CARPATHIAN J. MATH. Online version at http://carpathian.ubm.ro
33 (2017), No. 3, 373 - 380 Print Edition: ISSN 1584 - 2851 Online Edition: ISSN 1843 - 4401

A descent global optimization method based on smoothing techniques via Bezier curves

AHMET SAHINER, NURULLAH YILMAZ and GULDEN KAPUSUZ

ABSTRACT. In this study, we introduce a new global optimization method, named Esthetic Delving Method, based on the auxiliary function approach. First, we design the method theoretically and then present its implementable version. Finally, we apply the algorithm to the test problems in order to demonstrate its efficiency.

1. INTRODUCTION

We consider the following optimization problem

$$(P) \quad \min_{x \in \mathbb{R}^n} f(x)$$

where the objective function $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable.

The problem (P) is seen in many practical problems of engineering, finance, medical and other sciences. When the function f has many local minimizers, finding the global solution of problem (P) become very difficult. In recent years, many new methods have been developed for the solution of the problem (P).

There are different types of global optimization methods and algorithms based on deterministic, stochastic and heuristic ideas [4, 12, 19, 14]. Each of these methods outmaneuver each other from different aspects but the deterministic methods are the reliable ones [11, 20]. The auxiliary function method is the frequently used deterministic global optimization methods [16]. The prominent methods in auxiliary function approach are Filled Function Method [2, 3], Tunneling Algorithms [6, 17] and Cut-Peak Algorithms [13, 5]. In these methods, specific functions are used in order to reach the global minimum step by step [18, 10, 15, 16, 7]. The main difficulty for these methods is escaping from the current local minimizer in order to find the lower one.

Bezier curves were introduced in [1]. They constitute one of the important way for smooth modeling technique in Computer Aided Design. They are widely used in curve and surface designing problems [9].

In this study, we propose Esthetic Delving Method as a new global optimization technique by the help of Bezier curves. We observe that this new method decrease the computational costs satisfactorily after applying on test problems.

2. PRELIMINARIES

Throughout the paper, x_k^* denotes the $k-$th local minimizer and x^* denotes the global minimizer. We assume that the following conditions are satisfied for the objective function $f(x)$:

Received: 30.09.2016 . In revised form: 04.05.2017 . Accepted: 15.05.2017
2010 *Mathematics Subject Classification.* 90C26, 49J52, 65D07, 65D10.
Key words and phrases. *global optimization, Bezier curve, smoothing function, auxiliary function.*
Corresponding author: Ahmet Sahiner; ahmetsahiner@sdu.edu.tr

Assumption 1. The function f is coercive, i.e. $f(x) \to \infty$ as $\|x\| \to \infty$. Therefore, there exists a closed and bounded box $\Omega \subset \mathbb{R}^n$ containing all the minimizers of f.

Assumption 2. The function f has a finite number of local minimizers.

Also we recall the following definitions.

Definition 2.1. The basin $f(x)$ at an isolated minimum x_0 is a connected domain $B(x_0)$, consisting of all points such that the steepest descent trajectory of $f(x)$ converges to x_0, starting from the any of them.

Definition 2.2. The simple basin $f(x)$ at an isolated minimum x_0 is a connected domain $S(x_0) \subset B(x_0)$ such that for any $x \neq x_0$, $(x - x_0)^T \nabla f(x) > 0$.

The sets Ω_1^β and Ω_2 are defined as $\Omega_1^\beta = \{x : f(x) \geq f(x_k^*) - \beta, x \in \Omega\}$ and $\Omega_2 = \{x : f(x) < f(x_k^*),\ x \in \Omega\}$ for $\beta > 0$, respectively.

Definition 2.3. A Bezier curve is defined by a set of control point P_0 through P_n, where n is the order of Bezier cuve defined by

$$B(t) = \sum_{i=0}^{n} b_{i,n}(t) P_i, \quad 0 \leq t \leq 1,$$

where the polynomials

$$b_{i,n}(t) = \left(\begin{array}{c} n \\ i \end{array} \right) t^i (1-t)^{n-i}$$

are known as Bernstein basis polynomials of degree n.

