ISSN 1 746-7233, England, UK
World Journal of Modelling and Simulation
Vol. 14 (2018) No. 1, pp. 50-57

LCPI method to find optimal solutions of nonlinear programming problems

M.H. Noori Skandari , M. Ghaznavi*

Faculty of Mathematical Sciences
Shahrood University of Technology, Shahrood, Iran

(*Received April 27 2016, Accepted June 27 2017*)

Abstract. In this paper a class of nonlinear programming (NLP) problems is considered. The linear combination property of intervals (LCPIs) is used to convert an NLP problem into an equivalent NLP problem. The LCPI states that any continuous function on a compact and connected domain can be shown as a linear combination of its minimum and maximum. By the LCPI idea we obtain a new NLP problem, the variables of which are fewer than the variables of the main NLP problem. Also, the obtained equivalent problem is linear with respect to the new added variable. Therefore, solving the new NLP problem is easier than solving the main one. Finally, some numerical examples are provided to endorse the efficiency of idea in solving NLP problems.

Keywords: nonlinear programming, LCPI method, approximate optimal solution.

1 Introduction

NLP problems are utilized in many applications including mathematics, engineering, management sciences and finance. Moreover, such applications contain digital signal processing, structural optimization, neural networks, mechanical engineering and medical optimization [17, 18, 21]. Optimal solutions have important economical and practical impacts on these applications.

Except for trivial cases, it is hard to solve nonlinear programming problems exactly. Hence, numerical and approximate methods have been proposed to search for optimal solutions of these problems. The procedure of solving an NLP problem involves searching for optimal points in the corresponding search space.

Since nonlinear objectives have many local minimums and nonlinear constraints are hard to deal with, finding global optimal solution of an NLP is a challenging task.

A large number of optimization methods have been proposed to solve NLP problems [1, 4–7, 18]. Nonlinear optimization methods are classified into local and global methods.

Local optimization methods include gradient descent, Newton, quasi Newton and conjugate gradient methods, etc. (see [3–5, 13, 15, 16, 22] and references therein). These methods converge to a local minimum from some initial point. Such a local minimum is globally optimal only when the objective is quasi-convex and the feasible region is convex, which rarely happens in practice. For nonlinear optimization problems, a local minimum may be very far from the global minimum. To overcome local minimality, global optimization methods have been developed [10, 12, 14, 24, 25].

Global optimization methods look for globally optimal solutions . These methods perform global and local searches in regions. The mechanism of a global search method is to escape from local minimum, while it is not so for local search methods.

Nevertheless, a lower bound for global optimum of the objective function can not be given after any finite number of iteration, unless the objective satisfies certain conditions, such as Lipschitz conditions and boundedness of the search space.

* Corresponding author. *E-mail address*: Ghaznavi@shahroodut.ac.ir (M.Ghaznavi)

However, in the real world, many applicable nonlinear optimization models are large-scale and have many variables and constraints. Therefore, it is hard and time consuming to find their global and even local optimal solutions. Hence, solving an equivalent problem with fewer constraints is of interest to researchers. Also, converting a nonconvex, nonsmooth nonlinear problem into a convex and smooth one is very advantageous. Now, in this paper we use a very famous theorem stating that *any continuous function on a compact and connected domain can be shown as a linear combination of its minimum and maximum* [9, 19, 20, 23] and propose a method, named LCPI, to convert an NLP problem into an equivalent one which has fewer variables than the main problem and is very easier to solve than the main NLP problem. Also, we can calculate an approximate optimal solution with less errors and less CPU time.

The paper is organized as follows: In Section 2 the LCPI idea for converting a nonlinear optimization problem into an equivalent NLP is described. Some theorems are proved and an algorithm is proposed to solve the NLP problems. Section 3 is devoted to numerical examples. In this section, nonlinear problems are converted into equivalent problems with fewer variables. Finally, conclusions are given in Section 4.

2 LCPI for converting an NLP problem

In this section, we use LCPI for converting an NLP problem into an equivalent NLP problem. The LCPI states that any continuous function on a compact and connected domain can be shown as a linear combination of its minimum and maximum (see [9, 19, 20, 23]).

