Available online at www.worldscientificnews.com

World Scientific News

WSN 87 (2017) 205-221 EISSN 2392-2192

Worm Gear Drive optimization Considering Industry Constraints Based on Nature Inspired Algorithms

N. Godwin Raja Ebenezer[1,*], R. Saravanan[2], S. Ramabalan[3], S. Navaneethasanthakumar[4]

[1]Department of Mechanical Engineering, Loyola - ICAM College of Engineering and Technology, Chennai – 600034, India

[2]Department of Automobile Engineering, Dr. Mahalingam College of Engineering and Technology, Pollachi – 642 003. India

[3]Department of Mechanical Engineering, EGS Pillay Engineering College, Nagapattinam – 611002, India

[4]Department of Mechanical and Industrial Engineering, Bahi Dar Institute of Technology, Bahir Dar, Ethiopia

*E-mail address: greraja@gmail.com

ABSTRACT

This paper presents a novel method to obtain optimum design for a worm gear drive used in sugar industries taking into account certain constraints of industrial relevance. The objective of this research is to minimize volume of worm gear drive. Gear ratio, face width and pitch circle diameters of worm and worm wheel are considered as design variables. Industry relevant constraints viz. gear strength capacity, wear capacity, thermal capacity, dynamic load, self locking, and face width are considered. Besides this other constraints such as maximum power transmission capacity, centre distances, deflection of worm and beam strength of worm are also considered. Nature inspired optimization algorithms, namely, Simulated Annealing (SA), Firefly (FA), Cuckoo Search (CS) and MATLAB solvers fmincon and GA are used for solving this problem in MATLAB environment. Results of simulation are analysed and presented.

***Keywords*:** Gear optimization, Worm gear drive, Industry relevant constraints, Nature inspired algorithms SA, FA, CS, MATLAB solvers GA, fmincon

1. INTRODUCTION

Finding solution to worm gear drive problems is complex and a time consuming process as many design variables, constraints, empirical formulas, different graphs and tables are involved. On the other side the demand for compact, efficient and reliable worm gears is on the rise. Figure 1 shows the worm gear drive and Figure 2 shows the nomenclature.

Considerable researches had been undertaken on optimization problems of mechanical gear design using various algorithms by many scholars [1-4]. A fair deal of research had been carried out on optimization of single and two stage gear pairs. Single gear pair optimization has generally focussed on optimal bending and crushing stresses, displacement acting on the gear tooth with regard to space requirements, transmitted power, weight, tooth profile and material capabilities. Different optimization algorithms such as GA, Simulated Annealing, Ant-Colony Optimization, and Neural Network have been used for design optimization of gear problems. Savsani V et al [1] obtained minimum weight of a spur gear train using particle swarm optimization and simulated annealing. Daizhong Su and Wenjie Peng [5] developed an artificial neural network and genetic algorithm to optimize worm gear design based on FEA results [9-25].

Gologlu et al. minimized volume of two-stage helical gear train by taking into account normal module, number of teeth & face width as design variables and bending strength, contact stress as constraints [2]. Chong et al. applied GA for minimizing geometric volume of two stage gear train and simple planetary gear train. There was about 40% reduction in pitch diameter and face width, and the error of reduction in gear ratio was about 3% between objective and result [3]. Li et al. used an adaptive Genetic Algorithm (GA) to solve multi-objective optimization design of reducer [4].

Figure 1. A combined helical gear and worm gear set speed reducer

As the literature reveals, not much research has been undertaken on worm gear drive optimization. Besides this certain challenging industry relevant constraints, namely, gear strength capacity, wear capacity, thermal capacity, dynamic load, self locking, and face width are also not considered in the literature. Further, powerful nature inspired algorithms SA, FA and CS are not used in the literature. Therefore, in this work, optimization of worm drive involving industry relevant constraints based on superior nature inspired algorithms SA, FA and CS and MATLAB solvers fmincon, GA is carried out.

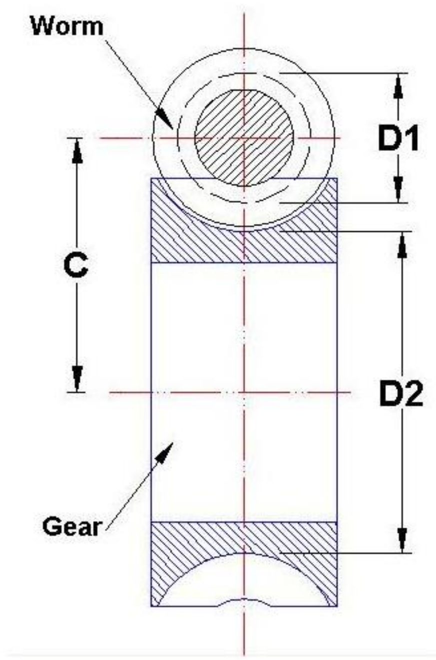

Figure 2. Nomenclature of the worm and gear

The organisation of the paper is as follows: In Section 2, Nature inspired optimization algorithms SA, FA and CS and MATLAB solvers fmincon, GA are presented. in Section 3, Mathematical formulation of the optimization problem for worm gear drive involving industrial constraints is presented. In Section 4 Results and discussion is presented. Finally, Conclusion and scope of the work is presented.