3. New Global Optimization Technique

In this section, we introduce a new global optimization method based on finding the global minimum of the given smooth objective function iteratively by the virtue of the following steps:

1. Apply a local solver to compute a local minimizer x_k^* and the value $f(x_k^*)$ of the function f.
2. Consider the function $\phi(x, x_k^*) := \min\{f(x), f(x_k^*)\}$ and apply a local solver to compute its minimizer x_{k+1}^*. It is clear that $f(x_{k+1}^*) \leq f(x_k^*)$.
3. Set $x_k^* = x_{k+1}^*$ and go to Step 2. Then by Assumption 2 after the finite number of iterations the algorithm will find a global minimizer of the function f.

The key point for this idea is that the minimizers of the function $f(x)$ higher than x_k^* are not the minimizer of the function $\phi(x, x_k^*)$.

Lemma 3.1. *Let f be a smooth function and x_k^* be its local minimizer. The local minimizers of the function f lower than x_k^* are the local minimizers of $\phi(x, x_k^*)$ and the higher minimizers of f are not minimizers of the function $\phi(x, x_k^*)$ higher than $f(x_k^*)$.*

It can be observed that, the function $\phi(x, x_k^*)$ is non-smooth. Therefore, it is not possible to find the local minimizers by applying gradient-based methods. In order to address this problem, we propose to smooth functions. Since we have

$$\phi(x, x_k^*) = f(x_k^*) + \min\{f(x) - f(x_k^*), 0\},$$

the function $\phi(x, x_i^*)$ can be re-written as

$$\phi(x, x_k^*) = f(x_i^*) + (f(x) - f(x_k^*))\, \chi_{A_k}(x), \tag{3.1}$$

where $A_k = \{x \in \mathbb{R}^n : f(x) - f(x_k^*) < 0\}$ and $\chi_{A_k} : \mathbb{R}^n \to \mathbb{R}$ is characteristic function of a set A_k which is defined as

$$\chi_{A_k}(x) = \begin{cases} 1 & x \in A_k, \\ 0 & x \notin A_k. \end{cases}$$

Based on the equation (3.1) we design the following smoothing function for $\phi(x, x_k^*)$:

$$\tilde{\phi}(x, x_k^*, \beta) = f(x_k^*) + (f(x) - f(x_k^*))\, \tilde{\chi}_{A_k}(x, \beta),$$

where

$$\tilde{\chi}_{A_k}(x, \beta) = \begin{cases} 0 & t > \beta, \\ q_1(t, \beta) & 0 \le t \le \beta, \\ q_2(t, \beta) & -\beta \le t \le 0, \\ 1 & t < -\beta, \end{cases}$$

is the smoothed version of characteristic function of A_k that is obtained by using Bezier curves in the transition region, where the functions $q_1(t, \beta)$ and $q_2(t, \beta)$ are defined as

$$q_1(t, \beta) = -\frac{t^2 + 2\beta t - \beta^2}{(2\beta^2)},$$

and

$$q_2(t, \beta) = \frac{(t/\beta - 1)^2}{2},$$

for $t = f(x) - f(x_k^*)$.

Theorem 3.1. *Let $\beta > 0$ and x_k^* be a minimizer of the function f then, we have the following:*

i. *The function $\tilde{\phi}(x, x_k^*, \beta)$ is continuously differentiable,*
ii. *$0 \le \tilde{\phi}(x, x_k^*, \beta) - \phi(x, x_k^*) \le \beta/2$ for all $x \in \mathbb{R}^n$,*
iii. *$\tilde{\phi}(x, x_k^*, \beta)$ approaches $\phi(x, x_k^*)$, when $\beta \to 0$.*

Proof. **i.** Since the function $\tilde{\chi}_{A_k}(x, \beta)$ is smooth, we conclude that the function $\tilde{\phi}(x, x_k^*, \beta)$ is also smooth.

ii. In case of $f(x) - f(x_k^*) > \beta$, the function $\tilde{\chi}_{A_k}(x, \beta) = \chi_{A_k}(x) = 0$. Therefore $\tilde{\phi}(x, x_k^*, \beta) - \phi(x, x_k^*) = 0$. Similarly, if $f(x) - f(x_k^*) < -\beta$, then $\tilde{\chi}_{A_k}(x, \beta) = \chi_{A_k}(x) = 0$. Hence we get $\tilde{\phi}(x, x_k^*, \beta) - \phi(x, x_k^*) = 0$.

