Weight optimization of a core form oil transformer by using heuristic search algorithms

A. Ozturk¹, H. Demir, L. Kuru, S. Tosun, E. Kuru

Duzce University, Department of Electrical and Electronics Engineering, Duzce, Turkey.

Accepted 18 April 2012

Abstract

There are two imported parameters affecting the weight of a transformer. These are the iron part and the copper part of a transformer [1]. Reducing the weight is the basic criteria to reduce the cost of a transformer. In this study, Heuristic Search Algorithms such as particle swarm optimization (PSO) method, Simulated Annealing (SA) and Gravitational Search Algorithm (GSA) which is a recently proposed method used on optimization problem are used to optimize the weight of a three phase core form oil transformer. The results are compared with the results of the classical method, described in [2]. In the Heuristic Search Algorithms method the total weight of the transformer (G_T), has been accepted as the goal function. The constraints are added to the fitness function as the penalty function to constrain fitness function values. The efficiency (η) and (Ls/a) has been taken as constraints. The fitness function has been obtained as a function of current density and transformer iron cross section convenience value. The critical values have been computed and the results have been evaluated. The total weight which has been founded by Heuristic Search Algorithms are less than the weight which is founded with the classical method. The weight of a 100 kVA three phase core form oil transformer, founded by using PSO, SA and GS are 359,07 (kg), 359,08 (kg), 359,09 (kg) respectively. Whereas the weight founded by classical method was 432 (kg). It has been seen that intuitive methods are an alternative and better way to estimate the minimum weight values. In last decades there has been a great deal of interest on the applications of optimization algorithms to the engineering the problems [3-11].

Keywords: Electrical machine, transformer, optimization, Heuristic Search Algorithms

1. Introduction

Electrical energy is the most consuming energy type on the earth [12]. This energy is transmitted and distributed by transformers. The aim of the transformer design is to completely obtain the dimensions of all the parts of the transformer based on the desired characteristics, available standards, and access to lower cost, lower weight, lower size, and better performance [13-15]. In this study, Heuristic Search Algorithms such as particle swarm optimization (PSO) method, Simulated Annealing (SA) and Gravitational Search Algorithm (GSA), a recently proposed method used on optimization problem, are used to optimize the weight of a three phase core form oil transformer which is designed previously [2]. There are two imported parameters affecting the weight of a transformer. These are the iron part and the copper part of a transformer [1]. In section 2, PSO method is explained. In section 3, the solutions of the problem are explained. In section 3.1 the solution of the problem is done by classical method and in section 3.2 the solution of the problem is done by PSO, SA and GSA Heuristic Search Algorithms. The total weight of a three phase core form oil transformer is obtained as a function of the total iron and the total copper weights of the transformer. This function has been accepted as the goal function. The constraints are added to the fitness function as the penalty function to constrain fitness function values. The efficiency (η) and (Ls/a) has been taken as the constraints. By using the given equations the fitness function has been obtained as a function of current density and transformer iron cross section convenience value. The minimum values for this function were computed using the PSO, SA and GSA methods and finally the results are given in the conclusion part.

2. Heuristic Search Algorithms

Some of the Heuristic search algorithms are Partial Swarm Optimization (PSO), Simulated Annealing (SA) and Gravitational Search Algorithm.
2.1 Partial swarm optimization (PSO)

PSO is a population-based optimization algorithm developed for the first time by [16]. It was inspired by the moving school of fish and swarm inspection. Basically, PSO is an algorithm based on swarm intelligence. It is similar to the computational techniques based on development such as genetic algorithms (GA). While PSO is searching for the optimum in the search space, it uses the population which holds the possible solutions for the function to be optimized [17]. Thus, each individual is called a particle. Particles make up the population called a swarm. In PSO, however, each member of the swarm has a variable speed which changes based on the conditions and determines its movement in the search space. In addition, each member has a memory holding the best spot previously visited. Each particle adjusts its position toward the best position in the swarm by taking the advantage of its previous experience. PSO is basically based on the fact that the positions of the particles in the swarm are brought closer to the best position in the swarm. The speed of “bringing closer” happens in a random fashion and in most cases the particles in the swarm find better positions in their new movements. This process continues until it gets to the goal.

