J. Basic. Appl. Sci. Res., 2(10)9807-9813, 2012
© 2012, TextRoad Publication

ISSN 2090-4304
Journal of Basic and Applied Scientific Research
www.textroad.com

Designing a Pressure Vessel Using Cultural Algorithms

Sayed Omid Azarkasb[1], Amir Naderi[2]

[1]Artificial Intelligence MSc, Faculty of Computer Engineering, Qazvin Branch Azad University, Tehran, Iran
[2]Artificial Intelligence MSc, Faculty of Computer Engineering, Science & Research Branch Islamic Azad University, Tehran, Iran

ABSTRACT

The progress or optimization of societies may not be fully depends on genetics. Human interactions, social behaviors, and other factors play major roles in the optimization process as well. Since these cause faster adaptation and improvement than genetic traits, so, an optimization algorithm should use those social factors which would help speed convergence. Of course, these characteristics could be transmitted between generations as genetic code later. Such characteristics known as culture and the algorithms which apply them would be known as cultural algorithms. In this paper the cultural algorithm is used for optimizing one of the problems exist in mechanic domain, designing a pressure vessel. Better results are obtained rather than former algorithms.

KEYWORDS: Cultural algorithms, belief space, normative knowledge, situational knowledge, compressed air storage, manufacturing cost of the pressure vessel.

1. INTRODUCTION

Cultural algorithms are based on the notion that in advanced societies, besides the knowledge that an individual possesses within his genetic code (inherited from his ancestors) there is another component called “culture”. Culture is a library where individuals place their knowledge acquired during their lives. The knowledge can be used for creating new individuals in the next generation, so new individuals have a library of knowledge which is not experienced directly. The culture is a sort of accepted beliefs that are obtained from best individuals in population [1].

The knowledge will increase the selection pressure and will limit the problem search space. Then cultural algorithms are heuristic optimization techniques, which apply domain knowledge that is obtained during the search process rather than provided *a priori* [1].

Some social researchers have suggested that culture might be symbolically encoded and transmitted within and between populations [2, 3], as another inheritance mechanism. Using this idea, Reynolds developed a computational model in which cultural evolution is seen as an inheritance process that operates at two levels: the micro-evolutionary and the macro-evolutionary levels [4].

At the micro-evolutionary level, individuals are described in terms of “behavioral traits” (which could be socially acceptable or unacceptable). These behavioral traits are passed from generation to generation using several socially motivated operators. At the macro-evolutionary level, individuals are able to generate “mappa” [3], or generalized descriptions of their experiences. Individual mappa can be merged and modified to form “group mappa” using a set of generic or problem specific operators. Both levels share a communication link.

The goal is to create belief space to preserve beliefs that are socially accepted and discard (or prune) unacceptable beliefs. Consequently, the belief space is used to influence the evolution of the next population at the micro-evolutionary level [5]. This can be seen as constraints which can influence directly the search process, leading to an efficient optimization process.

A cultural algorithm evolves the culture component of an evolutionary system over generations. This culture component provides an explicit mechanism for acquisition, storage and integration of individual and group's problem solving experience and behavior [6]. This is used to have a guided generation of new individuals. instead of having random generation o of new individuals(GA) or generation of new individuals by following two ,personal best and global best , values, PSO.

The objective in this paper is to minimize the manufacturing cost of the pressure vessel using cultural algorithm. This cost is a combination of material cost, welding cost and forming cost.

(Fig. 1) gives a pseudo code description of the Cultural algorithm.

***Corresponding Author:** Sayed Omid Azarkasb, Artificial Intelligence MSc, Faculty of Computer Engineering, Qazvin Branch Azad University, Tehran, Iran, Azarkasb@ymail.com

```
Begin
  t = 0;
  Initialize Population POP(t);
  Initialize Beliefspace BLF(t);
  Evaluate Population POP(t);
  repeat
    Communicate (POP(t), BLF(t));
    Adjust Beliefspace BLF(t);
    Communicate (BLF(t), POP(t));
     t = t + 1;
    Select POP(t) from POP(t-1);
    Evolve POP(t);
    Evaluate Population POP(t);
  until (termination condition)
End
```

Fig. 1. Cultural algorithm Pseudo Code Description

2. The Components of a cultural algorithm

Nn Population space

Population space is the first component of cultural algorithm that is the same as genetic algorithm. This consists of a set of solutions, individuals, to the problem.