Let us assume that $0 \le f(x) - f(x_k^*) \le \beta$. Since $0 \le \tilde{\chi}_{A_k}(x, \beta) \le 1/2$ and $\chi_{A_k}(x) = 0$, we have

$$\tilde{\phi}(x, x_k^*, \beta) - \phi(x, x_k^*) = (f(x) - f(x_k^*))\, (\tilde{\chi}_{A_k}(x, \beta) - \chi_{A_k}(x)) \le \beta/2.$$

Similarly, for $0 \le f(x) - f(x_k^*) \le \beta$ we have $1/2 \le \tilde{\chi}_{A_k}(x, \beta) \le 1$ and $\chi_{A_k}(x) = 1$ then, we have

$$\tilde{\phi}(x, x_k^*, \beta) - \phi(x, x_k^*) = (f(x) - f(x_k^*))\, (\tilde{\chi}_{A_k}(x, \beta) - \chi_{A_k}(x)) \le \beta/2.$$

iii. It can be easily obtained by letting $\beta \to 0$. □

Since, the function $\tilde{\phi}(x, x_k^*, \beta)$ behaves like a constant function at those x points for which the $f(x) > f(x_k^*)$. To minimize $\tilde{\phi}(x, x_k^*, \beta)$ may cause the increase in computational costs. Moreover, depending on the smoothing parameter, it can not be possible to escape from the current local minimizer. So, we add a term to the function $\phi(x, x_k^*, \beta)$. The added term (function) φ is defined on $\mathbb{R}_+$ and it satisfies the following properties:

i. $\varphi(t) > 0$,
ii. $\varphi'(t) < 0$,
iii. $\lim_{t\to\infty} \varphi(t) = 0$.

Examples of φ include $\frac{\pi}{2} - \arctan(t), \exp(-t)$ and $\frac{1}{1+t}$. Finally, the complete Esthetic Delving function is as the following:

$$\tilde{\phi}(x, x_k^*, \beta, \alpha) = f(x_k^*) + (f(x) - f(x_k^*))\, \tilde{\chi}_{A_k}(x, \beta) + \alpha\varphi(\|x - x_k^*\|^2),$$

where α is a real parameter.

Theorem 3.2. *Let x_k^* be a local minimizer of the function f, then the function $\tilde{\phi}(x, x_k^*, \beta, \alpha)$ has no stationary point for all $x \in \Omega_1^\beta$.*

Proof. For any $x \in \Omega_1^\beta$, $x \neq x_k^*$ we have $\tilde{\phi}(x, x_k^*, \beta, \alpha) = f(x_k^*) + \alpha\varphi(\|x - x_k^*\|^2)$ and $\nabla\tilde{\phi}(x, x_k^*, \beta, \alpha) = \alpha\nabla\varphi(\|x - x_k^*\|^2) \neq 0$. $\square$

Theorem 3.3. *Let x_k^* be a local minimizer of the function f. If the function $f(x)$ has local minimizer x_{k+1}^* lower than x_k^*, then there exist local minimizer $\bar{x}$ of $\tilde{\phi}(x, x_k^*, \beta, \alpha)$ such that $\bar{x} \in S(x_{k+1}^*)$.*

Proof. We first prove that the $\tilde{\phi}(x, x_k^*, \beta, \alpha)$ has a minimizer according to Assumptions 1 and 2. Assume that the function $f(x)$ has a local minimizer lower than x_k^*. It can be concluded that the set $\Omega_3 = \{x : f(x) \leq f(x_k^*),\ x \in \Omega\}$ is not empty. Moreover, Ω_3 is closed since $f(x)$ continuous and bounded since it is contained by Ω. Therefore, the function $\tilde{\phi}(x, x_k^*, \beta, \alpha)$ has minimizer in Ω_3.

