DOI: 10.2478/bhee-2017-0003

Review scientific paper/Pregledni naučni rad

COVERED OVERHEAD CONDUCTOR TEMPERATURE COEFFICIENT IDENTIFICATION USING A DIFFERENTIAL EVOLUTION OPTIMIZATION ALGORITHM

RAČUNANJE KOEFICIJENATA TEMPERATURE ZA POLUIZOLIRAN PROVODNIK POMOĆU OPTIMIZACIJSKOG ALGORITMA DIFERENCIJALNE EVOLUCIJE

Mirza Sarajlić[1], Marko Pocajt[1], Peter Kitak[1], Nermin Sarajlić[2], Jože Pihler[1]

Abstract: The paper describes a method for the temperature coefficient identification for a covered overhead conductor. Numerical and real models of the conductor were used in the paper. The numerical model is made using the Finite Element Method, and contains the material's temperature characteristics and boundary condition values. The real model presents a part of the covered overhead conductor, on which were mounted thermoelements to measure the temperature in laboratory conditions. The optimization algorithm Differential Evolution (DE) was used to identify the temperature coefficient in order to achieve minimum difference of the temperatures between the two models. The classic algorithm DE is compared to two improved DE algorithms, with dynamic changing of the population size during the optimization process. The two improved DE algorithms are: the DE algorithm with half-reducing of the population, and the DE algorithm with linear-reducing of the population. The results of all three algorithms are compared, and the most favourable choice of algorithm is given for such temperature coefficient calculations.

Keywords: covered overhead conductor, temperature, optimization, Differential Evolution, simulation, measurements

Sažetak: Rad opisuje način računanja koeficijenata temperature za poluizoliran provodnik. U radu je korišten numerički i stvarni model provodnika. Numerički model je sastavljen iz konačnih elemenata i obuhvata temperaturne osobine materijala i vrijednosti graničnih uvjeta. Stvarni model predstavlja segment poluizoliranog provodnika na kojeg su u laboratoriju postavljeni senzori i izvedena mjerenja temperature. Računanje koeficijenata temperature izvelo se pomoću optimizacijskog algoritma diferencijalne evolucije (DE) na taj način, da je postignuta manja razlika temperature između oba modela provodnika. Klasični algoritam DE usporediose s poboljšanim algoritmima DE, koji omogućavaju dinamičnu promjenu veličine populacije tokom optimizacijskog postupka. Upotrijebila su se dva poboljšana algoritma: algoritam DE s polovičnim smanjivanjem populacije i algoritam DE s linearnim smanjivanjem populacije. Dobiveni rezultati svih triju algoritama DE su uspoređeni i dat je najpovoljniji izbor algoritma za takvo računanje koeficijenata temperature.

Ključne riječi: poluizoliran provodnik, temperatura, optimizacija, diferencijalna evolucija, simulacija, mjerenja

INTRODUCTION

Heat transfer is a transfer of heat energy from one area to another [1], where the heat passes from the warmer to the colder area. During the heat transfer, there is an alteration in the internal energy between two areas with atomic or molecular displacement, or by means of electromagnetic waves. The transfer velocity is given by the heat flow P (W), which tells how much heat dQ is transferred in the unit of time dt:

$$P = \frac{dQ}{dt} \tag{1}$$

The heat flow density j (W/m^2) tells how much the heat flow dP (W) is per unit of surface dS (m^2):

$$j = \frac{dP}{dS} \tag{2}$$

Usually, the heat transfer is expressed through the following three processes [1]:

- Conduction – the transfer of heat in solid materials with direct contact, without agitation of materials.
- Convection – the transfer of heat to or from a liquid or gas solid surface nearby. Convection is basically conduction with the additional possibility of heat transfer by the movements of fluids or gas.
- Radiation – the transfer of heat by means of electromagnetic waves.

This paper describes a method for the temperature coefficient identification using numerical and real model of

[1]University of Maribor, Faculty of Electrical Engineering and Computer Science, Maribor, Slovenia

[2]University of Tuzla, Faculty of Electrical Engineering, Tuzla, BiH

mirza.sarajlic@um.si

Paper submitted: September 2017 Paper accepted: November 2017

the conductor. This process of identification is complex, hence the novelty of this paper is to identify the thermal conductivity and heat transfer coefficients using optimization algorithms with an innovative and scientific approach. The paper is organized in 6 sections. Section 1 provides a description of thermal conductivity coefficient and heat transfer coefficient. Section 2 presents the covered overhead conductor model. Section 3 describes the carried out measurements on the covered overhead conductor. Section 4 describes methods for identification of the thermal conductivity coefficient and heat transfer coefficient using the optimization process. Section 5 present the resulting coefficients obtained by optimization process and comparison between the calculated model and measured conductor temperatures. Section 6 concludes the paper.

1. HEAT TRANSFER COEFFICIENTS

This chapter presents thermal conductivity coefficient λ and heat transfer coefficient α.

