International Journal of Industrial Electronics, Control and Optimization © 2021 IECO
Vol. 4, No. 2, pp. 235-244, April (2021)

Optimal Design of a Brushless DC Motor Aiming at Decreasing Cogging Torque

Mina Salarian [1], Milad Niaz Azari [2,†], Mostafa Hajiaghaei- Keshteli [3]

[1,2] Department of Electrical Engineering, University of Science and Technology of Mazandaran, Behshahr, Iran
[3] Department of Industrial Engineering, University of Science and Technology of Mazandaran, Behshahr, Iran

ABSTRACT

One of the important issues in designing high-performance brushless direct current (BLDC) motors is reducing the cogging torque since it results in mechanical vibration, audible noises, and torque ripples, which adversely impact the performance of the motor, which is awkward high-accuracy applications. This paper proposes an optimum design for BLDC motors aimed at reducing the cogging torque based on the capability of metaheuristics algorithms in finding the optimal solution. For this purpose, a simplified cogging torque equation is used as the objective function whose design variables include air gap length, magnet height, slot height, slot opening, and motor axial length. These are the five most influential parameters of cogging torque. On the other hand, we employ not only the old metaheuristics algorithms like the Genetic Algorithm (GA) and Simulated Annealing (SA) but also more recent algorithms such as Keshtel Algorithm (KA) along with the hybrid ones to benefit from their strength. The simulation is performed in the Matlab package. First, five selected optimization algorithms are applied and the results are investigated. The results of all the algorithms show a significant reduction in the cogging torque. Eventually, the proposed algorithms are compared to one another in terms of their value of cogging torque. The results show the superiority of the KASA algorithm in comparison with the others.

Article Info

Keywords:
Brushless DC motors, Cogging torque, Design, Metaheuristics algorithms, Optimization, Reduction.

Article History:
Received 2020-07-21
Accepted 2020-09-24

NOMENCLATURE

$w'(\theta)$: air gap co-energy
N: order of cogging torque harmonics
T_{CN}: amplitude of the nth harmonic component of cogging frequency
g: air gap length
L: motor axial length
D: stator inner diameter
k_c: carter's coefficient of the air gap that takes into account slot openings
B_g: flux density
t_1: stator slot pitch
l_m: magnet height
h_s: magnet height
k_{ok}: factor of stator slot opening
k_{sk}: factor of stator skew
w_s: slot opening
θ: rotor position angle
Q: number of slots
μ_0: magnetic permeability of free space
α: pole arc to pole pitch ratio
φ_g: air-gap flux

†**Corresponding Author**: miladniazazari@mazust.ac.ir
Tel: +98-01134556000, Fax: +98-01134556004,
Faculty of Electrical and Computer Engineering, University of Science and Technology of Mazandaran, Behshahr, Iran

I. INTRODUCTION

BLDC motors are widely used in various fields from domestic appliances to industrial machinery and speed and position control systems due to their high efficiency, high-density power, and easy speed control. One of the important and inherent drawbacks of BLDC motors is the cogging torque, which results from the interaction of the stator teeth, slots, and permanent magnets (PM) in all PM machines. The cogging torque prevents the rotor to rotate smoothly, so it is a major source of undesirable mechanical vibrations and noises in PM machines. Besides, it is one of the most important factors in torque ripple and can affect the control accuracy. Hence, in high performance and accuracy applications of BLDC motors, it is imperative to reduce cogging torque [1]-[4].

There are different methods for cogging torque reduction, such as physical methods and changing the geometry of the stator and rotor. They include using certain number of auxiliary slots in particular position [5], selecting an optimal number for poles and stator's pitch, optimizing the rotor's pole geometry by experimental results, skewing the stator or the magnetic pole [6], modifying the shape of stator teeth by changing the stator teeth tang span [1], using the eccentric structure of stator teeth [7], and a new manufacturing method with respect to the influence of the stator and rotor stacks' tolerance [2].

Some analytical methods, e.g. computing cogging torque [8],[9], study alterations of cogging torque according to slots number of the pole of the rotor [10] and also making $N_s/GCD(N_s, N_P)$ ≥4 (GCD: great common divisor, N_s: slot number, N_P: pole number) [11]. The sine-shaped PMs can be used to cancel the cogging torque. On the other hand, if the number of slots /pole /phase is equal to 0.25, 0.5, and 1, respectively, the amplitude of the cogging torque can increase in three-phase SPM machines with the Sine+3rd-shaped PM [12].

Due to the nature of the problem, optimization techniques can be very useful in minimizing the cogging torque. For this purpose, the particle swarm optimization (PSO), genetic algorithms, and modified particle swarm optimization are used to find the optimal design [13],[14].

Changing the geometry of the stator and rotor may cause other problems in PM machines. For example, the slanting of PM, which is the most common approach to decreasing the cogging torque, can reduce the flux linkage and increase the unbalanced emf [13]. On the other hand, analytical methods have been used in the literature for cogging torque-related computations. However, analytical methods should be used along with other methods, such as optimization or FEA, for mitigating the cogging torque. In the optimization method used in this paper, the cogging torque's influential parameters are changed over a specified range. Not only does this change lower the cogging torque as much as possible but it also does not lead to manufacturing problems in machines. In this method, previous studies have applied two or three design variables. But, the objective function in this paper contains five influential parameters of cogging torque and uses them as design variables.

