IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Vol. 04, Issue 03 (March. 2014), ||V2|| PP 19-23

Weight and deflection optimization of Cantilever Beam using a modified Non-Dominated sorting Genetic Algorithm

Sasidhar Gurugubelli[1], Deepak Kallepalli[2]
[1]*Lecturer, Department of Mechanical Engineering, Adama Science and Technology University, Ethiopia, P.O.Box-1888*
[2]*Lecturer, Department of Mechanical Engineering, Adama Science and Technology University, Ethiopia, P.O.Box-1888*

Abstract: - Many problems having single or multiple objectives are routinely solved using GA methodologies. This paper is devoted to the simultaneous weight and deflection optimization of cantilever beam by using a modified Non-Dominated sorting Genetic Algorithm, subjected to maximum stress and maximum deflection constraints. The multi-objective optimization algorithm used in this paper has better sorting, incorporates elitism and no sharing parameter needs to be chosen a priori.

Keywords: - *Multi-objective Optimization, cantilever beam, Genetic algorithm, Pareto optimal set, non-dominated sorting, Genetic operators.*

I. INTRODUCTION

It is obvious that a smaller deflection is obtained by a larger amount of material. Hence, due to the importance of the weight of the structural material from an economical point of view, it would be more advantageous if one defines an optimization problem which aims minimizing the weight of the structure simultaneously with minimizing the deflection [1]. This will obviously be a multi-objective optimization problem, where the deflection and weight are kind of conflicting global functions. Amongst the several optimization methods there are a few which are more suitable for solving a multi-objective optimization problem with a large number of design variables and a relatively few number of constraints, which is the case in the structural topology optimization problem. Walker and Kumar have tackled complex multi-objective optimization problems by scalarising the multiple objective functions into a single objective using a weight vector [2, 3]. A disadvantage of this approach is that the resulting optimal lamination scheme depends on the chosen weight vector. In general, a multi-objective optimization algorithm yields a set of optimal solutions, instead of a single optimal solution [4]. The reason for the optimality of many solutions is that no one solution can be considered better than any other with respect to the objective functions. These optimal solutions are known as Pareto-optimal solutions [5]. Deb [6] has tried to solve the multi-objective optimization problems by using a fast and elitist multi-objective genetic algorithm (NSGA-II). The primary goals of a multi-criteria optimization algorithm are to guide the search towards the global Pareto-optimal front and to maintain population diversity in the Pareto-optimal solutions.

In the present paper, an improved methodology for the multi-objective optimization of cantilever beam structure. A modified form of multi-objective genetic algorithm, based on the elitist non-dominated sorting genetic algorithm (NSGA-II), is implemented to obtain Pareto-optimal designs for the chosen conflicting objectives. It explores the optimal design of a cantilever beam for minimization of weight and deflection, with the constraint that the developed maximum stress σ is less than the allowable strength S_y and the end deflection δ is smaller than a specified limit δ_{max}.

II. PROPOSED MULTI-OBJECTIVE OPTIMIZATION ALGORITHM

In the proposed modified Non- dominated sorting Genetic Algorithm the population is initialized first. Once the population in initialized, it is sorted based on non-domination into each front. The first front, $F_{1,}$ being completely non-dominant set in the current population and the second front being dominated by the individuals in the first front only and the front goes so on. Each individual in each front are assigned rank (fitness) value based on front in which they belongs to. Individuals in first front are given a rank value of 1 and individuals in second are assigned rank value as 2 and so on.

In addition to fitness value a new parameter called crowding distance is calculated for each individual. Parents are selected from the population by using tournament selection based on the rank and crowding distance. The selected population generates offspring from crossover and mutation operators, which will be discussed in detail in later section. The population with the current population and current offspring is sorted again based on non-domination and only the best *N* individuals are selected, where *N* is the population size. The selection is based on rank and on the crowding distance of the last front. The new generation is filled by each front subsequently until the population sizeexceeds the current population size. And hence the process repeats to generate the subsequent generations.

2.1. Population Initialization

The population is initialized based on the problem range and constraints if any.

