Intel (R) Core i7-8700K @ 3.70GHz, boost 4.50GHz, 6 physical cores, Hyper-Threading, DDR4 48.0 GB, 15/15/15/36, 2133 MHz, Windows [Version 10.0.19042].

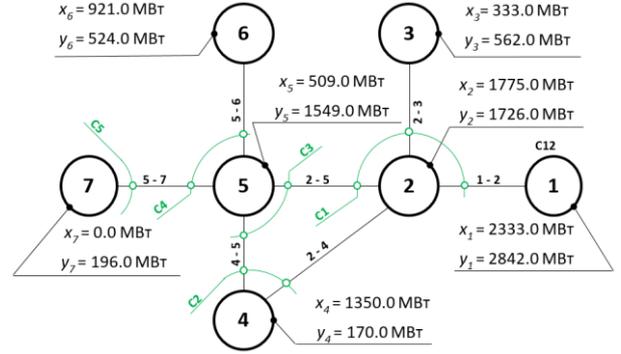


Fig. 3. Systems with 7 adequacy zones

The systems shown in Fig. 2 and Fig. 3, can be modeled using two models. Fig.2 – shown the system with 3 RZ, it's named  $S_1$  it could be modeled by  $M_1$  only for the schema without CS described as (C#) and modeled by  $M_2$  for whole schema include CS. Same features contains on the Fig. 3, which shown us the 7 RZ system named  $S_2$ . We indicated the characteristics of the lines for  $M_1$  and the CS for  $M_2$  for each of the schemes which could be take from tables below.

TABLE I. CHARACTERISTICS OF POWER LINES FOR  $S_1$   $M_1$  AND  $S_1$   $M_2$

Power Line (PL)	Loss factor	PL (MW) (for $M_1$ )		CS (MW) (for $M_{12}$ )		CS name
		Fwd	Bwd	Fwd	Bwd	
2-1(1)	0,00060	25	25	50	50	C12
2-1(2)	0,00060	25	25			
3-2(1)	0,00001	10	10	5	5	C23
3-2(2)	0,00001	10	10			
3-2(3)	0,00001	10	10	80	80	C31
1-3	0,00017	80	80			

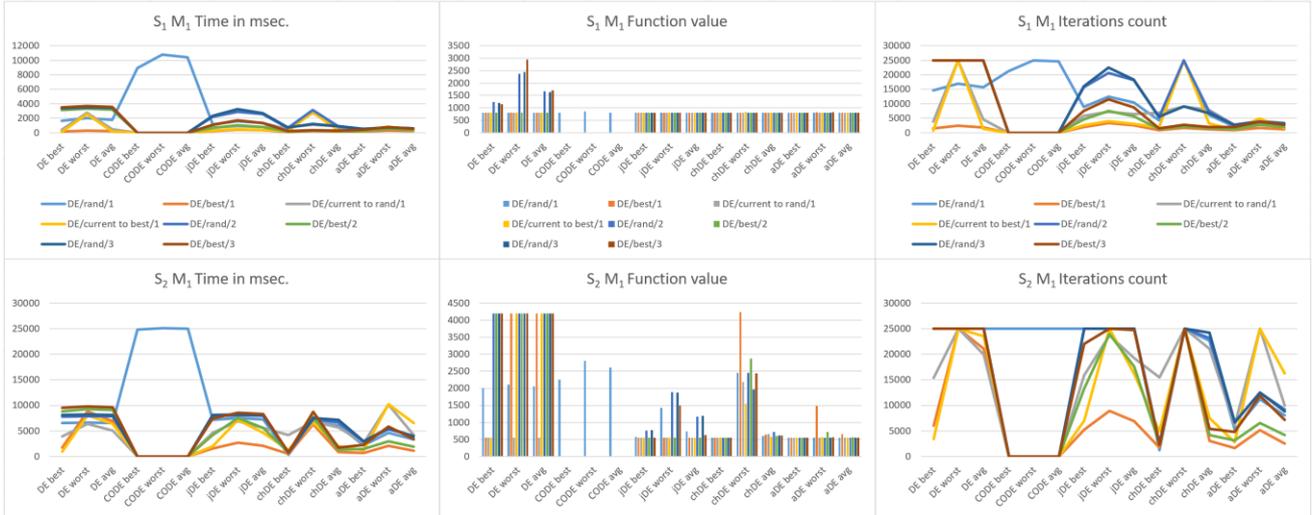
TABLE II. CHARACTERISTICS OF POWER LINES FOR  $S_2$   $M_1$

