Novelty Particle Swarm Optimisation for Truss Optimisation Problems

Hirad Assimi, Frank Neumann, Markus Wagner
University of Adelaide
Adelaide, Australia
firstname.lastname@adelaide.edu.au

Xiaodong Li
RMIT University
Melbourne, Australia
xiaodong.li@rmit.edu.au

ABSTRACT

Bilevel optimisation has been successfully applied to truss optimisation to consider topology and sizing in upper and lower levels, respectively. This study proposes novelty particle swarm optimisation for the upper level to discover new designs by maximising novelty. Our experimental investigations show that our approach outperforms current state-of-the-art methods and obtains multiple high-quality solutions.

CCS CONCEPTS

• **Applied computing** → **Engineering**;

KEYWORDS

Truss, bilevel optimisation, novelty search

ACM Reference Format:
Hirad Assimi, Frank Neumann, Markus Wagner and Xiaodong Li. 2021. Novelty Particle Swarm Optimisation for Truss Optimisation Problems. In *2021 Genetic and Evolutionary Computation Conference Companion (GECCO '21 Companion), July 10–14, 2021, Lille, France.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3449726.3459572

1 INTRODUCTION

Trusses are structural frameworks carrying applied external forces on nodes to support structures in civil, aerospace and robotic applications. In truss optimisation, it is important to quickly find preliminary designs for further detailed investigation and design [8].

Weight minimisation is the most common objective, and a solution consists of subset of available connections, namely the.

Truss optimisation problems are subject to multiple constraints such as stability, failure criteria, practice design codes and manufacturing specifications. Conventional optimisation methods showed limited efficiency in solving the problem [3].

Bilevel optimisation is an efficient design approach because it can model the interaction among different aspects of the problem more explicitly. In the bilevel formulation, the upper level optimisation problem determines the truss configuration, such as topology, where the lower level optimises bars' sizing.

It has been observed that there exist multiple distinct topologies with almost equal overall weight in the truss optimisation search space [3, 5]. Therefore finding multiple equally good truss designs with respect to the topology and sizes can enable practitioners to choose according to their preferences.

In this paper we consider bilevel optimisation of topology and size of trusses subject to discrete sizes. For the lower level optimisation, we use a reliable evolutionary optimiser. For the upper level, we employ novelty particle swarm optimisation to explore the upper level. In our experiments, We show that we can find multiple distinct high-quality solutions with respect to the topology – moreover, we demonstrate new best solutions for all investigated design problems.

2 BILEVEL TRUSS OPTIMISATION PROBLEM

Bilevel truss optimisation problem nests an upper level topology optimisation problem into a lower level size optimisation problem as follows:

$$
\begin{aligned}
\text{find} \quad & \vec{x}, \vec{y}, \quad \vec{x} \in \{0,1\}^m,\ \vec{y} \in S^m \\
\text{optimise} \quad & F(\vec{x}, \vec{y}) \\
\text{subject to} \quad & G_1(\vec{x}),\ G_2(\vec{x}),\ G_3(\vec{x}) \\
\text{where} \quad & G_1(\vec{x}) = \text{True} \iff \text{Essential nodes are in truss} \\
& G_2(\vec{x}) = \text{True} \iff \text{Truss is externally stable} \\
& G_3(\vec{x}) = \vec{y} \in \operatorname{argmin}\{W(\vec{x}, \vec{y}),\ g_j(\vec{x}, \vec{y}) \le 0,\ j = 1,\ 2,\ 3\}
\end{aligned}
$$

where $\vec{x}$ refers to the binary topology variable in the upper level where it shows if a truss bar is active (1) or excluded (0). We can show the upper bound of topology as the ground structure where all bars are active for m=8 as $\vec{x} = [11111111]$. $\vec{y}$ denotes the sizing variable in the lower level optimisation problem. The elements of $\vec{y}$ should be selected from an available size set (S). $F(\vec{x}, \vec{y})$ shows the objective function considered in the upper level such as weight minimisation used in [5] or maximising novelty in this study and, $W(\vec{x}, \vec{y}) = \rho \sum_{i=1}^{m} x_i y_i l_i$ refers to the truss weight where, ρ and l show the material density and length of a bar, respectively.

Solutions in the upper level should satisfy the topology constraints for feasibility namely $G_1(\vec{x})$ and $G_2(\vec{x})$, we refer the reader to [6] for detailed explanations.

3 BILEVEL NOVELTY SEARCH

NdPSO is a Novelty-Driven variant of PSO employing novelty search to drive particles toward novel solutions that are different from previously encountered ones [4]. The main idea is to explore the search space by ignoring objective-based fitness functions and rewarding novel individuals. NdPSO employs core principles of PSO and mainly replaces the objective function with novelty evaluation. Note that personal best (p_t) and global best (p_g) value in NdPSO

GECCO '21 Companion, July 10–14, 2021, Lille, France

ACM ISBN 978-1-4503-8351-6/21/07…$15.00
https://doi.org/10.1145/3449726.3459572

Algorithm 1: Novelty PSO for Bilevel Truss Optimisation

```
Randomly generate the initial population of Binary PSO
Set the velocity of particles in population
Evaluate the novelty score for each particle (z⃗^i)
y^i = loweroptimiser(z^i)
Store W(z⃗^i, y⃗^i)
Update p_t and p_g
Update the archive
repeat
    for i=1 to population size do
        Update position of particle
        Update velocity of particle
        Evaluate novelty score of the particle
        y_i = loweroptimiser(x_i)
        Store W(x⃗_i, y⃗_i)
        Update p_t^i and p_g according to novelty score
        Update the archive
until termination criterion is met
```

show a dynamic behaviour. We use NdPSO in the upper level of truss optimisation to discover novel topology designs.

