Appl. Comput. Math. **5** (2006), no.2, pp.218-226

PSO FOR FUZZY GOAL PROGRAMMING

ALIREZA REZAEE †, §

Abstract. In this paper, an interactive particle swarm optimization (IPSO) for general fuzzy non-linear goal programming (FNLGP) is proposed. It utilizes the attractive features of interactive approaches, concept of fuzzy logic, and computational intelligence paradigms to solve general FNLGP model. A criterion for setting PSO Parameters is proposed, it employs a function imposed on the magnitude of particle's velocity to control the PSO' s convergence. Experiments are carried out to verify the proposed algorithm.
Key words: Fuzzy,nonlinear,PSO.

1. Introduction

In most of real life situations the objectives are generally conflicted, non-commensurable and fuzzy in nature and many considerations of the vague nature of uncertainty should be taken in the formulation of the problem. Naturally the objective functions and constraints are fuzzy in their nature and involve many fuzzy parameters [19] whose possible values are fuzzy numerical data ,which can be represented by means of fuzzy subsets of the real line known as fuzzy numbers [4, 20]. The fuzzy models are more realistic and more sophisticated than the conventional (deterministic) models and help the analyst in modeling the real life problems. Sakawa [20] proposed an approach to transform multi-objective nonlinear programming problem with fuzzy parameters to a non-fuzzy problem. The application of fuzzy set theory to goal programming has been made by Hannan [8,9] , Tiwari et al. [22], and Huey-Kuo Chen [10].

Interactive methods are appropriate techniques for solving multi criteria decision making problems. Interactive methods can be classified as goal programming based or multi-criteria based. Many of these approaches are based upon a procedure that employs an interactive election of information from the decision maker. The main steps of any interactive approach are i) Finding a solution (feasible), ii) Interacting with the decision maker to obtain his reaction to the solution , iii) Repeating step i and step ii , until the satisfaction is reached or until any other termination criterion is exhausted. Many researchers have investigated the interactive goal programming [5, 13, 12]. Masud and Hwang [13] introduced an interactive approach which combine the attractive feature of both goal programming and interactive approaches for multi-criteria decision making problems.

On the other hand, intelligent optimization techniques such as evolutionary computation, swarm intelligence, immune systems have growing interest as a problem solver in the field of optimization and computer science. The main features of these techniques are parallelism, intelligence, stochastic and ability to handle complex problems. Several researchers presented different techniques in the field of multi-objective decision making problems [2, 7, 21, 3, 11, 23, 18] and in the field of optimization problems [14,15].

In this paper, section 2, introduce problem statement and solution concept of the fuzzy nonlinear goal programming model FNLGP. Section 3, describes the mechanism of Particle Swarm Optimization PSO and a proposed adaptable construction factor. In section 4, the proposed

†Amirkabir University,IRAN, arrezaee@aut.ac.ir
§*Manuscript received June 14, 2005.*

IPSO algorithm for solving general FNLGP is summarized. Section 5 describes experimental results of the proposed algorithm. Section 6 describes the main features of the proposed algorithm and points for further research.

2. Problem Statements and Solution Concept.

The fuzzy non-linear goal programming model FNLGP can be modeled as follows:

$$Goal\,1 : g_1(x) = \sum_{j=1}^{n} c_{1j}x_j \overset{\sim}{\geq} \tilde{B}_1$$

$$Goal\,2 : g_2(x) = \sum_{j=1}^{n} c_{2j}x_j \overset{\sim}{\geq} \tilde{B}_2$$

..

$$Goal\,m : g_m(x) = \sum_{j=1}^{n} c_{mj}x_j \overset{\sim}{\geq} \tilde{B}_m \quad subject\ \ to :$$

$$h_r(x) = \sum_{j=1}^{n} ar_jx_j \overset{\sim}{\leq} \underset{r}{\tilde{b}} \;\; r = 1, 2, ..., s$$

$$L_j \leq x_j \leq U_j\,, \quad j = 1, 2, ..., n$$

subject to:

$$h_r\,(x) = \sum_{j=1}^{n} \tilde{a}r_j\,x_j \leq \tilde{b}_r \quad r = 1, 2, ..., s$$
$$L_j \;\leq\; x_j \;\leq\; U_j \;,\quad j = 1, 2, ..., n$$

where the i^{th}goal consists of mi fuzzy non-linear sub-goals $g_{ij}\,(x) = \tilde{B}_{ij}$ with differential weight Wij ,

$$i = 1, 2, ..., m, \quad j = 1, 2, ..., mi,$$
$$h_r(x) \leq \tilde{b}, \qquad r = 1, 2, ..., s$$

are the system constraints . The fuzzy terms in the above formulated model are constituted in goal constraints (in aspiration levels and in coefficients of decision variables) and constituted in system constraints (fuzzy resources and fuzzy coefficients of decision variables).

