Accelerated Particle Swarm Optimizer for Optimizing Problems of Structural Engineering

Meenu Sharma (Dwivedi)¹, Shahnwaz Alam² and M. I. Khan³
¹,² and ³. Department of Mechanical Engineering, Integral University, Lucknow, India (U.P.), India.

Abstract
The aim of the present work is to find a solution to non-linear constrained optimization problems of structure. Constrained optimization difficulties are practical shortcomings. The loopholes of traditional numerical methods are being removed by heuristic methods as no requirement of the functional derivatives is desired and approaches to the global way out. This article presents a “penalty guided Accelerated Particle Swarm Optimization (APSO) algorithm” to search the problem’s optimal solution in the feasible region of whole search domain. There is numerical result and comparison of the structural design optimization problems. The way out by the current perspective proves to be the better than other techniques and it can be said that our findings show better solutions to engineering problems than those earlier obtained using current algorithms.

1. INTRODUCTION

Problems based on structural engineering are extremely nonlinear, employing mixed design variables under complex restraints, where any other way rendered by calculus becomes inconsistent [13]. Design optimization is the technique to discover the most favorable parameters, to have maxima minima of desired function, subjected to constraints. Such problem of optimization is given the name of constrained optimization problems or nonlinear programming problems.

Efficacy of restrained optimization algorithms is that it unveils the optimization problems related to engineering design. These nonlinear engineering problems have been explored by several investigators that used individual approaches to solve them: Branch and Bound using Sequential Linearization Algorithm [38], SQP [33], Recursive Quadratic Programming [37], Integer discrete continuous nonlinear Programming [39], Nonlinear mixed discrete Programming [1]. Problems generally have miscellaneous (e.g., continuous and discrete) input variables, nonlinear desired functions and nonlinear restraints. Restraints are well known in engineering design problems as without them the problem becomes very hard to solve.

A recent bio inspired metaheuristic is Accelerated Particle Swarm Optimization (APSO) which finds its way in highly competitive different types of problems. Moreover, a very little importance has been given to this technique in structural problems in comparison to other areas. The way to manage restraints, discussed in this work gives method for position and velocity enhancement of particle.

The augmented Lagrange multiplier method with Powell’s and Fletcher and Reeves Conjugate Gradient method is combined by Kannan and Kramer [12] to solve the optimization problems. Proposal of nonlinear branch and restrained algorithms related to
integer programming to satisfy the miscellaneous-integer optimization problems had been given by Sandgren [33]. Arora described constraint correction at the constraint’s cost method for solving numerical optimization issues [2]. The present technique renders fruitful approach for conceptual problems with little acceptable drawbacks.

Appropriate and effective method to find the optimization solution is Heuristic method. These are mere approximations, but they eradicate the differential coefficient of the desired functions and restraints in structural problems. The heuristics technique comprise of Accelerated Particle Swarm Optimization (APSO), genetic algorithms (GA), simulated annealing (SA), tabu search (TS), particle swarm optimization (PSO), Harmony search (HS), ant colony optimization (ACO) etc. In this, monotonicity of the design variables and activities of the constraints determined by the theory of monotonicity analysis are modeled in the fuzzy proportional-derivative controller optimization engine using generic fuzzy logics.

Accelerated Particle Swarm Optimization (APSO) is one of the meta heuristic ways proposed by Xin-She Yang at Cambridge University in 2008. APSO is able to discover an excellent optimal solution due to memory, multi character, local search and solution improvement mechanism. Current analysis declares APSO as better technique. Artificial bee colony is used in the present work and proves to be beneficial and efficient one.

2. LITERATURE REVIEW

The algorithm is examined (E02 and E04) by Guo et al. [5]. The solution brought for E02 is higher than real values and for E04 they are unmatched with current findings, as use of so many constraints becomes brainteaser.

The concept in the method lies in the interaction for social improvement. E01, E02 and E03 is deployed in current work and gives optimal known values. The number of fitness function evaluations was 19,259 for E01, 19,154 for E02 and 12,630 for E03. # This method is the preferred one, as it is accurate due to its convergent rate . The employed difficulty level considered is E01, E03 and E04 took 300,000, 200,000 and 50,000 critical examinations. The values obtained serves to be inconsistent under the said parameter.

The algorithm was employed for solving E01, E02, E03 and E04 with 320000, 80000, 36000 and 3600 tests respectively. The results of the evaluation gave expected results but the number of evaluations needed is too high.