2. NATURE INSPIRED ALGORITHMS AND MATLAB SOLVERS

2. 1. Simulated Annealing (SA)

It is a trajectory-based, random search technique for global optimization. It mimics the annealing process in material processing. SA algorithm starts with a random initial design vector (solution) X_i and high temperature T. A second design point is created at random in the vicinity of the initial point and the difference in the function values (ΔE) at these two points is calculated as:

$$\Delta E = \Delta f = f_{t+1} - f_1 \equiv f(X_{i+1}) - f(X_i) \quad (1)$$

If the new solution's objective function value is smaller, the new solution is automatically accepted and becomes the current solution from which the search will continue. Otherwise the point is accepted with a probability $e^{\left(-\Delta E/_{kT}\right)}$ where k is the Boltzmann's constant. This completes one iteration of the SA. Due to the probabilistic acceptance of a non improving solution, SA can escape from local optima. At a certain temperature T predetermined numbers of new points are tested. The algorithm is terminated when current value of temperature is small enough or when changes in function values (Δf) are sufficiently small.

Simulated Annealing Algorithm

Objective function f (x), x = (x_1 , ..., x_p)T
Initialize initial temperature T_0 and initial guess $x^{(0)}$
Set final temperature T_f and max number of iterations N
Define cooling schedule T → αT , (0 < α < 1)
while (T > T_f and n < N)
    Move randomly to new locations: $x_{n+1} = x_n + \epsilon$(random walk)
    Calculate $\Delta f = f_{n+1}(x_{n+1}) - f_n(x_n)$
    Accept the new solution if better
  if not improved
    Generate a random number r
    Accept if $p = \exp[-\Delta f / T] > r$
  end if
  Update the best x∗ and f∗
  n=n+1
end while

The initializing parameters and settings of SA for this research are:

Initial temperature, T_init = 1.0; Finial stopping temperature, T_min = 1e-10; Min value of the function, F_min = -1e+100; Maximum number of rejections, max_rej = 500; Maximum number of runs, max_run = 150; Maximum number of accept, max_accept = 50; Initial search period, initial_search = 500; Boltzmann constant k = 1; Energy norm (eg, Enorm = 1e-8) Enorm = 1e-5.

2. 2. Firefly Algorithm (FA)

FA was developed by Xin-She Yang at Cambridge University in 2007 [7]. There are three idealized rules incorporated into the original Firefly algorithm (FA) : i) all fireflies are unisex so that a firefly is attracted to all other fireflies; ii) a firefly's attractiveness is proportional to its brightness seen by other fireflies, and so, for any two fireflies, the dimmer firefly is attracted by the brighter one and moves towards it, but if there are no brighter fireflies nearby, a firefly moves randomly; and iii) the brightness of a firefly is proportional to the value of its objective function. According to the above three rules, the degree of attractiveness of a firefly is calculated by the following equation:

$$\beta = \beta_o e^{-\gamma r^2} \quad (2)$$

where β is the degree of attractiveness of a firefly at a distance r, β_o is the degree of attractiveness of the firefly at r = 0, r is the distance between any two fireflies, and γ is a light absorption coefficient. The distance r between firefly i and firefly j located at X_i and X_j respectively is calculated as a Euclidean distance:

$$r = \|X_i - X_j\| = \sqrt{\sum_{k=1}^{d}\left(X_i{}^k - X_j{}^k\right)^2} \tag{3}$$

The movement of the dimmer firefly i towards the brighter firefly j in terms of the dimmer one's updated location is determined by the following equation:

$$X_{i+1} = X_i + \beta_o e^{-\gamma r^2} = \left(X_j - X_i\right) + \alpha\left(\text{rand} - \frac{1}{2}\right) \tag{4}$$

The third term in (4) is included for the case where there is no brighter firefly than the one being considered and rand is a random number in the range of (0, 1).

The Firefly algorithm

Objective function f (x),x = (x_1 , ..., x_d)T
Generate initial population of fireflies x_i (i = 1, 2, ..., n)
Light intensity I_i at x_i is determined by f (x_i)
Define light absorption coefficient γ
while (t <Max Generation)
for i = 1 : n all n fireflies
  for j = 1 : n all n fireflies (inner loop)
  if (I_i < I_j), Move firefly i towards j; end if
    Vary attractiveness with distance r via exp[−γr]
    Evaluate new solutions and update light intensity
  end for j
end for i
Rank the fireflies and find the current global best g ∗
end while
Postprocess results and visualization

FA can find the global optima as well as the local optima simultaneously and electively and also suitable for parallel implementation. For (FA), the values of the parameters used are: 20 fireflies, Number of iterations = 250, $\alpha = 0.5$, $\gamma = 1$ and $\beta_o = 0.2$. These parameters have been chosen after adjustment to suit for solving the problem.