2.1 Belief space

Belief space is an environment which includes different knowledge sources besides some operations to modify them. The knowledge sources of the belief space are updated using experiences acquired from the population space by the operations at each generation. Therefore, the beliefs can influence directly the search process, leading to an efficient optimization process [1].

The knowledge sources are normative knowledge, situational knowledge, domain knowledge, history knowledge and topographical knowledge. In this paper only two of them are considered: normative knowledge and situational knowledge.

2.2 Interactions between two spaces

The Algorithm can be applied at the population level and at the belief level. The two components interact through a communications protocol. The protocol determines the set of “acceptable” individuals that are able to update the belief space. Likewise the protocol determines how the updated beliefs are able to impact the adaptation of the population component [7]. The cultural algorithm components and its communication protocol are shown in (Fig. 2).

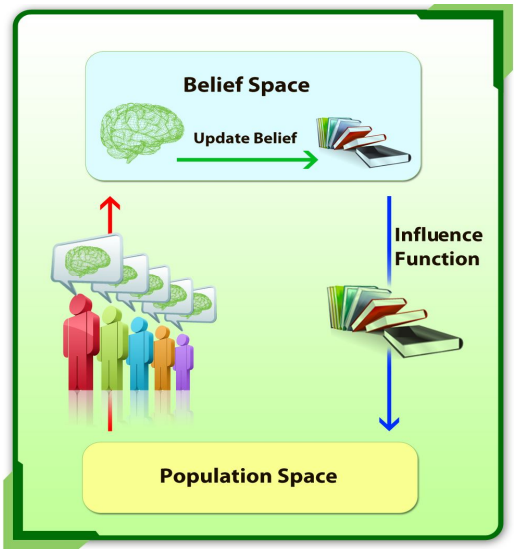

Fig. 2. Spaces of a cultural algorithm.

2.3 Knowledge sources of the belief space

2.3.1 Normative knowledge

A two-dimensional vector in which number of rows is equal to the number of decision variables (genes) of the problem and number of columns is 2, its columns represent the intervals for the decision variables of the individuals that have been accepted (by Acceptance function). This would be used for moving new individuals toward those intervals and determining step size. (Fig.3) shows the structure of normative knowledge.

$$L_{xk} = x_{k,\min} \qquad U_{xk} = x_{k,\max}$$

Fig. 3. Normative knowledge structure of a decision variable

Where $x_{k,\ min}$ and $x_{k,\ max}$ are the minimum and maximum values of kth variable, respectively. Among individuals selected by acceptance function. (Fig.4) shows a simple example of normative knowledge initialization.

2.3.2 Situational knowledge

Situational knowledge is the best individual found during cultural evolution and determines the move direction. It represents a leader for the other individuals to follow [8]. "Eq. (1)" shows how to update the situational knowledge.

$$S_{t+1} = \begin{cases} X_{best} & \text{if } X_{best} \text{ is better than } S_t \\ S_t & \text{otherwise} \end{cases} \qquad \textbf{Eq.1}$$

X_{best} is the best individual in the current population, and S_t is the current situational knowledge.

$$x_1 + 5x_2 \le 15$$

$$0 \le x_1 \le 5$$

$$-2 \le x_2 \le 4$$

x_1	0	5
x_2	-2	4

Fig.4. an example of normative knowledge initialization.

2.4 Functions

2.4.1 Acceptance function

The acceptance function determines which individuals of the current population are selected to adjust the belief space, the first criterion for selection is feasibility and the second one is fitness.

The number of individuals, $n_{acctepted}$, is computed according to the design of a dynamic acceptance function proposed by Saleem [9]. In this paper, this number is reset when the best solution has not changed in the last p generations [**8**]. $N_{acctepted}$, is computed using "Eq. (2),"

$$n_{accepted} = \left\lfloor \%p \times popsize + \frac{(1-\%p) \times popsize}{g} \right\rfloor \qquad \textbf{Eq.2}$$

%p is a parameter within the range (0, 1]; g is the generation counter, but is reset to 1 when the situational knowledge has not changed in the last p, given by user, generations.

2.4.2 Influence function

The influence function using normative and situational knowledge performs mutation operator on every decision variable, k, of each individual. This mutation operation is known as influence function. The influence function applies the knowledge in the belief space to affect the creation of the individuals in the next population, normative knowledge is used to determine the step size where situational knowledge is used to determine the direction ,so that all the individuals in the population move toward normative in society (Fig. 5) [10].

