



## APPLICATION OF THE METAHEURISTIC TUNICATE SWARM ALGORITHM IN SOLVING APPLIED MECHANICS PROBLEMS

Received: 05 October 2021 / Accepted: 29 November 2021

**Abstract:** In this paper, the principles of a metaheuristic algorithm based on tunicate swarm behavior are shown. The Tunicate Swarm Algorithm (TSA for short) was used for solving problems in applied mechanics (speed reducer, cantilever beam and three-dimensional beam optimization). In the end, a comparison of results obtained by TSA and results obtained by other methods is given.

**Key words:** metaheuristics, tunicate, algorithm, mechanics

**Primena metaheurističkog algoritma roja plastaša za rešavanje problema primenjene mehanike.** U ovom radu su izloženi principi rešavanja metaheurističkog algoritma zasnovanog na ponašanju roja plastaša. Primenom algoritma roja plastaša (TSA) rešeni su problemi u okviru primenjene mehanike (optimizacija jednostepenog reduktora, konzolne grede i trodimenzionalnog nosača). Na kraju rada će biti prikazani rezultati dobijeni metodom TSA u odnosu na prethodno dobijene rezultate sa drugim metodama.

**Ključne reči:** metaheuristika, plastaši, algoritam, mehanika.

### 1. INTRODUCTION

Most engineering problems can be formulated as optimization problems. To solve optimization problems, different methods have been studied in mathematical programming, operations research, etc.

The classification of metaheuristic algorithms [4, 6, 7, 8, 9, 10] is shown in figure 1.



Fig. 1. Classification of metaheuristic algorithms

In this paper, the Tunicate swarm algorithm is used to solve the following engineering problems.

The first problem to be solved is speed reducer optimization, having the goal of minimizing reducer weight in accordance with bending stress constraints of gear teeth, surface stresses, transverse deflections of shafts and stresses in shafts. This problem was first analyzed and solved by Coello using GA [1].

The second engineering problem is cantilever beam optimization, where minimal weight that fulfills the constraints is sought after. Gandomi has solved this problem using the Cuckoo Search Algorithm (CSA) [2].

The last engineering problem that will be considered in this paper is optimization of a cantilever beam. The goal of this optimization is to minimize cross-section heights for each beam element. Miodragovic used FA algorithm to solve this problem [3].

### 2. TUNICATE SWARM ALGORITHM

Tunicates, commonly called sea squirts, are a group of marine animals that spend most of their lives attached to docks, rocks or the undersides of boats. The name, "tunicate" comes from the firm, but flexible body covering, called a tunic. They have the shape of a small barrel, and have two openings, called siphons, enabling water to stream through them so that they can feed on planktons. The tunicate moves through the ocean by jet propulsion of water through the siphons. Tunicate is only animal to move around the ocean with such fluid jet like propulsion. Tunicates are often found at depth of 500–800 m and migrate upwards in the upper layer of surface water at night.

In the Tunicate Swarm Algorithm, tunicates are in search of food source. The food source is indicated by the best search agent, while the tunicates seeking the food source are all other search agents. The jet propulsion behavior must satisfy three conditions:

1. Avoiding conflicts between search agents
2. Movement towards the best neighbor
3. Converging towards the best search agent

In order to avoid conflicts between search agents, the following model is applied:

$$\vec{A} = \frac{\vec{G}}{M} \quad (1)$$

$$\vec{G} = c_2 + c_3 - \vec{F} \quad (2)$$

$$\vec{F} = 2 * c_1 \quad (3)$$

$$\vec{M} = [P_{\min} + c_1 * P_{\max} - P_{\min}] \quad (4)$$

where  $\vec{A}$  - new agent position,  $\vec{G}$  - gravity force,  $\vec{M}$  - social forces,  $\vec{F}$  - water flow advection,  $c_1, c_2, c_3$  - random numbers in range  $[0,1]$ ,  $P_{\min}, P_{\max}$  - initial and subordinate speeds for social interaction.

Usually, the values for  $P_{\min}, P_{\max}$  are 1 and 4, respectively.

To avoid the conflict between neighbors, the search agents move towards the direction of the best neighbor.

$$\vec{PD} = \left| \vec{FS} - rand * P_p(x) \right| \quad (5)$$

where  $\vec{PD}$  - distance between the food source and search agent,  $\vec{FS}$  - position of food source,  $rand$  - random number in range  $[0,1]$ ,  $P_p(x)$  - position of tunicate.