The membership functions of fuzzy terms in the above formulated model are defined as follows:

$$\mu_i(B_i) = \begin{cases} 1 & if \quad B_i \geq B_i^0 \\ 1 - \frac{[B_i^0 - B_i]}{\Delta B_i} & if \quad B_i^0 - \Delta B_i \leq B_i \leq B_i^0 \\ 0 & otherwise \end{cases}$$

$$\mu_{ij}(c_{ij}) = \begin{cases} 1 & if \quad C_{ij} \geq C_{ij}^0 \\ 1 - \frac{[c_{ji}^0 - c_{ij}]}{\Delta c_{ji}} & if \quad c_{ji}^0 - \Delta c_{ij} \leq c_{ij} \leq c_{ij}^0 \\ 0 & otherwise \qquad (P.2) \end{cases}$$

$$\mu_{rj}(a_{ij}) = \begin{cases} 1 & if \quad a_{rj} \le a_{rj}^0 \\ 1 - \frac{[a_{rj} - a_{rj}^0]}{\Delta a_{ij}} & if \quad a_{rj}^0 \le a_{ij} \le a_{rj}^0 + \Delta a_{rj} \\ 0 & otherwise \end{cases}$$

$$\mu_r(b_r) = \begin{cases} 1 & if \quad b_r \le b_r^0 \\ 1 - \frac{[b_r - b_r^0]}{\Delta b_r} & if \quad b_r^0 \le b_i \le b_r^0 + \Delta b_r \\ 0 & otherwise \end{cases}$$

where c^0, Δc_{ij}, B^0, Δ_{Bi}, a_{rj}, Δ_{arj}, b^0 and Δ_{br} are given $i = 1 , 2 , .. . , m , r = 1 ,2 , .. . , S , j = 1 , 2 , n,... ,$

Definition 1.

The α -level set of the fuzzy $\underset{ij}{\tilde{c}}, \underset{i}{\tilde{B}}, \underset{rj}{\tilde{a}}, and\ \tilde{b}_r$ are defined as the ordinary set $L\alpha$ for which the degree of their membership functions exceed the level α ,

$$L_\alpha = \{c_{ij}, B_i, a_{rj}, b_r |\ \mu_{ij}\ (c_{ij}) \ge \alpha\, ,\ \mu_i\,(B_i) \ge \alpha$$

$$\mu_{rj}\,(a_{rj}) \ge \alpha \ \ \mu_r\,(b_r) \ge \alpha$$

$i = 1 , 2 , .. . , m , r = 1 ,2 , .. . n,...1 , 2 ,, S , j = \} .$

Let α be the membership value of the fuzzy Bi , cij , br and arj , i.e.

$\mu_i\,(Bi) \ge \alpha,\ \ \mu_{ij}\,(c_{ij})\,\alpha \ge,$
$\mu_r\,(b_r)\,\alpha \ge,\ \ \mu_{rj}\,(a_{rj})\,\alpha \ge .$

then we can obtain:

$$B_i \ge B^0 - (1-\alpha)\,\Delta_{Bi},\ \ b_r \le b^0 + (1-\alpha)\,\Delta_{rb},$$

$$c_{ij} \ge c_{ij}^0 - (1-\alpha)\,\Delta_{cij},\ \ a_{rj} \le a_{rj}^0 + (1-\alpha)\,\Delta_{rja},$$

$$i = m, ..., 2, 1\ ,\ , ..., 2, 1 = rSj = .n, ..., 2, 1$$

Using definition 1 and at a certain degree of α [0 , 1] , the FNLGP model (P.1) can be converted into a non-fuzzy non-linear goal programming model (α -NLGP) as follows :

α -NLGP:

$$\sum_{j=1}^{n} \left[\overset{0}{c}_{ij} - (1-\alpha)\,\Delta c_{ij} \right] x_j \ge \overset{0}{B}_i - (1-\alpha)\,\Delta_{B_i}\ \ i = 1, ...m$$

