Investigating the Existence of Function Roots Using Particle Swarm Optimization

Konstantinos E. Parsopoulos
Department of Mathematics,
University of Patras Artificial Intelligence
Research Center (UPAIRC),
University of Patras,
GR–26110 Patras, Greece
kostasp@math.upatras.gr

Michael N. Vrahatis
Department of Mathematics,
University of Patras Artificial Intelligence
Research Center (UPAIRC),
University of Patras,
GR–26110 Patras, Greece
vrahatis@math.upatras.gr

Abstract- The existence of roots of functions is a topic of major significance in Nonlinear Analysis, and it is directly related to the problem of detection of extrema of a function. The topological degree of a function is a mathematical tool of great importance for investigating the existence and the number of roots of a function with certainty. For the computation of the topological degree according to Stenger's theorem, a sufficient refinement of the boundary of the polyhedron under consideration is needed. The sufficient refinement can be computed using the optimal complexity algorithm of Boult and Sikorski. However, the application of this algorithm requires the computation of the infinity norm on the boundary of the polyhedron under consideration as well as an estimation of the Lipschitz constant of the function. In this paper a new technique for the computation of the infinity norm on the polyhedron's boundary as well as for the estimation of the Lipschitz constant is introduced. The proposed approach is illustrated on several test problems and the results are reported and discussed.

1 Introduction

Topological degree is a concept of great importance in Nonlinear Analysis, because, under certain conditions, its value, computed on an open and bounded region $\mathcal{D}$, is equal to the number of zeros of a function within the interior of $\mathcal{D}$. This information can be exploited for the computation of all roots of the function within $\mathcal{D}$ with certainty, by computing sequences of domains with nonzero degree and decreasing diameter, that contain at least one zero of the function. A sufficient refinement of the boundary of the considered region is a prerequisite for the computation of the topological degree using the well known methods of Stenger [40] and Kearfott [14]. Consider the class $\mathcal{F}$ of Lipschitz functions with constant K, defined on the unit cube $\mathcal{C}$,

$$f : \mathcal{C} \rightarrow \mathbb{R}^n,$$

such that for every $f \in \mathcal{F}$ we have

$$\|f(x)\|_\infty \geqslant \delta > 0, \quad \forall x \in \vartheta\mathcal{C},$$

where $\vartheta\mathcal{C}$ is the boundary of $\mathcal{C}$. Then the following holds [39, p. 211]:

(a) if $K/(8\delta) \geqslant 1$, then the function may have zeros in $\mathcal{C}$;

(b) the case $1 \leqslant K/(2\delta) < 4$ is still an open problem;

(c) if $K/(2\delta) < 1$, then the function does not have any zeros in $\mathcal{C}$.

A worst–case lower bound,

$$m^* \simeq 2n \left\lfloor \frac{K}{8\delta} \right\rfloor^{n-1},$$

on the number of function evaluations required to compute the topological degree of any function in the class $\mathcal{F}$, has also been established [39].

In numerous cases the parameters δ and K are not a priori known. Usually, an underestimation of δ and an overestimation of K are sufficient for the computation of the topological degree, but the computation of these estimations is a hard task and in cases where this is possible, it is performed analytically.

In this paper, a new approach based on the Particle Swarm Optimization (PSO) algorithm is proposed. Specifically, PSO is employed to compute efficiently the parameter δ and an estimation of the Lipschitz constant, K. Then, by applying Boult and Sikorski's algorithm, a sufficient refinement of the boundary, which is required for the application of Stenger's theorem to compute the topological degree, is obtained.

PSO is a population–based stochastic optimization method, that requires function values solely. Thus, there is no need for derivatives or for analytic representation of the functions. The selection of PSO was based on its ability on solving efficiently a plethora of problems in science and engineering [2, 3, 8, 13, 18, 19, 20, 22, 25, 26, 30, 31, 32, 35, 36, 44].

Although the 2–dimensional case is mainly considered in this article, a generalization to higher dimensional problems is straightforward. The rest of the paper is organized as follows: in Section 2 some main concepts of the topological degree theory are briefly described and analyzed. In Section 3 the PSO algorithm is described and in Section 4 experimental results are reported. The paper concludes with Section 5.