We consider the following class of NLP problems:

$$\begin{aligned} &\min\ (\text{or } \max)\ F(x,y) \\ \text{subject to}\quad & g_i(x)=0, i=1,\cdots,m, \\ & h_j(x)\le 0, j=1,\cdots,p, \\ & y\in\mathcal{Y}, \end{aligned} \tag{1}$$

where $\mathcal{X}=\{x\in\mathbb{R}^n|g_i(x)=0, i=1,\cdots,m, h_j(x)\le 0, j=1,\cdots,p\},\ \mathcal{Y}\subseteq\mathbb{R}^k$ is a compact and connected set and $F:\mathcal{X}\times\mathcal{Y}\to\mathbb{R}$ is a continuous function.

Definition 1. *A feasible solution (x^*,y^*) is an optimal solution for NLP problem (1) if $F(x^*,y^*)\le F(x,y)$ for any feasible point $(x,y)\in\mathcal{X}\times\mathcal{Y}$.*

At first, we state the following theorems and utilize them to convert the NLP Problem (1) into an equivalent problem with a reduced number of variables. Moreover, by this idea we reduce the complexity of problem.

Theorem 1. *Let $F:\mathcal{X}\times\mathcal{Y}\to\mathbb{R}$ be a continuous function where $\mathcal{Y}$ is a compact and connected subset of $\mathbb{R}^k$. Then, for any arbitrary (but fixed) $x\in\mathcal{X}$, the set $\{F(x,y):y\in\mathcal{Y}\}$ is a closed interval in $\mathbb{R}$.*

Proof. Let $x\in\mathcal{X}$ be given and $Y\subset\mathbb{R}^k$ be a compact and connected set. Since continuous functions preserve compactness and connectedness, the set $\{F(x,y):y\in\mathcal{Y}\}$ is a compact and connected set in $\mathbb{R}$. Therefore, $\{F(x,y):y\in\mathcal{Y}\}$ has lower and upper bounds, thus it is a closed interval in $\mathbb{R}$.

Now, for any arbitrary $x\in\mathcal{X}$, assume that the lower and upper bounds of the closed interval $\{F(x,y):y\in\mathcal{Y}\}$ are $F_1(x)$ and $F_2(x)$, respectively. Thus, we have

$$F_1(x)\le F(x,y)\le F_2(x),\ \ \forall y\in\mathcal{Y}. \tag{2}$$

In other words:

$$F_1(x)=\min\{F(x,y)|y\in\mathcal{Y}\},\ x\in\mathcal{X} \tag{3}$$

and

$$F_2(x)=\max\{F(x,y)|y\in\mathcal{Y}\},\ x\in\mathcal{X}. \tag{4}$$

Theorem 2. *Let functions $F_1(x)$ and $F_2(x)$ be defined by relations (3) and (4), respectively. Then, they are uniformly continuous on $\mathcal{X}$.*

Proof. The proof is similar to that of Theorem 2.2 in [9].

Lemma 1. *Let $a, b \in \mathbb{R}$. For any $x \in [a, b]$ there exists $\lambda \in [0, 1]$ such that $x = \lambda a + (1 - \lambda)b$.*

Theorem 3. *Assume that $x \in X$ is given. Also, let $F : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ be a continuous function where $\mathcal{Y}$ is a compact and connected subset of $\mathbb{R}^k$. Then, for any $y \in \mathcal{Y}$, there exists a $\lambda \in [0, 1]$ such that*

$$F(x, y) = \lambda F_2(x) + (1 - \lambda)F_1(x),\ \lambda \in [0, 1]. \tag{5}$$

Proof. The proof is a result of Lemma 1 and relations (3) and (4).

Now, by Theorem 3, we can convert problem (1) into the following problem:

$$\begin{aligned} &\min\ (or\ \max)\ G(x, \lambda) = \lambda F_2(x) + (1 - \lambda)F_1(x) \\ &s.t.\ \ g_i(x) = 0,\ \ i = 1, \cdots, m, \\ &h_j(x) \le 0,\ j = 1, \cdots, p, \\ &0 \le \lambda \le 1. \end{aligned} \tag{6}$$

Applying the following theorem, we show that the optimal solutions of the main problem can be found by using those of the new NLP problem.