: subject to (P.2)

$$\sum_{j=1}^{n} \begin{matrix} [a_{rj}^0 + (1-\alpha)\Delta a_{rj}]x_j \le b_i^0 + (1-\alpha)\Delta b_r & r = 1, 2, ..., s \\ L_j \le x_j \le U_j & j = 1, 2, ..., n \end{matrix}$$

Apply the iterative approach and let Pi be the attainment model corresponding to the goal i or to the goals have the same priority level i, Pi can be defined as follows :

$P_i : Minimize \quad S_i = \sum_{j=1}^{m_i} w_{ij} n_{ij}$

Subject to:

$$\sum_{j=1}^{n} [\overset{0}{c}_{Lsj} - (1-\alpha)\Delta_c L_{Sj}]x_j + n \overset{0}{L}_s \geq B \overset{0}{L}_s - (1-\alpha)\Delta_B Ls$$

$$L = 1, 2, ...i$$
$$s = 1, 2, ..., m$$

$$\sum_{j=1}^{mL} w_{Lj} n_{Lj} = S_L^* \quad L = 1, 2, ..., i$$

$$\sum_{j=1}^{n} \left[a_{rj}^0 + (1-\alpha)\,\Delta_{\omega j}\right] x_j \leq b_r^0 + (1-\alpha)\Delta_{br}$$

(P.3)

$$r = 1, 2, ..., s$$

$$n_{Ls} \geq 0, \;\; L = 1, 2, ..., i \qquad s = 1, 2, ...m_L$$

$$L_j \leq x_j \leq U_j \qquad j = 1, 2, ...m_L.$$

Where *nLj, L = 1 , 2 , i. , j = 1 , 2 , , mL* are negative deviation variables .

Apply, the PSO algorithm presented in section 3 to solve model Pi *(P.3)*. Interact with the decision maker, If the decision maker is not satisfied with the obtained solution, he chooses another leveling degree of [0 ,1]. Otherwise, the algorithm is terminated.

3. TheParticle swarm Optimazation algorithm

PSO was introduced to study social and cognitive behavior [1, 17, 6] , but it has been applied as a problem solving technique in engineering design and computer science. PSO is a population based stochastic search techniques ,they work on a set of individuals of potential solutions to the problem under consideration called swarm, is used to explore the search space , each individual of the swarm has an adaptable velocity (position change), according to which it moves in the search space. Moreover, each individual has a memory, remembering the best position of the search space it has ever visited.

There are two main types of the PSO algorithm, one with a global neighborhood, and other with a local neighborhood. In the global neighborhood variant, each particle moves towards its best previous position and towards the best particle in the whole swarm. On the other hand, according to the local neighborhood variant, each particle moves towards its best previous position and towards the best particle in its restricted topological neighborhood, which is usually implemented as ring structure, with the last member is being as a neighbor of the first one.

In the global variant PSO, suppose that the search space is a n-dimensional, and the size of the swarm is N, the swarm is maintained during the search process according to the following two equations:

$$\nu_i^{t+1} = \chi(\omega\nu_i^t + c_1 r_{i1}^t (P_i^t - x_i^t) + c_2 r_{i2}^t (P_g^t - x_i^t)) \quad (Eq.1),$$

$$x_i^{t+1} = x_i^t + \nu_i^{t+1} \quad (Eq.2).$$

Where:

Xi = (xi1, xi2, . . . , xin) , *Vi = (ν i1, ν i2, . . . , ν in)* and *Pi = (pi1, pi2, . . . , pin)* are n-dimensional vectors represent the i-th particle, its velocity and its best previous position respectively, $i = 1, 2, \ldots, N$ and $t = 1, 2, \ldots .T$, is the iteration number. ω is called inertia weight; c*1*, c*2* are called cognitive and social learning rates respectively; r*i1*, *ri2* are random numbers, uniformly distributed within the range [0, 1]; g is the index of the best particle in the swarm; and χ is a constriction factor, is used to control the magnitude of velocity.