Assuming that the search space is found with D dimensions, the position of the $i^{th}$ particle in the swarm can be expressed in a D dimensional vector as in $X_i=(x_{i1}, x_{i2}, \ldots, x_{iD})^T$ and its speed can be expressed in a D dimensional vector as in $V_i=(v_{i1}, v_{i2}, \ldots, v_{iD})^T$. In addition, the best position of the particle ever visited can be expressed in a D dimensional vector as in $P_i=(p_{i1}, p_{i2}, \ldots, p_{iD})^T$. In the following swarm expressions numbered 1 and 2, $g$ denotes the index number of the best particle, and the upper scripted numbers denote the iteration numbers.

$$V_{id}^{n+1}=wV_{id}^n+c_1r_1^n(p_{gd}^n-x_{id}^n)+c_2r_2^n(p_{gd}^n-x_{id}^n) \quad (1)$$

$$X_{id}^{n+1}=x_{id}^n+v_{id}^{n+1} \quad (2)$$

Here, $d = 1, 2, \ldots, D,$ and $i = 1, 2, \ldots, PS$, where $PS$, the size of the population in the swarm. $r_1$ and $r_2$ are randomly selected values between 0 and 1. $n$ denotes the iteration number. $x_{id}$ and $v_{id}$ denote the position and the speed values, respectively. $w$ is the inertia weight and $C_1$ and $C_2$ are for the scaling factors.

Figure 1. The flow diagram for PSO.

2.2 Simulated Annealing

Early simulated annealing algorithms (SA) considered combinatorial systems, where the system’s state depends on the configuration of variables. Perhaps the best known is the traveling salesman problem, in which one tries to find the minimum trip distance connecting a number of cities [19]. The SA was proposed by Kirkpatrick et al to deal with complex non-linear problems [20]. They
showed the analogy between simulating the annealing of solid as proposed by Metropolis et al. [21]. The SA is an iterative improvement algorithm for a global optimization. The optimization process in SA is essentially a simulation of the annealing process of molten metal’s [22-25]. Annealing is cooled down slowly in order to keep the system of the melt in a thermodynamic equilibrium which will increase the size of its crystals and reduce their defects. As cooling proceeds, the atoms of solid become more ordered. If the cooling was prolonged beyond normal, the system would approach a “frozen” ground state at the lowest energy state possible. The initial temperature must not be too low and the cooling must be done sufficiently slowly so as to avoid the system getting stuck in a meta-stable state representing a local minimum of energy. SA aims to find global minimum without got trapped local minimums. So if object function is a maximization problem, problem is converted minimization problem multiplying minus 1. The simulated annealing makes use of the Metropolis et al. [26] algorithm which provides an efficient simulation according to a probabilistic criterion stated as:

\[
P(\Delta E) = \begin{cases} 
1, & \text{if } \Delta E \geq 0 \\
\frac{1}{e^{-\frac{\Delta E}{kT}}}, & \text{otherwise}
\end{cases}
\]  

(3)

Thus, if \(\Delta E \geq 0\), the probability, \(P\), is one and the change - the new point- is accepted. Otherwise, the modification is accepted at some finite probability. Each set of points of all atoms of a system is scaled by its Boltzmann probability factor \(e^{-\frac{\Delta E}{kT}}\) where \(\Delta E\) is the change in the energy value from one point to the next, \(k\) is the Boltzmann’s constant and \(T\) is the current temperature as a control parameter. This condition was represented by eq. 3. The general procedure for employing the SA as follows;

Step 1: Start with a random initial solution, \(X\), and an initial temperature, \(T\), which should be high enough to allow all candidate solutions to be accepted and evaluate the objective function.

