Multi-Objective Optimisation using the Bees Algorithm

D.T.Pham[a], A.Ghanbarzadeh[a,b]

[a] *Manufacturing Engineering Centre, Cardiff university, Cardiff CF24 3AA, UK*
[b] *Mechanical Engineering Department, Engineering Faculty, Shahid Chamran University, Ahvaz, Iran*

Abstract

This paper describes the first application of the Bees Algorithm to multi-objective optimisation problems. The Bees Algorithm is a search procedure inspired by the way honey bees forage for food. A standard mechanical design problem, the design of a welded beam structure, was used to benchmark the Bees Algorithm. The results obtained show the robust performance of the Bees Algorithm.

Keywords: Bees Algorithm, multi-objective optimisation, mechanical design.

1. Introduction

The goal of an optimisation problem can be stated as finding the combination of parameters (independent variables) which maximises or minimises the value of one or more dependent variables possibly subject to some constraints on the independent variable ranges. The values to be optimised are called objective functions. If there is only one function to optimise, the task is a single function optimisation problem. If more than one function should be optimised, the task is a multi-objective optimisation problem.

There is now increasing interest in multi-objective function optimisation as most engineering design problems involve multiple and often conflicting objectives. There are two ways of solving multi-objective optimisation problems. The first possibility is to form a linear combination of the different objective functions. The contribution of each function is associated to a weight, and each function is optimised using methods developed for single objective function problems. The other way of solving a multi-objective problem – the genuine way - is to consider all objective functions simultaneously. The following two main drawbacks are of concern when converting a multi-objective optimisation problem into a single objective optimisation problem. The first shortcoming is that not all the solutions are usually found. The second drawback is that the weight assigned to some objective functions may not be suitable, and the overall linear combination of functions may lack of significance. In multi-objective optimisation tasks, the goal is not find a single optimal solution, but to compute the set of all non-dominated solutions, that is , the Pareto optimal set. A solution belonging to the Pareto set is not better than any other solution belonging to the same set. For this reason, they are not comparable and each of them is called a feasible solution. Different techniques to solve multi-objective function optimisation tasks and their characteristics are explained in [1].

The authors have developed a new optimisation tool, called the Bees Algorithm [2], and have applied it to constrained and unconstrained single objective function optimisations [3-7]. An adapted version of this algorithm has been created to recognise and construct a Pareto set with as many non-dominated solutions as possible.

To show the performance of the algorithm, two test functions plus a well known engineering problem, welded beam design are used and the results are presented.

2. Multi-Objective optimisation

A maximum of a function *f* is a minimum of –*f*. Thus, the general optimisation problem may be stated mathematically as:

$$\begin{array}{lll} \text{minimise} & f_i(X), & i = 1, 2, ..., l \\ \text{subject to} & c_j(X) = 0, & j = 1, 2, ..., p \\ & h_k(X) \geq 0, & k = 1, 2, ..., q \\ & X = (x_1, x_2, ..., x_n)^T & \end{array} \quad (1)$$

Where $f_i(X)$ are the *l* objective functions, *X* is the column vector of the *n* independent variables, and $c_j(X)$ are *p* termed equality constraints, and those of form $h_k(X)$ are *q* inequality constraints. Taken together, $f_i(X)$, $c_j(X)$ and $h_k(X)$ are known as the problem function[1].

The word `minimise` means that we want to minimise all the objective functions simultaneously. If there is no conflict between the objective functions, then a solution can be found where every objective function reaches its optimum. To avoid such trivial cases, it is assumed that there is not a single solution that is optimal with respect to every objective function. This means that objective functions are at least partly conflicting. They may also have different units.

2.1. Pareto Optimality

The predominant solution concept in defining solutions for multi-objective optimisation problems is that of Pareto optimality [8]. A solution in the feasible solution space is called Pareto optimal if there is no other feasible solution in the solution space that reduces at least one objective function without increasing another one.

3. The Bees Algorithm

This section summarises the main steps of the Bees Algorithm (BA). For more details, the reader is referred to [2-4]. Figure 1 shows the pseudo code for the Bees Algorithm. The algorithm requires a number of parameters to be set, namely: number of scout bees (n), number of sites selected for neighbourhood search (out of n visited sites) (m), number of bees recruited for the selected sites (nep), the initial size of each patch (ngh) (a patch is a region in the search space that includes the visited site and its neighbourhood), and the stopping criterion.

The algorithm starts with n scout bees randomly distributed in the search space. The fitness of the sites (i.e. the performance of the candidate solutions) visited by the scout bees are evaluated in step 2.