The use of an objective function, which includes the five most influential parameters of the cogging torque, along with the use of new and hybrid algorithms, as well as comparing the results with well-known algorithms such as GA and SA, is a new work that has not been done before. This paper presents an optimal design of BLDC motors with the objective of minimizing the cogging torque by employing metaheuristics algorithms such as Genetic Algorithm (GA), Simulated Annealing (SA), Keshtel Algorithm (KA), and a hybrid of these algorithms. The algorithms are used in the Matlab package. Every algorithm is run individually and after all parameters are investigated, the results are compared. The analysis shows that the cogging torque can be minimized to a certain level and the KASA algorithm provides a more optimal solution than others.

For clarity, the paper is organized as follows. Section 2 describes the problem and objective function. Section 3 presents the proposed metaheuristics algorithms for optimizing the cogging torque and how they are implemented. Section 4 discusses the parameters setting used for optimizing, the design variables, and the constraints. Section 5 examined the results of the simulation of the algorithms and compares them. Also the obtained result were compared with the result of other similar methods. Finally, the concluding points and the recommendations for future studies are presented in Section 6.

II. COGGING TORQUE EXPLANATION AND THE OBJECTIVE FUNCTION

The BLDC motor is one of the most fascinating choices for industrial drives because of its attractive properties such as high torque density, reliable structure, and low maintenance requirements. But, the cogging torque components affect the BLDC motor's output. Hence, the reduction of cogging torque is an important issue in designing a BLDC motor [1]. This section defines the cogging torque equation and expresses the objective function aimed to minimize this torque. The design variable is introduced at the end of this section.

A. *Cogging Torque Explanation*

Cogging torque is a kind of reluctance torque that is generated due to the tendency to reduce the reluctance between the rotor PMs and the stator teeth. The cogging torque can be minimized by reducing the change in air gap reluctance or decreasing the flux of the air gap [1],[15]. Reluctance R is given by Equ. (1).

$$R = \frac{l}{\mu A} \tag{1}$$

To determine the flux path length l, in addition to the length of the air gap g, the hieght of the stator slots h_s should be considered too because h_s is the length of the flux that enters the slot opening. So, the flux path length is considered in the range of $g < l < g + h_s$

Based on changes in the reluctance, the cogging torque can be defined by Equ. (2) as below [1],[16].

$$T_{cog} = -\frac{1}{2}\varphi_g^2\frac{dR}{d\theta} \tag{2}$$

B. *The Objective Function*

Equ. (3) is a simplified cogging torque equation of PM machines, which is used as the objective function in this paper [17].

$$T_{cog} \approx -\frac{Lg}{2\mu_0}\frac{D}{2}K_1(X) \tag{3}$$

In which $K_1(X)$ is obtained by Equ. (4) and is a function of the distance X. In addition, Equ. (5) is used to express A.

$$k_1(x) = -\frac{4}{k_c}AB_g^2\sin[\alpha(X + \frac{a+b}{2})]\sin\left(\alpha\frac{a-b}{2}\right) \tag{4}$$

$$\mathrm{A} = 2\gamma\frac{g}{t_1}{k_{ok=1}}^2{k_{sk=1}}^2 \tag{5}$$

In which α can be defined by Equ. (6) as follows.

$$\alpha = \frac{2\pi}{t_1} \tag{6}$$

Equs. (7) and (8) determine a and b, respectively, but to express b it is necessary to define c_t by Eq. (9).

$$a = 0.5t_1 \tag{7}$$

$$b = 0.5\,w_s + c_t \tag{8}$$

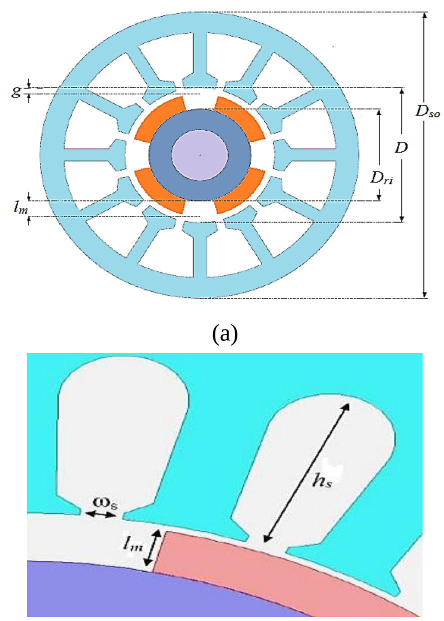
$$c_t = t_1 - w_s \tag{9}$$

In which γ can be defined by Equ. (10) as below [17].

$$\gamma = \frac{4}{\pi}\left[\frac{w_s}{2g}\tan^{-1}(\frac{w_s}{2g}) - \ln\sqrt{1+\frac{w_s}{2g}}^{\,2}\right] \tag{10}$$

C. *Design Variables*

The cogging torque can be minimized by selecting the most suitable value for the motor design parameters. Thus, selecting design variables to optimize is an important part of cogging torque reduction. Because of the transfer of the main energy in the air gap, not only are air gap and slot opening important variables but magnet length is also an important design variable to be optimized for cogging torque reduction [13]. Here, α, l_m, w_s, h_s and g are selected as the motor design variables. For better understanding, the geometric representation is shown in Fig. 1 [16].