Xin-She Yang proposed an APSO algorithm with two new perturbation operators aiming to prevent premature convergence. The life of particles is extended after the handling of contraints. The author reference three algorithms which obtained good results for the problems adopted in their study: two APSO based algorithms and a Differential Evolution (ED) algorithm. One of the APSO based approaches compared [8] used three of the problems adopted here (E01, E02 and E04), performing 200,000 objective function evaluations. The other APSO based method compared was tested with the same set of problems and the best known values were reached for E02 and E04 after 30,000 objective function evaluations. The ED algorithm [9] reported better results with 30,000 evaluations for the four problems. This equal number of evaluations was performed by the algorithm proposed by Xin-She Yang and gave better results in knowledge till now. This compels us to use the appropriate for comparison of the utility of said technique ( results are summarised and noted later).

2. ENGINEERING OPTIMIZATION MECHANISM

2.1. Constraint Handling Method

Due to presence of both inequalities it is not accessible for solution. Despite the popularity of penalty functions, they have shortcomings out of which the main one is necessity of having large number of parameters to be adjusted and the search for the equality of the objective and penalty functions is difficult. The finding process is considerably at low pace devoid of expected results. In order to eradicate this
problem, these algorithms using concept of parameter free penalty functions can be expressed as
\[
F(x) = \begin{cases} 
  f(x) & \text{if } x \in S \\
  f_w + \sum_{z=1}^{p+q} g_z(x) & \text{if } x \notin S
\end{cases}
\] (1)

where \( x \) are solutions and \( f_w \) is the least desired solution.

2.2. Structural Design Optimization

Mechanical design optimization problems can be formulated as a nonlinear programming (NLP) problem. Unlike generic NLP problems which only contain continuous or integer variables, mechanical design optimizations usually involve continuous, binary, discrete and integer variables. The binary variables are usually deployed to formulate the design problem to select alternative options. The discrete variables represent standardization constraints such as the diameters of standard sized bolts. Integer variables usually occur when the numbers of objects are design variables, such as the number of gear teeth. Adopting the different variables can be formulised as
\[
\text{Minimize } f(x) \\
\text{subject to } \quad h_k(x) = 0 ; \quad k = 1, 2, \ldots, p \\
g_j(x) \leq 0 ; \quad j = 1, 2, \ldots, q \\
l_i \leq x_i \leq u_i ; \quad i = 1, 2, \ldots, n
\] (2)

where \( x = [x_1, x_2, \ldots, x_n]^T \) denotes the decision solution vectors; \( f \) is the objective function; \( l_i \) and \( u_i \) are the minimum and maximum permissible values for the \( i \)-th variable respectively; \( p \) is equality number and \( q \) is the number of inequality constraints. Let \( S = \{ x \mid gz(x) \} \) or = 0, z = 1, 2, p+q, li < xi > ui} accessible solution and \( gz \) be the set of equalities and inequalities constraints.

3. ACCELERATED PARTICLE SWARM OPTIMIZATION (APSO)

3.1. PSO

There are lots of PSO variants and hybrid algorithms by combining PSO with popular combination of variants with other opted procedure. Trajectories of particles are examined by the technique. The swarming particle’s advancement consists of twice greater parts: a stochastic component and a deterministic component. \( g_\_ \) gives best value and \( x_\_ \) locating the particle for earlier results, while simultaneously it has a tendency to move randomly.

Let \( x_i \) and \( v_i \) be the position vector and velocity for particle \( i \), respectively. Vector is expressed as
\[
v_i^{t+1} = v_i^t + \alpha \epsilon_n + \beta (g^* - x_i^t),
\] (3)

where \( \alpha \) and \( \epsilon_n \) where vectorial values are between \( 0 \) and \( 1 \). Thenwe can take, say, \( \alpha = \beta = 2 \).

3.2. Accelerated PSO

Xin-She Yang at Cambridge University in 2008 put forward APSO which is metaheuristic algorithm.

For speedy convergence of method only best results are considered. Thus APSO [6,7], the velocity vector is expressed as
\[
v_i^{t+1} = v_i^t + \alpha \epsilon_n + \beta (g^* - x_i^t).
\] (4)

where second one is given new place by en. The new position is simply written as
\[
x_i^{t+1} = x_i^t + v_i^{t+1}.
\] (5)

For increasing the rate of convergence, we wrote the new location in a step
\[
x_i^{t+1} = (1 - \beta) x_i^t + \beta g^* + \alpha \epsilon_n.
\] (6)

The same order of convergences is employed by this simpler version. Typically, \( = 0.1L < \alpha 0.5L \) where \( L \) is the scale of each variable, while \( = 0.1 < \alpha 0.7 \) is sufficient for most applications. It is important to point out that velocity disappear in equation (6), and without dealing with initialization of velocity vectors. Therefore, APSO is much simpler. On comparison with many PSO variants, APSO uses only two parameters, and the mechanism is easy to grasp.