2. 3. Cuckoo search Algorithm (CS)

Cuckoo search (CS) is developed in 2009 by Xin-She Yang and Suash Deb [8]. CS is biologically inspired by the cuckoos' manner of looking for nests where they could lay eggs. In this optimization algorithm, each nest represents a potential solution. The cuckoo reproduction process in the algorithm is simplified by three rules:

1. Each cuckoo lays an egg in a randomly chosen nest; 2. The best nests carry over to the next generation of cuckoos; 3. The number of available host nests is fixed (limited), and the egg

laid by a cuckoo is discovered by the host bird with a probability, $p_a$which ranges 0.1. Birds can detect only the worst nests so that they are losing from the population.

The initial population of nests with the size, n which are randomly distributed over the search space, is generated first. The randomly chosen initial solutions of design variables are defined in the search space by the lower and upper boundaries. The new nest, for example i^{th}, is generated according to the following law,

$$x_i^{t+1} = x_i^{t} + \alpha \oplus levy\lambda \tag{5}$$

where $\alpha > 0$ is the step size whose value depends on the optimization problem, and t is the current generation. Step size is multiplied by the random numbers with Lévy's distribution, and such random motion is called Lévy flight. Levy flight has the step-lengths distributed according to the following probability distribution:

$$Lévy \sim u = t^{-\lambda}, 1 < \lambda \leq 3 \tag{6}$$

The numerical algorithm using the exponential law, was used for generation of Lévy distribution in CS. It is recommended that the step size should be, $L/100$v where L is the size of the space which is searched. There is a danger that Lévy flight may become too "aggressive" for large values of the step size and that new solutions may go out of the space which is searched.

Cuckoo Search via L´vy Flightse

```
Objective function f (x), x = (x1 , ..., xd )T
Generate initial population of n host nests xi
while (t <Max Generation) or (stop criterion)
  Get a cuckoo randomly/generate a solution by L´vy flightse
        and then evaluate its quality/fitness Fi
  Choose a nest among n (say, j) randomly
  if (Fi > Fj ),
    Replace j by the new solution
  end
  A fraction (pa ) of worse nests are abandoned
        and new ones/solutions are built/generated
  Keep best solutions (or nests with quality solutions)
  Rank the solutions and find the current best
end while
Post process results and visualization
```

Here the parameters used are: n = 25 nests, $\alpha = 1$ and $p_a = 0.25$ for implementation. Number of host nests (or the population size n) and the probability p_a are varied as: n = 5, 10, 15, 20, 30, 40, 50, 100, 150, 250, 500 and p_a = 0, 0.01, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.4, and 0.5. Simulation showed n = 25 and p_a = 0.25 are suitable for this problem.

2. 4. Genetic Algorithm (GA)

GA is based on natural selection, the process that drives biological evolution. It can be applied to problems in which the objective function is discontinuous, stochastic, or highly nonlinear. It repeatedly modifies a population of individual solutions. At each step, it selects individuals at random from the current population to be parents and uses them to produce the children for the next generation. Over successive generations, the population ‘‘evolves’’ toward an optimal solution. Table 1 shows the selected values for different parameters of GA
.

Table 1. Selected values for different parameters of GA

Parameters	Selected values
Population size	20
Initial range	0;1
Elite count	2
Cross over fraction	0.8
Generations	50

2. 5. Fmincon

FMINCON is a function included in MATLAB's Optimization Toolbox which seeks the minimizer of a scalar function of multiple variables, within a region specified by linear constraints and bounds. The syntax for fmincon is *x_opt* = fmincon (*fun*, *x0*, *A*, *b*, *Aeq*, *beq*, *lb*, *ub*) where *fun* is the "function handle"; that is, the name of an M-file that defines the function, preceded by an "@" sign; *x0* is an initial value for the optimizer; *A*, *b* define a linear inequality constraint A * x <= b on the solution. *Aeq*, *beq* define a linear equality constraint Aeq * x = beq on the solution. *lb*, *ub* define bounds on the solution, lb <= x <= ub.

3. WORM GEAR DRIVE OPTIMUM DESIGN

Here a worm gear drive used in sugar industries is considered. The specifications and input for the worm gear drive are shown in Table 2 [6].