1.1. Thermal conductivity coefficient λ

Figure 1 presents an example of conductivity, where there are two heat reservoirs, and each reservoir has a different temperature (T_1 – hot, T_2 – cold). The reservoirs are connected with a conductor of length l and cross-section S. It is necessary to consider the thermal conductivity coefficient λ, which depends on the type of material. In a steady state, the heat flow flows from a reservoir with a higher temperature to a lower temperature reservoir. Equation (3) describes the heat flow P, taking into account the thermal conductivity coefficient λ, surface S, temperature difference ΔT and conductor's length l.

$$P = \lambda S \frac{\Delta T}{l} \tag{3}$$

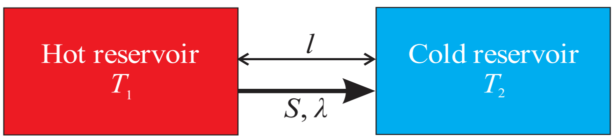

Figure 1: Heat transfer between two reservoirs

Thermal conductivity λ (W/m·K) is a feature of material, which tells how well the material conducts heat. The thermal conductivity coefficient depends on the type of material, temperature and air pressure. Thermal conductivity is divided in three groups: solid material (metals), fluids and gas. The thermal conductivity values of some materials are shown in Table I. Solid materials have the highest thermal conductivity. The conductivity decreases with additions and impurities.

Table I: Thermal conductivity values at 20 °C and atmospheric pressure [2]

Gas	λ (W/m·K)
Air	0.026
CO	0.026
He	0.180
CO2	0.016
Fluids	
Water	0.598
Motor oil	0.145
Ethanol	0.182
Quicksilver	9.304
Metals	
Fe (clean)	80
Al (clean)	230
Cu (clean)	400
Steel (15% Cr; 0.1% C)	26

1.2. Heat transfer coefficient α

Transfer of heat energy between a solid surface and a fluid is called convection. There are two types of convection: natural convection and forced convection. During the forced convection, there is a mixing of fluid particles due to inertial forces, which dominate over the lifting forces, and the heat transfer is more intense than the natural convection. Forced convection occurs when the fluid movement is caused by a fan, pump, or some other source.

The basic convection equation is:

$$j = \alpha (T_s - T_f) \tag{4}$$

where α is the heat transfer coefficient, T_s is the surface temperature and T_f is the fluid temperature.
The heat transfer coefficient α is calculated using (5):

$$\alpha = \frac{\lambda}{\delta} \tag{5}$$

where λ is the fluid's thermal conductivity and δ is the thickness of the boundary layer between the surface and fluid.

When determining the heat transfer coefficient α, factors that influence the heat transfer should be taken into account, which are: type of fluid, type of convection, position along the surface, temperature, surface, geometry and roughness.

2. COVERED OVERHEAD CONDUCTOR MODEL

The covered overhead conductor model is built in EleFAnT [3] using the Finite Element Method (FEM) [9], where the conductor's temperature coefficients are calculated. The conductor model is designed as a 2D problem. Figure 2 shows a 2D model of a covered overhead conductor with the finite element mesh, where the orange colour shows the conductor. Table II indicates the specifications of the real model of a covered overhead conductor.

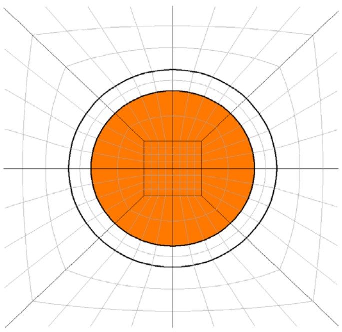

Figure 2: 2D model of a covered overhead conductor

Table II: Covered overhead conductor specifications [2]

Cross-section (mm^2)	70
Number of wires in the conductor	7
Wire diameter (mm)	10.7
Thickness of the inner semi-conductor layer (mm)	0.2
Thickness of the inner XLPE layer (mm)	1.2
Thickness of the outer UV resistant XLPE layer (mm)	1.1
Wire diameter with insulation	
(min – max) (mm)	15.7–16.7
Mass (kg/km)	279
Rated voltage (kV)	24
Maximum current load at conductor's temperature 80 °C, at air temperature 20 °C, IEC 1597 (A)	325
Maximum resistance of conductor at 20 °C (Ω/km)	0.434
Maximum short circuit current, 1 s (kA)	6.65
Aluminum alloy	AlMgSi

Figure 3 shows a covered overhead conductor with insulation, based on an artificial mass or a polyethylene base. That insulation is named XLPE, which is mesh polyethylene, formed by polyethylene vulcanization. Its electric features are the same as that of polyethylene and have better thermal properties. It is non-flammable and cannot be melted. The permissible thermal load is 250 °C. The contemporary production of covered overhead conductors with the XLPE insulation is based on the simultaneous application of all three polyethylene layers, which are called three-layer extrusion. It provides high-quality contact between the layers, despite the repetitive heating and cooling of the conductor. The production of the XLPE material is based on assembling polyethylene linear molecular chains into a 3D molecular mesh. These are the three best known methods of meshing: electron beam meshing, peroxide meshing and meshing based on siloxane bonds. These conductors are less sensitive to excessive short-term loads because of the higher thermal capability. Current XLPE covered conductors, designed to operate under humid conditions, contain moisture protection both longitudinally and radially.

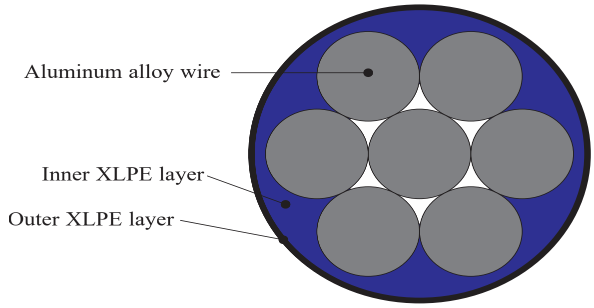

Figure 3: Covered overhead conductor outline

The basis of a conductor is aluminium wires, from which the conductor's rope is interlaced. The cross-section of the aluminium conductor should be 1.61 times higher than that of the copper conductor for the same current and voltage load. Due to the larger surface area, the aluminium conductor is heated by 18% less than copper. Aluminium conductors are half lighter than copper conductors for the same cross-section.