Power Line (PL)	Loss factor	PL (MW) (for $M_1$ )	
		Fwd	Bwd
1 – 2	0,00040	180	180
2 – 3	0,00002	75	50
2 – 4	0,00018	200	200
2 – 5	0,00022	800	1000
4 – 5	0,00001	1200	1200
5 – 6	0,00052	300	300
5 – 7	0,00031	150	150

TABLE III. CHARACTERISTICS OF POWER LINES FOR  $S_2$   $M_1$

CS name	PL	Loss factor	PL (MW)	
			Fwd	Bwd
C1	1 – 2	0,00040	1310	1460
	2 – 3	0,00002		
	2 – 5	0,00022		
C2	2 – 4	0,00018	1400	1400
	4 – 5	0,00001		
C3	4 – 5	0,00001	2000	2200
	2 – 5	0,00022		
C4	5 – 7	0,00031	450	450
	5 – 6	0,00052		
C5	5 – 7	0,00031	150	150

## M<sub>1</sub>- model



## M<sub>2</sub>- model



Fig. 4. graphs of the bundles (DE method) + (mutation strategy) work for systems S<sub>1</sub> and S<sub>2</sub> modeled using M<sub>1</sub> and M<sub>2</sub>.

### A. First step of experimental part

At the first stage, we execute sequential runs of described methods of DE with each of the presented mutation strategies for problems S<sub>1</sub> and S<sub>2</sub> modeled using M<sub>1</sub>. As the initial parameters for the DE methods we choose: the maximum number of generations 25000, the mutation factor  $F = 0.6$ , the probability of crossing  $CR = 0.9$ , the multiplier of the population size 15. The penalty parameter  $\gamma$  for the penalty functions was 10. In this tests we used the stopping criterion as the minimum difference in the values of the objective functions of the obtained and the previous best vector. Accumulating parameters were determined for each of the **bundles** (DE method) + (mutation strategy), namely:

- best, worst and average operating time of the bundle;
- best, worst and average value of the calculated objective function for the bundle;
- best, worst and average number of spent iterations for the bundle.

First of all, the solution of problems was obtained using the GAMS system. As a result we got optimal value for S<sub>1</sub> M<sub>1</sub> – 798.757 MW, and 549.233 MW for S<sub>2</sub> M<sub>1</sub>. At the same

time, all stabilized bundles stopped at 798.752 MW for S<sub>1</sub> M<sub>1</sub> and 549.194 MW for S<sub>2</sub>, M<sub>1</sub>.

The number of tests for each bundle was 50 calculations. As a result of the program's work Fig. 3, we got an extensive picture of the interaction of different bundles of methods and mutations for task M<sub>1</sub>. We selected stable bundles based on whether the worst solutions out of all deviate from the optimum. If the best solution and the worst solution were equal and did not exceed the optimum, such a bundle was considered stable.

The graphs Fig. 4 shown the largest drawdown in terms of operating time is observed for the coDE method. The method has almost reached the iteration limit, and therefore the optimal value was obtained only for the S<sub>1</sub> system. In this tests coDE did not obtain the final solution for S<sub>2</sub>. In the description part of the coDE method, we noted that this method uses three mutation strategies, so the calculations were carried out only for them, without the intervention of other mutation strategies. As for the rest of the DE methods, there are ambiguous values. The instability of the results of the work of the bundle of the standard differential evolution method and the DE/rand/2, DE/rand/3, DE/best/3 mutation strategies for the S<sub>1</sub> system is observed. The same can be seen for the S<sub>2</sub> system, but the DE/rand/1 mutation is also unstable. As for the operation of the remaining links with the S<sub>2</sub> system,

it should be noted that there are not optimal results of the same mutations for the jDE method. Also, despite the fact that there are correct solutions for all bundles of the chDE method, the chDE worst indicates that obtaining correct optimal results is not guaranteed. The aDE method shown optimal results almost for everyone mutation strategies.

A total of 9 stable bundles with correct results of the objective function values for both systems came out. To rank bundles by speed, we also decided to analyze their running time and the number of iterations spent. The estimation of the operating time was carried out for  $S_2$ , since all stably working bundles were already present in the set of bundles for  $S_1$ . This allowed us to additionally cut off less efficient bundles that do not work well for EPS with a large number of variables.