2.2. Non-Dominated Sorting

The initialized population is sorted based on non-domination.An individual is said to dominate another if the objective functions of it is no worse than the other and at least in one of its objective functions it is better than the other. The fast sort algorithm [2] is given below:

- For each individual *p* in main population P do the following

→Initialize $S_p = Ø$. This set would contain all the individuals that arebeing dominated by *p*.
→Initialize $n_p = 0$. This would be the number of individuals that dominate *p*.
→For each individual *q* in *P*
* if *p* dominated *q*, then
. add *q* to the set *Sp* i.e. *Sp*=*Sp* U {*q*}
* else if *q* dominates *p* then
. increment the domination counter for *p* i.e. $n_p = n_p + 1$
→If $n_p = 0$, i.e., no individuals dominate *p*, then *p* belongs to the first front; Set rank of individual *p* to one, i.e., $p_{rank}= 1$.
Update the first front set by adding *p* to front one, i.e., $F_1 = F_1 U \{p\}$

- This is carried out for all the individuals in main population P.
- Initialize the front counter to one. *i* = 1
- Thefollowing is carried out while the i^{th} front is nonempty i.e. $F_i \neq Ø$

Q = Ø. The set for storing the individuals for $(i + 1)^{th}$ front.
For each individual *p* in front F_i
*For each individual *q* in S_p(S_p is the set of individuals dominated by *p*)
. $n_q = n_q - 1$, decrement the domination count for individual *q*.
. if $n_q = 0$, then none of the individuals in the subsequent fronts would dominate *q*. Hence set $q_{rank} = i + 1$.
Update the set *Q* with individual *q*,i.e.,*Q* = *Q U q*.
–Increment the front counter by one.
–Now the set *Q* is the next front and hence $F_i = Q$.

In our modified NSGA-II, after sorting all the population, the first front i.e., the population with rank 1 is only considered. Remaining fronts are discarded. This algorithm is better than the original NSGA since it utilizes the information about the set that an individual dominate (S_p) and number of individuals that dominate the individual (n_p).

2.3. Crowding Distance

Once the non-dominated sort is complete, the crowding distance is assigned. Since the individuals are selected based on rank and crowding distance, all the individuals in the population are assigned a crowding distance value. Crowding distance is assigned front wise and comparing the crowding distance between two individuals in different fronts is meaningless. The crowing distance is calculated as below:

- For each front F_i, *n* is the number of individuals.
- Initialize the distance to be zero for all the individuals i.e. $F_i\ (d_j) = 0$, where j corresponds to the j^{th} individual in front F_i.
- For each objective function m

* Sort the individuals in front F_i based on objective *m*,i.e., *I* = sort (F_i,*m*).
* Assign infinite distance to boundary values for each individual in i.e. F_i
* For *k* = 2 to (n - 1)

$$I(d_k) = I(d_k) + \frac{I(k+1).m - I(k-1).m}{f_m^{max} - f_m^{min}}$$

. *I (k).m* is the value of the m^{th} objective function of the k^{th} individual in *I*

The basic idea behind the crowing distance is finding the Euclidian distance between each individual in a front based on their m objectives in the *m* dimensional hyper space. The individuals in the boundary are always selected since they have infinite distance assignment.

2.4. Tournament Selection

Once the individuals are sorted based on non-domination and with crowding distance assigned, the selection is carried out using a *crowded-comparison-operator.* The comparison is carried out as below based on

(1) Non-domination rank p_{rank}, i.e., individuals in front F_i will have their rank as $p_{rank} = i$, and

(2) Crowding distance $F_i\ (d_j)$

Operate the crowded-comparison-operator between pand q, i.e.,$p \prec_n q$ if

– $p_{\text{rank}} < q_{\text{rank}}$
– or if p and q belong to the same front F_i then
$F_i(d_p) > F_i(d_q)$ i.e. the crowing distance should be more.
2.5. Genetic Operators
The genetic operators, viz., *Simulated Binary Crossover (SBX), polynomial mutation* [2, 4] are used to create off-springs. The child population is then analyzed and ranked.

2.6. Recombination and Selection

This offspring is sorted again based on non-domination and only the best individuals with ranking 1 are selected. The selection is based on rank and the crowding distance of the last front. The new generation is filled by each front subsequently until the population size exceeds the current population size. If by adding all the individuals in front F_j the population exceeds N, then individuals in front F_j are selected based on their crowding distance in the descending order till the population size becomes N. And hence the process repeats to generate the subsequent generations.

III. MULTI-OBJECTIVE OPTIMIZATION OF CANTILEVER BEAM PROBLEM FORMULATION

In the multi-objective formulation for the optimization of beam problem, Let us consider a cantilever bean whose one end is fixed and the cross-section of the beam is circular of diameter ‘*d*’ and length ‘*l*’ (Fig.1). The beam is optimized for minimum deflection and minimum weight under constraints on maximum stress and maximum deflection [7].