(b)

Fig. 1. Motor geometry: (a) parameters of the motor and (b) slot view.

III. SOLUTION APPROACH

A. *Genetic Algorithm (GA)*

GA is a population-based technique and a famous evolutionary algorithm. In general, GA has four main steps, i.e. reproduction, selection mechanism, mutation, and crossover. Mutation and crossover are two search operators of GA that not only produce a new generation but also search for the potential area of problems.

Moreover, their focus is on intensification and diversification, respectively. Due to the nature of chromosomes, GA chromosomes is used as an array of variables. On the other hand, mutation and crossover alter chromosomes to create a set of new solutions and solve the huge and complex problems [18],[19]. The main parameters of GA to determine the general behavior are expressed as follows: the number of population is 10, the probability of crossover is 0.3, and the probability of mutation is 0.3.

B. Simulated Annealing (SA)

SA is based on the process of annealing metals, which is a kind of local search optimization techniques. Besides, SA is one of the well-known single solution algorithms. The SA process is as follows. The first step is choosing a solution randomly which is presented as the best solution. At the next step, an initial temperature must be fixed. Then, a neighbor solution with a near close position to the best solution is generated during a process and it is evaluated. If it is conclusively or by a probability better than the best solution, the new solution replaces the previous one. Afterward, the best solution and the temperature are updated. This cycle is continued until the updated temperature becomes greater than the threshold [20]. The main parameters of SA to determine the general behavior are that the initial temperature is 50 and α is 0.97.

C. Keshtel Algorithm (KA)

KA was introduced by Hajiaghaei-Keshteli and Aminnayeri [21] inspired by how a dabbling duck whose common name is Keshtel in the north of Iran is fed. Generally, Keshtels work together as a group while feeding and when a Keshtel finds a rich source of food, the side Keshtels approach it and seek food by swirling in a circle. As soon as Keshtel finds food, it sinks its head under water and rotates its body around its beak like a moving circle. The other nearby Keshtels approach it and rotate in the same direction of the first Keshtel while it is in the center of the circle. They continue to rotate until no food remains. Again when a Keshtel finds a good source of food, this process is repeated. An important result of this algorithm is a solution with higher quality; a result that satisfies us to use it for our problem.

In order to identify the counterpart of the proposed algorithm, initial Keshtels are divided into three categories (i.e. N1, N2, and N3) to search the feasible space. N1 includes lucky Keshtels – the ones that have found good foods in their first try. In contrary to N1, N3 contains the worst solutions, which are generated randomly in each iteration. If better food is found for each lucky Keshtel, a new lucky will be replaced. Otherwise swirling will be continued. For N2 type Keshtels, they move between two lucky Keshtels [22]. The flowchart of the KA algorithm is shown in Fig. 2.

D. Hybrid of KA and SA (KASA)

Hybridization is considered as combining two search algorithms to find the best solution for a given problem. The KASA algorithm is based on KA. Thus, KA is the main loop and the role of SA is to help the algorithm for an intelligent probe.

Furthermore, the properties of the exploration step are investigated by the SA loop. Generally, this algorithm is used to enhance the strength of intensification and diversification steps. Using SA as a sub-loop in per iteration can modify the randomization of the procedure. Accordingly, in KA, each Keshtel is evaluated and compared with the previous one and the better Keshtel replaces but the opposite of KA in SA uses a temperature rate controller to accept the rejected solutions during the iterations. The flowchart of the KASA algorithm is shown in Fig. 3. Also, more details of this metaheuristic KASA pseudo-code are displayed in Fig. 4 [23].

E. Hybrid of GA and SA (GASA)

GASA is a hybrid optimization strategy created by combining the GA and the SA. Two weaknesses of the SA algorithm are the limiting behavior of convergence and computation time. The hybrid approach is an efficient way to overcome these problems. In addition, it can help improve SA performance. Adding this sampling to the GA not only can help avoid the search getting stuck in a local optimum solution but it can also extend the search scope [24]. SA can locally optimize each individual solution. On the other hand, GA works on the optimal solution in a global sense. These two powerful properties are the motivation for the combination of GA and SA.

The GASA algorithm has two phases. The first phase is performed by GA, which is a global search. The second phase is named the local search phase, which is applied by SA. The pseudo-code of GASA is explained in Fig. 5. Also, the flowchart of the GASA algorithm is depicted in Fig. 6 [25].