Step 2: Set \(i = i + 1\) and generate new solution \(X_{i}^{new} = X_{i} + r * SL_{i}\) where \(r\) is random number and \(SL_{i}\) at each move should be decreased with the reduction of temperature. Evaluate \(F_{i}^{new} = F(X_{i}^{new})\)

Step 3: Choose accept or reject the move. The probability of acceptance (depending on the current temperature) if \(F_{i}^{new} < F_{i-1}\), go to Step 5, else accept \(F_{i}\) as the new solution with probability \(e^{-\frac{\Delta E}{kT}}\), where \(\Delta E = F_{i}^{new} - F_{i-1}\) new and go to Step 4.

Step 4: If \(F_{i}\) was rejected in Step 3, set \(F_{i}^{new} = F_{i-1}\). Go to Step 5.

Step 5: If satisfied with the current objective function value, \(F_{i}\), stop. Otherwise, adjust the temperature \(T_{new} = T_{r}\) where \(r\) is temperature reduction rate called cooling schedule and go to Step 2. The process is done until freezing point is reached. The major advantages of the SA are an ability to avoid becoming trapped in local optimum and dealing with highly nonlinear problem with many constraints and multiple points of optimum [27].

2.4 Gravitational Search Algorithm

Gravitational search algorithm (GSA) is a method used on optimization problems [28,29,30,31]. The GSA is constructed on the law of Newtonian Gravity: “Every particle in the universe attracts every other particle with a force that is directly proportional to the product of their masses and inversely proportional to the square of the distance between them”. In the algorithm, all the individuals can be viewed as objects with masses. The objects attract each other by the gravity force, and the force makes all of them move towards the ones with heavier masses. The objects transform information by the gravitational force, and the objects with heavier masses become heavier. The mass of each particle is calculated according to its fitness value, as in eq. 4.5 where \(t\) is time (iteration), \(fiti\) presents fitness value of the \(i\)-th particle, \(Mi\) is the normalized mass, \(nmass\) is the number of particles, and \(worst(t)\) and \(best(t)\) are defined for a minimization problem as in eq. 6,7. The vector of force, applied by mass \(j\) to mass \(i\), \(fij(t)\), is calculated as in eq. 8. Here \(xi\) is the position vector of the \(i\)-th agent, \(c\) is a small threshold, and \(G(t)\) is the gravitational constant, initialized at the beginning of the algorithm and is reduced with time to control the search accuracy. \(Rij(t)\) is the Euclidian distance between two agents \(i\) and \(j\). Using the 2nd Newton law of motion, the total gravitational acceleration of the \(i\)-th agent, can be calculated as in eq. 8.9; where \(randj\) is a uniform random number within the interval [0,1], considered to add some stochastic behavior to the acceleration. Now, the velocity and position of the agents are updated as in eq.10, 11 [29]. Where \(i\) \(rand_i\) is another uniform random number in the
interval [0,1]. To prevent the particles go out of the search space, the positions are bounded to the limits of each variable. To perform a good compromise between exploration and exploitation, has proposed to reduce the number of attractive agents in eq. 6 with lapse of time. Therefore, in GSA only the \( k_{\text{best}}(t) \) of the moving agents will attract the others. \( k_{\text{best}}(t) \) is a decreasing function of time, with the initial value \( k_0 = n_{\text{mass}} \). At the end there will be just one agent applying force to the others.

\[
m_i(t) = \frac{f_{i(t)} - \text{worst}_{t}}{\text{best}_{t}} - \text{worst}_{t}, \quad i = 1, 2, 3, \ldots n_{\text{mass}}
\]

\[
M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{n_{\text{mass}}} m_j(t)} \quad 0 \leq M_i(t) \leq 1
\]

worst(t) = \( n_{\text{mass}} \max \{f_{i(t)}\} \)

best(t) = \( n_{\text{mass}} \min \{f_{i(t)}\} \)

\[
f_{ij}(t) = G(t) \frac{M_i(t) M_j(t)}{R_{ij}(t) + \varepsilon} (x_j - x_i)
\]

\[
a_i(t) = G(t) \sum_{j=1}^{n_{\text{mass}}} \text{rand} \frac{M_j(t)}{R_{ij}(t) + \varepsilon} (x_j - x_i)
\]

\[
v_i(t + 1) = \text{rand} v_i(t) + a_i(t)
\]

\[
x_i(t + 1) = x_i(t) + v_i(t + 1)
\]