The PSO parameters is considered important factors of the PSO working procedure, The inertia weight ω plays an important role the PSO 's convergence behavior [6]. A large inertia weight facilitates global exploration (searching new areas), while a small one tends to facilitate local exploration. However, experimental results indicated that as a good choice forω is initially 1.2 and a gradual decreased towards 0. The parameters *r1* and *r2* are used to maintain the diversity of the population, and they are uniformly distributed in the range [0, 1]. The parameters c*1* and c*2* are used to bias the search of a particle toward its best previous position and the best the best particle in the swarm. Experimental results [17] recently indicate that it is better to choose a larger cognitive parameter, *c1*, than a social parameter, *c2*, but with $c1 + c2 \leq 4$.

In this research, the constriction factor χ is setting as a function of the iteration number and the magnitude of the particle's velocity, instead of using a constant value like *Vmax* , this function is defined as::

$$\chi = f(m, t) = r.m\,(1 - t/T)\,,$$

where t is the iteration number , T maximum number of iterations; r is a random numbers, uniformly distributed within the range [0, 1]; m is the magnitude of the particle's velocity. This function returns a value in range [0 , m] , which lead to the probability of χ value being close to 0 as t increase. This feature leads to control the magnitude of the particle's velocity in adaptable way, to search uniformly the search space when t is small and very locally as t increased during the search process and assure the PSO 's convergence behavior. The use of this function alleviate the effect of inertia weight on the PSO 's convergence behavior.

The Algorithm of PSO:

The working procedure of proposed PSO is summarized in the following Algorithm:

Step 1: Set t = 1; Generate a feasible population of particles of size N.

Step 2: For each particle:

Calculate fitness value.

If the particle's fitness value is better than its previous best fitness *(pBest)* ,set current value as the new *pBest.*

Step3: Choose the particle with the best fitness value of all the particles as the *gBest.*

Step4: For each particle: Calculate particle velocity, using *Eq.1,* Update particle position, using *Eq.2,*

Step5: Check particles feasibility.

Step6: If the stopping criterion is satisfied, stop: otherwise go to step 2.

4. Proposed algorithm IPSO for FNLGP:

The solution procedure of the proposed IPSO algorithm for FNLGP can be summarized in the following steps:

Step 1: Interact with the decision maker (DM) to gather the information $B^0i,\ b^0r,\ cij,\ arj,\ \Delta Bi,\ \Delta br,\ \Delta Cij,\ \Delta Cij,\ \Delta arj,\ i.=1,2,...,m,$ $j=1,2,...,n,\ r=1,2,...,S$ of fuzzy terms constituted in the model.

***Step 2*:** Construct the membership functions of the fuzzy terms.

Step 3: Formulate the FNLGP model (P.1).

***Step 4*:** The decision maker (DM) chose α [0 , 1] .

Step 5: Formulate the $\tilde{\alpha}$ NLGP model (P.2).

Step 6: Apply the Iterative approach, formulate the non-linear programming model Pi (P.3).

Step7: Using the PSO algorithm to solve the attainment non-linear programming model Pi (P.3). The obtained solution is an -optimal solution for the original FNLGP model.

Step8: Interact with the decision maker (DM), if he is not satisfied with obtained solution, then go to step 4.

Step 9: Stop.

5. Experiments results.

The following FNLGP model is used to highlight the working process of the proposed IPSO during decision making process.

$$\begin{aligned}
&Goal1 : g_1(x) = c_{11}x_1^2 + c_{12}x_2 + c_{13}x_3^2 \geq B_1\\
&Goal2 : g_{21}(x) = \underset{211}{\tilde{c}}\, x_1^2 - \underset{212}{\tilde{c}}\, x_2 + \underset{213}{\tilde{c}}\, x_3 \geq \tilde{B}_2\\
&\quad g_{22}(x) = \tilde{c}_{221}x_1^2 + \underset{222}{\tilde{c}}\, x_2 + \tilde{c}_{223}x_3^2 \geq \underset{2}{\tilde{B}}\\
&Subjectto:\\
&\quad h_1(x) = \underset{11}{\tilde{a}}\, x_1^2 + \underset{12}{\tilde{a}}\, x_2^2 x_3 + \underset{13}{\tilde{a}}\, x_3 \leq \tilde{b}
\end{aligned}$$

Where the second goal has two sub-goals with differential weights equal 3 and 1, respectively.