Let us further assume that the function $f(x)$ has a local minimizer x_{k+1}^* lower than x_k^*, and $\bar{x}$ be a local minimizer of the function $\tilde{\phi}(x, x_k^*, \beta, \alpha)$, then we have

$$\nabla\tilde{\phi}(\bar{x}, x_k^*, \beta, \alpha) = \nabla f(\bar{x}) + \alpha\nabla\varphi(\|\bar{x} - x_k^*\|^2) = 0.$$

Therefore, we have $\nabla f(\bar{x}) = -\alpha\nabla\varphi(\|\bar{x} - x_k^*\|^2)$. From the above definition of the function $\varphi(\|x - x_k^*\|^2)$, we have

$$(\bar{x} - x_k^*)\nabla f(\bar{x}) = (\bar{x} - x_k^*)\left(-\alpha\nabla\varphi(\|\bar{x} - x_k^*\|^2)\right) > 0.$$

Since we obtain $\bar{x}$ depending on $\varphi(\|x - x_k^*\|^2)$, for sufficiently small α values $\bar{x}$ is close enough to x_{k+1}^*, $(\bar{x} - x_k^*)$ is close enough to $(x_{k+1}^* - x_k^*)$ and $\|\bar{x} - x_k^*\| \geq \|x_{k+1}^* - x_k^*\|$. This gives us the vectors $(\bar{x} - x_k^*)$ and $(\bar{x} - x_{k+1}^*)$ are almost in the same directions. Therefore, we obtain

$$(\bar{x} - x_k^*)\nabla f(\bar{x}) = (\bar{x} - x_{k+1}^*)\left(-\alpha\nabla\varphi(\|\bar{x} - x_k^*\|^2)\right) > 0.$$

It is clear that $\bar{x} \in S(x_{k+1}^*)$ and it completes the proof. $\square$

Algorithm:

Step 0.
- Set $k = 1$, $\beta = 0.1$, $\alpha = 1$, $\epsilon = 10^{-2}$, the maximum number of directions D, the directions d_i for $i = 1, 2, \ldots, D$, the number of maximum iterations N and determine boundary of Ω.
- Choose the function $\varphi(t)$.

Step 1. Find the $k-$th local minimizer x_k^* of the objective function $f(x)$ starting from the any random point x_0.

Step 2. Construct the the function

$$\tilde{\phi}(x, x_k^*, \beta, \alpha) = f(x_k^*) + (f(x) - f(x_k^*))\, \tilde{\chi}_{A_k}(x, \beta) + \alpha\varphi(\|x - x_k^*\|^2),$$

Step 3. Set $i = 1$ and use $x_0 = x_k^* + \epsilon d_i$ as a starting point and find the minimizer of $\tilde{\phi}(x, x_k^*, \beta, \alpha)$ and denote it as x_s.

Step 4. If $x_s \in \Omega$, then go to Step 5; otherwise, go to Step 6.

Step 5. Take $x_0 = x_s$ and go to Step 1.

Step 6. If $i \geq D$ or $k \geq N$ stop the algorithm and take the global minimizer $x^* = x_k^*$ otherwise set $i = i + 1$ and go to Step 3.

If each time different local minimizers are found then by Assumption 2 after the finite number of iterations the algorithm will find a global minimizer of the function f. Then some large number $N > 0$ can be determined as a maximum number of iterations used as a stopping criteria. The function $\tilde{\phi}(x, x_k^*, \beta, \alpha)$ needn't to have a local maximizer at x_k^* but it may have a local maximizer at any point in Ω. The parameters α and β are determined in the beginning and they are not changed in the loop. For local search, any of local solver can be used. We use Quasi Newton Method as a local solver.

4. Numerical examples

In this section, we apply our algorithm to test problems. The proposed algorithm is programmed in Matlab R2011A. For these tables we use some symbols in order to abbreviate the expressions. The iteration number is denoted by k, the starting point by x_0, the local minimum point of the $k-$th iteration by x_k^*, the value of f at a local minimum point x_k^* by f_k^*. We call our algorithm as EDA, the algorithm proposed in [8] is called by CDFA. We use the following test problems in Table 1 which are taken from [8].