3. MEASUREMENTS

Measuring elements (sensors) for measuring temperature were placed on the conductor and its surroundings. The measurements were conducted at the high voltage laboratory ICEM, Maribor.

The measurements were carried out on a measuring line for continuous loads, and alternating or direct current tests with small currents (up to 50 A) and high voltages (up to 1,500 V), or for loads with high currents (up to 10 kA) and low voltages (up to 12 V). Thus, a test of a permanent load, i.e. heating, can be carried out. The measuring line consists of a low voltage supply, low voltage switchgear, regulation transformer, secondary disconnector, heating transformer and measuring equipment.

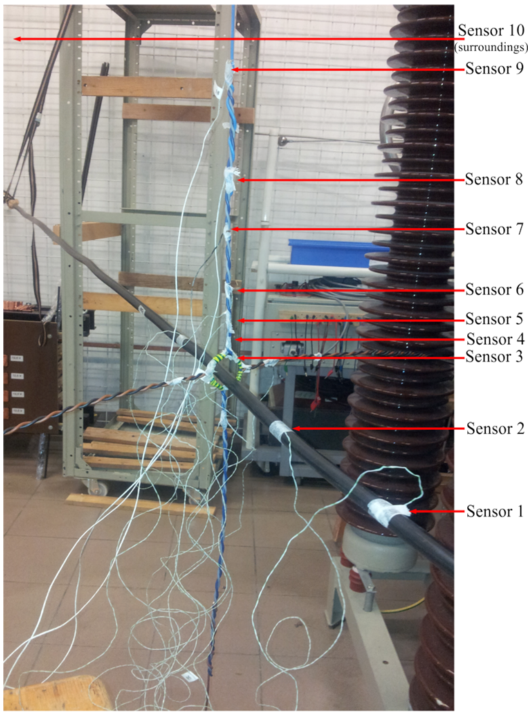

Figure 4: Covered overhead conductor with thermosensors

In the conductor, the current was 200 A, and when the temperature stabilized, the current was raised to 300 A, which created two measurement points. Figure 4 shows a covered overhead conductor with distributed thermosensors.

The sensors are arranged vertically over and under the conductor on the left and right sides. The most sensors are set over the conductor, because the natural convection influence is best recorded in that way.

3.1. The results of temperature measurements

Figure 5 shows the temperature of each sensor over time. At a load of 300 A, the conductor has reached a temperature close to 80 °C. The increase of temperature for the first half hour of measurement can be seen and when the current change occurred (dotted line in Figure 5 shows the point of the change). The highest temperatures have been reached on the conductor's surface. The oscillations were present in temperature (Figure 5) due to the rise of hot air layers, after which the temperature has stabilized. Those oscillations are a result of the natural convection presence. Table III shows the final temperature values on the covered conductor, which are later on used in the optimization process.

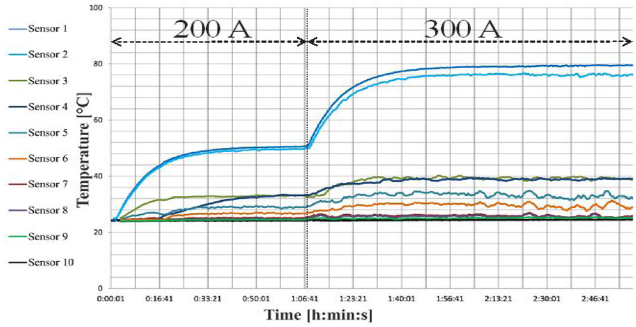

Figure 5: Measured temperature on the covered overhead conductor

Table III: Measured temperature values at 300 A

Sensor	Temperature (°C)
1	79.2
2	76.1
3	39.9
4	39.1
5	33.1
6	29.5
7	26.7
8	25.6
9	25.2
10	24.5

4. IDENTIFICATION OF THE THERMAL CONDUCTIVITY COEFFICIENT AND HEAT TRANSFER COEFFICIENT USING THE OPTIMIZATION PROCESS

Figure 6 shows an optimization process flow diagram for identifying the optimum parameters of the conductor's numerical model. The calculation begins with a randomly generated initial population. For each population member a numerical analysis is performed, which contains the calculation of the temperature field. Calculation of the magnetic field is performed only once, since the current and the power losses do not change. The current density is calculated from the magnetic field, where the skin effect current is taken into account, and power losses p, which is then used in the thermal field when calculating the temperature T. Determining the optimization parameters is performed in the optimization algorithm. A stopping condition (predefined minimal difference between measured and calculated temperature) is checked for each evaluation of the objective function. Once the stopping condition is satisfied, these optimization parameters represent the model parameters. If the stopping condition is not satisfied, the optimization algorithm makes a new set of parameters, and the whole procedure is repeated for the new iteration. The process of creating potential solutions (new set of optimization parameters) is described in 4.1, 4.2 and 4.3.