Table 2. Specifications [6] and input for the worm gear drive

Parameters	Worm	Worm wheel
Pitch circle diameter D_1, D_2 (mm)	240	1519.16
Number of starts/teeth, Z_1, Z_2	1	65

Centre distance, C (mm)	901	
Face width, b (mm)	160	
Axial pitch, p (mm)	75.408	-
Speed n_1, n_2 (rpm)	48	0.7385
Lead angle, λ (deg)	5.71	-
Module, m (mm)	24	
Pressure angle, Φ_n (deg)	20	-
Permissible bending stress, $[\sigma_b]$ (N/mm^2)	80	
Material	Phosphor bronze	
Modified Lewis form factor, Y	0.393	
Worm gear wear factor, K_w	0.552	
Coefficient of friction, f	0.099	
Power, (kW)	0.8	
Efficiency, η (%)	0.85	
Temperature of the gear, t_2 (°C)	93	
Ambient temperature, t_1 (°C)	35	
Heat transfer coefficient, C_H $Nm/s/m^2/°C$	7	

3. 1. Objective Function

The objective is to minimize volume of worm and worm wheel assembly. The volume equation from [6] is:

$$Volume(V) = 14.310926 * \frac{{x_3}^2 x_4}{x_1} + 455.5309 * x_2 x_4 + 0.8382 * \left(\frac{256{x_4}^2 + {x_1}^2{x_2}^2}{x_1 x_4}\right)^2 + 104634.67 \qquad (19)$$

3. 2. Design Variables

The design variables are: Gear ratio (i), Face width (b), Pitch circle diameter of worm (D_1) and Pitch circle diameter of worm wheel (D_2). The function for design variables of the worm drive can be written as $F(x) = F(i.b, D_1, D_2) = F(x_1, x_2, x_3, x_4)$.

Upper and lower bounds of design variables from (24) are: $40 \leq i \leq 70$,$100 \leq b \leq 200$, $150 \leq D_1 \leq 280$ and $1000 \leq D_2 \leq 1500$.

3. 3. Industrial constraints

No other research has involved industrial constraints such as face width, conditions for self locking, wear capacity, strength capacity and thermal capacity. But they are considered in this work for worm gear.

3. 4. Face width constraint

The face width of a gear is the length of teeth in an axial plane. The face width b of the gear should not exceed half the worm outside diameter.

$$b \leq d_{a1}, \tag{20}$$

where $d_{a1} = D_1 + 2h_{a1} = D_1 + 2m$ (21)

By substituting the value of m from Table 5 in equation (21) and then from equations (20) and (21) we get,

$$b \leq 0.5(D_1 + 48) \tag{22}$$

Now constraint 1 is given by, $C_1(x): x_1 - 0.5(x_3 + 48) \leq 0$ (23)

3. 5. Self locking constraint

Mathematically, the tangent of the helix angle of the worm gear is less than the coefficient of friction between the worm and the gear. This condition is referred to as self locking of worm gears [9].

Coefficient of friction, $\mathrm{f} \geq \cos\Phi_n \tan\lambda$ (24)

where $\tan\lambda = \frac{l}{\pi D_1}$, and $l = \pi m Z_1$ (25)

Substituting the values of Φ_n , f , m from Table 2 in equation (25), we get,

$$0.099 \geq \frac{22.5}{D_1} \tag{26}$$

Now constraint 2 can be written as, $C_2(x): \frac{22.5}{x_3} - 0.099 \leq 0$ (27)

3. 6. Worm gear strength and wear capacity constraint

For better performance, dynamic load F_d must be greater than strength and wear capacity, given by [9],

$F_b \geq F_d$ and (28)

$F_w \geq F_d$ (29)

where strength capacity, $F_b = [\sigma_b] \times b \times m \times Y$ and (30)

wear capacity is given by the relation, $F_w = D_2 \times b \times K_w$ (31)

Dynamic load F_d is given by,

$F_d = F_{2t} \times \left[\frac{6.1 + V_2}{6.1}\right]$ where (32)

$V_2 = V_s \sin\lambda$ and (33)

$V_s = \frac{\pi D_1 n_1}{60000 \cos\lambda}$ (34)

$F_{2t} = F_{1t} \times \left[\frac{\cos\Phi_n \cos\lambda - f\sin\lambda}{\cos\Phi_n \sin\lambda + f\cos\lambda}\right]$ where (35)

$F_{1t} = \frac{60 \times 10^6 (kW)}{D_1 \pi n_1}$ (36)

Using the values $(kW), d_1, n_1, \Phi_n, \lambda$ from Table 2 and from equations (28~32), we get, $754.56\ b \geq 6430.01$ and $0.552 D_2 b \geq 6430.01$

Now, the strength and wear capacity constraints (Constraints 3 and 4) are:

$C_3(x): 6430.01 - 754.56 x_2 \leq 0$ (37)

$C_4(x): 6430.01 - 0.552 x_2 x_4 \leq 0$ (38)

3. 7. Thermal capacity constraint

Amount of power lost in friction is converted into heat. To prevent overheating, heat generated should not exceed the rate of heat transfer [9].