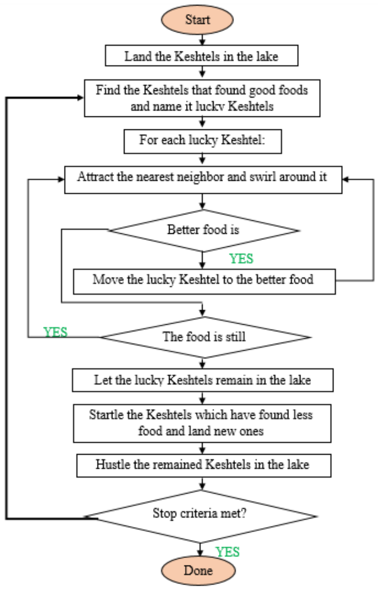


Fig. 2. The flowchart of the KA algorithm

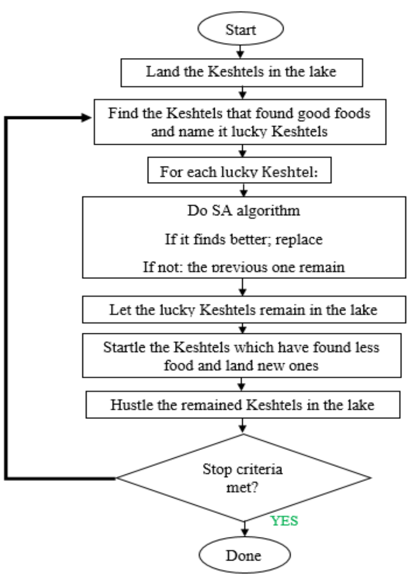


Fig. 3. The flowchart of the KASA algorithm

IV. PARAMETERS SETTING

A notable section in implementing all algorithms is the parameter-setting. The model of the motor used in this paper is a 4 pole, 220 v, and 550 w motor. The characteristics of the BLDC motor that we require to optimize are presented in Table I.

The constraints considered as ranges of design variables are defined in Table II.

```
Initialize Keshtels population.
Calculate the fitness and sort them in three types: M1, M2 and M3
Y*=the best solution.
while (z< maximum number of iteration)
    for each M1
    Calculate the distance between this lucky Keshtel and all Keshtels.
Select the closest neighbor.
    while (S< maximum number of swirling)
    Do the swirling.
    if the fitness of this new position is better than prior
    Update this lucky Keshtel.
      break
  endif
  S=S+1
  endwhile
endfor
for each M2
  Move the Keshtel between the two lucky Keshtels.
endfor
for each M3
    Create a neighbor from the one of luck Keshtels randomly.
    if the function value of the new solution is better than prior
      Replace the new solution as old solution.
  else
  Calculate δ, δ = |f_old − f_new|.
          if rand< exp(−δ/T )
              Replace the new solution.
          endif
        endif
      endfor
      Merge the M1, M2 and M3
  Sort the Keshtels and form M1, M2 and M3 for next iteration and Update Pareto optimal solutions.
      Update the Y* if there is better solution.
      z=z+1
  end while
  return Y*
```

Fig 4. The pseudo-code of KASA

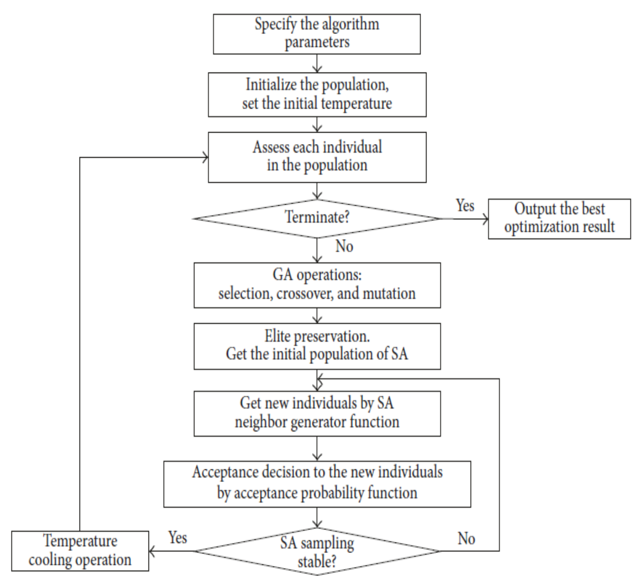


Fig. 5. The flowchart of the GASA algorithm

V. SIMULATION

Using the MATLAB software, the GA, SA, KA, KASA, and GASA algorithms are applied to optimize the objective function according to the specifications and constraints. In all algorithms, the iteration is set on 200.

```
MAX_GENS: max number of generations
NUM_GENES: number of genes, POP_SIZE: population size
RANDOM: decimal between 0 and 1, T: temperature
Pnew: new placement, Pold: old placement, Pcrossover: probability of
crossover rate
Pmutation: probability of mutation rate, Plocal: probability of local
improvement rate
Preserve: the percent of population reserved in the generation
e: constant of 2.732
begin
        initialize_population ()
        while (generation < MAX_GENS) do
            evaluate_population_fitness ();
            reproduce_population (Preserve);
            for i = 1 to POP_SIZE/2 do
                crossover (Pcrossover);
            for j = 1 to NUM_GENES do
                mutate (Pmutation);
            for i = 1 to POP_SIZE do
                local_improvement (Plocal);
            elitism();
        end while
        select_the_best_one();
        T = set_temperature();
/* following algorithm is pseudo-code of SA*/
        while (exit_criterion() == FALSE) do
            while (inner_criterion() == FALSE) do
                Pnew = generate_movement ()
                ΔC = C (Pnew)−C (Pold);
                RANDOM = generate_number();
                if (RANDOM < e exp (−ΔC/T))
                    Pold = Pnew;
            end while
        end while
end algorithm
```