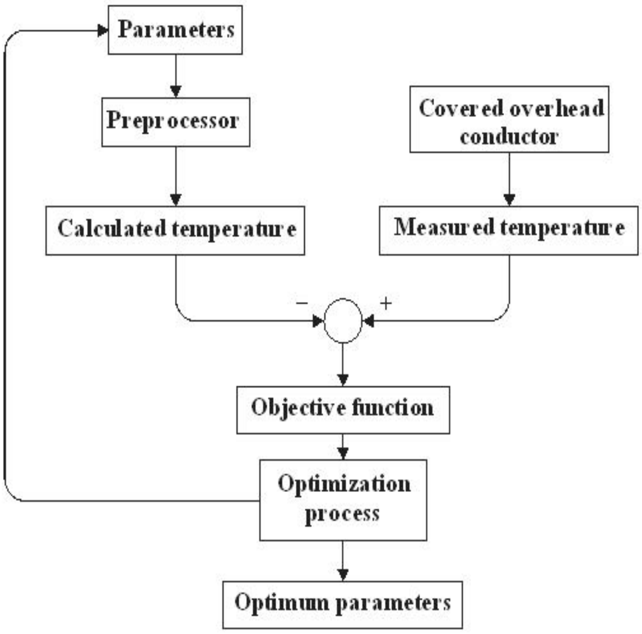

Figure 6: Optimization process' flow diagram

Figure 7a) shows the arrangement and the value of the current density, where it can be seen that the current density is greatest on the conductor's surface ($4.292 \cdot 10^6$ A/m^2). Figure 7b) shows the power losses, where it can be seen that the losses are biggest on the conductor's surface ($511.9 \cdot 10^3$ W/m^3).

One of the most important things about the optimization process is the objective function. In this paper, the objective function q consists of the minimum temperature difference between calculated ($T_{i,calc}$) and measured values ($T_{i,meas}$). Index i presents the observed measurement point. The objective function q is written as follows:

$$q = 0.6 \cdot q_1 + 0.4 \cdot q_2 \tag{6}$$

where q_1 and q_2 are partial objective functions, that are written as

$$q_1 = \frac{1}{2} \sum_{i=1}^{2} \left| T_{i,calc} - T_{i,meas} \right| \tag{7}$$

$$q_2 = \frac{1}{8} \sum_{i=3}^{10} \left| T_{i,clac} - T_{i,meas} \right| \tag{8}$$

The partial objective function q_1 presents two sensors, which were placed directly on the conductor. The partial objective function q_2 presents the remaining eight sensors, which were placed in the surroundings. Weights 0.6 and 0.4 in (6) were chosen based on the analysis of the influence of certain optimization parameters on the temperature calculation.

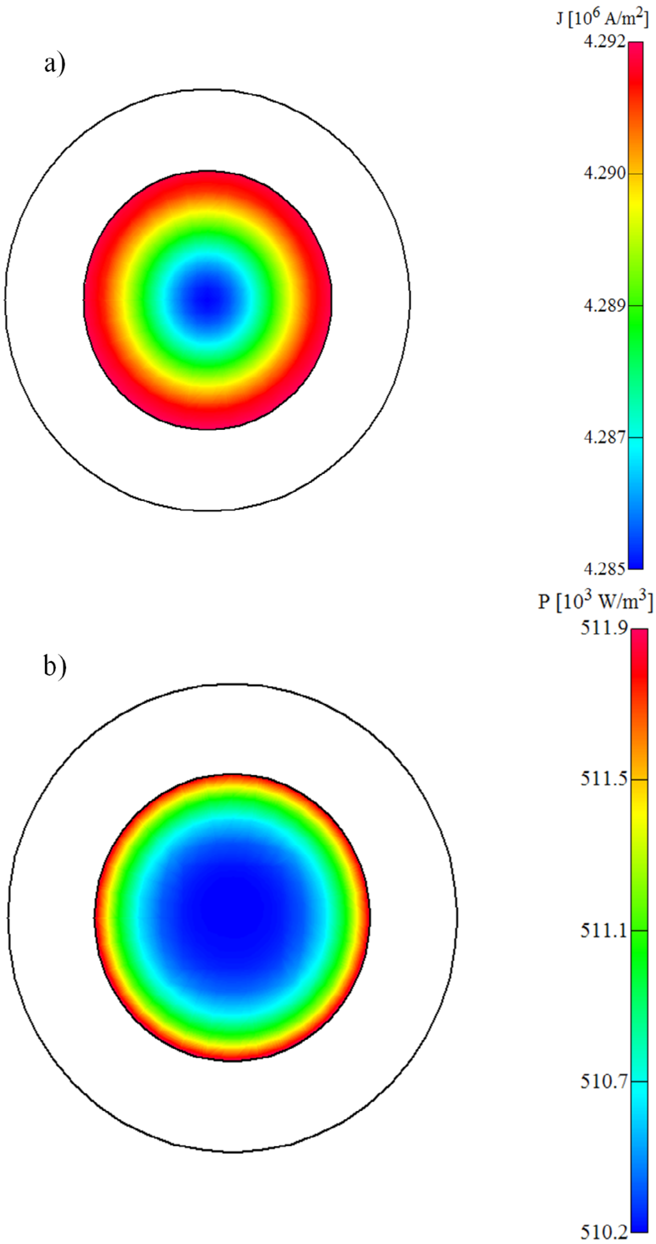

Figure 7: The arrangement of the covered overhead conductor's: a) Current density and b) Power losses

4.1. The classical DE algorithm

This subsection describes the classical DE algorithm [4, 5, 9]. The general problem with optimization algorithms is to find vector x in order to optimize $f(x)$; $x = (x_1,x_2,\ldots,x_D)$. D is a dimension of function f. The variable domains are defined with the lower and upper bound: $x_{j,lower}$, $x_{j,upper}$; $j \in \{1,\ldots,D\}$. The initial population is chosen randomly between lower ($x_{j,lower}$) and upper bound ($x_{j,upper}$), which is defined for each variable x_j. The bounds are defined by the user regarding the problem.