TABLE IV. RANKED LIST OF EFFECTIVE BUNDLES FOR THE  $M_1$  MODEL

Method	Mutation strategy	Msec.	Iterations
jDE	DE/best/1	2062	6887
aDE	DE/rand/1	3337	8031
aDE	DE/rand/3	3886	8792
aDE	DE/rand/2	3967	9176
aDE	DE/current to rand/1	4208	9970
jDE	DE/current to best/1	4611	16181
DE	DE/current to rand/1	5069	19954
jDE	DE/current to rand/1	5545	19238
jDE	DE/best/2	5562	17553

As we can see from Table 4, jDE and DE/best/1, as well as aDE and DE/rand/1, can provide a quick solution for the  $M_1$  problem. The ranking was based on the average time the method worked on the task until the stop criterion was reached.

### B. Second step of experimental part

At this stage, we made calculations for systems  $S_1$  and  $S_2$  modeled using  $M_2$ . The initial DE parameters, the stop criterion and accumulated data are identical to those of the first stage. We checked the  $M_2$  model using the GAMS system. The optimal value for  $S_1$   $M_2$  was 823.754 MW, and for  $S_2$ ,  $M_2$  - 462.632 MW. The calculations that we carried out for the bundles stabilized at 823.607 MW for  $S_1$ ,  $M_2$  and 462.685 MW for  $S_2$ ,  $M_2$ .

According to the research results, there is the instability of the coDE method calculations. The desired optimal value of the objective function was achieved only for the  $S_1$  system with a smaller number of parameters. The graphs on Fig. 4 shows same results for other bundles of DE and mutation strategies that we got on first stage. We again formed a ranked list of 7 bundles of methods and strategies of mutations with stable values of objective functions and the best computation times.

TABLE V. RANKED LIST OF EFFECTIVE BUNDLES FOR THE  $M_2$  MODEL

Method	Mutation strategy	Msec.	Iterations
aDE	DE/rand/1	1930	7180
aDE	DE/rand/3	2070	7311
aDE	DE/rand/2	2116	7590
aDE	DE/current to rand/1	2463	9151
jDE	DE/current to rand/1	2710	12615
jDE	DE/best/2	3312	13576
DE	DE/current to rand/1	3520	18485

Table 5 demonstrates the most effective combinations of mutation methods and strategies. The fastest and most accurate solution to the  $M_2$  problem can be provided by the bundles aDE and DE/rand/1, as well as aDE and DE/rand/3. The ranking was based on the average time the method worked on the task until the stop criterion was reached.

### C. The result of experimental part

Thus, we tested the optimization of different models  $M_1$  and  $M_2$  with systems of different dimensions  $S_1$ ,  $S_2$  and obtained results regarding the effectiveness of certain bundles of methods and mutation strategies in relation to them. Then for a stable and most time-efficient solution to the  $M_1$  problem can be provided by the jDE and DE/best/1, as well as aDE and DE/rand/1. To solve the  $M_2$  problem, the most advantageous will be the use of aDE and DE/rand/1 as well as aDE and DE/rand/3. Based on this, we can also conclude that the aDE method in bundled with the DE/rand/1 mutation strategy is the most universal and can be used in modeling both problems with stable results.

## V. CONCLUSIONS

In frame of this investigation, we analyzed the various methods of differential evolution, including DE, coDE, jDE, chDE and aDE. We also identified and implemented the various variants of mutation strategies for these methods, namely: DE/best/1, DE/current to rand/1, DE/current to best/1, DE/rand/2, DE/best/2, DE/rand/3, DE/best/3. The main task of this investigation was to determine the combination of method DE with mutation strategy, which would most effectively solve the problem of minimizing the power shortage with quadratic losses and controlled sections. Within the framework of experimental calculations, for each posed problem ( $M_1$ ,  $M_2$ ), EPSs of different dimensions were developed. After that, at least 50 independent calculations were carried out where all the bundles of methods and strategies of mutation were applied. Based on the results of the work, we determine the most universal, stable within the framework of the values of the obtained objective functions and efficient in terms of the computation time, the bundle of aDE and DE/rand/1. In other cases, good results were shown by the bundles aDE and DE/rand/3, jDE and DE/best/1. The least effective methods in this case were the coDE and chDE methods. The fastest calculations were obtained using the chDE method, however, the presence of suboptimal solutions indicates the instability of the method with these models.