TABLE 1. The list of test problems

Problem No.	Function Name	Dimension n	Region	Optimum value
1	Two dimensional function $c = 0.5c = 0.2$ and 0.05	2	$[-3,3]^2$	0
2	3-hump back Camel function	2	$[-3,3]^2$	0
3	6-hump back Camel function	2	$[-3,3]^2$	-1.0316
4	Treccani function	2	$[-3,3]^2$	0
5	Goldstein-Price function	2	$[-3,3]^2$	3.0000
6	Shubert function	2	$[-10,10]^2$	-186.73091
7	Shekel function	4	$[0,10]^4$	-10.1532
8	$n-$dimensional function	$7, 10$	$[-10,10]^n$	0

TABLE 2. Result for Problem 1 for $c = 0.2,\ 0.5$ and 0.05

		EDA		CDFA	
k	x_0	x_k^*	f_k^*	x_k^*	f_k^*
1		$(3.7387, -1.2649)$	0.6165	$(5.7221, -1.8806)$	2.5070
2	$(6, -2)$	$(2.7380, -0.7884)$	0.0887	$(3.7387, -1.2649)$	0.6165
3		$(1.8784, -0.3458)$	$1.8135e-014$	$(1.5909, -0.2703)$	$2.8126e-009$
1	$(0, 0)$	$(0.0420, -0.0948)$	0.5175	$(0.0420, -0.0948)$	0.5175
2		$(1.5872, -0.2606)$	$5.0239e-014$	$(1.0000, 0)$	$5.7949e-016$
1		$(7.7280, -2.8341)$	6.5031	$(8.7299, -3.2965)$	9.0733
2	$(10, -10)$	$(6.7248, -2.3724)$	4.3943	$(7.7280, -0.4022)$	6.5031
3		$(1.8513, -0.4021)$	$1.7899e-014$	$(1.8513, -0.4021)$	$4.3885e-011$

The results on the total iteration numbers of EDA on Problem 1-8 and the comparison of EDA with CDFA are presented in Tables 2-9. Different conditions for the same test problems are separated by using double lines in the Tables $2, 3, 4, 8$ and 9.

It can be seen from the tables, both EDA and CDFA can find the global optimal solution for all test problems. For Problem 1, both EDA and CDFA use the same number of iterations to find the optimal solution but the EDA presents better solution in terms of function values in general. For Problem 3, the EDA uses fewer number of iterations in comparing with CDFA. For Problem 4, both EDA and CDFA report same number of iterations and close function values. For Problem 5, again EDA and CDFA report same number of iterations and same function values. For Problem 6, the EDA uses fewer number of iterations

TABLE 3. Results of the Problem 2 for $x_0 = (-2, 1)$ and $x_0 = (2, 1)$

		EDA		CDFA	
k	x_0	x_k^*	f_k^*	x_k^*	f_k^*
1	$(-2,-1)$	$(-1.7476, -0.8738)$	0.2986	$(-1.7476, -0.8738)$	0.2936
2		$(-0.0000, -0.0000)$	$9.7984e-014$	$(-0.0000, -0.0000)$	$4.0157e-010$
1	$(2,1)$	$(1.7476, 0.8738)$	0.2986	$(1.7476, 0.8738)$	0.2936
2		$(0.0000, -0.0000)$	$9.7984e-014$	$(-0.0000, -0.0000)$	$3.9567e-010$

TABLE 4. Results of the Problem 3 for $x_0 = (-2, 1)$,$x_0 = (2, -1)$ and $x_0 = (-2, -1)$

		EDA		CDFA	
k	x_0	x_k^*	f_k^*	x_k^*	f_k^*
1	$(-2,1)$	$(0.0898, 0.7127)$	-1.0316	$(-1.6071, 0.5687)$	2.1043
2				$(0.0898, 0.7127)$	-1.0316
1	$(2,-1)$	$(-0.0898, -0.7127)$	-1.0316	$(1.6071, -0.5687)$	2.1043
2				$(-0.0898, -0.7127)$	-1.0316
1	$(-2,-1)$	$(-0.0898, 0.7127)$	-1.0316	$(1.7036, -0.79608)$	-0.21546
2				$(-0.0898, -0.7127)$	-1.0316

TABLE 5. Numerical results of the Problem 4

		EDA		CDFA	
k	x_0	x_k^*	f_k^*	x_k^*	f_k^*
1	$(-1,0)$	$(-1.0000, 0.0000)$	1.0000	$(-1.0000, 0.0000)$	1.0000
2		$(-0.0000, -0.0000)$	$1.4037e-016$	$(-0.0000, -0.0000)$	$2.4048e-017$