1. Initialise population with random solutions.
2. Evaluate fitness of the population.
3. While (stopping criterion not met) //Forming new population.
4. Select sites for neighbourhood search.
5. Determine the patch size.
6. Recruit bees for selected sites and evaluate fitnesses.
7. Select the representative bee from each patch.
8. Amend the Pareto optimal set.
9. Abandon sites without new information.
10. Assign remaining bees to search randomly and evaluate their fitnesses.
11. End While.

Fig 1. Pseudo code of the Bees Algorithm

In step 4, the m non-dominated sites are designated as “selected sites” and chosen for neighbourhood search. If there are more than m non-dominated sites in the population, the first m will be selected as it is not possible to differentiate between them. If there are less than m non-dominated sites, from the dominated ones which have been dominated only once, the rest will be selected and this procedure is continued until a sufficient number of sites have been selected. In step 5, a large patch size is chosen initially. For each patch, the initial size is kept unchanged as long as the recruited bees can find better solutions in the neighbourhood. If the neighbourhood search does not yield any progress, the patch size is decreased. This strategy aims at making the local search more exploitative, searching more densely the area around the local optimum. Henceforth, this step will be called the “shrinking method”.

In step 6, the algorithm searches around the selected sites. In the basic version of the Bees Algorithm more bees were associated to search in the vicinity of the best e sites and selection of the best sites was made according to the fitness associated with them. In the multi-objective optimisation version of the Bees Algorithm, it is not possible to always rank the solution candidates, so all the selected sites have the same number of recruited bees to search around the neighbourhood. In step 7, the representative bee will be the original one unless it is dominated by one of the recruited ones; in that case the representative will be the new non-dominated bee. Step 8, has been added to the basic Bees Algorithm to enable it to deal with multi-objective optimisation problems. If the representative is a non-dominated solution, it will be added to the Pareto optimal set. In addition, if this solution is dominating the other solutions in the created Pareto optimal set, the dominated ones will be removed from the set.

In step 9, in the case when no improvement is gained using the shrinking method, it is assumed that the patch is centred on a local peak of performance of the solution space. Once the neighbourhood search has found a local optimum, no further progress is possible. Consequently, the exploration of the patch is terminated. Henceforth this step is called "abandon sites without new information". In step 10, the remaining bees in the population are placed randomly around the search space to scout for new potential solutions.

At the end of each iteration, the colony has two parts to its new population: representatives from the selected patches, and scout bees assigned to conduct random searches. These steps are repeated until a stopping criterion is met.

4. Welded beam design problem

Researchers have used the design of welded beam structures [9] as benchmarks to test their optimisation algorithms. The welded beam design problem involves two nonlinear objective functions and seven constraints.

A uniform beam of rectangular cross section needs to be welded to a base to be able to carry a load of 26689 *N* (6000 *lbf*) The configuration is shown in Fig. 2. The beam is made of steel 1010.

The length L is specified as 356 *mm* (14 *in*). The objectives of the design are to minimise the cost of fabrication and beam deflection while finding a feasible combination of weld thickness h, weld length l, beam thickness t and beam width b. The objective functions can be formulated as [9] :

$$\min \quad f_1 = (1+c_1)h^2 l + c_2 tb(L+l) \tag{2}$$

$$\min \quad f_2 = \delta \tag{3}$$

where

$f_1 =$ Cost function including setup cost, welding labour cost, and material cost;

$f_2 =$ Beam end deflection;

c_1 = Unit volume of weld material cost $= 6.3898 \times 10^{-6} \$ / mm^3 (0.10471 \$ / in.^3)$;

$c_2 =$ Unit volume of bar stock cost $= 2.9359 \times 10^{-6} \$ / mm^3 (0.04811 \$ / in.^3)$;

L = Fixed distance from load to support $= 356\ mm$ $(14\ in.)$.

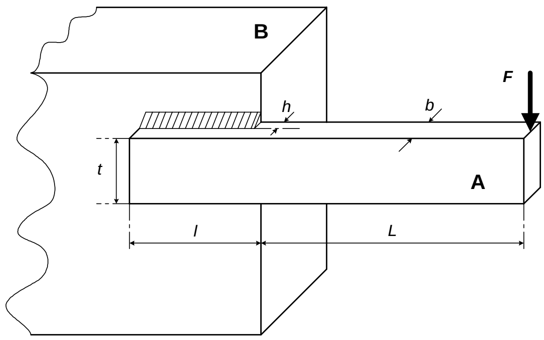

Fig 2. A welded beam

Not all combinations of h, l, t, and b which can support F are acceptable. There are limitations which should be considered regarding the mechanical properties of the weld and bar, for example, shear and normal stresses, physical constraints (no length less than zero) and maximum deflection. The constraints are as follows:

