Bat-Inspired Optimization Approach for the Brushless DC Wheel Motor Problem

Teodoro C. Bora[1], Leandro dos S. Coelho[1], Luiz Lebensztajn[2]

[1] Automation and Systems Laboratory, PPGEPS, Pontifical Catholic University of Paraná
Rua Imaculada Conceição 1155, Zip code: 80215-901, Curitiba, PR, Brazil

[2] Laboratório de Eletromagnetismo Aplicado, LMAG-PEA, Escola Politécnica da Universidade de São Paulo
Av. Prof. Luciano Gualberto 158, Zip code: 05508-900, São Paulo, SP, Brazil

E-mails: teodoro.bora@pucpr.br, leandro.coelho@pucpr.br, leb@pea.usp.br

Abstract— This paper presents a metaheuristic algorithm inspired in evolutionary computation and swarm intelligence concepts and fundamentals of echolocation of micro bats. The aim is to optimize the mono and multiobjective optimization problems related to the brushless DC wheel motor problems, which has 5 design parameters and 6 constraints for the mono-objective problem and 2 objectives, 5 design parameters and 5 constraints for multi-objective version. Furthermore, results are compared with other optimization approaches proposed in the recent literature, showing the feasibility of this newly introduced technique to high nonlinear problems in electromagnetics.

I. INTRODUCTION

The Bat Algorithm (BA) is a very interesting approach recently proposed by Yang [1]. It is based in the nature behavior of micro bats when looking for food. These bats use echolocation to guide their search. This feature is idealized and implemented by the BA to optimization problems. This idealized idea of bats is that they generate sound waves with some given frequencies and pulse rates. When it tends to get close to its prey the pulse rate increases while the loudness decreases. BA algorithm mimics this idea, i.e., the design parameters are the bat position and the pray is the objective.

In this way, it is possible to say that the BA tries to merge the main features of two different optimization algorithms: the Particle Swarm Optimization (PSO) and the Simulated Annealing (SA), as shown in [1].This paper not only investigates the BA ability for mono-objective optimization problems, but also proposes a BA extension to multiobjective tasks. A Brushless DC Motor (BLDC) Wheel Optimization Problem is used as benchmark for the mono-objective and multi-objective optimization problem.

II. IMPLEMENTED MONO AND MULTI OBJECTIVES BA

Fig. 1 presents the proposed BA implementation pseudocode for the mono-objective approach. For a prescribed iteration (t), a set of bats ($\{B_1, B_2, ...B_N\}$) is generated. Each bat is defined by your position ($\boldsymbol{x}_i^t$), velocity ($\boldsymbol{v}_i^t$), frequency (F_i), loudness (A_i^t) and the rate of pulse emission (r_i^t). In order to tune the frequency, F_{min} and F_{max} are defined as lower and upper bound emission frequency. $\boldsymbol{\lambda}$ is a bat position, which can be one taken randomly among the best solutions or from the entire population (see Fig.1). Finally, α and γ are parameters to be set before the run. In this work, their values are assigned to 0.9, for both cases. The variable $\boldsymbol{x}_{i,old}^t$ is the same of $\boldsymbol{x}_i^t$ in equation (4), but after the use of (5), it is stored in the same memory position but denoted, mathematically as $\boldsymbol{x}_{i,new}^t$ only to avoid misunderstand.

While (t < max number of iterations)
For $i = 1: N_b$
Generate B_{new} using (2), (3) and (4)
If $rand > r_{new}$
Select one among the best solutions and generate a local solution around this one, using (5)
Else
Select randomly a solution and generate a local solution around this one, using (5)
End if
Evaluate the bats
If $(rand < A_i) \wedge (B_{new} < B_i)$
$B_i = B_{new}$
Increase r_i and reduce A_i, using (6)
End if
End for
Rank bats to find the best solutions in population.
Find the best bat
End while

Fig. 1. Proposed Mono-Objective BA pseudocode.

One can realize that the proposed algorithm uses the same dynamics of a PSO, but the loudness and rate of pulse make the BA works like the standard PSO combined with an intensive local search, as given in [1], which is very similar to some SA approaches for local search.

$$F_i = F_{min} + (F_{max} - F_{min})\beta, \beta \in [0,1], \beta \sim N \quad (2)$$

$$\boldsymbol{v}_i^t = \boldsymbol{v}_i^{t-1} + (\boldsymbol{x}_* - \boldsymbol{x}_i^{t-1}) \quad (3)$$

$$\boldsymbol{x}_i^t = \boldsymbol{x}_i^{t-1} + \boldsymbol{v}_i^t \quad (4)$$

$$\boldsymbol{x}_{i,new}^t = \boldsymbol{x}_{i,old}^t + \left(\boldsymbol{\lambda} - \boldsymbol{x}_{i,old}^t\right)\epsilon\langle A_i^{t-1}\rangle, \quad \epsilon \in [0,1], \epsilon \sim U \quad (5)$$

$$A_i^t = \alpha A_i^{t-1}, r_i^t = r_i^0[1 - e^{-\gamma t}], r \in [0,1], r \sim U \quad (6)$$

The proposed multiobjective BA variation follows the pseudocode presented in Fig. 2. It is possible to say that it is based on the mono-objective BA with the selection structure of the NSGA-II (Non-Dominated Sorting Genetic Algorithm – version II) as found in [3]. In this way, it allows to provide the multiobjective optimization problem the both good features of these algorithms, i.e. the good and fast convergence of BA and the parameter free NSGA-II operators of elitism and diversity.