DE is a population based algorithm and the vector $x_{i,G}$; $i = 1,2,\ldots, NP$ is individual in that population. NP represents population size and G generation. Figure 8 shows the main steps of the DE algorithm. During one generation, DE executes mutation and crossover for each vector and, thus, creates a test vector. Between the original vector and the test vector, the vector with the best value of the objective function is selected through selection. The classical DE algorithm is described in detail in the literature [5, 6].

1. Set parameters: F, CR, NP and counter of generation $G=1$
2. Generate random initial population: $x_{i,0}$; $i=1,2,\ldots, NP$
3. while stopping criterium is not satisfied
   4. for each vector $x_{i,G}=\{x_{1i,G}, x_{2i,G}, \ldots, x_{Di,G}\}$; $i=1,2,\ldots, NP$
      a) Choose random 3 indices r_1, r_2, r_3 within the area $[1, NP]$ None of these indices can not be equal to the index i
      b)(Mutation): Generate mutation vector $v_{i,G+1}$
      $$v_{i,G+1} = x_{r_1,G} + F \cdot (x_{r_2,G} - x_{r_3,G})$$
      c)(Crossover): Generate new vector $u_{i,G+1}$
      $$u_{ji,G+1} = \begin{cases} v_{ji,G+1} & \text{if} \quad rand(j) \le CR \quad \text{or} \quad j = rn(i), \\ x_{ji,G} & \text{if} \quad rand(j) > CR \quad \text{and} \quad j \ne rn(i), \end{cases}$$
      where $r(j) \in [0,1]$ j-times evaluation of randomly generated number $rn(i) \in (1, 2, \ldots, D)$ randomly chosen index
      d)(Selection):
      $$x_{i,G+1} = \begin{cases} u_{ji,G+1} & \text{if} \quad f(u_{i,G+1}) < f(x_{i,G}), \\ x_{i,G} & \text{otherwise.} \end{cases}$$
   End for
   $G=G+1$
   End while
5. Display the best results

Figure 8: The classical DE algorithm

4.2. The DE algorithm with the reduction of half of the population

This algorithm is based on [7], where the authors named it as a dynamic population size. At the beginning of the optimization process, a greater number of individuals is needed due to the diversity of the population. During the optimization process, the population reduces gradually, and it is the smallest at the end of the process. The population size reduction step is defined as:

$$_{i\ G} \begin{cases} x_{\frac{NP}{2}+i,G} \ ; \text{if } f(x_{\frac{NP}{2}+i,G}) \quad f(x_{i\ G}) \\ \qquad \text{and } G^{\ 2}\ G \ , \\ _{i\ G} \ ; \quad \text{otherwise.} \end{cases} \tag{9}$$

$$NP_{G+1} = \begin{cases} \dfrac{NP_{G+1}}{2} & \text{if} \quad G = G_R, \\ NP_G & \text{otherwise.} \end{cases} \tag{10}$$

The generation, whose population is going to be reduced is indicated with G_R. The algorithm, described in [7], compares an individual from the first half of the current population and the corresponding individual from the other half of the current population on the basis of their objection function value (9), and the better between them is set on the position i of the current population. The first half of the current population is foreseen for the population which is going to be the parent population in the next generation (10).

In this paper, the reduction of the population is performed as follows: The first 5 iterations the algorithm execute calculations using the whole population (pop_{NP}). The algorithm sorts the individuals with regard to the objective function from the best to the worst. A new sorted population goes to the next iteration. In the 5th iteration, the algorithm takes half of the population and calculates with the reduced population ($pop_{NP1/2}$) by the same principle. In the 20th iteration, the algorithm takes half of the $pop_{NP1/2}$ and performs calculations with the new population $pop_{NP1/4}$. In the 40th iteration, the algorithm takes half of the $pop_{NP1/4}$ and performs calculations to the end of the optimization process with the new population $pop_{NP1/8}$.

That means, if the initial population size is $NP = NP_{initial} = 40$:
- After the 5th iteration $NP = NP_{initial}/2 = 20$;
- After the 20th iteration $NP = NP_{initial}/4 = 10$ and
- After the 40th iteration $NP = NP_{initial}/8 = 5$.

Figure 9a) shows the reduction of NP, and figure 9b) shows the trace of objective function. Figures 9a) and 9b) are obtained minimizing the Rastrigin function [10] using the DE algorithm with the reduction of half of the population.

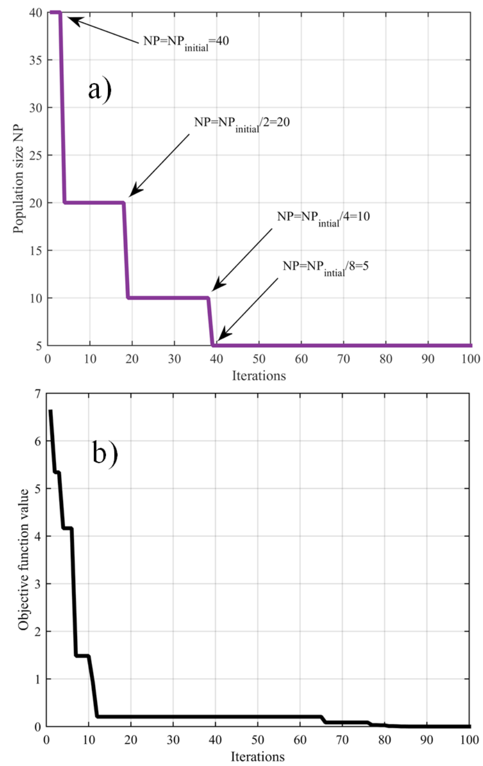

Figure 9: a) The reduction of half of the population and b) Objective function trace during the optimization process

4.3. The DE algorithm with the linear reduction of the population

This algorithm is based on the linear reduction of the population. The reduction of population, i.e. determining the new population size is defined as:

$$NP = NP_{\max} - \frac{NP_{\max} - NP_{\min}}{iter_{\max}} \cdot iter \tag{11}$$

where NP_{max} is the maximum population size; NP_{min} is the minimum population size; $iter_{max}$ is the maximum number of iterations; *iter* is the current iteration. The user defines NP_{max}, NP_{min} and $iter_{max}$ at the beginning of the optimization process.