Theorem 4. *Let $(\hat{x}, \hat{\lambda})$ be an optimal solution of (6). Then, there exists $\hat{y} = (\hat{y_1}, \cdots, \hat{y_k})$ such that $(\hat{x}, \hat{y})$ is an optimal solution of NLP (1).*

Proof. Let $(x, \lambda) \in \mathcal{X} \times [0, 1]$. Then:

$$\hat{\lambda}F_2(\hat{x}) + (1 - \hat{\lambda})F_1(\hat{x}) \le \lambda F_2(x) + (1 - \lambda)F_1(x), \tag{7}$$

and $g_i(\hat{x}) = 0, \forall i$ and $h_j(\hat{x}) \le 0, \forall j$. We define $\psi(y) = F(\hat{x}, y),\ \forall y \in Y$. Now, by Theorem 3, for any $y \in Y$, there exists $\lambda \in [0, 1]$ such that

$$\psi(y) = \lambda F_2(\hat{x}) + (1 - \lambda)F_1(\hat{x}).$$

Assume that $\alpha = \hat{\lambda}F_2(\hat{x}) + (1 - \hat{\lambda})F_1(\hat{x})$. Since $\psi(y)$ is continuous, by the intermediate value theorem, there exists $\hat{y} \in \mathcal{Y}$ such that $\psi(\hat{y}) = \alpha$, that is $F(\hat{x}, \hat{y}) = \hat{\lambda}F_2(\hat{x}) + (1 - \hat{\lambda})F_1(\hat{x})$. On the other hand by (7), we have

$$F(\hat{x}, \hat{y}) = \hat{\lambda}F_2(\hat{x}) + (1 - \hat{\lambda})F_1(\hat{x}) \le \lambda F_2(x) + (1 - \lambda)F_1(x) = F(x, y),$$

i.e. $(\hat{x}, \hat{y})$ is an optimal solution for NLP (1).

Theorem 5. *Let $(\hat{x}, \hat{y})$ be an optimal solution of NLP (1). Then, there exists $\hat{\lambda} \in [0, 1]$ such that $(\hat{x}, \hat{\lambda})$ is an optimal solution of NLP (6).*

Proof. The proof is similar to that of Theorem 4.

It is important to note that, the new equivalent NLP problem (6) has the following prominences, in comparison with the main one:

- The number of variables has reduced from $n + k$ to $n + 1$.
- The new problem is linear with respect to the new variable λ, in which the main problem is nonlinear with respect to $y_1, \cdots, y_k$.
- By this transformation, the time of solving an NLP problem can be reduced.

Remark 1. Note that the main problem might be nonconvex, whereas the new problem might be convex. Therefore, we can use the KKT optimality conditions to find the optimal solutions of the new problem and hence, the optimal solutions of the main problem. Let the functions $F_1(.), F_2(.), g_i(.), i = 1, \dots, m$ and $h_j(.), j = 1, \dots, p$ be differentiable. The KKT optimality conditions for problem (6) can be calculated as follows:

$$\begin{cases} \lambda \nabla F_2(x) + (1-\lambda)\nabla F_1(x) + (F_2(x) - F_1(x)) + \\ \sum_{i=1}^{m} \mu_i \nabla g_i(x) + \sum_{j=1}^{p} \xi_j \nabla h_j(x) + \delta - \varepsilon = 0, \\ \sum_{i=1}^{m} \mu_i g_i(x) = 0, \\ \sum_{i=1}^{m} \xi_j h_j(x) = 0, \\ \delta(\lambda - 1) = 0, \\ \varepsilon\lambda = 0, \\ \mu_i \ free, \forall i = 1, ..., m, \\ \xi_j \geq 0, \forall j = 1, ..., p, \\ \delta \geq 0, \varepsilon \geq 0, \end{cases} \tag{8}$$

where $\nabla F(x) = (\frac{\partial F}{\partial x_1}, \cdots, \frac{\partial F}{\partial x_n})$.

By the given theorems and discussions, the main algorithm for solving the NLP problem (1) can be proposed by the following steps:

Algorithm

Step 1: Define sets X and Y;
Step 2: Using relations (3) and (4), calculate functions $F_1(.)$ and $F_2(.)$ and obtain the optimal solution y^* for the NLP problem (1);
Step 3: Obtain the optimal solution (x^*, λ^*) for the NLP problem (6).

