Nature Inspired Optimisation – 06-26949
Nature-Inspired Optimisation (Extended) – 06-26948

Course Work II (10%): Programming Assignment
Engineering Design using Multi-Objective Evolutionary Algorithms

Due date: 29th March 2015: 12:00pm

Optimal design of a pinned-pinned sandwich beam

The problem is first studied by Achille Messac at Syracuse University [1] (paper can be found here). The problem aims to design a pinned-pinned sandwich beam using three materials to support a motor as shown in Figure 1. The beam is of lenth L, of with b, and symmetrical about its midplane. As illustrated in Figure 1, the variables d_1 and d_2 locate the contact of materials one and two, and two and three, respectively. The variable d_3 locates the top of the beam. The mass density ρ, Young's modulus E, and cost per unit volume c for materials one, two and three are denoted by the triplets (ρ_1, E_1, c_1), (ρ_2, E_2, c_2) and (ρ_3, E_3, c_3), which is listed in Table 1. A vibratory disturbance (at v Hz) is imparted from the motor onto the bean. The problem is to choose a set of optimal design (decision) variables $\mathbf{x} = [d_1, d_2, d_3, b, L]$ in order to (a) minimise the fundamental frequency of the vibratory disturbance; and (b) minimise the total cost of building the beam.

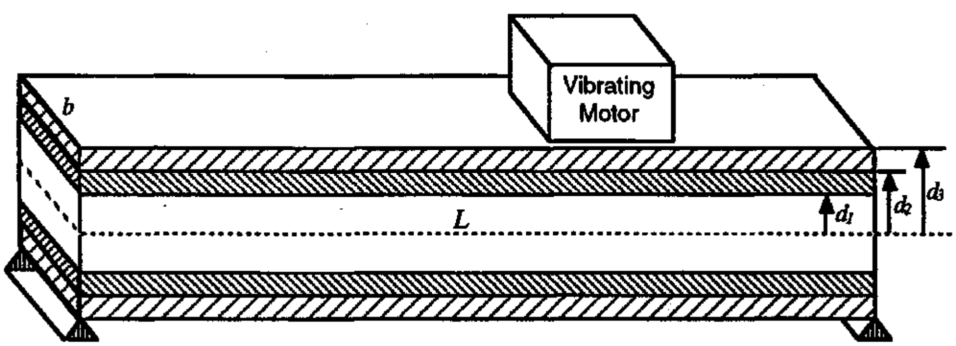

Figure 1: Pinned-pinned sandwich beam design with vibrating motor (Source: Physical Programming: Effective Optimization for Computational Design [1])

Formally, the problem is defined as:

Minimize

$$
\begin{aligned}
f_1(\mathbf{x}) &= \frac{\pi}{2L^2}\left(\frac{EI}{\mu}\right)^{\frac{1}{2}} \\
f_2(\mathbf{x}) &= 2bL[c_1 d_1 + c_2(d_2 - d_1) + c_3(d_3 - d_2)] \qquad (1)
\end{aligned}
$$

where f_1 defines the fundamental frequency and f_2 defines the total cost, and

$$
\begin{aligned}
EI &= \frac{2b}{3}[E_1 d_1^3 + E_2(d_2^3 - d_1^3) + E_3(d_3^3 - d_2^3)] \qquad (2) \\
\mu &= 2b[\rho_1 d_1 + \rho_2(d_2 - d_1) + \rho_3(d_3 - d_2)] \qquad (3)
\end{aligned}
$$

subject to:

$$g_1(\mathbf{x}) \equiv 0.3 \le b \le 0.55 \tag{4}$$
$$g_2(\mathbf{x}) \equiv 3 \le L \le 6 \tag{5}$$
$$g_3(\mathbf{x}) \equiv 2000 \le \mu L \le 2800 \tag{6}$$
$$g_4(\mathbf{x}) \equiv 0.3 \le d_3 \le 0.6 \tag{7}$$
$$g_5(\mathbf{x}) \equiv 0.01 \le d_1 \le 0.58 \tag{8}$$
$$g_6(\mathbf{x}) \equiv 0.01 \le d_2 - d_1 \le 0.58 \tag{9}$$
$$g_7(\mathbf{x}) \equiv 0.01 \le d_3 - d_2 \le 0.57 \tag{10}$$

where g_3 is the mass and g_4, g_5 and g_6 are the widths of material layers 1-3, respectively.

Table 1: Properties of materials.

Material	E_i(Pa)	c_i($/m^3)	ρ_i (kg/m^3)
1	1.60E+09	500	100
2	7.00E+10	1500	2770
3	2.00E+11	800	7780

Detailed Instructions

The assignment requires you to solve the above problem using at least two multi-objective optimisation algorithms in a programming language of your choice (e. g., Java, C, C++, Matlab). You are encouraged to use libraries such as MOEA. You also need to use the hypervolume indicator as the performance metric for comparing the performance of the two algorithms. You need to perform 30 independent runs of each algorithms. Each run should have an upper limit of 10,000 function evaluations. For each of the runs you need to record the final hypervolume value. Plot the non-dominated solutions with the best hypervolume value obtained by each algorithm among 30 runs in the objective space.

Additionally, in your report, you are required to describe and discuss your implementation and your experimental findings in a report. The report should in particular discuss the following aspects (amongst any other points which you might consider important):

- Simply from the two objective functions as defined in equation (1), can you justify the above problem is a non-trivial multi-objective optimisation problem?
- How did you implement the algorithm and the fitness function (e. g., programming language, design decisions, ...)? What data structures, algorithms or libraries did you use? What other software/scripts did you use to facilitate your experiments/analysis?
- How did you handle the constraints?
- Discuss and explain the hypervolume indicator and at least one other performance metric for comparing the performance of different multi-objective optimisation algorithms.

Requirements and Submission

Submit your coursework through the Canvas system. In case you encounter any problems with the Canvas system, please email support.

Your submission should contain

- a **single pdf file** with your report and
- a **single zip file** containing your source code and raw data. If you are using other languages than Matlab you should also provide a compiled executable,

No late submissions are allowed.

1000–2000 words are appropriate for the report (excluding any figures, tables and the list of references), however, note that this is not a hard limit. You should add a declaration that the report and source code is entirely written by yourself, unless it is noted otherwise.

The implementation and report should be completed independently and any sources used need to be clearly cited in your report. You can use a reference system of your choice, but you must be consistent. Please refer to the University's guide to referencing: `https://intranet.birmingham.ac.uk/as/libraryservices/library/referencing/icite/referencing/index.aspx`

Please also refer to the School's guidance notes on plagiarism:

`http://www.cs.bham.ac.uk/internal/students/handbook/current/#PLAG`

Assignment Marks

Marking will be based on the quality of the implementation and the report in its content, structure and language. It will be split according to the following parts of the assignment:

- Implementation (40%, based on source code and report)
- Experiments and their presentation (40%, based on raw data and report)
- Discussion of findings (20%, based on report)

Mark descriptors will roughly follow those used in project marking:

`http://www.cs.bham.ac.uk/internal/staff/handbook/ProjectGradeDescriptors.html`

References

[1] Messac, A., Physical Programming: Effective Optimization for Computational Design, AIAA Journal, Vol. 34, No. 1, Jan. 1996, pp. 149-158.