### 3. The Solution of the problem

The weight of the three phase core form oil transformer is determined first by the classical method which was described in [2] previously. This method is a method in which known equations have been calculated with some assumptions to find the transformer weight. In the second method, the Heuristic Search Algorithms, PSO, SA and GSA are suggested. PSO is such a method that is searching for the optimum in the search space and uses the population which holds the possible solutions for the function to be optimized.

#### 3.1 The Solution of the problem by classical method

The weight, calculated by the classical method was described in [2]. In that classical method, \( C \); transformer iron cross section convenience value, and \( s \); current density, were accepted as constants. The results were evaluated regarding to these accepted values. The iron and copper parts of a transformer are the two basic parameters affecting the weight of a transformer. The total weight \( (G_T) \) (kg) can be expressed as the sum of total iron weight \( (G_{fe}) \) (kg) and total copper weight \( (G_{cu}) \) (kg). \( G_T \) is the objective function of the PSO algorithm. The total iron weight can be expressed as the yoke weight \( (G_{feb}) \) and the three legs \( (G_{feb}) \) weight. The total copper weight is the sum of the total primary winding weight \( (G_{cu1}) \) and the total of secondary winding weight \( (G_{cu2}) \). These values were given in eq. 12-18.

\[
G_{cu} = G_{cu1} + G_{cu2}
\]

\[
G_{fe} = G_{feb} + G_{fej}
\]

\[
G_{T} = G_{cu} + G_{fe}
\]

\[
G_{cu1}, G_{cu2}, G_{feb}, G_{fej}
\] are calculated as follows:

\[
G_{cu1} = 3.10^{-5} \delta_{cu}.w_1.q_1.L_{m1}
\]

\[
G_{cu2} = 3.10^{-5} \delta_{cu}.w_2.q_2.L_{m2}
\]

\[
G_{feb} = 3.10^{-3} \delta_{fe}.q_{fe}.L_s
\]

\[
G_{fej} = 3.10^{-3} \delta_{fe}.q_{fej}.2(2M+0.8D)
\]

\( \delta_{cu}, \delta_{fe} \) are the specific copper and iron weight respectively. \( w_1, w_2 \) are the primary and secondary turns of the windings respectively. \( L_{m1} \) (cm) , \( L_{m2} \) (cm) are the average length of the primary and secondary windings respectively. \( q_1 \) (mm²), \( q_2 \) (mm²), \( q_{fe} \) (cm²), \( q_{fej} \) (cm²) are the primary winding cross section, secondary winding cross section, iron cross section from the leg and the iron cross section from the upper-lower part of the core;

\[
q_{fe} = C. \sqrt{\frac{10^{13}}{3f}}
\]

\[
q_{fej} = 1.1 \cdot q_{fe}
\]

\[
w_1 = \frac{U_1}{\sqrt{5.444.f.0.10^{-8}}}
\]

\[
w_2 = \frac{U_2}{\sqrt{5.444.f.0.10^{-8}}}
\]

\[
L_{m1} = \pi.D_{m1}
\]

\[
L_{m2} = \pi.D_{m2}
\]
\[ D_{m2} = D + 2(\Delta_2 + \delta_2) + a_2 \] (27)

\[ D_{m1} = D_{m2} + a_2 + 2(\Delta_1 + \delta_1) + a_1 \] (28)

\( D_{m1} \) (cm), \( D_{m2} \) (cm) are the average cross section of the primary and secondary windings respectively. \( U_1 \), \( U_2 \) (V) are the primary and secondary voltage respectively and \( f \) is the frequency. These values were given in eq. 19-28.

\[ \theta = q_{fe} \cdot B \] (29)

\( \theta \) (Maxwell) is the magnetic flux and \( B \) (Tesla) is the flux density. These values were given in equation eq. 29.