3 Numerical examples

In this section, we solve three NLP problems by using the proposed approach. Here, the obtained nonlinear programming (NLP) problems are solved by FMINCON toolbox in MATLAB software based on the SQP method, interior point method and active-set method. Also, a PC with RAM $4GB$ is utilized.

Example 1. Consider the following NLP problem:

$$\begin{array}{ll} \text{Minimize} & F(x,y) = ln\left((x_1 - 0.5)^2 y_1 + x_2^4 + (y_2 - 1)^2 + e^{y_1} + 2\right) \\ \text{subject to} & x_1^2 + x_2^2 = 0.25, \\ & 2x_1 - \cos(\pi x_2) = 0, \\ & 0 \leq y_1 \leq 1, \ \ 0 \leq y_2 \leq 1, \end{array} \tag{9}$$

where $(x_1^*, x_2^*, y_1^*, y_2^*) = (0.5, 0, 0, 1)$ is the exact optimal solution and exact optimal objective value is $F(x^*, y^*) = ln3$. We define $X = \{(x_1, x_2) : \ x_1^2 + x_2^2 - 0.25 = 0, \ 2x_1 - \cos(\pi x_2) = 0\}$, and $Y = \{(y_1, y_2) : \ 0 \leq y_1 \leq 1, \ 0 \leq y_2 \leq 1\}$. By relations (3) and (4), for any arbitrary $(x_1, x_2) \in X$, we have

$$\begin{aligned} F_1(x_1, x_2) &= \underset{y=(y_1,y_2)\in Y}{\text{Minimum}} \ \ ln\left((x_1 - 0.5)^2 y_1 + x_2^4 + (y_2 - 1)^2 + e^{y_1} + 2)\right) \\ &= ln(x_2^4 + 3), \end{aligned} \tag{10}$$

and

$$F_2(x_1,x_2) = \underset{y=(y_1,y_2)\in Y}{\text{Maximum}} \;\; ln\left((x_1-0.5)^2 y_1 + x_2^4 + (y_2-1)^2 + e^{y_1} + 2)\right) = ln\left((x_1-0.5)^2 + x_2^4 + e + 3\right). \tag{11}$$

By relation (10), it is obvious that $(y_1^*, y_2^*) = (0,1)$.
Now, the corresponding minimization problem (6) is as follows:

$$\begin{array}{ll} \text{Minimize} & G(x,\lambda) = \lambda(ln((x_1-0.5)^2 + x_2^4 + e + 3)) + (1-\lambda)ln(x_2^4+3)) \\ \text{subject to} & x_1^2 + x_2^2 - 0.25 = 0, \\ & 2x_1 - \cos(4\pi x_2) = 0, \\ & 0 \le \lambda \le 1. \end{array} \tag{12}$$

We solve the NLP problem(12) and obtain the approximate optimal solution $(x_1^*, x_2^*, \lambda^*)$.

The optimal solutions $(x_1^*, x_2^*, y_1^*, y_2^*)$ are illustrated in Table 1. Moreover, the optimal objective value $F(x^*, y^*)$ is given in Table 2.

Table 1: Optimal values of variables in Example 3.1

Approximate optimal solution x_1^*	Approximate optimal solution x_2^*	Optimal solution y_1^*	Optimal solution y_2^*
0.50000000	0.00000005	0.00000000	1.00000000

Now, we solve the main NLP problem (9) by SQP method, interior point method and active-set method, directly. The results are given in Tab. 2. As one can see, the absolute error and CPU time of solving the NLP problem (9) by the proposed algorithm, are less than the absolute error and CPU time of the utilizing SQP, interior point and active-set methods, directly.