$Q = 1000(1 - \eta)(kW)$ and (39)

$Q = C_H(A_{gear} + A_{worm})(t_2 - t_1)$ (40)

where A_{worm}=lead of the worm $\times D_1$ and

$A_{gear} = \frac{\pi}{4} {D_2}^2$ (41)

As the worm considered has a single start thread, lead is equal to axial pitch. In equations (39), (40), (41) substituting the known values, the condition for avoidance of overheating is obtained as constraint 5.

$$C_5(x): 120 - 406(0.785{x_4}^2 + 0.075408x_3) \leq 0 \tag{42}$$

3. 8. Centre distance constraint

To achieve compact design of gear pairs, the centre distance of optimized gear pair should be less than that of actual gear pair. So constraint 6 is [6],

$$C_6(x): \frac{\frac{x_1 x_2}{2\sqrt{\frac{65x_3}{x_4}+1}}+x_3}{2} - 901 \leq 0 \tag{43}$$

3. 9. Centre distance and Pitch Circle diameter of worm drive constraint

The condition for maximum power transmission capacity from [6] is constraint 7, and constraint 8, given by,

$$C_7(x): \frac{\left[\frac{\frac{x_1 x_2}{2\sqrt{\frac{65x_3}{x_4}+1}}+x_3}{2}\right]^{0.875}}{3} - x_3 \leq 0 \tag{44}$$

$$C_8(x): x_3 - \frac{\left[\frac{\frac{x_1 x_2}{2\sqrt{\frac{65x_3}{x_4}+1}}+x_3}{2}\right]^{0.875}}{1.7} \leq 0 \tag{45}$$

3. 10. Deflection of the worm constraint

If the worm shaft bends too much, the teeth will not mesh properly. This will result in excessive wear and early failure. To avoid this, the maximum deflection [6] is constraint 9, given by,

$$C_9(x): \frac{90.887873 \times \frac{{x_4}^4}{x_1}}{\left[{x_2}^2 - 4\frac{{x_4}^2}{{x_1}^2}\right] \times \left[x_3 - 2.4\frac{x_4}{x_1}\right]^4} - 0.001x_3 \leq 0 \tag{46}$$

3. 11. Beam strength of worm gear constraint

Beam strength of worm gear tooth is the maximum tangential load the worm gear tooth can take without tooth breakage. Constraint 10 based on Lewis equation is [6],

$$C_{10}(x): \frac{66116467.3 \times x_1}{x_4 x_2 x_3} - 83.33333 \leq 0 \tag{47}$$

4. RESULTS AND ANALYSIS

Table 3 shows optimal values of gear ratio given by algorithms and MATLAB solvers are same. However for face width slightly different optimal values are given by SA, FA, CS and MATLAB solvers GA and fmincon (126.98, 124.347, 126.94, 130.97, and 126.94 mm). The highest value (130.97 mm) is by GA and the lowest value (124.347 mm) by CS and fmincon.

Table 3. Comparison of optimal values by different algorithms of this work (bold values) and literature [6]

Parameters	SA	FA	CS	GA	fmincon	Literature [6]
Gear ratio	40	40	40	40	40	46
Face width (mm)	126.998	127.347	126.994	130.97	126.994	197
Pitch circle diameter of worm (mm)	250	249.253	250	242.397	250	188
Pitch circle diameter of worm wheel (mm)	1000	1000	1000	1000.058	1000	1000
Volume of worm and worm wheel (mm^3)	0.8844462×10^8	0.844981×10^8	0.844442×10^8	0.849692×10^8	0.844442×10^8	1.45442×10^8
Centre distance between worm and worm wheel (mm)	431.01	431.70	430.78	441.18	430.78	717.24
Deflection of worm (mm)	0.1279	0.1291	0.12794	0.1401	0.12794	0.157242
Strength capacity (N)	98091.29	98368.59	95824.59	98824.72	95824.59	Not reported
Wear capacity (N)	17525.74	17521.37	17525.17	17524.19	17525.17	Not reported
Thermal capacity (Nm/s)	326.36	326.33	326.36	326.13	326.36	Not reported

Optimal pitch circle diameter of worm by SA, CS and fmincon are same (250 mm). FA and GA give 249.253 mm, 242.397 mm. Optimal pitch circle diameter by all algorithms (except GA) is 1000 mm. Optimum strength capacity by GA is the highest (98824.72N). SA gives the highest optimum wear capacity (17525.74 N) and thermal capacity (326.36 Nm/s). Figure 3 shows the graphical representation of the results.

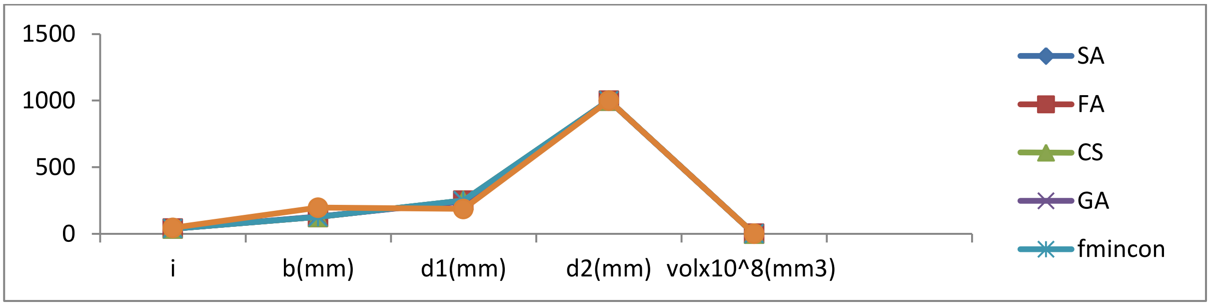

Figure 3. Graphical representation of the result obtained by various algorithms.