$$g_1 = \tau_d - \tau \geq 0 \tag{4}$$

$$g_2 = \sigma_d - \sigma \geq 0 \tag{5}$$

$$g_3 = b - h \geq 0 \tag{6}$$

$$g_4 = l \geq 0 \tag{7}$$

$$g_5 = t \geq 0 \tag{8}$$

$$g_6 = P_c - F \geq 0 \tag{9}$$

$$g_7 = h - 0.125 \geq 0 \tag{10}$$

where

τ_d = Allowable shear stress of weld $= 9.38 \times 10^7\ Pa$ $(13600 Psi)$;

τ = Maximum shear stress in weld;

σ_d = Allowable normal stress for beam material

$= 2.07 \times 10^8 Pa$ $(30000 Psi)$;

σ = Maximum normal stress in beam;

P_c = Bar buckling load;

F = Load = $26689N$ $(6000 lbf)$;

δ = Beam end deflection.

The first constraint (g_1) ensures that the maximum developed shear stress is less than the allowable shear stress of the weld material. The second constraint (g_2) checks that the maximum developed normal stress is lower than the allowed normal stress in the beam. The third constraint (g_3) ensures that the beam thickness exceeds that of the weld. The fourth and fifth constraints (g_4 and g_5) are practical checks to prevent negative lengths or thicknesses. The sixth constraint (g_6) makes sure that the load on the beam is not greater than the allowable buckling load. The last constraint (g_7) checks that the weld thickness is above a given minimum.

Normal and shear stresses and buckling force can be formulated as [10]:

$$\sigma = \frac{2.1952}{t^3 b} \quad (11)$$

$$\tau = \sqrt{(\tau')^2 + (\tau'')^2 + (l\tau'\tau'') \Big/ \sqrt{0.25(l^2 + (h+t)^2)}} \quad (12)$$

where

$$\tau' = \frac{6000}{\sqrt{2}hl} \quad (13)$$

$$\tau'' = \frac{6000(14+0.5l)\sqrt{0.25(l^2+(h+t)^2)}}{2\left\{0.707hl\left(l^2/12+0.25(h+t)^2\right)\right\}} \quad (14)$$

$$P_c = 64746.022(1-0.0282346t)tb^3 \quad (15)$$

τ' and τ'' are primary and secondary shear stresses.

5. Results

The empirically chosen parameters for the Bees Algorithm are given in Table 1. The search space was defined by the following intervals [11]:

$$0.125 \le h \le 5 \quad (16)$$

$$0.1 \le l \le 10 \quad (17)$$

$$0.1 \le t \le 10 \quad (18)$$

$$0.1 \le b \le 5 \quad (19)$$

With the above search space definition, constraints g_4, g_5, and g_7 are already satisfied and do not need to be checked in the code.

Table 1 Parameters of the Bees Algorithm for the welded beam design problem

Bees Algorithm parameters	Symbol	Value
Population	n	150
Number of selected sites	m	30
Initial patch size	ngh	0.1
Number of bees recruited for selected sites	nep	50
Number of iterations	g	1000

Fig 3. shows the non-dominated solutions obtained using the Bees Algorithm. The total number is 215 non-dominated solutions distributed along the Pareto front. Deb has investigated this problem [12] using the non-dominated sorting GA (or NSGA) and another different predecessor NSGA, called NASGA-II for finding multiple Pareto optimal solution (Fig 4.).