Table 2: Comparing the results of solving problem (9) directly and by new algorithm

	Solving problem (9) by SQP method	Solving problem (9) by interior point method	Solving problem (9) by activeset method	Solving problem (9) by new algorithm
CPU time (s)	1.863	1.109610	0.664829	0.514
Objective value	1.11923157	1.11923157	1.11923157	1.09861228
Absolute error	0.02061928	0.0206132	0.0206192	0.00000000

Example 2. Consider the following NLP problem:

$$\begin{array}{ll} \text{Minimize} & F(x,y) = x_1^2 \tan(\frac{\pi}{6} y_1 + \frac{\pi}{4}) + x_2^2 \cos(\frac{\pi}{2} y_2) \\ \text{subject to} & x_2 - \cos(4\pi x_1) = 0, \quad -1 \le x_1 \le 1, \;\; 0 \le x_2 \le 1, \\ & y_1^2 + y_2^2 \le 1, \quad 0 \le y_1 \le 1, \;\; 0 \le y_2 \le 1. \end{array} \tag{13}$$

where $(x_1^*, x_2^*, y_1^*, y_2^*) = (0,1,0,1)$ is the exact optimal solution and $F(x^*, y^*) = 0$ is the exact optimal objective value.

Assume $X = \{(x_1,x_2): \; x_2 - \cos(4\pi x_1) = 0, \;\; -1 \le x_1 \le 1, \;\; 0 \le x_2 \le 1\}$ and $Y = \left\{(y_1,y_2): \; y_1^2 + y_2^2 \le 1, \quad 0 \le y_1 \le 1, \;\; 0 \le y_2 \le 1\right\}$. By relations (3) and (4), for any arbitrary $(x_1,x_2) \in X$, we have

$$F_1(x_1,x_2) = \underset{y=(y_1,y_2)\in Y}{\text{Minimum}} \;\; \left(x_1^2 \tan(\frac{\pi}{6} y_1 + \frac{\pi}{4}) + x_2^2 \cos(\frac{\pi}{2} y_2)\right) = x_1^2, \tag{14}$$

and

$$F_2(x_1,x_2) = \underset{y=(y_1,y_2)\in Y}{\text{Maximum}} \;\; \left(x_1^2 \tan(\frac{\pi}{6} y_1 + \frac{\pi}{4}) + x_2^2 \cos(\frac{\pi}{2} y_2)\right) = x_1^2 \tan(\frac{5\pi}{12}) + x_2^2, \tag{15}$$

Here, the arguments of minimum in problem (14) are $y_1^* = 0$ and $y_2^* = 1$.
The equivalent NLP problem (6) is as follows:

$$\begin{array}{ll} \text{Minimize} & G(x,\lambda) = \lambda(x_1^2 \tan(\frac{5\pi}{12}) + x_2^2) + (1-\lambda)x_1^2 \\ \text{subject } to & x_2 - \cos(4\pi x_1) = 0, \quad -1 \leq x_1 \leq 1, \\ & 0 \leq x_2 \leq 1, \qquad 0 \leq \lambda \leq 1. \end{array} \tag{16}$$

Now, we solve the NLP problem (16) to find the approximate optimal solution $(x_1^*, x_2^*, \lambda^*)$.

The results of resolving the NLP problems (14) and (16) are given in Table 3. Also, the compared results of solving problem (13) directly (i.e. by SQP, interior point and active-set methods) and by our algorithm are given in Table 4. The results show high accuracy of the method with respect to the utilizing the other methods.

Table 3: Comparing the results of solving problem (9) directly and by new algorithm

Approximate optimal solution x_1^*	Approximate optimal solution x_2^*	Optimal solution y_1^*	Optimal solution y_2^*
0.00000000007	1.0000000000	0.00000000	1.00000000

Table 4: Comparing the results of solving problem (13) directly and by new algorithm

	Solving problem (13) by SQP method	Solving problem (13) by interior point method	Solving problem (9) by active-set method	Solving problem (13) by new Algorithm
CPU time (s)	0.626	3.914989	1.972576	0.507
Objective value	0.1×10^{-8}	5.11×10^{-9}	9.67×10^{-16}	0.5×10^{-18}
Absolute error	0.1×10^{-8}	5.11×10^{-9}	9.67×10^{-16}	0.5×10^{-18}

In the following example, we solve a nonsmooth NLP problem. As it can be seen, using the LCPI technique, we can convert some nonsmooth problems to equivalent smooth ones. It is obvious that solving the equivalent smooth problem is very faster than solving the main nonsmooth NLP problem.