CS and fmincon give the best values ($0.844442*10^8 mm^3$). Also, minimum volume of the worm gear drive by both CS and fmincon are same.

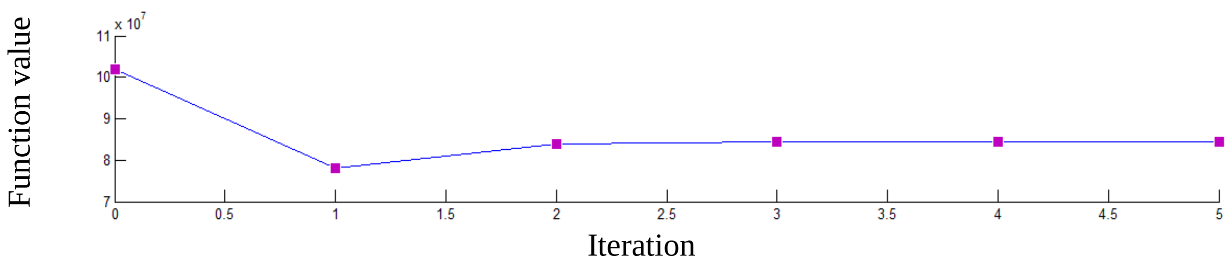

Figure 4. Function value Vs Iteration by MATLAB fmincon solver

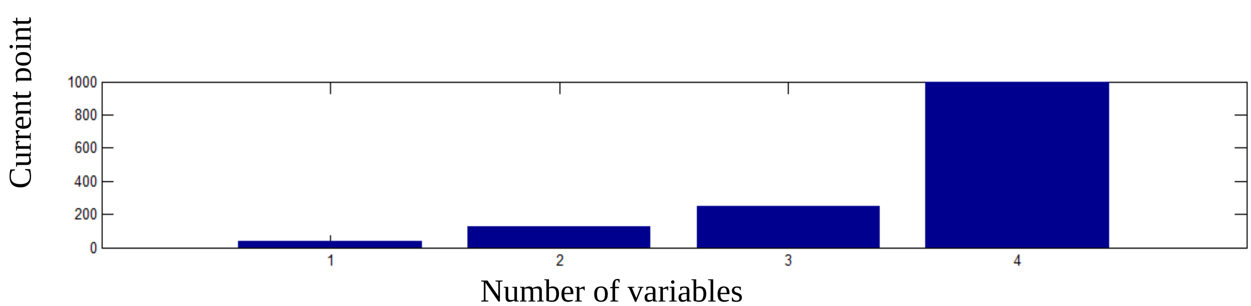

Figure 5. Current point Vs Number of variables by MATLAB fmincon solver

Figure 5 and Figure 7 show optimized values of design variables by fmincon and GA. These graphs plot the vector entries of the individual with the best fitness value in each generation. Figure 4 and Figure 6 plot the best function value in each generation versus iteration number.