$c^0 11 = 4,\ \Delta C11 = 2,\ c^0 12 = 5,\ \Delta C12 = 2,\ c^0 13 = 5, \Delta C13 = 4, B^0 1 = 7, \Delta B1 = 4,$
$C^0 211 = 6,\ C211 = 4,\ {}^0 212 = 5,\ \ \Delta C212 = 2,\ c^0 213 = 9,\ \Delta C213 = 2,\ B^0 21 = 3$
0
$\Delta B21 = 2,\ c221 = 5/2,\ \ \Delta c221 = 1,\ c^0 222 = 3,\ \Delta c222 = 20$

At a certain degree of certainty α=0.55, The FNLGP model is converted non-fuzzy model as follows:

$$0.55 - NLGP:$$

$$G1 : 3.1x_1^2 + 4.1x_2 + 3.2^2 \geq 5.2$$

,

$$G2 : 4.2x_1^2 - 4.1x_2 + 8.1x_3 \geq 3.1$$

$$G3 : 3.7x^2 + 2.1x_2 + 2.15x_3^2 \geq 4.15$$

Subject to

$$0.95x_1^3 + 1.95x^2 x_3 + 1.9x_3 \leq 0.9$$

x. X

For goals at the first priority level, formulate the following model:

$$\begin{aligned}
&P1 : Min\ \ S1 = n1 \\
&subject\,to : \\
&3.1\ x_1^2 + 4.1x_2 + 3.2x_3^2 + n_1 \geq 5.2\,, \\
&4.2\,x_1^2 - 4.1x_2 + 8.1x_3 + n_{21} \geq 3.1, \\
&3.7x_1^2 + 2.1x_2 + 2.15x_3^2 + n_{22} \geq 4.15, \\
&0.95x_1^3 + 1.95x_2^2 x_3 + 1.9x_3 \leq 0.9, \\
&\quad 0 \leq x_j \leq 3 \qquad j = 1, ...3, \\
&\quad n1, n21, n22 \geq 0
\end{aligned}$$

Apply the PSO algorithms to solve P1,
the obtained α-optimal solution of model P1 is:

X* = (x1, x2, x3, n1, n21, n22) = (0.92, 0.73, 0.00, 0.00, 2.80, 0.29), and S*1 = 0.
For goals at the second priority level formulate the following model P2:

$$\begin{aligned}
&P2 : Min\ S2 = 3n_{21} + n_{22} \\
&subject\,to : 1 \qquad 1 \\
&3.1\ x_1^2 + 4.1x_2 + 3.2x_3^2 + n_1 \geq 5.2\,, \\
&4.2\,x_1^2 - 4.1x_2 + 8.1x_3 + n_{21} \geq 3.1, \\
&3.7x_1^2 + 2.1x_2 + 2.15x_3^2 + n_{22} \geq 4.15, \\
&0.95x_1^3 + 1.95x_2^2 x_3 + 1.9x_3 \leq 0.9, \\
&\quad 0 \leq x_j \leq 3 \qquad j = 1, ...3, \\
&\quad n1, n21, n22 \geq 0
\end{aligned}$$

Apply the PSO algorithms to solve P2, The obtained solution of P2 is: X* = (x1, x2, x3, n1, n21, n22) =(0.94, 0.60, 0.04 , 0.00 , 1.57, 0.01), and S*2 = 4.7155. This is the -optimal solution for the original FNLGP model.

The same model is solved using Genetic algorithms , with the same population size (N = 40) and maximum number of iterations (T =10000), the obtained solution is: X* = (x1, x2, x3, n1, n21, n22) = (0.93, 0.62 , 0.05, 0.00,

1.59, 0.00) and S*2 = 4.7764; and the computational time of optimization process using Genetic algorithms (GA) is greater than PSO

6. Conclusions and future work.

In this paper, an interactive PSO for general fuzzy nonlinear goal programming is proposed. It combines attractive features of interactive approach, concept of fuzzy set, features PSO to solve general FNLGP model. It allows to the decision maker to satisfy his goals and his preferred decision through interaction with model during the decision making process. It employs a function to control the magnitude of the particle's velocity in an adaptable way, instead of using a constant value like $Vmax$, which lead to probe the search space uniformly, when t is small and very locally as t increased during the search process, and alleviate the effect of inertia weight on the PSO 's convergence behavior. The proposed algorithm handles constraints in a direct way Instead of using penalty functions for handling constraints. During the evolution process, if a particle does not satisfy all constraints of the model, then the particle feasibility is maintained.