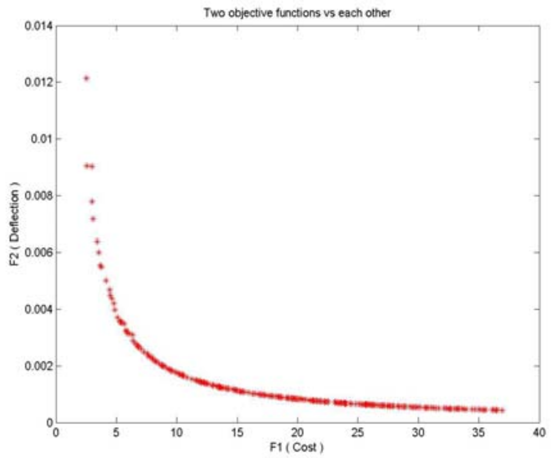

Fig 3. Non-dominated solutions obtained using the Bees Algorithm

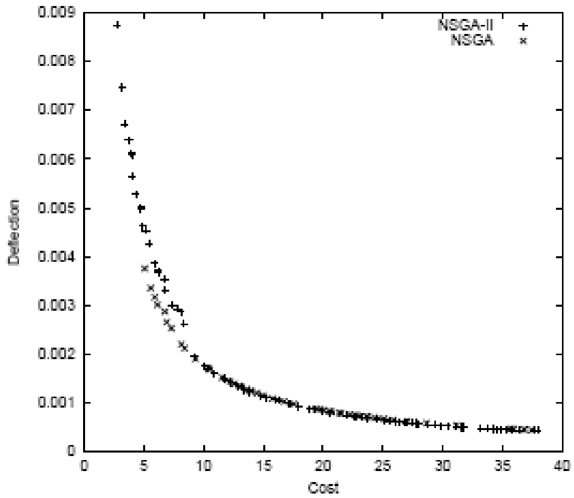

Fig 4. Non-dominated solutions obtained using the two different versions of genetic algorithms

In comparison with the number of solution found by non-dominated sorting genetic algorithms, it can bee seen that the Bees Algorithm can find more non-dominated solutions.

5. Conclusion

This paper, has described a modified version of the Bees Algorithm and its application to the search for multiple Pareto optimal solutions in a mechanical engineering problem. The Bees Algorithm found many trade-off solutions compared to the number of solutions obtained using non-dominated sorting genetic algorithms [12].

The Bees Algorithm is a computationally fast multi-objective optimiser tool for complex engineering multi-objective optimisation problems.

Indeed the Bees Algorithm can solve a multi-objective optimisation problem without any special domain information, apart from that needed to compute objective functions.

Acknowledgements

The research described in this paper was performed as part of the Objective 1 SUPERMAN project, the EPSRC Innovative Manufacturing Research Centre Project and the EC FP6 Innovative Production Machines and Systems (I*PROMS) Network of Excellence.

References

[1] Deb K. *Multi-Objective Optimization using Evolutionary Algorithms*. 2001, Chichester, UK: Wiley.

[2] Pham D.T., Ghanbarzadeh A., Koc E., Otri S., Rahim S., and Zaidi M. *Technical Note: Bees Algorithm*. 2005, Manufacturing Engineering Centre, Cardiff University: Cardiff.

[3] Pham D.T., Ghanbarzadeh A., Koc E., Otri S., Rahim S., and Zaidi M. *The Bees Algorithm, A Novel Tool for Complex Optimisation Problems*. *Proc 2nd Int Virtual Conf on Intelligent Production Machines and Systems (IPROMS 2006)*. 2006. Oxford: Elsevier, pp. 454-459.

[4] Pham D.T., Ghanbarzadeh A., Koc E., and Otri S. *Application of the Bees Algorithm to the Training of Radial Basis Function Networks for Control Chart Pattern Recognition*. *Proc 5th CIRP International Seminar on Intelligent Computation in Manufacturing Engineering (CIRP ICME '06)*. 2006. Ischia, Italy, pp. 711-716.

[5] Pham D.T., Otri S., Ghanbarzadeh A., and Koc E. *Application of the Bees Algorithm to the Training of Learning Vector Quantisation Networks for Control Chart Pattern Recognition*. *Proc Information and Communication Technologies (ICTTA'06)*. 2006. Syria, pp. 1624-1629.

[6] Pham D.T., Koç E., Ghanbarzadeh A., and Otri S. *Optimisation of the Weights of Multi-Layered Perceptrons Using the Bees Algorithm*. *Proc 5th International Symposium on Intelligent Manufacturing Systems*. 2006. Turkey, pp. 38-46.

[7] Pham D.T., Castellani M., and Ghanbarzadeh A. *Preliminary design using the Bees Agorithm*. *Proc Eighth International Conference on Laser Metrology, CMM and Machine Tool Performance, LAMDAMAP*. 2007. Cardiff: euspan Ltd, UK, pp. 420-429.

[8] Arora J.S. *Introduction to Optimum Design*. 2004, New York: Elsevier.

[9] Rekliatis G.V., Ravindrab A., and Ragsdell K.M. *Engineering Optimisation Methods and Applications*. 1983, New York: Wiley.

[10] Shigley J.E. *Mechanical Engineering Design*. 1973, Ney York: McGraw-Hill.

[11] Deb K. *Optimal Design of a Welded Beam via Genetic Algorithm.* AIAA Journal, 1991. **29**(11): pp. 2013-2015.

[12] Deb K., Pratap A., and Moitra S. *Mechanical Component Design for Multiple Objectives Using Elitist Non-Dominated Sorting GA*. 2000, Indian Institute of Technology: Kanpur, India. pp. 10.