The search agents move towards the food source using the following equation:

$$P_p(x') = \begin{cases} \vec{FS} + \vec{A} * \vec{PD}, & \text{if } rand \geq 0.5 \\ \vec{FS} - \vec{A} * \vec{PD}, & \text{if } rand < 0.5 \end{cases} \quad (6)$$

Where  $P_p(x')$  is the updated position of the search agent with respect to the food source. In order to simulate the swarm behavior, the first two optimal best solutions are recorded, and all the other solutions update their position with respect to those positions. This is done using the following equation:

$$\vec{P}_p(x+1) = \frac{\vec{P}_p(x) + \vec{P}_p(x+1)}{2 + c_1} \quad (7)$$

The TSA consists of following steps:

**Step 1:** Initialize the tunicate population  $\vec{P}_p$ .

**Step 2:** Choose the initial parameters and maximum number of iterations.

**Step 3:** Calculate the fitness value of each search agent.

**Step 4:** After computing the fitness value, the best search agent is explored in the given search space.

**Step 5:** Update the position of each search agent using the equation for  $\vec{P}_p(x+1)$ .

**Step 6:** Adjust the updated search agent which goes beyond the boundary in a given search space.

**Step 7:** Compute the updated search agent fitness value. If there is a better solution than the previous optimal solution, then update  $P_p$ .

**Step 8:** If the stopping criterion is satisfied, then the algorithm stops. Otherwise, repeat the Steps 5–8.

**Step 9:** Return the best optimal solution which is obtained so far.

### 3. EXPERIMENTAL ENGINEERING EXAMPLES FOR OPTIMIZATION

This chapter will present certain examples of engineering problems, such as: optimization of speed reducer, cantilever beam and three dimensional beam. The basis of the problem, the objective function, variable parameters that should be found as well as the constraints that should be respected will be shown.

The goal of speed reducer optimization is minimizing the reducer weight whilst fulfilling all the defined constraints.

In Figure 2 a schematic view of speed reducer is shown.

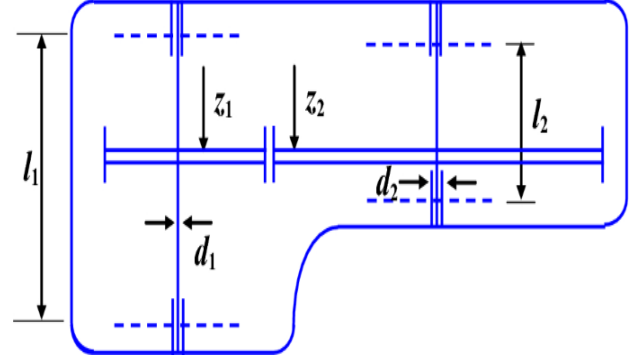


Fig. 2. Speed reducer design problem

Project variables for the speed reducer problem are: the width between the shafts ( $x_1$ ), the module of the teeth ( $x_2$ ), the number of teeth in the pinion ( $x_3$ ), the length of the first shaft between the bearings ( $x_4$ ), the length of the second shaft between the bearings ( $x_5$ ), the diameter of the first shaft ( $x_6$ ) and the diameter of the second shaft ( $x_7$ ).

Goal function to be minimized is defined as:

$$f(x) = 0.7854x_1x_2^2(3.3333x_3^2 + 14.933x_3 - 43.0934) - 1.508x_1(x_6^2 + x_7^2) + 7.4777(x_6^3 + x_7^3) + 0.7854(x_4x_6^2 + x_5x_7^2) \quad (8)$$

subject to the following constraints:

$$g_1(x) = \frac{27}{x_1x_2^2x_3} - 1 \leq 0 \quad (9)$$

$$g_2(x) = \frac{397.5}{x_1x_2^2x_3^2} - 1 \leq 0 \quad (10)$$

$$g_3(x) = \frac{1.93x_5^3}{x_2x_3x_7^4} - 1 \leq 0 \quad (11)$$

$$g_4(x) = \frac{1.93x_5^3}{x_2x_3x_7^4} - 1 \leq 0 \quad (12)$$

$$g_5(x) = \frac{\left( \left( \frac{475x_4}{x_2x_3} \right)^2 + 16.9 \times 10^6 \right)^{1/2}}{110x_6^3} - 1 \leq 0 \quad (13)$$

$$g_6(x) = \frac{\left( \left( \frac{475x_4}{x_2x_3} \right)^2 + 157.5 \times 10^6 \right)^{1/2}}{85x_7^3} - 1 \leq 0 \quad (14)$$

$$g_7(x) = \frac{x_2x_3}{40} - 1 \leq 0 \quad (15)$$

$$g_8(x) = \frac{5x_2}{x_1} - 1 \leq 0 \quad (16)$$

$$g_9(x) = \frac{x_1}{12x_2} - 1 \leq 0 \quad (17)$$

$$g_{10}(x) = \frac{15x_6 + 1.9}{x_4} - 1 \leq 0 \quad (18)$$

$$g_{11}(x) = \frac{1.1x_7 + 1.9}{x_5} - 1 \leq 0 \quad (19)$$

$$2,6 \leq x_1 \leq 3,6 ; 0,7 \leq x_2 \leq 0,8 ; 17 \leq x_3 \leq 28$$

$$7,3 \leq x_4 \leq 8,3 ; 7,3 \leq x_5 \leq 8,3 ; 2,9 \leq x_6 \leq 3,9$$

$$5,0 \leq x_7 \leq 5,5 \quad (20)$$

In Figure 4 a schematic view of cantilever beam is shown.