Our proposed approach works as follows (see Algorithm 1). Initially, the binary PSO generates a random population of binary strings. The particles' velocities are drawn randomly from $[-v, v]$. Then, the novelty score is computed for particles with respect to the archive. Because all particles are feasible, *loweroptimiser* computes the corresponding optimal size for the upper level topology using an evolutionary optimiser in lower level for size optimisation [1]. Next, we update the archive with the current population. Then, the position and velocity of particles are updated, and the above process repeats till the termination criterion is met.

4 EXPERIMENTAL RESULTS

We apply our method to two truss problem known as 15-bar and 72-bar trusses which are a symmetric and non-symmetric problems. We refer the reader to [2] for details on the benchmarks.

Table 1 shows our findings by the proposed bilevel novelty search compared with other methods. We can see that designs (b) and (c) find the same weight, and they are symmetric around the vertical axis with respect to the topology and size of bars. Both designs remove 6 bars from the design space, and symmetrically they eliminate nodes 2 and 4 from the design space, respectively. Design (d) eliminates five bars and node 2 from the design space. Design (a) is the best-found design eliminates five bars in the design space and provides a lighter solution compared with other methods.

Table 2 shows our findings for 72-bar truss problem. We can see that designs (b) and (c) identify five groups of bars as redundant, including four common groups. This will lead to the elimination of 16 bars (out of 72) from the design space. Design (a) combines the identified redundant bars in designs (b) and (c) and removes 20 bars in total from the design space and achieves a lighter solution.

Table 1: Comparison of optimised designs for 15-bar truss.

A : mm^2	[10]	[2]	This Study			
			(a)	(b)	(c)	(d)
A_1	308.6	113.2	113.2	-	113.2	-
A_2	174.9	113.2	113.2	-	113.2	-
A_3	338.2	113.2	-	113.2	-	113.2
A_4	143.2	113.2	-	113.2	-	113.2
A_5	736.7	736.7	736.7	736.7	736.7	736.7
A_6	185.9	113.2	-	113.2	113.2	113.2
A_7	265.9	113.2	143.2	143.2	143.2	143.2
A_8	507.6	736.7	736.7	736.7	736.7	736.7
A_9	143.2	113.2	113.2	-	113.2	-
A_{10}	507.6	113.2	-	113.2	-	-
A_{11}	279.1	113.2	113.2	145.9	145.9	145.9
A_{12}	174.9	113.2	113.2	-	-	-
A_{13}	297.1	113.2	-	-	-	113.2
A_{14}	235.9	334.3	334.3	334.3	334.3	334.3
A_{15}	265.9	334.3	334.3	334.3	334.3	334.3
Best weight (kg)	142.12	105.74	89.899	90.223	90.223	91.874

Table 2: Comparison of optimised designs for 72-bar truss.

A : in.^{2}	[9]	[7]	[2]	This Study		
				(a)	(b)	(c)
Best Weight (lb)	400.66	387.94	385.54	368.16	369.15	370.15

REFERENCES

[1] Ali Ahrari and Kalyanmoy Deb. 2016. An improved fully stressed design evolution strategy for layout optimization of truss structures. *Computers & Structures* 164 (2016), 127–144.

[2] M Cheng. 2016. A Hybrid Harmony Search algorithm for discrete sizing optimization of truss structure. *Automation in Construction* 69 (Sept. 2016), 21–33.

[3] Kalyanmoy Deb and Surendra Gulati. 2001. Design of truss-structures for minimum weight using genetic algorithms. *Finite Elements in Analysis and Design* 37, 5 (2001), 447 – 465. https://doi.org/10.1016/S0168-874X(00)00057-3 Genetic algorithms and finite elements in engineering.

[4] Diana F. Galvao, Joel Lehman, and Paulo Urbano. 2016. Novelty-Driven Particle Swarm Optimization. In *Artificial Evolution*, Stéphane Bonnevay, Pierrick Legrand, Nicolas Monmarché, Evelyne Lutton, and Marc Schoenauer (Eds.). Springer International Publishing, Cham, 177–190.

[5] Md. Jakirul Islam, Xiaodong Li, and Kalyanmoy Deb. 2017. Multimodal Truss Structure Design Using Bilevel and Niching Based Evolutionary Algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '17)*. Association for Computing Machinery, New York, NY, USA, 274–281. https://doi.org/10.1145/3071178.3071251

[6] Hamid Khayyam, Ali Jamali, Hirad Assimi, and Reza N. Jazar. 2020. *Genetic Programming Approaches in Design and Optimization of Mechanical Engineering Applications*. Springer International Publishing, Cham, 367–402. https://doi.org/10.1007/978-3-030-18963-1_9

[7] Kang Seok Lee, Zong Woo Geem, Sang-ho Lee, and Kyu-woong Bae. 2005. The harmony search heuristic algorithm for discrete structural optimization. *Engineering Optimization* 37, 7 (Oct. 2005), 663–684.

[8] Mathias Stolpe. 2016. Truss optimization with discrete design variables: a critical review. *Structural and Multidisciplinary Optimization* 53, 2 (2016), 349–374.

[9] S J Wu and P T Chow. 1995. Steady-state genetic algorithms for discrete optimization of trusses. *Computers & Structures* 56, 6 (1995), 979–991.

[10] Y N Zhang, P Liu, B Liu, C Y Zhu, and Y Li. 2005. *Application of improved hybrid genetic algorithm to optimized design of architecture structures.* Huanan Ligong Daxue Xuebai(Ziran Kexue ….