Example 3. Consider the following nonsmooth NLP problem:

$$\begin{array}{ll} \text{Minimize} & F(x,y) = e^{-|y_1-1|x_1} + |y_2-1|\,(x_2-1)^2 \\ \text{subject to} & x_1 + x_2^2 \leq 1, \quad x_1 \geq 0, \ x_2 \geq 0, \\ & 0 \leq y_1 \leq 1, \quad 0 \leq y_2 \leq 2. \end{array} \tag{17}$$

where $(x_1^*, x_2^*, y_1^*, y_2^*) = (1, 0, 0, 1)$ is the exact optimal solution and $F(x^*, y^*) = e^{-1}$ is the exact optimal objective value.

Let

$$X = \left\{(x_1, x_2): \ x_1 + x_2^2 \leq 1, \quad x_1 \geq 0, \ x_2 \geq 0\right\}$$

and

$$Y = \{(y_1, y_2): \ 0 \leq y_1 \leq 1, \quad 0 \leq y_2 \leq 2\}.$$

Using relations (3) and (4), for any arbitrary $(x_1, x_2) \in X$, we have

$$F_1(x_1,x_2) = \underset{y=(y_1,y_2)\in Y}{\text{Minimum}} \left(e^{-|y_1-1|x_1} + |y_2-1|\,(x_2-1)^2\right) = e^{-x_1} \tag{18}$$

and

$$F_2(x_1,x_2) = \underset{y=(y_1,y_2)\in Y}{\text{Maximum}} \left(e^{-|y_1-1|x_1} + |y_2-1|\,(x_2-1)^2\right) = 1 + (x_2-1)^2. \tag{19}$$

Here, the arguments of minimum in problem (18) are $y_1^* = 0$ and $y_2^* = 0$.
By relations (18) and (19), the equivalent minimization problem (6) is obtained as follows:

$$\begin{array}{ll} \text{Minimize} & G(x,\lambda) = \lambda(1+(x_2-1)^2) + (1-\lambda)e^{-x_1} \\ \text{subject to} & x_1 + x_2^2 \leq 1, \quad x_1 \geq 0,\ x_2 \geq 0,\ 0 \leq \lambda \leq 1. \end{array} \tag{20}$$

Now, we solve the obtained smooth NLP problem (20) and find the approximate optimal solution $(x_1^*, x_2^*, \lambda^*)$. The obtained results are given in Table 5.

Table 5: Optimal values of variables in Example 3

Approximate optimal solution x_1^*	Approximate optimal solution x_2^*	Optimal solution y_1^*	Optimal solution y_2^*
1.000000000	0.000000002	0.00000000	1.00000000

Moreover, we calculate the objective function $F(x^*, y^*)$ that is compared with the exact value of problem (17) in Table 6. The obtained optimal objective values by the SQP, interior point and active-set methods are far from the exact objective value. The results show the high accuracy of the method with respect to the other methods.

Table 6: Comparing the results of solving problem (17) directly and by new algorithm

	Solving problem (17) by SQP method	Solving problem (17) by interior point method	Solving problem (17) by active-set method	Solving problem (17) by new Algorithm
CPU time (s)	0.377	4.245378	0.690984	0.236
Objective value	0.610863134	0.4139844	0.3690997	0.367879441
Absolute error	0.242983693	0.046104958	0.00122025	0.000000000

4 Conclusion

In this work, we showed that a class of NLP problems can be converted into the equivalent nonlinear problems such that the complexity of the new obtained problems are reduced and the number of decision variables is fewer than that of the main problems. Thus by the LCPI idea, we can reduce the time needed for solving the problems. Also, we showed that by using LCPI method, we can convert some nonsmooth (or onconvex) NLP problems into equivalent smooth (or convex) NLPs. Therefore, by the proposed approach we can solve a wide class of NLP problems, faster and more easily.
However, applying the idea of this paper for solving nonlinear mutliobjective optimization problems [2, 8, 11] can be a worthwhile direction for the future investigation.

References

[1] M. Avriel. *Nonlinear Programming: Analysis and Methods*. *Dover Publishing*,2003.