Fig. 6. The pseudo-code of GASA

TABLE I
The Specifications of a Sample BLDC Motor

Parameter	Value	Definition
T_C	719.1 mN.m	Cogging torque
P	4	Number of pole
S	24	Number of slot
D	75 mm	Stator inner diameter
D_{ri}	26 mm	Rotor inner diameter
D_{so}	120 mm	Stator outer diameter
D_{ro}	74 mm	Rotor outer diameter
G	1 mm	Air gap length
L	65 mm	Axial length
L_m	3.5 mm	Magnet height
α	0.7	Pole arc to pole pitch ratio
φ	3 mwb	Magnetic flux under pole
h_s	8.2 mm	Slot height
w_s	2.5 mm	Slot opening

Due to the random nature of metaheuristic algorithms, the values of the parameters and cogging torque show the average value of each of them after 20 times of the algorithm implementation. Table III, IV and V present the design parameters required to implement the SA, GA and KA algorithms, respectively.

TABLE II
The Ranges of the Design Variable

Variable	Min	Max
α	0.65	0.85
l_m (mm)	2	5
g (mm)	0.5	2
w_s (mm)	2	4
h_s (mm)	7	10

TABLE III
Design Parameter for SA

Parameters	Value
T_0	50
Alpha	0.97
Iterations	200

TABLE IV
Design Parameter for GA

Parameters	Value
No of variables	5
Population	10
Iterations	200
Probability of crossover (P_c)	0.3
Probability of mutation (P_m)	0.3

TABLE V
Design Parameter for KA

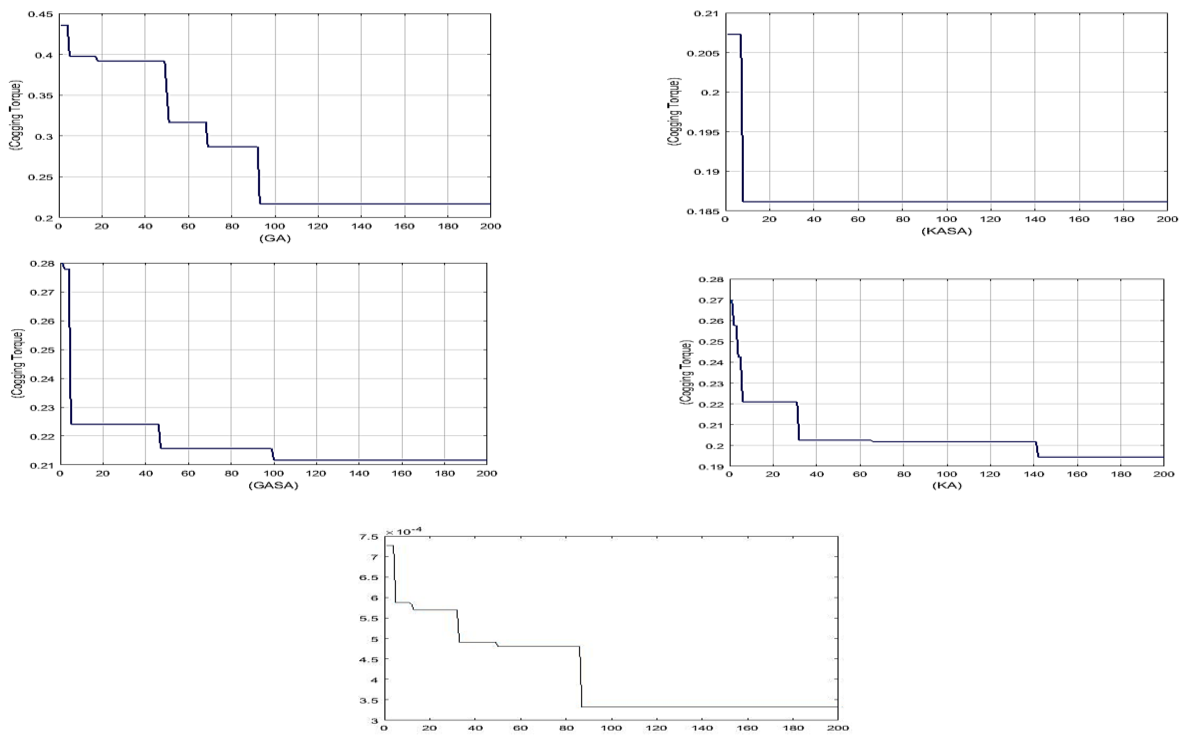
Parameters	Value
No of variables	5
Population	150
Iterations	200
Condition stop swirling	1
No of swirling	40
Percentage of N_1 population	0.5
Percentage of N_2 population	0.5

A. Simulation Results

The optimized values for parameters are shown in Table VI. According to Table VI, all algorithms able to reduce the cogging torque compared to the cogging torque of sample motor. However the best solution is presented by the KASA algorithm that has the minimum value of the cogging torque.