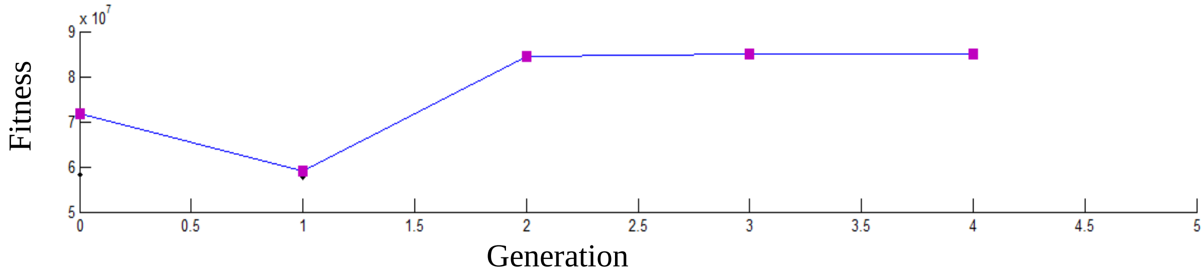

Figure 6. Fitness value Vs Generation by GA

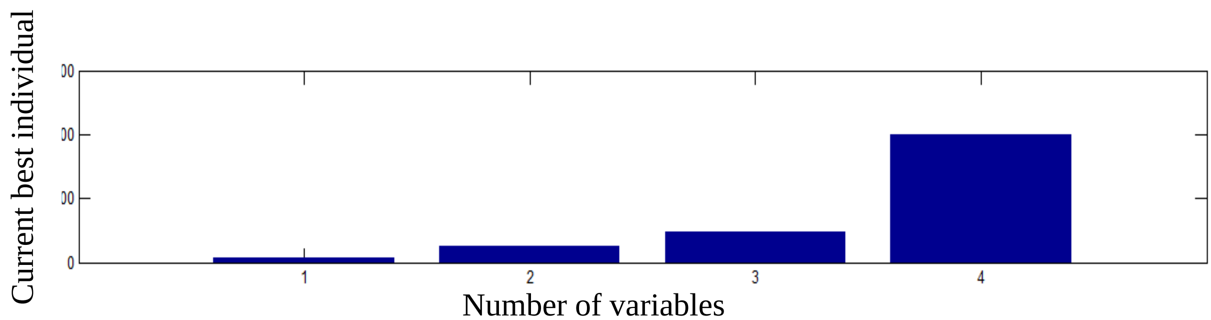

Figure 7. Current best individual Vs Number of variables by GA

Figure 4 reveals at zeroth iteration function value is 10.1×10^7 $mm^3$and it is decreases upto 8×10^7 at first iteration, after that it is increases up to 8.4×10^7 mm^3 and then remains constant for next generations. From Figure 6, it is observed that at zero generation fitness value is 7.1×10^7 mm^3 and it is decreases up to 6×10^7 at first generation after that it is increases upto 8.4×10^7 and remains constant for next generations. Optimal values by algorithms and solvers are presented in Table 3. As CS gives best results, it is compared with literature [6] in Table 4. A 42.06 % reduction in volume of worm and worm wheel is observed.

Table 4. Percentage change in optimal parameter values by CS of this work (bold values) and literature [6]

Parameters	Literature [6]	CS	% Change in value
Gear ratio	46	40	13.04
Face width (mm)	197	126.994	35.53
Pitch circle diameter of worm (mm)	188	250	32.97
Pitch circle diameter of worm wheel (mm)	1000	1000	0
Volume of worm and worm wheel (mm^3)	$1.45442*10^8$	$0.844442*10^8$	42.06

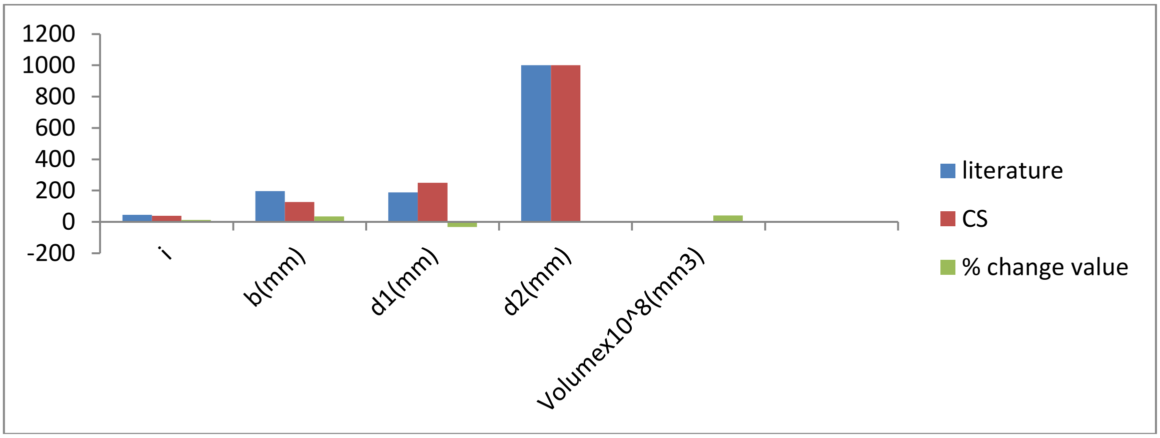

Figure 8. Graphical representation of the percentage change in optimal parameter values obtained by CS algorithm and literature [6]

Percentage change in optimal design parameters values is shown in Figure 8. It is observed that a decrease in value of 13.04 % in gear ratio, 35.53% in face width, and an increase of 32.97% in pitch circle diameter of worm. Interestingly, there is no change in value in pitch circle diameter of the worm wheel.

5. CONCLUSION

In this paper, optimum design for a worm gear drive considering industry relevant constraints using Nature inspired algorithms is obtained. All constraints are well satisfied. The important findings are:

(1) There is 42.06 % reduction in volume of worm gear drive as compared to literature. It is worthwhile to mention that this reduction in volume is achieved after including industry relevant constraints.

(2) There is a decrease in design parameter value of 13.04 % in gear ratio, 35.53% in face width, and an increase of 32.97% in pitch circle diameter of worm. This research also reports the value of thermal capacity, strength capacity and wear capacity of worm gear drive which are not reported in any of the literature.

(3) This work is readily applicable and suitable for optimization of other similar gear drives employed in industries.

References

[1] V. Savsani, R. V. Rao and D. P. Vakharia, Optimal weight design of a gear train using particle swarm optimization and simulated annealing algorithms. *Mechanism and machine theory*, 45 (2010) 531–541

[2] C. Gologlu and M. Zeyveli, A genetic approach to automate preliminary design of gear drives. *Computers & Industrial Engineering* 57 (2009) 1043-1051.