\( C \) is the transformer iron cross section convenience value and is considered as the first variable for PSO. \( 4 \leq C \leq 6 \) \( s \) is the current density value and is considered as the second variable for PSO. \( 2.2 \leq s \leq 3.5 \)

\( \Delta_1, \delta_1 \) (cm) are the primary insulator coil thickness and oil canal respectively. \( \Delta_2, \delta_2 \) (cm) are the secondary insulator coil thickness and oil canal respectively. \( a_1, a_2 \) (cm) are the thickness of the primary and secondary windings. \( I_1, I_2 \) (A) are the primary and secondary current, \( S \) (VA) is the complex power that was given eq. 30.

\[ S = \sqrt{3} \cdot U \cdot I \] (30)

\( M \) (cm) is the length between the axis of the legs, \( D \) (cm) is the transformer core diameter, \( L_s \) (cm) is the height of the window, \( a \) (cm) is the width of the window and can be calculated as eq. 31, 34.

\[ M = 0.851 \cdot D + a \] (31)

\[ D = 2 \sqrt{\frac{q_{fe}}{0.677, \pi}} \] (32)

\[ L_s = \frac{2 \cdot w_1 \cdot I_1}{A_s} \] (33)

\[ a = \frac{4 \cdot w_1 \cdot q_{11}}{100 \cdot k_{cu} \cdot L_s} \] (34)

\( k_{cu} \) is the window copper fill factor. \( A_s \) (A/cm) is the specific ampere turn. Efficiency \( \eta \) can be calculated as eq.35.

\[ \eta = \frac{S}{S + p_k} \] (35)

\( p_c \) is the total copper loss and \( p_f \) is the total iron loss. The sum of these are \( p_k \) (watt). This values was given in eq. 36;

\[ p_k = p_c + p_f \] (36)

The copper loss of the primary winding is \( p_{c1} \) whereas the copper loss of the secondary winding is \( p_{c2} \), the iron loss of the yoke is \( p_{f1} \) and the iron loss of the legs is represented as \( p_{f2} \). \( r_1, r_2 \) are the primary and secondary winding resistances respectively. Synthesis of these values were given eq. 37-43;

\[ p_{c1} = 3.1^2 \cdot r_1 \] (37)

\[ p_{c2} = 3.1^2 \cdot r_2 \] (38)

\[ p_{f1} = p_{10} \cdot \xi_2 \left( \frac{B_j}{10000} \right)^2 \cdot G_{f1} \] (39)

\[ p_{f2} = p_{10} \cdot \xi_2 \left( \frac{B_j}{10000} \right)^2 \cdot G_{f2} \] (40)

\[ p_c = p_{c1} + p_{c2} \] (41)

\[ p_f = p_{f1} + p_{f2} \] (42)
P_{10} is an additional loss factor whereas \( \xi_2 \) is the loss factor. \( r_1 \) and \( r_2 \) are the primary and secondary winding resistances respectively.

\[
B_i = B / 1.2
\] (43)

The weight of a 100 kVA three phase core form oil transformer founded by classical method was 432 (kg). The values for this result is given in table 1.

3.2 The Solution of the problem by PSO, SA and GSA
In this study, transformer iron cross section convenience value, and \( s \), current density have been taken as variables whereas in the classical method, \( C \) and \( s \), were accepted as constants. The total weight \( G_T \) is the objective function. The minimum values were found with the objective function. This function is obtained as a function of two variables \( C \) and \( s \) using the above equations. The circumstances where the minimum values were considered as appropriately functional are shown in equation 2. However, the copper and iron weight equations are obtained according to references [2]. If there are constraints, those constraints are added to the fitness function as the penalty function to constrain fitness function values. \( C \) (cm\(^2\cdot\text{joule}^{-1/2}\)) is the transformer iron cross section convenience value and \( s \) is the current density value are considered as the first and second variable for PSO. The commonly accepted boundary values for \( C \) and \( s \) of three phase core type transformers are \( 4 \leq C < 6; \ 2.2 < s < 3.5 \). These boundaries are accepted in this study. Efficiency and \( L_s/a \) are constraints, those constraints are added to the fitness function as the penalty function to constrain fitness function values. In this study, since solutions are evaluated for constant weight function values, the efficiency and \( L_s/a \) must be constant as general usage. This situations has been taken as constraints.