TABLE VI
The Optimized Values of the Algorithms' Parameters

Parameter	GA	SA	KA	KASA	GASA	Sample motor
α	0.8194	0.7071	0.8474	0.8470	0.7810	0.7
l_m (mm)	4.2078	4.3757	4.2303	4.2358	4.0550	3.5
g (mm)	1.9616	1.6757	1.9389	1.9859	1.8970	1
w_s (mm)	2.0609	2.0604	2.0620	2.0261	2.0352	2.5
h_s (mm)	9.1328	9.3536	9.7590	9.9384	9.7935	8.2
T_c N.m	0.2737	0.3334	0.2097	0.1957	0.2979	0.7191



(SA)

Fig. 7. Graph of cogging torque changes during 200 iteration for GA, SA, KA, GASA and KASA algorithms

Besides, all variable parameters are in their specific range. Also, the maximum value belongs to the SA algorithm. As shown in Table VI, the cogging torque is not completely eliminated, but it can be reduced significantly. Fig. 7 displays graph of the cogging torque changes during 200 iteration for GA, SA, KA, GASA and KASA algorithms and it can be seen that the KASA algorithm converges to best solution faster than the others. The cogging torque can be minimized in the design of the BLDC motor by using the optimal values of the parameters obtained. For more explanation, the results of optimization are compared in Fig. 8, and Fig. 9 shows the results of the design variables by the optimization algorithms. The computational time of the proposed algorithms presents in Table VII that KASA and SA have maximum and minimum time. This result was predictable.

B. *Comparison with Some Similar Methods*

In Ref. [13] PSO and GA are applied to optimize a BLDC motor with the aim of minimizing the cogging torque.

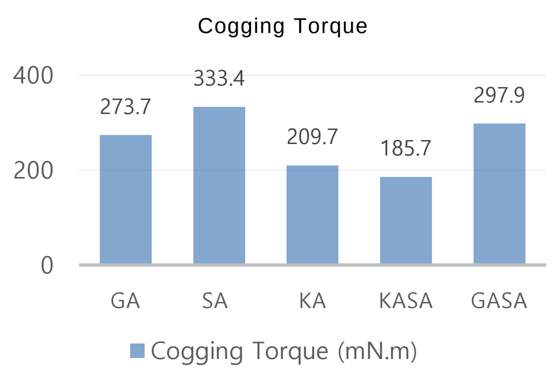


Fig. 8. The comparison of the cogging torque optimization results by the GA, SA and KA, GASA and KASA algorithms.

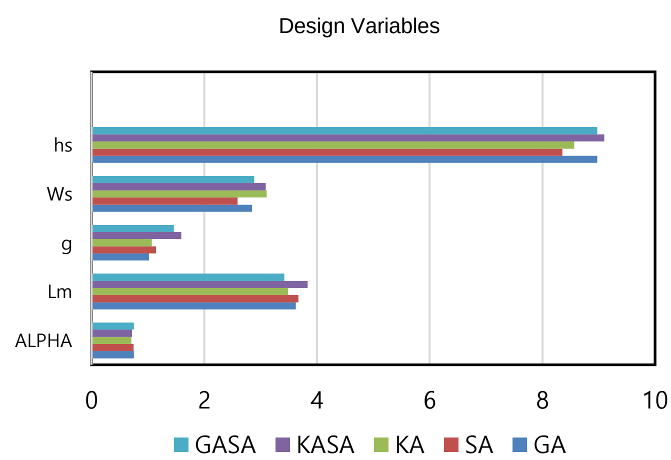


Fig. 9. The results of the design variables by the GA, SA, KA, GASA, and KASA algorithms

TABLE VII
The Computational Time of the Proposed Algorithms

	GA	SA	KA	GASA	KASA
Elapsed Time (Second)	1.442	0.285	13.253	5.180	135. 557

In this procedure, slot opening, air gap length and magnet length are chosen as design variables. The result shows reduction of cogging torque from 0.0510 to 0.025 using PSO and 0.0510 Nm to 0.04 Nm using GA.
Ref. [26] uses GA as an optimization tool to decrease the cogging torque in permanent magnet synchronous motor. Pole span, magnet thickness and shape of the magnets are design variables. In this method the cogging torque reduces from 0.368 Nm to 0.0169 Nm.

In this paper, the results show the superiority of the KASA algorithm in comparison with the other algorithms. Here pole arc to pole pitch ratio, slot height, air gap length, slot opening and magnet length are selected as the motor design variables. KASA reduces the cogging torque from 0.7191 Nm to 0.1957 Nm.
For comparison these three methods, the value of cogging torque reduction is defined as indice and it can be obtained by subtraction of the optimized motor cogging torque from the sample motor cogging torque. Table VIII shows this comparison and according to Table VIII KASA has maximum reduction of the cogging torque.