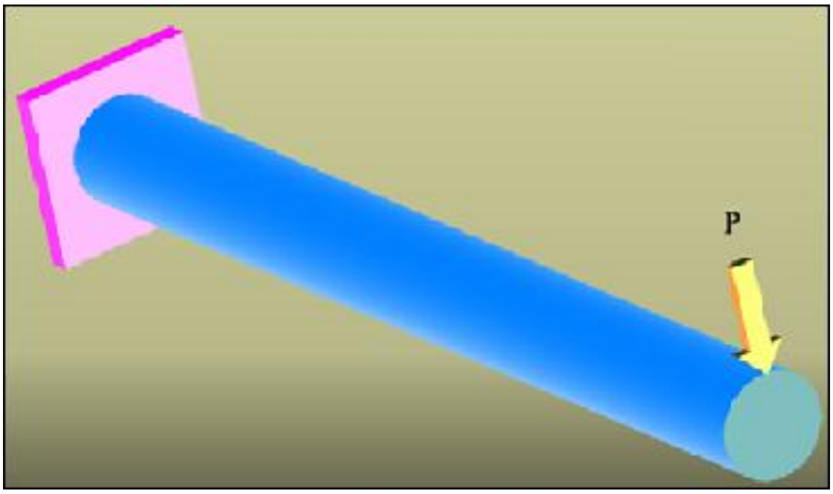

Figure 1 A schematic diagram of cantilever beam

3.1 Input data required for cantilever beam design [7]

The cantilever beam is having a load 1000 N on its free end. The material properties are given as density, ρ = 7800 Kg/m^3, and young’s modulus, E = 207 Gpa. The design variables for Weight and deflection minimization of cantilever beam are shown in Table 1.
The constraint parameters used in the formulation are:

Maximum stress, S_y = 300Mpa and maximum end deflection, $\delta_{\max}$ = 5mm.

Table 1 Design variables with lower and upper bounds

S.No.	Design variable (x)	upper limit (mm)	lower limit (mm)	Bounds
1	Diameter of beam, d	50	10	$10 \le d \le 50$
2	Length of beam, *l*	1000	200	$200 \le l \le 1000$

3.2 Mathematical formulation

The formulation for cantilever beam is as follows:
Objective function:

Minimize: F1= weight = $\frac{\rho\pi d^2 l}{4}$. ... (1)

Minimize: F2 = deflection = $\frac{64 p\, l^3}{3E\pi\, d^4}$. … (2)

Constraints:

(1) Constraint on maximum stress: $\sigma \le S_y$

i.e., $\frac{32pl}{\pi d^3} \le 300$. … (3)

(2) Constraint on maximum deflection: $\delta \le \delta_{max}$

i.e., $\frac{64 p\, l^3}{3E\pi\, d^4} \le 5$. … (4)

Numerical results

The results of this standard engineering multi-objective optimization problem were compared with those obtained by the proposed algorithm. Fig. 2 shows Pareto optimal solutions obtained by the proposed algorithm and the NSGA-II. Table 2 compares the four best Pareto optimal solutions obtained by the proposed algorithm and the NSGA-II.