There are many issues still needed for further researches such as, a parametric analysis through PSO to FNLGP, the integration between PSO and the other computational intelligence paradigms for solving FNLGP, studying the stochastic convergence of the PSO, and implementation of the proposed algorithm to real life applications.

References

[1] Bonabeeau, E., Dorigo, m., & Theraulaz, G. . " Swarm Intelligence from natural to Artificial Systems", Oxford University Press, 1999.

[2] Carlos, A., Coello, "A Comprehensive Survey of Evolutionary-Based Multi-objective Optimization Techniques", Knowledge and Information Systems. An International Journal, 1(3):1999, pp. 269-308.

[3] Deb, K., "Multi-objective optimization using evolutionary algorithms", John Wiley & sons, LTD, 2001.

[4] Dubois, D., and Prade , H., " Fuzzy sets and systems theory and application " . New York , Acadimic press, 1980 .

[5] Dyer , J.S., " Interactive goal programming " . Management science 19 ,1972 , pp. 62-70 .

[6] Ebrhart, R., & Xiaohui, Hu, "Swarm Intelligence", Morgan Kaufmann, 2002.

[7] Eckart, Z., Deb., K., Lothar, T., " Comparison of Multiobjective Evolutionary Algorithms: Empirical Results" , Evolutionary Computation, 8(2): 2000, pp. 173-195.

[8] Hannan, E.L., " Linear programming with multiple goals". Fuzzy Sets and system 6, 1981 , pp.235-248.

[9] Hannan, E.L., "On fuzzy goal programming", Decision Sci. 12 , 1981,pp. 522-531.

[10] Huey-kuo, chen, "A note on Fuzzy programming algorithm by Tiwari et al. " . Fuzzy Sets and Systems 62 , 1994 , pp. 287-290 .

[11] Kennedy, J., Ebrhart, R., & Shi, Y., " Multi-objective optimization using Dynamic Neighborhood Particle Swarm Optimization ", Proceedings of the IEEE Congress on First International Conference on. evolutionary computation , Hawii USA, 2001.

[12] Lee, S.M., and Shin, J.P., " Interactive goal programming on microcomputer to establish priorities for small business ". Journal of the operation research 37 ,1986, pp.571-577 .

[13] Masud , A.S., and Hwang, " Interactive sequential goal programming " . Journal of the operation research 32 , 1981 , pp.391-400

[14] Michalewicz,Z., "Evolutionary Computation Techniques for Nonlinear Programming Problems", International Transactions in Operational Research,1994, Vol.1, No.2, pp.223-240.

[15] Michalewicz, Z., Dasgupta, D., Le Riche, R.G., and Schoenauer, M., "Evolutionary Algorithms for Constrained Engineering Problems". Computers & Industrial Engineering Journal, 1996, Vol.30, No.2, pp.851-870.

[16] Narasimhan, R., " Goal programming in a fuzzy environment ". Decision Sci. 17 ,1980 , pp.325-336.

[17] Parsopoulos, K.E., and Vrahatis, M.N., "Recent approaches to global optimization problems through Particle Swarm Optimization", Neural Computing , Kluwer Academic Publishers, 2002.

[18] Rahmat, Y., Gies, D., and Robinson, J., " Particle Swarm Optimization (PSO): A Novel Paradigm for Antenna Designs", 2003.
[19] Rasmy, M.H., and Soliman, O.S., "An Interactive Approach For Solving Fuzzy Linear Goal Programming Problem " , The Annual Conf., ISSR ,Cairo University, 1998, Vol.(33), PP 33-46
[20] Sakawa, M., " Interactive decision making for multi-objective nolinear programming problems with fuzzy parameter ". Fuzzy Sets and Systems, 1989 , Vol. (29), 315-326 .
[21] Srinivas, N., and Deb, K., "Multi-objective Optimization Using Nondominated Sorting in Genetic Algorithms". Evolutionary Computation, 2(3): 1999 , pp. 221-248.
[22] Tiwari, R.N., and Dharmar, S., " Priority structure in Fuzzy goal programming " . Fuzzy Sets and Systems 19 ,1986 ,pp. 251-259.
[23] Xiaohui, H., and Ebrhart, R., "Multiobjective Optimization Using Dynamic Neighborhood Particle Swarm Optimization", 2002.

Alireza Rezaee, for a photograph and biography, see Apl. Compt. Math. (2006),vol.5, no.1, p.118.