TABLE VIII
Comparison of Three Methods through Reduction of Cogging Torque Value

		Ref [13]	Ref [27]	This paper
Reduction of Cogging Torque (N.m)	PSO	0.026		
	GA	0.011	0.3518	
	KASA			0.5234

VI. CONCLUSION

The cogging torque has a detrimental effect on the performance of the PM machines, so the reduction of the cogging torque is an important issue in BLDC motors to gain high performance.

This paper proposed an optimum design of the BLDC motors with the aim of reducing the cogging torque by employing metaheuristics algorithms such as GA, SA, KA, KASA, and GASA. The analysis shows:

- All selected algorithms able to decrease the cogging torque value
- All variable parameters are in their specific range for all algorithms so they can be used for manufacturing.
- The KASA algorithm presents the minimum cogging torque value
- The SA algorithm presents the maximum cogging torque value in comparison with the other algorithms.

In the future works, the authors will offer an optimal solution for the BLDC motor that, in addition to reducing the cogging torque, will also consider the value of the output torque.

REFRENCES

[1] Kamal, D. Amudhavalli and T. Thyagarajan, "Reformed Stator Design of BLDC Motor for Cogging Torque Minimization Using Finite Element Analysis," *In 2018 4th International Conference Electrical Energy Systems (ICEES), IEEE*, Chennai, pp.481-484, 2018.

[2] J. Ou, Y. Liu, R. Qu and M. Doppelbauer, "Experimental and theoretical research on cogging torque of PM synchronous motors considering manufacturing tolerances," *IEEE Trans. Industrial Electronics*, Vol. 65, No. 5, pp. 3772-3783, 2017.

[3] M.N. Azari, M. Samami and S.A. Pahnehkolaei. "Optimal Design of a Brushless DC Motor, by Cuckoo Optimization Algorithm (research note)," *IJE Trans. Applications*, Vol. 30, No. 5, pp. 668-677, 2017.

[4] Hajiaghaei-Keshteli. "An Improved Red Deer Algorithm to Address a Direct Current Brushless Motor Design Problem," *Scientia iranica*, 2019.

[5] D.U. Xiao-bin, H.U.A.N.G. Kai-sheng and H.U.A.N.G. Xin, "Reduction of Cogging Torque of Permanent Magnet Machines Based on Harmonic Analysis," *Small & Special Electrical Machines*, Vol. 47, No. 1, p. 37, 2019.

[6] M. POPESCU, E. TUDOR, S. NICOLAIE, C.I. ILIE, , L. POPOVICI and C. DUMITRU, "Experimental Results Regarding Cogging Torque Reduction for the Permanent Magnet Synchronous Motors PMSM," *In 2019 11th International Symposium on Advanced Topics in Electrical Engineering (ATEE), IEEE*, pp. 1-4, 2019.

[7] K. Wang, Y. Liang, D. Wang and C. Wang, "Cogging torque reduction by eccentric structure of teeth in external rotor permanent magnet synchronous motors," *IET Electric Power Applications*, Vol.13, No. 1, pp. 57-63, 2018.

[8] Ebadi, M. Mardaneh, A. Rahideh and N. Bianchi, "Analytical Energy-Based Approaches for Cogging Torque Calculation in Surface-Mounted PM Motors," *IEEE Transactions on Magnetics*, Vol. 55, No. 5, pp. 1-10, 2019.

[9] L. Jing, J. Chen, Z. Huang and J. Gong "Exact Analytical Method for Air-Gap Main Magnetic Field Computation and Cogging Torque of SMPM Motors," *Progress In Electromagnetics Research*, Vol. 81, pp. 75-84 2019.

[10] L. Jing, J. Gong and Y. Lin, "Analysis and Reduction of Cogging Torque of Line-Start Permanent Magnet Motors," *Progress In Electromagnetics Research*, Vol. 78, pp. 115-124, 2019.

[11] H.Y. Sun and K. Wang, "Effect of Third Harmonic Flux Density on Cogging Torque in Surface-Mounted Permanent Magnet Machines," *IEEE Trans, Industrial Electronics*, Vol. 66, No. 8, pp. 6150-6158, 2018.

[12] K. Wang, and Z.Q. Zhu, "Analysis of Cogging Torque in Surface-Mounted Permanent Magnet Machines with Shaped Magnets," *In Third Harmonic Utilization in Permanent Magnet Machines, Springer, Singapore*, pp. 121-141, 2019.

[13] Kamal, T. Thyagarajan, M. Selvakumari and D. Kalpana "Cogging torque minimization in brushless DC motor using PSO and GA based optimization," *In 2017 Trends in Industrial Measurement and Automation (TIMA), IEEE, Chennai*, pp. 1-5, 2017.

[14] Z. Xue, H. Li, Y. Zhou, N. Ren and W. Wen, "Analytical prediction and optimization of cogging torque in surface-mounted permanent magnet machines with modified particle swarm optimization," *IEEE Trans. Industrial Electronics*, Vol. 64, No. 12, pp. 9795-9805, 2017.