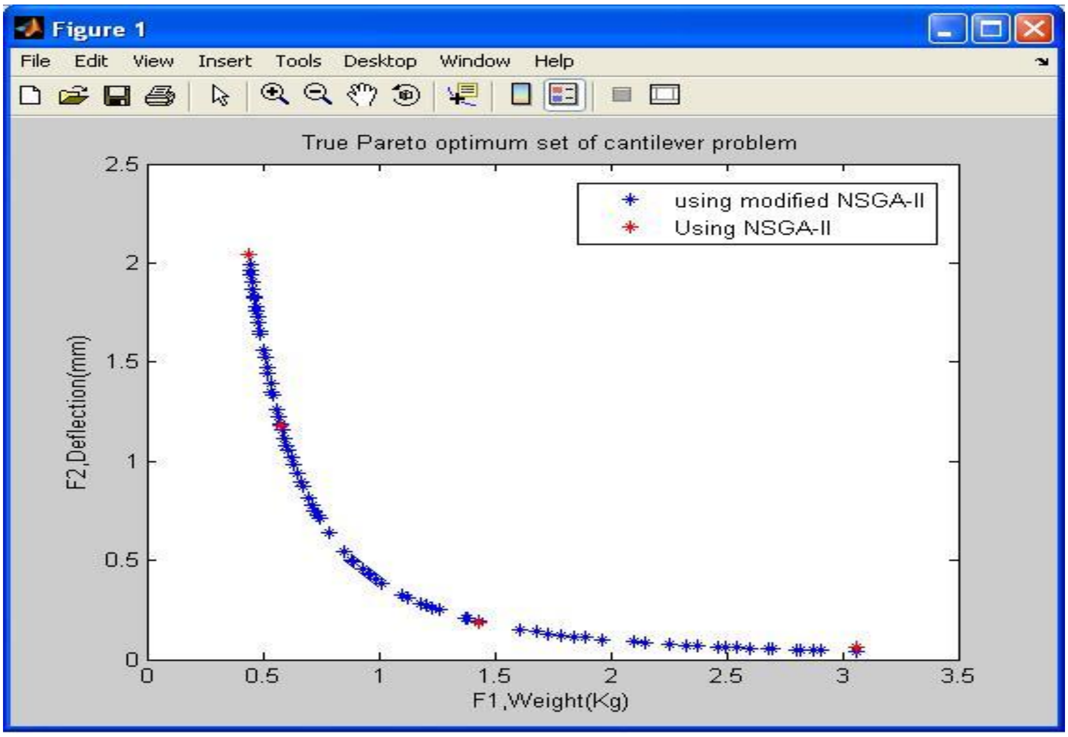

Figure 2 Comparison of pareto-optimal solutions of the present work with NSGA-II

Table 2 Comparison of objective function values with NSGA-II

Pareto optimal solutions	Weight, F1 (Kg)		% change in F1	Deflection, F2 (mm)		% change in F2
	By NSGA-II [7]	By proposed algorithm		By NSGA-II [7]	By proposed algorithm	
1	0.44	0.44	0.00	2.04	1.99	-2.45
2	0.58	0.58	0.00	1.18	1.18	0.00
3	1.43	1.46	+2.09	0.19	0.18	-5.26
4	3.06	3.06	0.00	0.06	0.04	-33.3

IV. CONCLUSIONS

In this paper a modified version of the NSGA-II is proposed for the multi-objective optimization of cantilever beam design problem. The proposed algorithm has been extended to include an archive of the non constraint dominated set, which is updated at each generation. In addition to the non dominated set, a new parameter called crowded distance is used to obtain a termination criterion that automatically stops the algorithm when the maximum number of generations has been reached.

The results obtained by the proposed algorithm are compared with the results NSGA-II. From the results given in the Table 2 it is found that, the objective function values are considerably reduced in modified NSGA-II algorithm. Among the four best solutions given in the literature, the 3rd optimal solution obtained by the proposed algorithm shows an increase of 2.05% only in first objective (weight), and a maximum reduction of about 33.3.% in the second objective (deflection) with respect to the 4th solution. The deflection values on an average 10.25% less than NSGA-II values. The results demonstrate the effectiveness of the proposed methodology for the multi-objective optimization of beams and also for structures.

REFERENCES

[1] B. Hassani and S.M.Tavakkoli, "A Multi-Objective structural optimization using optimality criteria and cellular automata" *Asian journal of civil engineering*; Vol.8, No.1, 2007, pp.77-88.

[2] Walker M, Smith RE, "A technique for the multi-objective optimization of laminated composite structures using genetic algorithms and finite element analysis", *Composite structures*; Vol.62, No.8, 2003, pp.123-128.

[3] Kumar N, Tauchert TR, "Multi-objective design of symmetrically laminated plates", *Journal of mechanical design*; Vol.114, No.4, 1992, pp.620-625.

[4] Kalyanmoy Deb and R. B. Agarwal, "Simulated Binary Crossover for Continuous Search Space". *Complex Systems*, Vol.9, April 1995, pp. 115-148.

[5] Deb K, *Multi-objective optimization using evolutionary algorithms*, John Wiley and Sons Ltd, 2001.

[6] Deb K, "A fast and elitist multi-objective genetic algorithm NSGA-II" *Evolution of Computers, IEEE transaction*, Vol.6, No.2, April 2002, pp.182-196.

[7] Costa L, Fernandes L, Figueiredo I, Judice J, Leal R, Oliveira P, 'Multiple and single-objective approaches to laminate optimization with genetic algorithms", Structural and multidisciplinary optimization; Vol.27, 2004, pp.55-65.