In this paper, NP_{max} = 40 and NP_{min} = 10 is chosen, so the linear reduction could be carried out from 40 to 10 individuals. Figure 10a) shows the linear reduction of the population. Figure 10b) shows the trace of the objective function. Figures 10a) and 10b) are obtained minimizing the Rastrigin function [10] using the DE algorithm with the linear reduction of the population.

Likewise, with the algorithm described in subsection 4.2, in this algorithm, the individuals are sorted from the best to the worst with respect to the objective function.

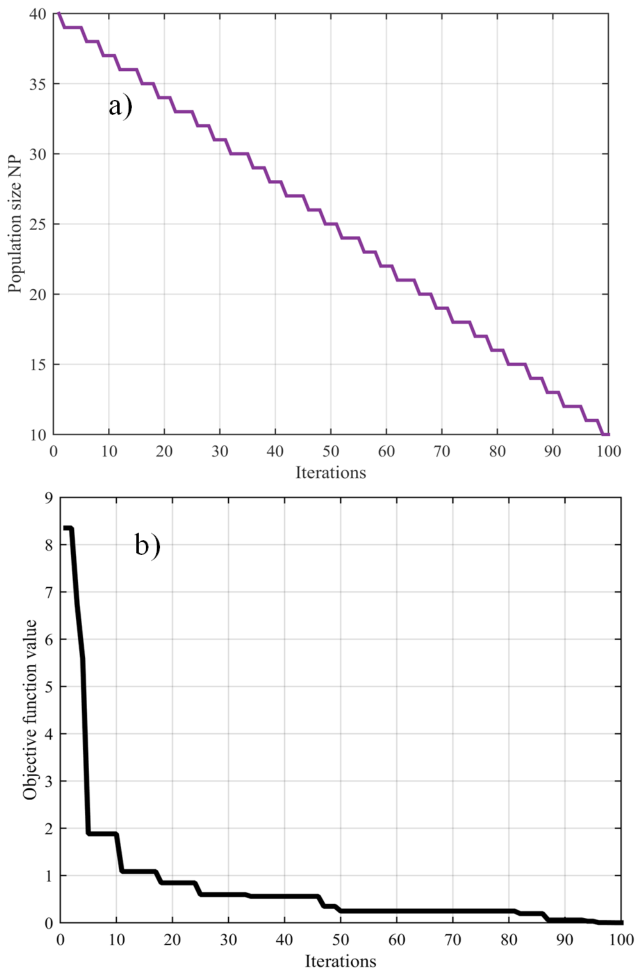

Figure 10: a) The linear reduction of the population and b) Objective function trace during the optimization process

5. OPTIMIZATION RESULTS

After running each optimization algorithm, the following results are obtained: Objective function value (*bestval*), number of objective function evaluations (*nfeval*), number of iterations and CPU time of the calculation.

Table IV shows the optimization process' control parameters. The boundaries of the searched parameters are shown in Table V, where XV_{min} is the lower boundary and XV_{max} is the upper boundary of the searched parameter.

There are four searched parameters, that are labelled as p_1, p_2, p_3 and p_4. The parameter p_1 represents the heat transfer coefficient for air. The parameters p_2, p_3 and p_4, represent the thermal conductivity coefficient for the insulation, conductor and air, respectively.

Meanings of parameters from Table IV:

- *VTR* – Value to Reach;
- *Stopping* – number of iterations for stopping the optimization process, when the objective function does not change;
- *D* – number of parameters;
- *itermax* – maximum number of iterations;
- *F* – mutation factor;
- *CR* – crossover factor.

Table IV: Optimization process' control parameters

VTR	*Stopping*	*D*	*itermax*	*F*	*CR*
1e-6	70	3	250	0,6	0,7

Table V: Minimum and maximum values of the searched parameters [8]

	p_1 (W/m²K)	p_2 (W/mK)	p_3 (W/mK)	p_4 (W/mK)
XV_{min}	2.5	0.15	210	0.02
XV_{max}	20	0.35	250	0.03

Table VI shows the results obtained with the used optimization algorithms, where RHP stands for the Reduction of Half of the Population, and LRP stands for the Linear Reduction of the Population.

Table VI: Results after the optimization

Algorithm	*bestval*	*nfeval*	Iterations	CPU time (s)
Classical DE	9.21	11329	177	1737
RHP	9.21	2129	173	330
LRP	9.21	6782	140	1055
	Identified parameters			
	p_1	p_2	p_3	p_4
Classical DE	8.97	0.35	250	0.02
RHP	8.98	0.35	250	0.02
LRP	9.02	0.35	235	0.02

From the obtained results, it can be seen that all three algorithms have the same objective function value (9.21). The algorithm RHP has the lowest number of objective function evaluations (2129), and the algorithm Classical DE has the highest number of objective function evaluations (11329). The algorithm LRP has the lowest number of iterations (140), and algorithms Classical DE (177) and RHP (173) have the highest number of iterations. CPU time of the optimization process' duration is the shortest with the RHP algorithm (329.95 s) and the biggest with the Classical DE algorithm (1737 s).