[2] B.A.Ghaznavi-ghosoni, E.K, M.Soleimani-damaneh. Scalarization for characterization of approximate strong/weak/proper efficiency in multi-objective optimization. *Optimization*, 2013, **62**(6): 703–720.

[3] P. Baldi. Gradient descent learning algorithm overview: a general dynamical systems perspective. *IEEE Transactions on Neural Networks*, 1995, **6**(1): 182–95.

[4] M.S. Bazaraa, H.D. Sherali, C. M. Shetty. *Nonlinear programming theory and algorithms*.3rd Edition, Wiley, 2006.

[5] D.G. Luenberger, Y. Ye. *Linear and Nonlinear Programming, International Series in Operations Research & Management Science 116 (Third ed.)*. New York: Springer, 2008.

[6] S. Effati, A. Ghomashi, A.R. Nazemi, Application of projection neural network in solving convex programming problems. *Applied Mathematics & Computation*, 2007, **188**(2): 1103–1114.
[7] S. Effati, M. Hoseini, A.R. Nazemi, A new numerical method by revised measure theory for solving the nonlinear initial value problems. *Applied Mathematics & Computation*, 2007, **186**(1): 780–788.
[8] M. Ehrgott. *Multicriteria Optimization*. Springer,Berlin,2005.
[9] H.R.Erfanian, M.H.Noori Skandari, A.V. Kamyad, Solving a class of separated continuous programming problems using linearization and discretization. *Int. J. Sensing*, Computing & Control 2011,**1**(2): 117–124.
[10] C.A. Floudas, P.M. Pardalos, *A Collection of Test Problems for Constrained Global Optimization Algorithms*. volume 455 of Lecture Notes in Computer Science Springer-Verlag, 1990.
[11] M. Ghaznavi. Optimality conditions via scalarization for approximate quasi efficiency in multiobjective optimization. *Filomat*, 2017, **31**(3): 671–680.
[12] R. Horst, H. Tuy. Global optimization:deterministic approach. *Springer-Verlag Berlin*, 1993.
[13] H.R.Erfanian, M.H.Noori Skandari, A.V.Kamyad. A novel approach for solving nonsmooth optimization problems with application to nonsmooth equations. *Journal of Mathematics*,Hindawi Publishing Corporatio 2013.
[14] V. Jeyakumar, G.Y. Li. Necessary global optimality conditions for nonlinear programming problems with polynomial constraints. *Mathematical Programming*, 2011, **126**: 393–399.
[15] J. A. Kinsella. Comparison and evaluation of variants of the conjugate gradient method for efficient learning in feed-forward neural networks with backward error propagation. *Neural Networks*, 1992, **3**:27–35.
[16] M.S. Moller. A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, 1993, **6**(4): 525–533.
[17] D.S. Johnson, M.R. Garey. Computers and intractability freeman san francisco. *CA*, 1979.
[18] A. R. Nazemi. Solving general convex nonlinear optimization problems by an efficient neurodynamic model. *Engineering Applications of Artificial Intelligence*, 2013, **26**: 685–696.
[19] M.H. Noori Skandari, E. Tohidi. Numerical solution of a class of nonlinear optimal control problems using linearization and discretization. *Applied Mathematics*, 2011, **2**(5): 646–652.
[20] M. Noori Skandari, H.R.Erfanian, A.V. Kamyad. A new approach for a class of optimal control problems of volterra integral equations. *Intelligent Control& Automation*, 2011, **2**:121–125.
[21] R. Steuer. Multiple criteria optimization. *Theory Computation and Application*,Krieger Publishing Company, 1989.
[22] E.G. Sturua, S K Zavriev. A trajectory algorithm based on the gradient method i. the search on the quasioptimal trajectories. *Journal of Global Optimization*, 1991, **1991**(4): 375–388.
[23] E. Tohidi, M.H. Noori Skandari. A new approach for a class of nonlinear optimal control problems using linear combination property of intervals. *Journal of Computations & Modelling*,2011, **1**(2): 145–156.
[24] A. Torn, A. Zilinskas. *Global Optimization*. Springer-Verlag, 1989.
[25] T. Weise. Global Optimization Algorithms: Theory and Application. 2008, Available: http: //www.it-weise.de/projects/bookNew.pdf.