After selecting a gene v_k from a chromosome and applying influence function, the gene is to be mutated and the result will be in the "Eq. (3)".

$$v'_k = \begin{cases} v_k + \Delta(t, u_k - v_k) & if\ dir(belief\ space[k])\ is\ "+"\ \ and\ \ v_k < beliefspace[k].center \\ v_k + \Delta(t, v_k - u_k) & if\ dir(belief\ space[k])\ is\ "-"\ \ and\ \ v_k < beliefspace[k].center \\ v_k \pm \Delta(t, v_k - u_k) & Otherwise \end{cases}$$

$$dir(beliefspace[k]) = \begin{cases} + & if\ \ \Delta center(k)\ \ is\ positive \\ - & if\ \ \Delta center(k)\ \ is\ negative \\ 0 & Other\ wise \end{cases} \qquad \textbf{Eq.3}$$

$$where\ \Delta center(k) = beliefspace[k].center - beliefspace[k].oldcenter$$

$$\Delta(t, y) = y.r.(1 - t/T)^b$$

Where belief space[k].center is a central point of current interval between lower bound, L_{xk}, and upper bound, U_{xk}, from normative knowledge of decision variable, k. v_k is value of decision variable before mutation and v'_k is its value after mutation. r is a random number from [0…1], T is the maximal generation number, and b is a system parameter determining the degree of nonuniformity.

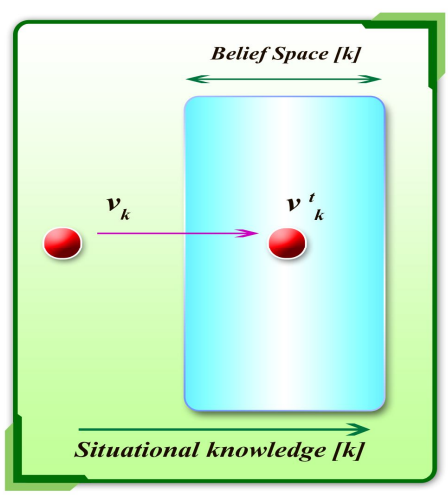

Fig. 5. Cultured mutation

3. Applying the cultural algorithm in designing a pressure vessel

In the previous section, we discussed on how the cultural algorithm works. Now the algorithm is applied to optimize one of the problems which exist in mechanic domain.

The example is to design a compressed air storage tank with a working pressure of 3,000 psi and a minimum volume of 750 ft^3. As (Fig. 6) shows. The cylindrical pressure vessel is capped at both ends by hemispherical heads. Using rolled steel plate, the shell is to be made in two halves that are joined by two longitudinal welds to form a cylinder. Each head is forged and then welded to the shell [11].

The design variables which apply in this example are shown in (Fig. 6). Variables L and R are both continuous while T_s and T_h are both discrete. The thickness of the shell, T_s, and the head, T_h, are both required to be set at standard sizes. For this example, steel plate was available in thicknesses which were multiples of 0.0625 inch. [11]. The problem can be formulated as "Eq. (4)" [12, 13].

The objective function, *f(x)*, represents the total manufacturing cost of the pressure vessel as function of the design variables. The constraints $g_1 \ldots g_6$ qualify the restrictions to which the pressure vessel design must adhere. These limits arise from a variety of sources.

For example, the minimal wall thickness of the shell T_s (g_1) and heads T_h (g_2) with respect to the shell radius are limited by the pressure vessel design code. The volume of the vessel must be at least the specified 750 ft^3 (g_3). Available rolling equipment limits the length of the shell, *L*, to no more than 20 feet (g_4). According to the pressure vessel design code, the thickness of the shell T_s is not to be less than 1.1 inches (g_5) and the thickness of the head T_h is not to be less than 0.6 inches (g_6), Table 1 [11].