Improving the accelerated PSO is to reduce the randomness as iterations proceed. This clears that
monotonically decreasing function can be used such as
\[
\alpha = \alpha_0 e^{-\gamma t}, \quad (7)
\]
\[
\alpha = \alpha_0 \gamma^t, \quad (0 < \gamma < 1), \quad (8)
\]
with \(\alpha_0 \approx 0.5 \sim 1\) is foremost value and \(t\) is time steps.

\(0 < \gamma < 1\) is holding value [7]. Eg. We use
\[
\alpha = 0.7t \quad (9)
\]
Where \(t \in [0, t_{max}]\) and \(t_{max}\) is the maxima.

4. NUMERICAL EXAMPLES
The presented algorithm is implemented in MATLAB (Math works) and the program has been run on a T6400 @ 2GHz Intel Core(TM) 2 Duo processor with 2GB of Random Access Memory (RAM).

5. PARAMETER AND RESULTS
ANALYSIS
Engineering design optimization problems with using set of 4 evaluates the suitability of algorithm. The clear declaration of problems is in the appendix.

APSO used standards: swarm size = 8 particles, neighbourhood size = 4, inertia factor \(w = 0.9\), personal learning factor and social learning factors for \(c_1, c_2\) and \(c_3\) were set to 1.8. Numerous experiments empirically derived parameter settings. A fair comparison of results is done with special data. Those values were obtained by Hernandez Aguirre et al. and Mezura et al. Given results into the tables shown next as “COPSO” and “Mezura”, respectively.

The said method used 30000 evaluations, while it is larger value. Table 1 and 2 gives the result of our work. We reference those results into the tables shown. The values obtained are shown in table 2 and 3 for E01 problem and 1 for standard APSO.

Table (1): Accelerated PSO

| Best estimates: | 2.1906 | 3.6934 | fmin = -0.79867 | Iteration = 1 |
| Best estimates: | 2.1623 | 3.8147 | fmin = 0.77543 | Iteration = 2 |
| Best estimates: | 2.2154 | 2.9678 | fmin = -0.97877 | Iteration = 3 |
| Best estimates: | 2.1942 | 2.5934 | fmin = -0.87685 | Iteration = 4 |
| Best estimates: | 2.122 | 2.6495 | fmin = -0.96437 | Iteration = 5 |
| Best estimates: | 2.2039 | 2.71 | fmin = -1.2139 | Iteration = 17 |
| Best estimates: | 2.2033 | 2.7119 | fmin = -1.2141 | Iteration = 18 |
| Best estimates: | 2.2028 | 2.7116 | fmin = -1.2141 | Iteration = 19 |

Table (2): PSO Solution vector for E01 (welded beam)

| Best Particle Value >> | 8.8412 |
| Best Particle Value >> | 8.8252 |
| Best Particle Value >> | 8.8125 |
| Best Particle Value >> | 8.8046 |
| Best Particle Value >> | 8.2979 |
| Best Particle Value >> | 8.2979 |
| Best Particle Value >> | 8.2241 |
| Best Particle Value >> | 8.2241 |
| Best Particle Value >> | 8.2166 |

Table (3): Obtained Result (of a welded beam) using Accelerated PSO for E01

| DESIGN PARAMETER of E01 | x1 | x2 | x3 | x4 |
| Best estimates: gbest = 1.0196 | 9.9195 | 6.1517 | 0.68812 | Iteration = 1 |
| Best estimates: gbest = 1.0196 | 9.9195 | 6.1517 | 0.68812 | Iteration = 2 |
| Best estimates: gbest = 0.4388 | 10 | 6.75007 | 0.445817 | Iteration = 3 |
| Best estimates: gbest = 0.4388 | 10 | 6.75007 | 0.445817 | Iteration = 4 |
| Best estimates: gbest = 0.23248 | 9.0743 | 8.6722 | 0.3535 | Iteration = 5 |
| Best estimates: gbest = 0.33435 | 6.4459 | 6.5906 | 0.41839 | Iteration = 6 |
| Best estimates: gbest = 0.33435 | 6.4459 | 6.5906 | 0.41839 | Iteration = 7 |
| Best estimates: gbest = 0.33435 | 6.4459 | 6.5906 | 0.41839 | Iteration = 8 |
| Best estimates: gbest = 0.33435 | 6.4459 | 6.5906 | 0.41839 | Iteration = 9 |
| Best estimates: gbest = 0.21293 | 7.5447 | 6.3711 | 0.4582 | Iteration = 10 |
| Best estimates: gbest = 0.21293 | 7.5447 | 6.3711 | 0.4582 | Iteration = 11 |
| Best estimates: gbest = 0.21293 | 7.5447 | 6.3711 | 0.4582 | Iteration = 12 |
| Best estimates: gbest = 0.21174 | 6.9496 | 7.1404 | 0.35104 | Iteration = 13 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 14 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 15 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 16 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 17 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 18 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 19 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 20 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 21 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 22 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 23 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 24 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 25 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 26 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 27 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 28 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 29 |
| Best estimates: gbest = 0.19646 | 6.3427 | 7.7966 | 0.28285 | Iteration = 30 |
6. ENGINEERING PROBLEMS

E01: Welded beam design optimization problem

Low cost beam (welded) is the required point and that to with some constraints [15] Figure E01 pictures out the welded beam.