## REFERENCES

- [1] Kovalev G.F., Nadezhnost' sistem jelektrojenergetiki [Reliability of electric power systems] G.F. Kovalev, L.M. Lebedeva; Ed. by N.I. Voropai. – Novosibirsk: Nauka, 2015. – 224 p.
- [2] Iakubovskiy D.V., Krupenev D.S., Boyarkin D.A. An Analysis of Shortage Minimization Models to Assess Power System Adequacy // Energy Systems Research. Vol.1. No.3. 2018. P.25-32, doi: 10.25729/esr.2018.03.0003J
- [3] Kovalev G.F., Lebedeva L.M., Model' ocenki nadezhnosti jelektrojenergeticheskikh sistem pri dolgosrochnom planirovanii ih raboty [Model for assessing the reliability of power systems in the long-term planning of their work] Elektrichestvo. — 2000. — ' 11. — 17–24 p.
- [4] Chukreev Yu.Ya. Modeli obespechenija nadezhnosti jelektrojenergeticheskikh sistem [Models of power system reliability support] – Syktyvkar: Komi RC of UrB RAS, 1995. – 176 p. Komi NTs UrO RAN.
- [5] Billinton R., Li W. Reliability Assessment of Electric Power Systems Using Monte Carlo Methods // Springer Science + Business Media, LLC. 1994.

- [6] Wenyuan Li. Probabilistic Transmission System Planning // Wiley-IEEE Press. May 2011. 376 p.
- [7] Working Group 601 of Study Committee C4, Review of the current status of tools and techniques for risk-based and probabilistic planning in power systems // International Conference on Large High Voltage Electric Systems. March 2010.
- [8] Fernandez Blanco Carramolino, R., Careri, F., Kavvadias, K., Hidalgo Gonzalez, I., Zucker, A. and Peteves, E., Systematic mapping of power system models: Expert survey, EUR 28875 EN // Publications Office of the European Union, Luxembourg, 2017, doi:10.2760/422399, JRC109123.
- [9] Antonopoulos G; Chondrogiannis S; Kanellopoulos K; Papaioannou I; Spisto A; Efthimiadis T; Fulli G., Assessment of underlying capacity mechanism studies for Greece, EUR 28611 EN // Publications Office of the European Union, Luxembourg, 2017, doi: 10.2760/51331, JRC106307
- [10] ENTSO-E, Mid-term Adequacy Forecast 2018, Appendix 1: Methodology and Detailed Results, Available at: [https://eepublicdownloads.entsoe.eu/clean-documents/sdc-documents/MAF/MAF\\_2018\\_Methodology\\_and\\_Detailed\\_Results.pdf](https://eepublicdownloads.entsoe.eu/clean-documents/sdc-documents/MAF/MAF_2018_Methodology_and_Detailed_Results.pdf), (accessed 18 March 2020).
- [11] PLEXOS Market Simulation Software Available at: <https://energyexemplar.com/solutions/plexos/>, (accessed 3 April 2020).
- [12] Analysis power shortage minimization methods in the modern processing software for adequacy assessment of electric power systems, Dmitrii Iakubovskii and Dmitry Krupenev, E3S Web Conf., 209 (2020) 06008, DOI: <https://doi.org/10.1051/e3sconf/202020906008>
- [13] Storn, Rainer. (2008). Differential Evolution Research – Trends and Open Questions. 10.1007/978-3-540-68830-3\_1.
- [14] Georgioudakis, Manolis & Plevris, Vagelis. (2020). A Comparative Study of Differential Evolution Variants in Constrained Structural Optimization. Frontiers in Built Environment. 6. 10.3389/fbuil.2020.00102.
- [15] Noman, Nasimul & Bollegala, Danushka & Iba, Hitoshi. (2011). An Adaptive Differential Evolution Algorithm. 2011 IEEE Congress of Evolutionary Computation, CEC 2011. 2229 - 2236. 10.1109/CEC.2011.5949891.
- [16] Opara, Karol & Arabas, J.. (2018). Comparison of mutation strategies in Differential Evolution – A probabilistic perspective. Swarm and Evolutionary Computation. 39. 10.1016/j.swevo.2017.12.007.
- [17] Qamar Abbasa, Jamil Ahmadb, Hajira Jabeenc, The analysis, identification and measures to remove inconsistencies from differential evolution mutation variants doi: 10.2306/scienceasia1513-1874.2017.43S.052
- [18] J. Brest, S. Greiner, B. Boškovič, M. Mernik, and V. Žumer, “Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems,” IEEE Transactions on Evolutionary Computation, vol. 10, no. 6, pp. 646–657, 2006.
- [19] Z. Guo, C. Bo, Y. Min, and B. Cao, “Self-adaptive chaos differential evolution,” in In Proceedings of International Conference on Natural Computation (ICNC), 2006, pp. 972–975.
- [20] G. Yu, X. Wang, and P. Li, “Application of chaotic theory in differential evolution algorithms,” in In Proceedings of International conference on Natural Computation (ICNC), 2010, pp. 3816–3820.