TABLE 6. Numerical results of the Problem 5

		EDA		CDFA	
k	x_0	x_k^*	f_k^*	x_k^*	f_k^*
1	$(-1,-1)$	$(-0.6000, -0.4000)$	30.0000	$(-0.6000, -0.4000)$	30.0000
2		$(0.0000, -1.0000)$	3.0000	$(0.0000, -1.0000)$	3.0000

compared to CDFA. For Problem 7, both methods report same number of iterations and same function values. For Problem 8 with $n = 7$ the EDA uses more number of iterations but for $n = 10$ the EDA uses fewer number of iterations compared CDFA. For Problem 2, both EDA and CDFA use the same number of iterations to find the optimal solution but the EDA presents better solution in terms of function values for all conditions.

5. CONCLUSION

We have introduced a new global optimization algorithm based on the Bezier curves. The new method presents satisfactory results on test problems. By comparing our method with the CDFA, we can conclude that our method gives better results than the CDFA in terms of number of iterations and final function values.

On the other hand, we present a new formulation and a new smoothing approach. This approach can be generalized to other non-smooth functions. For future works, we are planing to study on designing smoothing function for the class of non smooth functions.

TABLE 7. Numerical results of the Problem 6

		EDA		CDFA	
k	x_0	x_k^*	f_k^*	x_k^*	f_k^*
1	$(1,1)$	$(2.0467, 2.0467)$	$2.2918e-015$	$(2.0467, 2.0467)$	0
2		$(5.4829, 2.7859)$	-38.2960	$(3.2800, 4.8581)$	-46.511
3		$(5.4829, 4.8581)$	-186.739	$(4.2760, 4.8581)$	-79.411
4				$(5.4892, 4.8581)$	-186.739

TABLE 8. Numerical results of the Problem 7 for $x_0 = (1,1,1,1)$ and $x_0 = (6,6,6,6)$

		EDA		CDFA	
k	x_0	x_k^*	f_k^*	x_k^*	f_k^*
1	(1,1,1,1)	$(1.0001, 1.0002, 1.0001, 1.0002)$	-5.0552	$(1.0001, 1.0002, 1.0001, 1.0002)$	-5.0552
2		$(4.0000, 4.0001, 4.0000, 4.0001)$	-10.1532	$(4.0000, 4.0000, 4.0000, 4.0000)$	-10.1529
1	(6,6,6,6)	$(5.9987, 6.0003, 5.9987, 6.0003)$	-2.6829	$(5.9987, 6.0002, 5.9987, 6.0002)$	-2.6822
2		$(4.0000, 4.0001, 4.0000, 4.0001)$	-10.1532	$(4.0000, 4.0001, 4.0000, 4.0001)$	-10.1529

TABLE 9. Numerical results of the Problem 8 for $n = 7$ and $n = 10$

		EDA		CDFA	
k	x_0	x_k^*	f_k^*	x_k^*	f_k^*
1	$(2, 2, \ldots, 2)$	$(0.0100, 1.0000, 1.0000, 1.0000,$ $1.0000, 1.0000, 1.0000)$	0.4443	$(1.0000, 1.0000, 1.0000, 1.0000,$ $1.0000, 1.0000, 1.0000)$	$2.3538e-013$
2		$(1.0000, 1.0000, 1.0000, 1.0000,$ $1.0000, 1.0000, 1.0000)$	$1.4992e-14$		
1	$(6, 6, \ldots, 6)$	$(2.9798, 2.9948, 2.9949, 2.9949,$ $2.9949, 2.9949, 2.9949, 2.9949,$ $2.9949, 2.9949)$	12.5250	$(1.0101, 0.0103, 0.0103, 0.0104,$ $0.0103, 0.0102, 1.0000, 6.0000,$ $6.0000, 6.0000)$	2.6653
2		$(1.0000, 1.0000, 1.0000, 1.0000,$ $1.0000, 1.0000, 1.0000, 1.0000$ $1.0000, 1.0000)$	$5.6436e-14$	$(1.1615, 1.1651, 0.4418, 0.9258,$ $0.9638, -0.4809, 0.9926, 6.0000,$ $6.0000, 6.0000)$	2.4443
3				$(1.9900, 1.0000, 1.0000, 1.0000,$ $1.0000, 1.0000, 1.0000, 6.0000,$ $6.0000, 6.0000)$	0.4443
4				$(1.0000, 1.0000, 1.0000, 1.0000$ $1.0000, 1.0000, 1.0000, 1.0000,$ $1.0000, 1.0000)$	0

Acknowledgements. The authors would like to thank to the editor and the anonymous referees for their valuable comments and suggestions. This study has been supported by the Teaching Staff Training Project Units of Suleyman Demirel University (OYP-05545-DR-13) and Scientific Research Projects Unit of Suleyman Demirel University (SDU-BAP-4733–YL1–16).