For E01, for E02, APSO gave best values and COPSO the known equivalent, but Mezura gave value up to 4 decimal points, and no report of required precision attained. For E03, APSO reached the best value, COPSO fits less accurate data, and Mezura reached an unaccessible record. APSO and COPSO managed for E04, although Mezura reported a value that is worse than the best known. For E01 and E04 low value was obtained by COPSO; for E02 and E03, our APSO found the minimum values.

Table 1: Best results obtained by APSO, COPSO and Mezura

<table>
<thead>
<tr>
<th>PROBLEMS</th>
<th>OPTIMAL</th>
<th>APSO</th>
<th>COPSO</th>
<th>MEZURA</th>
</tr>
</thead>
<tbody>
<tr>
<td>E01</td>
<td>1.724852</td>
<td>1.724852</td>
<td>1.724852</td>
<td>1.724852</td>
</tr>
<tr>
<td>E02</td>
<td>6.059714335</td>
<td>6.059714335</td>
<td>6.059714335</td>
<td>6.059714335</td>
</tr>
<tr>
<td>E03</td>
<td>NA</td>
<td>2.996348165</td>
<td>2.996372448</td>
<td>2.996348094*</td>
</tr>
<tr>
<td>E04</td>
<td>0.012665</td>
<td>0.012664</td>
<td>0.012665</td>
<td>0.012689</td>
</tr>
</tbody>
</table>

*Infeasible Solution. NA Not Available

Table 2: APSO, COPSO and Mezura for mean and St. Dev. Results obtained

<table>
<thead>
<tr>
<th>PROBLEMS</th>
<th>MEAN</th>
<th>ST. DEV.</th>
</tr>
</thead>
<tbody>
<tr>
<td>E01</td>
<td>APSO</td>
<td>COPSO</td>
</tr>
<tr>
<td>E02</td>
<td>2.0372</td>
<td>1.7248</td>
</tr>
<tr>
<td>E03</td>
<td>6.0980498</td>
<td>6.0710133</td>
</tr>
<tr>
<td>E04</td>
<td>2.9964084</td>
<td>2.9964085</td>
</tr>
</tbody>
</table>

*Infeasible Solution
variables $x_1$ and $x_2$ are usually integral multiples of 0.0625 inch, but for this application are assumed continuously.

A cylindrical vessel is capped at both ends by hemispherical heads see Figure (E02). Using rolled steel plate, the shell is made in two halves joined by two longitudinal welds to form a cylinder. The total cost minimization is the aim, including the cost of the materials forming the welding [19]. The design variables are: thickness $x_1$, thickness of the head $x_2$, the inner radius $x_3$, and the length of the cylindrical section of the vessel $x_4$. The variables $x_1$ and $x_2$ are discrete values, which are integer multiples of 0.0625 inch.

Solution: $\mathbf{x}^* = (0.8125, 0.4375, 42.098446, 176.636596)$ where $f(\mathbf{x}^*) = 6,059.714335$

**E03: S RDO problem**

The image of speed reducer [12] is in Fig. (E03), have the face width $x_1$, module of teeth $x_2$, number of teeth on pinion $x_3$, length of the first shaft between bearings $x_4$, length of the second shaft between bearings $x_5$, diameter of the first shaft $x_6$, and diameter of the first shaft $x_7$. The minimization of weight of the instrument is subject to constraints on bending stress of the gear teeth, surface stress, transverse deflections of the shafts and stresses in the shaft.

**E04: T/CSDO Problem**

The algorithm [2, 24] reduces the weight of a tension/compression spring (Figure E04), conditioned to low deflection, shear stress, surge frequency, and with a role of outer dia & variable.

7. **CONCLUSIONS AND FUTURE WORK**

An Accelerated Particle Swarm Optimization (APSO) algorithm is way to deal with conditional search. The mentioned technique employs an easy way, a ring to-
ology, with suitable formula for orientation of particle. APSO showed better technique for engineering design optimization. Comparison of the acquired results suits the desired solution. With Low objective function our method is more fruitful than any other path followed. The technique is valuable and suited design problems. Exploring other APSO models and in performing a more detailed statistical analysis of the performance of our proposed approach is the future aspect of this finding.

References


[22] Q. He and L. Wang, An effective co-evolutionary particle swarm optimization for con-


[34] T. Ray and P. Saini, Engineering design optimization using a swarm with an intelligent information sharing among individuals, Engineering Optimization, 33 (2001), 735–748.