First constraint: \( 0.9 < \text{efficiency} < 1 \)
Second constraint: \( 2 < L_s/a < 4.5 \)

<table>
<thead>
<tr>
<th>Table 1. The values of a three phase core type oil transformer using classical method.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>WEIGHT OPTIMIZATION RESULTS OF A CORE FORM OIL TRANSFORMER BY CLASSICAL METHOT</strong></td>
</tr>
<tr>
<td><strong>Variables and Other Values</strong></td>
</tr>
<tr>
<td>Iron cross section convenience value</td>
</tr>
<tr>
<td>Current density value</td>
</tr>
<tr>
<td>Window height</td>
</tr>
<tr>
<td>Window width</td>
</tr>
<tr>
<td>Primary winding cross section</td>
</tr>
<tr>
<td>Secondary winding cross section</td>
</tr>
<tr>
<td>Transformer core diameter</td>
</tr>
<tr>
<td>Primary winding turn</td>
</tr>
<tr>
<td>Secondary winding turn</td>
</tr>
<tr>
<td>Iron cross section</td>
</tr>
<tr>
<td>Primary winding weight</td>
</tr>
<tr>
<td>Secondary winding weight</td>
</tr>
<tr>
<td>The Three legs weight of the Transformer</td>
</tr>
<tr>
<td>The yoke weight of the Transformer</td>
</tr>
<tr>
<td><strong>Total Weight of the Transformer</strong></td>
</tr>
<tr>
<td><strong>Efficiency</strong></td>
</tr>
</tbody>
</table>

In this study a weight optimization of a 100 kVA three phase core form oil transformer has been done by using heuristic search algorithms (PSO,SA and GSA). The values taken for this study are:

\[
U_1=220V, \ U_2=110V, \ F=50H, \ B=11000 \ \text{Gauss}, \ T=75^\circ, \ A_s=387 \ \text{A/cm}, \ K_{ou}=0.19, \ \xi_2=1.15, \ p_{10}=1.3, \ \Delta_1+\delta_1=1.2cm, \ \Delta_2+\delta_2=0.4 \ \text{cm}. \]

The resulting objective equation \( F_1 \) and efficiency are given eq. 44, 45 that are expressed as a function of \( C \) and \( s \);
F₁(C,s)=Gₜ     \hspace{1cm} (44)

Efficiency;

\[ \eta = \frac{100000}{100000 + p_k} \] \hspace{1cm} (45)

In SA the initial value for T is chosen as $1*10^{19}$. The last value for T is chosen as $T<0.001$. At every Temperature the iteration number has been taken as 10. As result at every temperature the best value has been evaluated. In GSA optimization the number of mass has been taken as 50 and the number of iterations has been taken as 250. 50 population and 70 iteration has been done by PSO and the values for C1, C2, w are taken as 2, 2.1, 0.73 respectively. Solution of the PSO, SA, GSA algorithms were used to obtain the minimum weight values of the three phase core form oil transformer. Figure 3, 4, 5 illustrates the transformer total weight fitness functions values with PSO, SA and GSA respectively. Figure 6 illustrates the Transformer efficiency fitness functions values with PSO. The weight of a 100 kVA three phase core form oil transformer, founded by using PSO, SA and GS are 359.07(kg), 359.08(kg), 359.09(kg) respectively. The values evaluated are given in table 2.
4. Conclusions