REFERENCES

[1] Farin, G. E., Hoschek, J. and Kim, M-S., *Handbook of Computer Aided Geometric Design*, Elsevier, 2002, pp. 4–6

[2] Ge, R. P., *A filled function method for finding global minimizer of a function of several variables*, Math. Programm., **46** (1990), 191–204

[3] Ge, R. P., *The theory of filled function method for finding global minimizer of a nonlinearly constrained Minimization problem*, J. Comput. Math., **5** (1987), No. 1, 1–9

[4] Horst, R. and Tuy, H., *Global Optimization*, Springer, Verlag, 1993

[5] Huang, Z.-H., Miao, X. H. and Wang, P., *A revised cut-peak function method for box constrained continuous global optimization*, Appl. Math. Comput., **194** (2007), 224–233

[6] Levy, A. V., and Montalvo, A., *The tunneling algorithm for the global minimization of functions*, SIAM J. Sci. Stat. Comput., **6** (1985), No. 2, 15–29

[7] Liu, X., *Several filled functions with mitigators*, Appl. Math. Comput., **133** (2002), 375–387

[8] Lin, H., Gao, Y. and Wang, Y., *A continuously differentiable filled function method for global optimization*, Numer. Algor., **66** (2014), 511–523

[9] Nowak, I., Smolka, J. and Nowak, A. J., *Application of Bezier surfaces to the 3-D inverse geometry problem in continuous casting*, Inv. Prob. Sci. Eng., **19** (2011), No. 1, 75–86

[10] Sahiner, A., Gokkaya, H., Yigit, T., *A new filled function for nonsmooth global optimization*, AIP Conf. Proc., **972** (2012), 1479

[11] Sergeyev, Y. D. and Kvasov, D. E., *A deterministic global optimization using smooth diagonal auxilary functions*, Commun. Nonlinear Sci. Numer. Simulat., **21** (2015), 99–111

[12] Strogin, R. G. and Sergeyev, Y. D., *Global Optimization with Nonconvex Constraints, Sequential and Paralel Algorithms*, Kluwer Academic Publishers, Dordecht, 2000

[13] Wang, Y., Fang, W. and Wu, T., *A cut-peak function method for global optimization*, J. Comput. Appl. Math., **230** (2009), 135–142

[14] Weise, T., *Global Optimization Algorithms – Theory and Application*, http://www.it-weise.de/projects/book.pdf 2009

[15] Wu, Z. Y., Bai, F. S., Lee, H. W. J. and Yang, Y. J., *A filled function method for constrained global optimization*, J. Glob. Opt., **39** (2007), 495–507

[16] Wu, Z. Y., Li, D. and Zhang, S., *Global descent methods for unconstrained global optimization*, J. Glob. Optim., **50** (2011), 379–396

[17] Xu, Y. T., Zhang, Y. and Wang, S. G., *A modified tunneling function method for non-smooth global optimization and its application in artificial neural network*, Appl. Math. Model., **39** (2015), No. 21, 6438–6450

[18] Zhang, L-S., Ng, C-K., Li, D., Tian, W-W., *A New Filled Function Method for Global Optimization*, J. Glob. Opt., **28** (2004), 17–43

[19] Zhigljavsky, A. and Zilinskas, A., *Stochastic Global Optimization*, Springer, New York, 2008

[20] Zilinskas, J., *Brach and bound with simplicial partitions for global optimization*, Math. Model. Anal., 13 (2008), No. 1, 145–159

Department of Mathematics
Suleyman Demirel University
Isparta, Turkey
E-mail address: ahmetnur32@gmail.com, ahmetsahiner@sdu.edu.tr