For the LRP algorithm there is a slight deviation (1%) in the value of searched parameters p_1 and p_3. The values of the other searched parameters are the same for all the used algorithms.

Figure 11 shows comparisons of the results obtained from the used algorithms. Figure 11a) shows the objective function traces from the used algorithms, where it can be seen that the LRP algorithm's objective function decreases faster than in the other two algorithms. Figure 11b) shows a comparison of the obtained nfeval values from the used algorithms. Figure 11c) shows a comparison of the number of iterations of the used algorithm. Figure 11d) shows the CPU time of the calculation of the used algorithms.

Table VII shows a comparison of calculated and measured temperature of the covered overhead conductor model. When calculating the conductor model temperature, the values of searched parameters obtained after optimization was used.

Table VII: Results of the calculated temperature after the optimization process

Sensor	T_{meas} (°C)	T_{calc1} (°C)	T_{calc2} (°C)	T_{calc3} (°C)
1	79.2	79.4	79.4	79.4
2	76.1	71.1	71.1	71.1
3	39.9	40.3	40.3	40.3
4	39.1	39.1	39.1	39.1
5	33.1	33.1	33.1	33.1
6	29.5	29.9	29.9	29.9
7	26.7	26.8	26.8	26.8
8	25.6	25.9	25.9	25.6
9	25.2	25.1	25.1	25.4
10	24.5	24.5	24.5	24.5

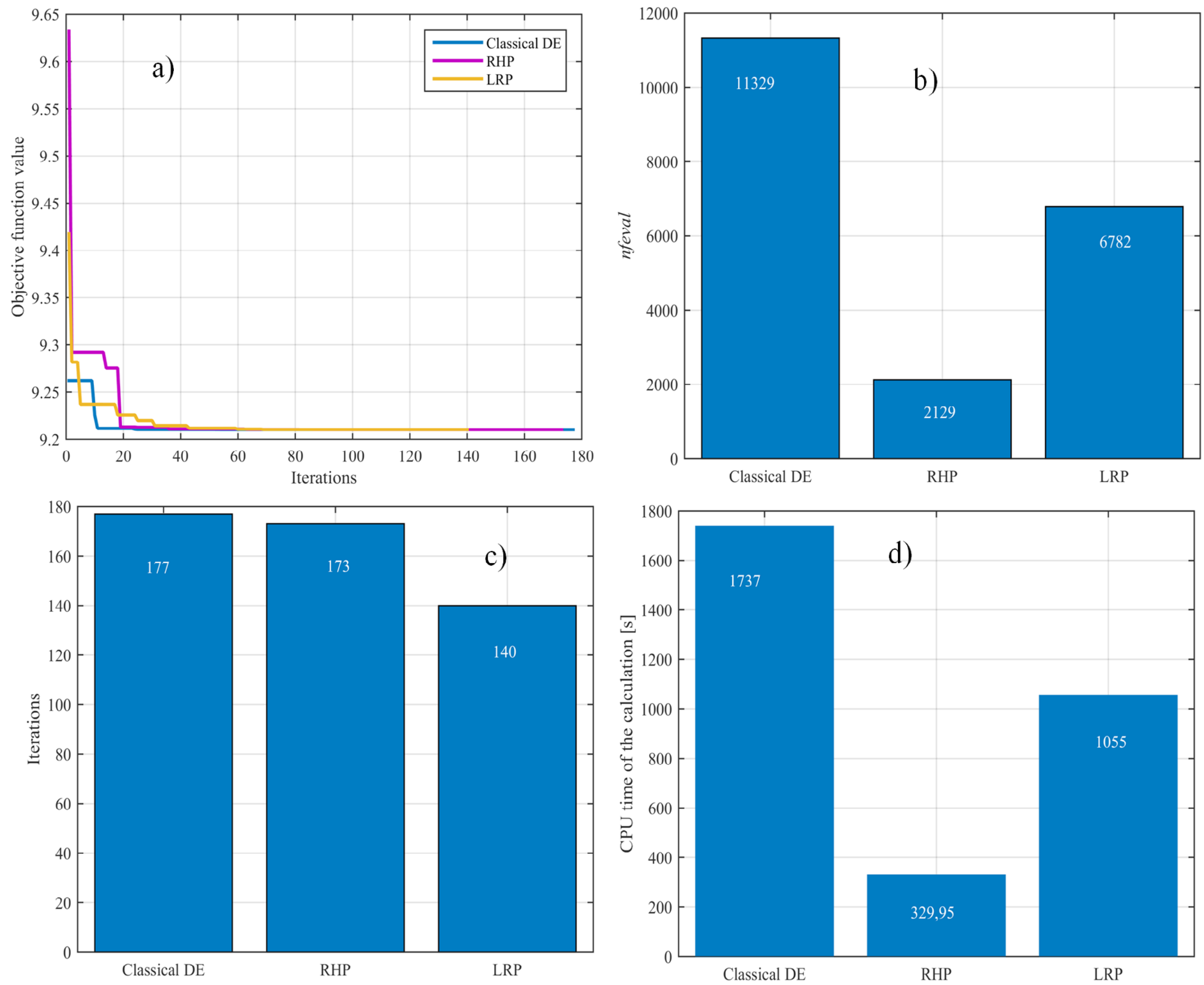

Figure 11: a) The objective function traces; b) nfeval values; c) Iteration number and d) CPU time of the calculation from different algorithms

Indices 1, 2, 3 at T_{calc} represent values of the searched parameters that are obtained using the different algorithms, which are: index 1 represents Classical DE, index 2 represents RHP, and index 3 represents LRP. From Table VII it can be concluded that there are small deviations between calculated and measured temperatures.