In this study a weight optimization of a 100 kVA three phase core form oil transformer has been done by using Heuristic Search Algorithms such as particle swarm optimization (PSO) method, Simulated Annealing (SA) and Gravitational Search Algorithm (GSA). In table 3, the results are compared with the results of the classical method. The total weight which has been founded by using Heuristic Search Algorithms are less than the weight which is founded with the classical method. The weight of a 100 kVA three phase core form oil transformer, founded by using PSO, SA and GS are 359,07(kg), 359,08(kg), 359,09(kg) respectively whereas the weight founded by classical method was 432(kg). The results show that heuristic search algorithms which can directly reach the minimum values are an alternative and better way to estimate the minimum weight values.

Fig. 5. Transformer total weight fitness functions values with GSA.

Fig. 6. Transformer efficiency fitness functions values with PSO.
### Table 2. The results evaluated of a three phase core type transformer by PSO, SA and GSA

<table>
<thead>
<tr>
<th>VARIABLES AND OTHER VALUES</th>
<th>SYMBOL</th>
<th>UNIT</th>
<th>PSO</th>
<th>SA</th>
<th>GSA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iron cross section convenience value</td>
<td>C</td>
<td>cm²/joule⁻¹/₂</td>
<td>4.12</td>
<td>4.0935</td>
<td>4.1658</td>
</tr>
<tr>
<td>Current density value</td>
<td>s</td>
<td>A/mm²</td>
<td>3.09</td>
<td>3.0868</td>
<td>3.1158</td>
</tr>
<tr>
<td>Window height</td>
<td>Ls</td>
<td>mm</td>
<td>540.2</td>
<td>544.16</td>
<td>532.883</td>
</tr>
<tr>
<td>Window width</td>
<td>a</td>
<td>cm</td>
<td>13.14</td>
<td>13.1976</td>
<td>13.0501</td>
</tr>
<tr>
<td>Primary winding cross section</td>
<td>q₁</td>
<td>mm²</td>
<td>0.93</td>
<td>0.9282</td>
<td>0.9321</td>
</tr>
<tr>
<td>Secondary winding cross section</td>
<td>q₂</td>
<td>mm²</td>
<td>80.67</td>
<td>81.0561</td>
<td>80.0015</td>
</tr>
<tr>
<td>Transformer core diameter</td>
<td>D</td>
<td>cm</td>
<td>14.14</td>
<td>14.1102</td>
<td>14.2131</td>
</tr>
<tr>
<td>Primary winding turn</td>
<td>w₁</td>
<td>turn</td>
<td>3621</td>
<td>3622</td>
<td>3624</td>
</tr>
<tr>
<td>Secondary winding turn</td>
<td>w₂</td>
<td>turn</td>
<td>42</td>
<td>42</td>
<td>42</td>
</tr>
<tr>
<td>Iron cross section</td>
<td>qfe</td>
<td>cm²</td>
<td>79.97</td>
<td>80.0126</td>
<td>79.6548</td>
</tr>
<tr>
<td>Primary winding weight</td>
<td>Gcu₁</td>
<td>kg</td>
<td>62.79</td>
<td>63.548</td>
<td>61.7586</td>
</tr>
<tr>
<td>Secondary winding weight</td>
<td>Gcu₂</td>
<td>kg</td>
<td>45.65</td>
<td>46.0417</td>
<td>44.9939</td>
</tr>
<tr>
<td>The Three legs weight of the Transformer</td>
<td>Gfeb</td>
<td>kg</td>
<td>131</td>
<td>131</td>
<td>131</td>
</tr>
<tr>
<td>The yoke weight of the Transformer</td>
<td>Gfej</td>
<td>kg</td>
<td>119.63</td>
<td>118.6871</td>
<td>121.3473</td>
</tr>
<tr>
<td>Total Weight of the Transformer</td>
<td>Gtotal</td>
<td>kg</td>
<td>359.07</td>
<td>359.0837</td>
<td>359.0961</td>
</tr>
<tr>
<td>Efficiency</td>
<td>η</td>
<td>%</td>
<td>97</td>
<td>0.9700</td>
<td>0.9700</td>
</tr>
</tbody>
</table>