2 Main Concepts of Topological Degree Theory

Numerous problems in different areas of science and technology can be reduced to the study of a set of solutions of an equation of the form $F(x) = p$, within an appropriate space. Topological degree theory has been developed as means of examining this solution set and obtaining information on the existence of solutions, their number and their nature. Degree theory is widely used in the study of nonlinear differential (ordinary and partial) equations. It is useful, for example, in bifurcation theory and in providing information about the existence and stability of periodic solutions of ordinary differential equations, as well as, the existence of solutions of certain partial differential equations. Several of these applications involve the use of various fixed point theorems which can be provided by means of topological degree [21, 47, 48, 49, 51].

Let the function

$$F_n = (f_1, f_2, \dots, f_n) \colon \overline{\mathcal{D}}_n \subset \mathbb{R}^n \to \mathbb{R}^n,$$

be twice continuously differentiable on the closure of an open and bounded domain $\mathcal{D}_n$ of the n–dimensional Euclidean space $\mathbb{R}^n$, with boundary $\vartheta\mathcal{D}_n$. Suppose further that the solutions of the equation

$$F_n(x) = p, \quad p \in \mathbb{R}^n,$$

where p is a given vector, are not located on $\vartheta\mathcal{D}_n$, and that they are simple, i.e., the determinant, $\det J_{F_n}$, of the Jacobian matrix of F_n at these solutions is non–zero.

Definition 2.1 *The **topological degree of** F_n **at** p **relative to** $\mathcal{D}_n$ is denoted by* $\deg[F_n, \mathcal{D}_n, p]$ *and it is defined by the equation*

$$\deg[F_n, \mathcal{D}_n, p] = \sum_{x \in F_n^{-1}(p) \cap \mathcal{D}_n} \operatorname{sgn}\big(\det J_{F_n}(x)\big), \quad (1)$$

where $\operatorname{sgn}(\psi)$ *defines the well known three valued sign function:*

$$\operatorname{sgn}(\psi) = \begin{cases} -1, & \text{if} \quad \psi < 0, \\ \ \ 0, & \text{if} \quad \psi = 0, \\ \ \ 1, & \text{if} \quad \psi > 0. \end{cases} \quad (2)$$

The topological degree is invariant under changes of the vector p in the sense that, if $q \in \mathbb{R}^n$ is any vector, then it holds that [24, p.157]:

$$\deg[F_n, \mathcal{D}_n, p] \equiv \deg[F_n - q, \mathcal{D}_n, p - q],$$

where $F_n - q$ denotes the mapping $F_n(x) - q$, $x \in \mathcal{D}_n$. Thus, for simplicity, we consider the case where the topological degree is defined at the origin $\Theta_n = (0, \dots, 0)$ in $\mathbb{R}^n$.

The topological degree $\deg[F_n, \mathcal{D}_n, \Theta_n]$ can be represented by the Kronecker integral which is defined as follows:

$$\deg[F_n, \mathcal{D}_n, \Theta_n] = \frac{\Gamma(n/2)}{2\pi^{n/2}} \int\int_{\vartheta\mathcal{D}_n} \cdots \int \frac{\sum_{i=1}^{n} A_i dx_1 \dots dx_{i-1} dx_{i+1} \dots dx_n}{\left({f_1}^2 + {f_2}^2 + \cdots + {f_n}^2\right)^{n/2}}, \quad (3)$$

where A_i is defined by

$$A_i = (-1)^{n(i-1)} \det \Big[F_n \ \frac{\partial F_n}{\partial x_1} \ \cdots \ \frac{\partial F_n}{\partial x_{i-1}} \ \frac{\partial F_n}{\partial x_{i+1}} \ \cdots \ \frac{\partial F_n}{\partial x_n} \Big], \quad (4)$$

where,

$$\frac{\partial F_n}{\partial x_k} = \left(\frac{\partial f_1}{\partial x_k}, \frac{\partial f_2}{\partial x_k}, \dots, \frac{\partial f_n}{\partial x_k} \right),$$

is the k–th column of the determinant $\det J_{F_n}$ of the Jacobian matrix J_{F_n}.