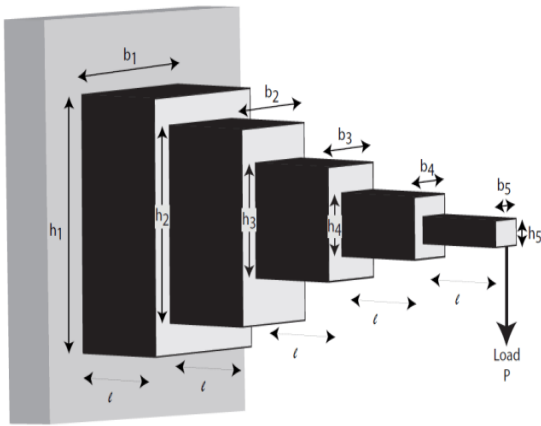


Fig. 4 . Cantilever beam design problem

As seen in Figure , the cantilever beam consists of five hollow, box shaped bearings with a square shaped frame. Project variables are lengths of the five squares( $x_1, x_2, x_3, x_4, x_5$ ) which make up the cantilever beam.

Objective function to be minimized, as defined in [5]:

$$f(\vec{x}) = 0,6224(x_1 + x_2 + x_3 + x_4 + x_5), \quad (21)$$

Whilst the only constraint for this problem being:

$$g(\vec{x}) = \frac{61}{x_1^3} + \frac{27}{x_2^3} + \frac{19}{x_3^3} + \frac{7}{x_4^3} + \frac{1}{x_5^3} - 1 \leq 0, \quad (22)$$

$$0,01 \leq x_1, x_2, x_3, x_4, x_5 \leq 100, \quad (23)$$

The last problem consists of minimizing cross-section heights of all elements of a cantilever beam, which is shown in Figure 3 .

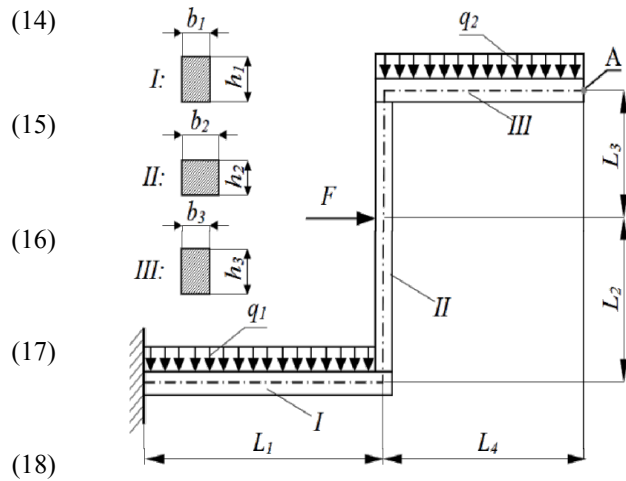


Fig. 3. Three dimensional beam design problem

A vertical shift of point A is defined in advance, having a specified upper limit. The beam is under continual load ( $q_1, q_2$ ) on horizontal parts of the beam, as well as horizontal force F, which affects the vertical part of the beam.

Goal function to be minimized is defined as:

$$f(\mathbf{X}) = 0.8x_1 + x_2 + 0.8x_3, \quad (24)$$

subject to the following constraints:

$$u_1(\mathbf{X}) = \left[ \frac{11.2480 \cdot 10^{-3}}{x_1^3} + \frac{3.5399 \cdot 10^{-3}}{x_2^3} + \frac{0.3840 \cdot 10^{-3}}{x_3^3} \right] \leq 0.05[m], \quad (25)$$

$$0.1 \leq x_1 \leq 0.9,$$

$$0.1 \leq x_2 \leq 0.9, \quad (26)$$

$$0.1 \leq x_3 \leq 0.9,$$

#### 4. OPTIMIZATION RESULTS

In this section, the results obtained by using TSA algorithm on previously defined engineering problems is given.

A detailed display of the results obtained by TSA and a comparison with several results obtained by other methods, for the problem of speed reducer, three dimensional beam and cantilever beam are shown in Table 1,2 ,3.