[15] C.Y. Hsiao, S.N. Yeh and J.C. Hwang, "A novel cogging torque simulation method for permanent-magnet synchronous machines," *Energies*, Vol.4, No. 12, pp. 2166–2179, 2011.

[16] R. Ilka, Y. Alinejad-Beromi and H. Yaghobi, "Cogging torque reduction of permanent magnet synchronous motor using multi-objective optimization," *Elsevier B.V. Mathematics and Computers in Simulation*, Vol. 153, pp. 83-95, 2018.

[17] J.F. Gieras, "Analytical approach to cogging torque calculation of PM brushless motors," *IEEE Trans, Industry Applications*, Vol. 40, No.5, pp. 1310-1316, 2004.

[18] Abdi, A.M. Fathollahi-Fard and M. Hajiaghaei-Keshteli, "A set of calibrated metaheuristics to address a closed-loop supply chain network design problem under uncertainty," *International Journal of Systems Science: Operations & Logistics*, pp. 1-18 2019.

[19] H. Moradi CheshmehBeigi and A. Mohamadi, "Torque Ripple Minimization in SRM Based on Advanced Torque Sharing Function Modified by Genetic Algorithm Combined with Fuzzy PSO," *International Journal of Industrial Electronics, Control and Optimization*, Vol. 1, No. 1, pp. 71-80, 2018.

[20] S. Sadeghi-Moghaddam, M. Hajiaghaei-Keshteli and M. Mahmoodjanloo, "New approaches in metaheuristics to solve the fixed charge transportation problem in a fuzzy environment," *Neural Computing and Applications*, Vol. 31, No. 1, pp. 477-497, 2019.

[21] M. Hajaghaei-Keshteli, M. Aminnayeri, M.J.A.S.C., "Solving the integrated scheduling of production and rail transportation problem by Keshtel algorithm," *Elsevier B.V. Applied Soft Computing*, Vol. 25, pp.184-203, 2014.

[22] A.M.F. Fard and M. Hajaghaei-Keshteli, "A tri-level location-allocation model for forward/reverse supply chain," *Elsevier B.V. Applied Soft Computing*, Vol. 62, pp. 328-346,

[23] M. Hajiaghaei-Keshteli and A.M. Fathollahi-Fard, "A set of efficient heuristics and metaheuristics to solve a two-stage stochastic bi-level decision-making model for the distribution network problem," *Elsevier Ltd. Computers & Industrial Engineering*, Vol. 123, pp. 378-395, 2018.

[24] Han, M., Li, P. and Sun, J. "The algorithm for berth scheduling problem by the hybrid optimization strategy GASA", *In 2006 9th International Conference on Control,*

Automation, Robotics and Vision, IEEE, Singapore, pp. 1-4, 2006.

[25] M. Yang, A.E.A. Almaini and P. Wang, "FPGA placement optimization by two-step unified genetic algorithm and simulated annealing algorithm," *Journal of Electronics (China)*, Vol. 23, No. 4, pp. 632-636, 2006.

[26] V. Sarac, "Performance optimization of permanent magnet synchronous motor by cogging torque reduction," *Journal of Electrical Engineering*, Vol. 70, No. 3, pp. 218-226, 2019.

Mina Salarian was born in Nur, Mazandaran, Iran in 1992. She completed her B.Sc. in Electrical Engineering from University of Gilan: Rasht, Iran in 2015. She is currently studying her M.Sc. in Electrical Engineering in field of Power Electronics and Electric Machines at University of Science and Technology of Mazandaran**:** Behshahr, Iran.
Email: minasalaryan@gmail.com

Milad Niaz Azari was born in Babol, Iran, in 1984. He received his B.S degree in electrical engineering from Noshirvani University of Technology, Babol, Iran in 2007. He Graduated in M.S degree in the department of Electrical Engineering, Amir Kabir University of Technology, Tehran, Iran in 2009. Also he graduated in PhD degree in Amirkabir University of Technology, Tehran, Iran in 2013. Since 2014, he has been at University of Science and Technology of Mazandaran, Behshahr, Iran, as an assistant Professor in the department of Electrical Engineering. His areas of interest are electrical machines design and power electronics.
Email: miladniazazari@mazust.ac.ir

Mostafa Hajiaghaei-Keshteli was born raised in Babol, Iran. He earned his B.Sc. from Iran University of Science & Technology, Tehran, Iran (2004); M.Sc. from Iran University of Science & Technology, Tehran, Iran (2006); and Ph.D. from Amirkabir University of Technology (Tehran- Polytechnic) Tehran, Iran (2012); all in Industrial Engineering. He is currently Industrial Engineering an Assistant Professor at University of Science & Technology of Mazandaran, Iran. He has over 10 years of experiences in Business Development, System Analysis, Inventory and Project Magnet. Mostafa also has worked for many corporations in Iran and has held the positions of consulter, planning and project manager and VP. The main focus of his research is in the area of Inventory Control, Supply Chain Network, Transportation and Meta-heuristics.
Email: mostafahaji@mazust.ac.ir