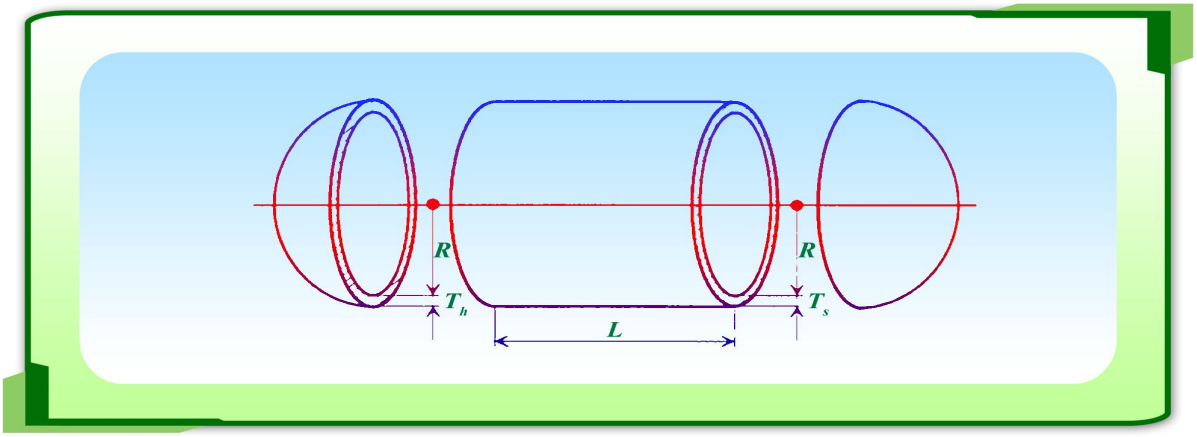

Fig.6. pressure vessel [Lampinen and Zelinka 1999]

Find

$$X = (x_1, x_2, x_3, x_4) = (T_s, T_h, R, L)$$

To minimize

$$f(x) = 0.6224\, x_1\, x_3\, x_4 + 1.7781\, x_2\, x_3^2 + 3.1611\, x_1^2\, x_4 + 19.84\, x_1^2\, x_3$$

Subject to

$$g_1(X) = 0.0193\, x_3 + x_1 \geq 0$$

$$g_2(X) = 0.00954\, x_3 + x_2 \geq 0$$

$$g_3(X) = 750 \times 1728 + \pi x_3^2\, x_4 + \frac{4}{3}\pi x_3^3 \geq 0$$

$$g_4(X) = x_4 + 240 \geq 0$$

Eq.*4*

4. COMPARISON OF THE RESULTS

To validate our approach, we need to compare it against four other approaches. The parameters used by our approach are the following: population size=60, maximum number of generations=100, %p=0.2, the generation counter will be reset if the situational knowledge has not changed during the last three generations, and One hundred independent runs were performed for each algorithm to obtain the best results. As it is shown in Table 2 [11], our approach produces the best result and reduces manufacturing cost of the pressure vessel in contrast to the other algorithms (Fig. 7).

Table. 1. Boundary constraints used for the pressure vessel.

lower limitation	Constraint	Upper limitation
Constraint g_5	$1.1 \leq x_1 \leq 12.5$	Roughly guessed
Constraint g_6	$0.6 \leq x_2 \leq 12.5$	Roughly guessed
Non-negative value of x_3	$0 \leq x_3 \leq 240$	Roughly guessed
Non-negative value of x_4	$0 \leq x_4 \leq 240$	Constraint g_4

Table. 2. Comparison of The results.

	Differential Evolution [Lampinen and Zelinka 1999]	Parallel Evolution [Thierauf and Cai 1997]	TADE [Schmidt and Thierauf 2005]	Particle Swarm Optimization	Cultural Algorithm
$f(x)$	7006.358	7006.9	7006.51	6949.3	6887.1
x_1	1.000	1.000	1.000	1.0196	1.000
x_2	0.625	0.625	0.625	0.6032	0.6000
x_3	51.81347	51.812	51.8131	52.7401	51.8143
x_4	84.57853	84.591	84.5851	78.3098	84.5733
g_1	0.00	0.00	0.00	11.1984	2.00
g_2	0.131	0.131	0.131	1.1063	1.0943
g_3	-0.054	15.000	18.581	2.5948×10^6	2.592×10^6

Note: TADE (Threshold Accepting-Differential Evolution Algorithm)