Variables	GWO[4]	GOA[5]	WCA[6]	TSA
$x_1$	3.502	3.5	3.5	<b>3.505</b>
$x_2$	0.7	0.7	0.7	<b>0.7</b>
$x_3$	17	17	17	<b>17</b>
$x_4$	7.333	7.3	7.3	<b>7.3</b>
$x_5$	7.8	7.8	7.715	<b>7.8</b>
$x_6$	3.351	3.350	3.350	<b>3.352</b>
$x_7$	5.288	5.287	5.286	<b>5.301</b>
$f(x)$	2998.299	2996.964	2994.471	<b>3007.327</b>

Table 1. Comparison of results for the speed reducer

Variables	ALO[7]	GOA[5]	MMA[8]	TSA
$x_1$	6.018	6.011	6.010	<b>5.973</b>
$x_2$	5.311	5.312	5.300	<b>5.258</b>
$x_3$	4.488	4.483	4.490	<b>4.514</b>
$x_4$	3.497	3.502	3.490	<b>3.529</b>
$x_5$	2.158	2.163	2.150	<b>2.203</b>
$f(x)$	1.339	1.339	1.340	<b>1.340</b>

Table 2. Comparison of results for the cantilever beam

Variables	ANSYS[9]	GOA[7]	FA[3]	TSA
$x_1$	0.80458	0.80453	0.89071	<b>0.8041</b>
$x_2$	0.44054	0.56995	0.56197	<b>0.5686</b>
$x_3$	0.51398	0.34588	0.32444	<b>0.3480</b>
$f(x)$	1.4618	1.4903	1.4900	<b>1.4903</b>

Table 3. Comparison of results for the three dimensional beam

## 5. CONCLUSION

This paper deals with the TSA optimization algorithm and applies it to a few engineering design examples: speed reducer, cantilever beam, and three dimensional beam.

The obtained results were compared to latest papers published in SCI list journals.

This algorithm was used to obtain optimal or near-optimal results, as shown in the examples. Further developing of this algorithm can be used to redefine and upgrade this method, as to gain better results.

## 6. REFERENCES

- [1] Carlos A. Coello Coello: Constraint-handling in genetic algorithms through the use of dominance based tournament selection. *Advanced Engineering Informatics*, 16, 193–203, 2002.
- [2] A. Gandomi, X. S. Yang, A. H. Alavi, *Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems*, Springer-Verlag London, 2011.
- [3] G. Miodragović, *Advanced bio-inspired algorithms development for solving optimization problems in applied mechanics*, doctoral thesis, Faculty of Mechanical and Civil Engineering Kraljevo, University of Kragujevac (2015).
- [4] B. Milenković, M. Krstić, Dj. Jovanović: *Primena algoritma sivog vuka za rešavanje inženjerskih optimizacionih problema*. *Tehnika*, 2021., Vol.76,Br.1. str.50-57, ISSN 0040-2176, DOI:10.5937/tehnika2101050M
- [5] S. Shahrzad, Seyedali Mirjalili and Andrew Lewis, *Grasshopper Optimisation Algorithm: Theory and Application*, *Advances in Engineering Software*, Volume 105, March 2017, pp 30-47.
- [6] H.Eskandar,A.Sadollah,A.Bahreininejad,M.Hamdi: *Water cycle algorithm – A novel metaheuristic optimization method for solving constrained engineering optimization problems*. *Computers and Structures* 2012.

- [7] S. Mirjalili, *The Ant Lion Optimizer*, *Adv Eng Software* 2015;83:80-98.
- [8] H. Chickermane, H. Gea, *Structural optimization using a new local approximation method*, *Int J Number Methods Eng* 1996; 39:829-46.
- [9] F. Fedorik, *Using optimizations algorithms by designing structures*, doctoral thesis, Institute of Structural Mechanics, Faculty of Civil Engineering Brno, University of Technology, (2013), 119-140.
- [10] <https://www.mathworks.com/matlabcentral/fileexchange/75182-tunicate-swarm-algorithm-tsa>

**Authors: Mladen Krstić<sup>1</sup>, Branislav Milenković<sup>2</sup>, Đorđe Jovanović<sup>3</sup>,**

<sup>1</sup>University of Kragujevac, Faculty of Mechanical and Civil Engineering, Dositejeva 19, 36000 Kraljevo, Serbia, Phone.: +381 36 383-269, Fax: +381 36 383-380.

<sup>2</sup>University UNION-Nikola Tesla, Faculty of Applied Science, Dušana Popovića 22a, 18000 Niš, Serbia, Phone.: +381 18 278-000, Fax: +381 18 553-000.

<sup>3</sup>Mathematical institute of Serbian Academy of Sciences and Arts, Kneza Mihaila 36, 11000 Beograd, Serbia, Phone.: +381 11 236-0170, Fax: +381 11 218-6105.

E-mail: [mladenkrstic994@gmail.com](mailto:mladenkrstic994@gmail.com)  
[bmilenkovic@fpn.rs](mailto:bmilenkovic@fpn.rs)  
[dj.jovanovic@mi.sanu.ac.rs](mailto:dj.jovanovic@mi.sanu.ac.rs)