6. CONCLUSION

This paper describes the identification process of the covered overhead conductor's heat transfer coefficients regarding the conductor's calculated and measured temperature. The temperature measurements were carried out in the laboratory using thermosensors, which were placed on the conductor and its surroundings. The conductor model was designed as a 2D problem using the Finite Element Method.

The heat transfer coefficient for air and the thermal conductivity coefficient for the insulation, conductor and air, were determined by using the DE optimization algorithm. Aside from the classical DE optimization algorithm, two improved DE algorithms were used, based on the dynamic changing of the population: the DE algorithm with the reduction of half of the population (RHP), and the DE algorithm with the linear reduction of the population (LRP).

After analyzing the obtained results, it can be concluded:

- All three algorithms have the same objective function value after the optimization process.
- All three algorithms gave the same values for the searched parameters p_2, p_3 and p_4.
- There is a slight deviation (1%) in the value of searched parameters p_1 and p_3 for the LRP algorithm.
- The RHP algorithm has the lowest number of objective function evaluations.
- The LRP algorithm has the lowest number of iterations.
- The RHP algorithm has the shortest CPU time of optimization process' duration.
- The RHP algorithm has a 5-times lower number of objective function evaluations and CPU time of optimization process' duration than the classical DE algorithm.
- Exceptional matching of the calculated temperature with the measured parameters.

The purpose of this article was to test three different optimization algorithms. All of the tested algorithms are suitable for identification of thermal parameters in applications as described in this article. However, in terms of objective function evaluation using FEM, which is time consuming, the crucial information is CPU time of optimization process duration. Hence, the most appropriate algorithm among the tested ones is the RHP algorithm.

REFERENCES

[1] T. L. Bergman, F. P. Incropera: Introduction to Heat Transfer, Wiley, 2011

[2] M. Pocajt: Thermal conductivity and heat transfer coefficient calculation of conductor based on optimization algorithm, Master's thesis, University of Maribor: Faculty of Electrical Engineering and Computer Science, Maribor, 2015

[3] Program tools EleFAnT, Graz, Austria, Inst. Fundam. Theory Electr. Eng., Univ. Technol. Graz, 2000

[4] R. Storn, K. Price: Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces, vol. 3. ICSI Berkeley, 1995

[5] R. Storn, K. Price: Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces, Journal of global optimization, vol. 11, no. 4, pp. 341–359, 1997

[6] K. V. Price, R. M. Storn, J. A. Lampinen: Differential evolution: a practical approach to global optimization, Berlin; New York: Springer, 2005

[7] J. Brest, M. Sepesy Maučec: Population size reduction for the differential evolution algorithm, Applied Intelligence, vol. 29, no. 3, pp. 228–247, Dec. 2008

[8] List of thermal conductivities, available at: https://en.wikipedia.org/wiki/List_of_thermal_conductivities

[9] M. Sarajlic, P. Kitak, J. Pihler: New design of a medium voltage indoor post insulator, IEEE Transactions on Dielectrics and Electrical Insulation, vol. 24, no. 2, pp. 1162–1168, 2017

[10] The Rastrigin function, availabe at: http://www.sfu.ca/~ssurjano/optimization.html

BIOGRAPHY

Mirza Sarajlić was born in Tuzla, Bosnia and Herzegovina in 1991. He received his B.Sc., M.Sc. in Electrical Engineering from the Faculty of Electrical Engineering and Computer Science, University of Maribor, Slovenia, in 2013, and 2015, respectively. Currently, he is pursuing his Ph.D. in Electrical Engineering with research interests related to switchgear and optimization methods. He is a member of CIGRE and IEEE.

Marko Pocajt received his B.Sc, M.Sc. in Electrical Engineering from the Faculty of Electrical Engineering and Computer Science, University of Maribor, Slovenia in 2012 and 2015, respectively.

Peter Kitak was born in Ptuj, Slovenia in 1974. He received his B.Sc. and Ph.D. degrees in Electrical Engineering from the Faculty of Electrical Engineering and Computer Science, University of Maribor, Slovenia, in 1999 and 2006, respectively. His special fields of interest are modelling insulators and switchgears, numerical calculation of electromagnetic fields, and optimization methods. He is a member of CIGRE and IEEE.

Nermin Sarajlić was born in Tuzla, Bosnia and Herzegovina in 1965. He graduated from the Faculty of Electrical Engineering, University of Tuzla, Bosnia and Herzegovina in 1987. He acquired a Master of Sciences degree at the Faculty of Electrical Engineering and Mechanical Engineering, University of Tuzla, Bosnia and Herzegovina in 1997, and acquired a Ph.D. degree at the Faculty of Electrical Engineering, University of Tuzla, Bosnia and Herzegovina in 2002. His field of interest is the calculation of the coupled electromagnetic-temperature fields; Cryptography, cryptanalysis.

Jože Pihler was born in Ptuj, Slovenia, in 1955. He received his B.Sc., M.Sc., and Ph.D degrees in Electrical Engineering from the University of Maribor, Maribor, Slovenia, in 1978, 1991, and 1995, respectively. Since 1988, he has been a Researcher and Professor with the Faculty of Electrical Engineering and Computer Science, Department of Electrical Engineering, Institute for Power Systems, Maribor. His special fields of interest are switching devices and switchgear. Prof. Pihler is a member of CIGRE and IEEE.