### Table 3. The values evaluated of a three phase core type transformer by PSO, SA, GSA compared with classical method

<table>
<thead>
<tr>
<th>VARIABLES AND OTHER VALUES</th>
<th>SYMBOL</th>
<th>UNIT</th>
<th>CLASSICAL METHOD</th>
<th>PSO</th>
<th>SA</th>
<th>GSA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iron cross section convenience value</td>
<td>C</td>
<td>cm²/joule⁻¹/₂</td>
<td>5.6</td>
<td>4.12</td>
<td>4.0935</td>
<td>4.1658</td>
</tr>
<tr>
<td>Current density value</td>
<td>s</td>
<td>A/mm²</td>
<td>2.6</td>
<td>3.09</td>
<td>3.0868</td>
<td>3.1158</td>
</tr>
<tr>
<td>Window height</td>
<td>Ls</td>
<td>mm</td>
<td>401</td>
<td>540.2</td>
<td>544.16</td>
<td>532.883</td>
</tr>
<tr>
<td>Window width</td>
<td>a</td>
<td>cm</td>
<td>15.6</td>
<td>13.14</td>
<td>13.1976</td>
<td>13.0501</td>
</tr>
<tr>
<td>Primary winding cross section</td>
<td>q₁</td>
<td>mm²</td>
<td>1.11</td>
<td>0.93</td>
<td>0.9282</td>
<td>0.9321</td>
</tr>
<tr>
<td>Secondary winding cross section</td>
<td>q₂</td>
<td>mm²</td>
<td>96</td>
<td>80.67</td>
<td>81.0561</td>
<td>80.0015</td>
</tr>
<tr>
<td>Transformer core diameter</td>
<td>D</td>
<td>cm</td>
<td>16.5</td>
<td>14.14</td>
<td>14.1102</td>
<td>14.2131</td>
</tr>
<tr>
<td>Primary winding turn</td>
<td>w₁</td>
<td>turn</td>
<td>2675</td>
<td>3621</td>
<td>3622</td>
<td>3624</td>
</tr>
<tr>
<td>Secondary winding turn</td>
<td>w₂</td>
<td>turn</td>
<td>31</td>
<td>42</td>
<td>42</td>
<td>42</td>
</tr>
<tr>
<td>Iron cross section</td>
<td>qfe</td>
<td>cm²</td>
<td>145</td>
<td>79.97</td>
<td>80.0125</td>
<td>79.6548</td>
</tr>
<tr>
<td>Primary winding weight</td>
<td>Gcu₁</td>
<td>kg</td>
<td>63.1</td>
<td>62.79</td>
<td>63.548</td>
<td>61.7586</td>
</tr>
<tr>
<td>Secondary winding weight</td>
<td>Gcu₂</td>
<td>kg</td>
<td>46.2</td>
<td>45.65</td>
<td>46.0417</td>
<td>44.9939</td>
</tr>
<tr>
<td>The Three legs weight of the Transformer</td>
<td>Gfeb</td>
<td>kg</td>
<td>128.5</td>
<td>131</td>
<td>131</td>
<td>131</td>
</tr>
<tr>
<td>The yoke weight of the Transformer</td>
<td>Gfej</td>
<td>kg</td>
<td>194.5</td>
<td>119.63</td>
<td>118.6871</td>
<td>121.3473</td>
</tr>
<tr>
<td>Total Weight of the Transformer</td>
<td>Gtotal</td>
<td>kg</td>
<td>432</td>
<td>359.07</td>
<td>359.0837</td>
<td>359.0961</td>
</tr>
<tr>
<td>Efficiency</td>
<td>η</td>
<td>%</td>
<td>97</td>
<td>97</td>
<td>0.9700</td>
<td>0.9700</td>
</tr>
</tbody>
</table>
References