[3] T. H. Chong and J. S. Lee, A design method of gear trains using a genetic algorithm, *International journal of the korean society of precision engineering*, Vol. 1, No. 1, Pp. 62-70, 2000.

[4] R. Li, T. Chang, J. Wang and X. Wei, Multi-objective optimization design of gear reducer based on adaptive genetic algorithm, 978-1-4244-1651-6, IEEE, 2008.

[5] Daizhong Su and Wenjie Peng, Optimum Design of Worm Gears with Multiple Computer Aided Techniques, ICCES, vol. 6, no. 4, pp. 221-227, 2008.

[6] Y. K. Mogal and V. D. Wakchaure, A Multi-objective Optimization Approach for Design of Worm and Worm Wheel Based on Genetic Algorithm. *Bonfring International Journal of Man Machine Interface*, Vol. 3, No. 1, March 2013.

[7] X. S. Yang, Firefly algorithms for multimodal optimization, in: Stochastic Algorithms: Foundations and Applications, SAGA 2009, *Lecture Notes in Computer Science*, 2009 5792, 169-178.

[8] Yang X. S. and Deb S,, Engineering optimization by cuckoo search, *Int. J. Math. Modelling & Numerical Optimisation*, 2010 1, pp. 330-343.

[9] Seyedali Mirjalili, Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowledge-Based Systems*, Volume 89, November 2015, Pages 228-249

[10] Amir Hossein Gandomi, Xin-She Yang, Amir Hossein Alavi, Mixed variable structural optimization using Firefly Algorithm. *Computers & Structures*, Volume 89, Issues 23–24, December 2011, Pages 2325-2336

[11] A. Rezaee Jordehi. Brainstorm optimisation algorithm (BSOA): An efficient algorithm for finding optimal location and setting of FACTS devices in electric power systems. *International Journal of Electrical Power & Energy Systems*, Volume 69, July 2015, Pages 48-57

[12] Aziz Ouaarab, Belaïd Ahiod, Xin-She Yang. Discrete cuckoo search algorithm for the travelling salesman problem. *Neural Computing and Applications*, June 2014, Volume 24, Issue 7–8, pp 1659–1669

[13] J. Belwin Edward, N. Rajasekar, K. Sathiyasekar, N. Senthilnathan, R. Sarjila. An enhanced bacterial foraging algorithm approach for optimal power flow problem including FACTS devices considering system loadability. *ISA Transactions* Volume 52, Issue 5, September 2013, Pages 622-628

[14] Bacanin, N., Tuba, M.: Artificial bee colony (ABC) algorithm for constrained optimization improved with genetic operators. *Stud. Inf. Control* 21(2), 137–146 (2012)

[15] Brajevic, I., Tuba, M.: An upgraded artificial bee colony algorithm (ABC) for constrained optimization problems. *J. Intell. Manuf.* 24(4), 729–740 (2013)

[16] Dai, C., Chen, W., Song, Y., Zhu, Y.: Seeker optimization algorithm: a novel stochastic search algorithm for global numerical optimization. *J. Syst. Eng. Electron.* 21(2), 300–311 (2010)

[17] Dominguez, A.R., Nandi, A.K.: Detection of masses in mammograms via statistically based enhancement, multilevel-thresholding segmentation, and region selection. *Comput. Med. Imaging Graph.* 32(4), 304–315 (2008)

[18] Dorigo, M., Gambardella, L.M.: Ant colonies for the travelling salesman problem. *Biosystems* 43(2), 73–81 (1997)

[19] Gandomi, A.H., Yang, X.S., Alavi, A.H.: Mixed variable structural optimization using firefly algorithm. *Comput. Struct.* 89(23–24), 2325–2336 (2011)

[20] Gandomi, A.H., Yang, X.S.: Evolutionary boundary constraint handling scheme. *Neural Comput. Appl.* 21(6, SI), 1449–1462 (2012)

[21] Gandomi, A.H., Yang, X.S., Alavi, A.H.: Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. Eng. Comput. 29(1), 17–35 (2013)

[22] Gandomi, A.H., Yang, X.S., Talatahari, S., Alavi, A.H.: Firefly algorithm with chaos. *Commun. Nonlinear Sci. Numer. Simul.* 18(1), 89–98 (2013)

[23] Hammouche, K., Diaf, M., Siarry, P.: A comparative study of various meta-heuristic techniques applied to the multilevel thresholding problem. *Eng. Appl. Artif. Intell.* 23(5), 676–688 (2010)

[24] Jaynes, E.T.: Information theory and statistical mechanics. *Phys. Rev. Ser. II* 106(4), 620–630 (1957)

[25] Sathya, P.D., Kayalvizhi, R.: Modified bacterial foraging algorithm based multilevel thresholding for image segmentation. *Eng. Appl. Artif. Intell.* 24(4), 595–615 (2011)

(Received 30 September 2017; accepted 17 October 2017)