While (t < max number of interactions)
For $i = 1{:}N_b$
Generate $B_{new,i}$, using (2), (3) and (4)
If $rand > r_{new,i}$
Select a non-dominated (first front) solution and generate a local solution around this one, using (5)
Else
Select randomly a solution and generate a local solution around this one, using (5)
End if
Evaluate the bats
End for
$R_t = B_t \cup B_{new}$
Sort the solution in ranks (Elitism) with the Fast Non-Dominated Sort Algorithm [3] generating $Front$
$B_{t+1} = \emptyset, j = 1$
While $|B_{t+1}| + Front_j < No.Bats$
Crowding – Distance Assignment ($Front_j$) [3]
$B_{t+1} = B_{t+1} \cup Front_j$
$j = j + 1$
End while
Sort $Front_j$ in the descending order in relation to the crowding distance values
$B_{t+1} = B_{t+1} \cup Front_j\big(1{:}(No.Bats - |B_{t+1}|)\big)$
Increase r_i and reduce A_i, using (6)
Find the best bat
End while

Fig. 2. The Proposed Multi-Objective BA Pseudocode.

III. The Optimization Problem and Results

In this work, we will optimize the efficiency of a BLDC motor, which is presented in [2], where the scripts are provided for research purposes. The analytical problem model has 78 non-linear equations. Electric, magnetic and thermal phenomena are taken into account. The problem is very well detailed in [2], so it will omitted here. The mono-objective problem has 5 design parameters and 6 constraints and the multi-objective problem has 2 objectives, 5 design parameters and 5 constraints.

Table I shows a comparison between several algorithms for the mono-objective problem, where the result presented for the BA in the best found in 30 runs. The metrics for these BA's runs are presented in Table II. The mean of the best results gives an efficiency of 95.2% which is very close to the best result. The standard deviation is also very small. Even though the best efficiency found is not the better one, the constraint handle was well performed by penalty in the BLDC design. This feature can be modified by the adjustment of run parameters (maximum frequency, α and γ constants) or penalty coefficients.

The multiobjective problem was also run 30 times, with 250 generation and 100 bats. The results are shown in Table III, where the best harmonic feasible solution of all run is presented. Fig. 3 shows the Pareto's front after one of these runs. The multiobjective BA can found relatively good results. Besides, the final Pareto front spreads well, in this context the proposed BA may find a good result in a harmonic way and a relatively good set of solutions in a general manner.

TABLE I
The Optimization Results from [2] and the Bat Algorithm (BA)

Method	SQP [2]	GA [2]	GA & SQP [2]	ACO [2]	PSO [2]	BA
$D_s(mm)$	201.2	201.5	201.2	201.2	202.1	202.2
$B_e(T)$	0.6481	0.6480	0.6481	0.6481	0.6476	0.6535
$\delta(A/mm^2)$	2.0437	2.0602	2.0615	2.0437	2.0417	2.0514
$B_d(T)$	1.8	1.799	1.8	1.8	1.8	1.8
$B_{cs}(T)$	0.8959	0.8817	0.8700	0.8959	0.9298	0.9792
$\eta(\%)$	95.32	95.31	95.31	95.32	95.32	95.31
$Eval$	90	3380	1644	1200	1600	1590
$M_{tot}(kg)$	15	15	15	15	15	14.95
$I_{max}(A)$	125	125	125	125	125	130.5
$D_{int}(mm)$	76	76	76	76	76	81.5
$D_{ext}(mm)$	238.9	239.2	238.9	238.9	239.8	240.3
$T_a(°C)$	95.35	95.21	95.31	95.35	94.98	94.95
$discr$	0.0235	0.0251	0.0246	0.0235	0.0253	0.0254

TABLE II
The Mono-objective Case Using BA in 30 runs (best results)

$\eta(\%)$	M_{tot}	I_{max}	D_{int}	D_{ext}	T_a	$Discr$
Mean						
95.23	14.97	151.91	88.11	242.85	94.93	0.0249
Standard Deviation						
0.0568	0.2905	43.8935	9.3930	3.2893	1.5040	0.0091

TABLE III
The multiobjective case using BA in 30 runs (best results)

Objectives		Constraints				
$\eta(\%)$	M_{tot}	I_{max}	D_{int}	D_{ext}	T_a	$Discr$
Best harmonic feasible solution						
95.083	13.887	147.853	80.375	232.852	100.224	0.017

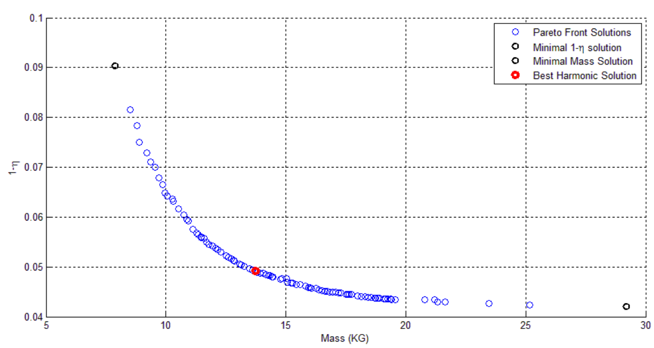

Fig. 3. The Multiobjective Case: the Pareto Front 1- η and the Mass

IV. Conclusion

The Bat Algorithm approach is a new approach for optimization problems which seems to be a robust tool and in the analyzed problem competitive with several traditional optimization methods because it merges some features of two different optimization algorithms: PSO and SA. It has shown a good performance for mono-objective problems and the proposed extension to multi-objective problems has also provided a high quality Pareto Front.

References

[1] X. S. Yang, "A new bat-inspired algorithm," *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, Springer, Studies in Computational Intelligence, vol. 284, pp. 65-74, 2010.

[2] Benchmark web site: *http://l2ep.univ-lille1.fr/come/benchmark-wheel-motor/Math.htm*

[3] K. Deb, "Multi-objective optimization using evolutionary algorithms," Wiley-Interscience Series in Systems and Optimization, 2001.