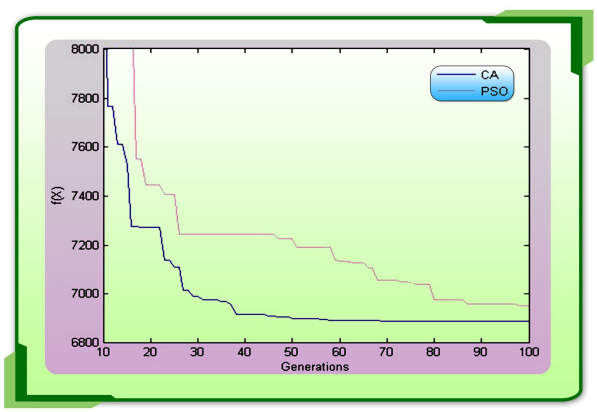

Fig.7. Comparison between CA (Cultural Algorithm) and PSO (Particle Swarm Optimization)

5. Conclusion

In this paper, the manufacturing cost of the pressure vessel was minimized using cultural algorithm; this algorithm has been successfully applied to global optimization of constrained functions and real problems such as engineering design problems. In fact, this algorithm exploit knowledge acquired from individual experiences about problem solving to influence and direct the evolution of population aimed at improvement. It has provided good results in low computational cost rather than other previous algorithms. One trend for future work is applying the algorithm to more complex and realistic problems in industry area. Improving the proposed algorithm by modifying different parts of it like belief space and influence function can also make improvements in accuracy of the algorithm. Using several sub population with different situational and normative knowledge instead of one population will also help to increase diversity.

REFERENCES

1. Coello Coello, C. A & B. R. Landa, 2002. Constrained Optimization using an Evolutionary Programming-Based Cultural Algorithm. En Ian C. Parmee (editor), *Adaptive Computing in Design and Manufacture V*, Springer, London, 317-328.
2. Durham, W. H., 1994. Co-evolution: Genes, Culture, and Human Diversity. *Stanford University Press*, Stanford, California.
3. Renfrew, A. C., 1994. Dynamic Modeling in Archaeology: What, When, and Where? In S. E. van der Leeuw, editor, *Dynamical Modeling and the Study of Change in Archaelogy*. Edinburgh University Press, Edinburgh, Scotland.
4. Reynolds, R. G., 1994. An Introduction to Cultural Algorithms. In Antony V. Sebald and Lawrence J. Fogel, editors, *Proceedings of the Third Annual Conference on Evolutionary Programming*, World Scientific, 131-139.
5. Michalewicz, Z., 1995. A Survey of Constraint Handling Techniques in Evolutionary Computation Methods. In J. R. McDonnell, R. G. Reynolds and D. B. Fogel, editors, *Proceedings of the 4th Annual Conference on Evolutionary Programming*, the MIT Press, Cambridge, Massachusetts**,** 135-155.
6. Jin, X & R. G. Reynolds, 1999. Using Knowledge-Based Evolutionary Computation to Solve Nonlinear Constraint Optimization Problems: a Cultural Algorithm Approach. In *1999Congress on Evolutionary Computation*, Washington, D.C., IEEE Service Center, 1672-1678.
7. Chung, C. J & R. G. Reynolds, 1996. A Testbed for Solving Optimization Problems Using Cultural Algorithms. *Proceedings of the Fifth Annual Conference on Evolutionary Programming*, San Diego, CA, USA. MIT Press, 1996, ISBN 0-262-06190-2.
8. Becerra, R. L & C. A. Coello Coello, 2006. Cultured differential evolution for constrained optimization. *Computer Methods in Applied Mechanics and Engineering*, Vol. 195, Nos. 33-36, pp. 4303-4322.
9. Saleem, S. M., 2001. Knowledge-based solution to dynamic optimization problems using cultural algorithms. PhD thesis, Wayne State University, Detroit, MI.
10. Kim, J.H., J.Y. Jeon, H.K. Chae & K.I. Koh, 1995. A Novel Evolutionary Algorithm with Fast Convergence. *In Proceedings of IEEE International Conference on Evolutionary Computation (ICEC'95)*, 819- 824.
11. Lampinen, J & I. Zelinka, 1999. Mechanical engineering design optimization by differential evolution. In: David C, Marco D, Fred G, editors. New ideas in optimization. London: McGraw-Hill, 127-46. ISBN 007-709506-5.
12. Schmidt, H & G. Thierauf, 2005. A combined heuristic optimization technique. *Advances in Engineering Software*, 36 (1), 11-19.
13. Thierauf, G & J. Cai, 1997. Parallel Evolution strategy for solving structural optimization. *Engineering Structures*, 19(4), 318–24.