Since $\deg[F_n, \mathcal{D}_n, \Theta_n]$ is equal to the number of zeros of $F_n(x) = \Theta_n$ that give positive determinant of the Jacobian matrix minus the number of zeros that give negative determinant of the Jacobian matrix, the total number $\mathcal{N}^r$ of zeros of $F_n(x) = \Theta_n$ can be obtained by the value of $\deg[F_n, \mathcal{D}_n, \Theta_n]$ if all these zeros have the same sign of the determinant of the Jacobian matrix. Note that, by assumption, all the zeros of $F_n(x) = \Theta_n$ are simple. To this end, Picard [33, 34] considered the following extension of the function F_n and the domain $\mathcal{D}_n$:

$$F_{n+1} = (f_1, \dots, f_n, f_{n+1}) \colon \mathcal{D}_{n+1} \subset \mathbb{R}^{n+1} \to \mathbb{R}^{n+1}, \quad (5)$$

where $f_{n+1} = y \det J_{F_n}$, and $\mathcal{D}_{n+1}$ is the direct product of the domain $\mathcal{D}_n$ with an arbitrary interval of the real y–axis containing the point $y = 0$. Then the zeros of the following system of equations:

$$\begin{cases} f_1(x_1, x_2, \dots, x_n) & = & 0, \\ \qquad \vdots & & \vdots \\ f_n(x_1, x_2, \dots, x_n) & = & 0, \\ y \det J_{F_n}(x_1, x_2, \dots, x_n) & = & 0, \end{cases} \quad (6)$$

are the same as the zeros of $F_n(x) = \Theta_n$ provided that $y = 0$. On the other hand, it is easily seen that the determinant of the Jacobian matrix of Eq. (6) is equal to $[\det J_{F_n}(x)]^2$ which is always nonnegative (positive at the simple zeros). Thus we may conclude the following:

Theorem 2.1 [33, 34]. *The total number* $\mathcal{N}^r$ *of zeros of* $F_n(x) = \Theta_n$ *is given by*

$$\mathcal{N}^r = \deg[F_{n+1}, \mathcal{D}_{n+1}, \Theta_{n+1}], \quad (7)$$

under the hypotheses that F_n *is twice continuously differentiable and that all the zeros are simple and lie in the strict interior of* $\mathcal{D}_{n+1}$.

For the case of complex zeros the following theorem states that the total number of complex zeros can be obtained by the value of the topological degree without Picard's extension:

Theorem 2.2 [52]. *Let* $\mathcal{D}_2 \subset \mathbb{C}$ *be an open bounded region and let* $f : \mathcal{D}_2 \to \mathbb{C}$ *be analytic. Suppose that* f *has no roots on* $\vartheta\mathcal{D}_2$ *and assume that all roots of* f *that lie in* $\mathcal{D}_2$

are simple. Then the total number $\mathcal{N}^r$ of roots of f is equal to $\deg[F_2, \mathcal{D}_2, \Theta_2]$, *where*

$$F_2(x_1,x_2) = (f_1(x_1,x_2), f_2(x_1,x_2)) = \\ = \Big(\Re(f(x_1 + \mathbf{i}\, x_2)), \Im(f(x_1 + \mathbf{i}\, x_2))\Big), \quad (8)$$

and $\Re(z)$, $\Im(z)$, *are the real and the imaginary part of* $z \in \mathbb{C}$, *respectively.*

Since Stenger's remarkable and pioneering work [40], many approaches have been developed and studied to compute the topological degree of a function (see e.g., [1, 5, 6, 14, 15, 41, 42, 43, 50, 52, 53]). Stenger's method expresses the topological degree of a continuous mapping

$$F_n = (f_1, \dots, f_n) \colon \overline{\mathcal{D}}_n \subset \mathbb{R}^n \to \mathbb{R}^n,$$

defined on a bounded domain $\mathcal{D}_n$ in $\mathbb{R}^n$ as a constant times a sum of determinants of various $n \times n$ matrices. The value of the topological degree gives information about the existence of a solution of the equation $F_n(x) = \Theta_n$ within $\mathcal{D}_n$. In particular, Kronecker's theorem [24] states that the equation $F_n(x) = \Theta_n$ has at least one zero in $\mathcal{D}_n$ if the degree is not zero relative to $\mathcal{D}_n$. Although, the value of the topological degree gives qualitative information about the existence of solutions, it does not give quantitative information about the solution values. On the other hand, using the nonzero value of topological degree we are able to obtain upper and lower bounds for solution values. To this end, by computing a sequence of bounded domains with nonzero value of topological degree and decreasing diameter, we are able to obtain a region with arbitrarily small diameter that contains at least one solution of the equation [46, 48].

The accurate computation of the topological degree of a mapping F_n at Θ_n relative to the bounded domain $\mathcal{D}_n$ using Stenger's, or other related methods [14, 41, 42], is heavily based on suitable assumptions, including the appropriate representation of the oriented boundary of $\mathcal{D}_n$. In particular, if the boundary of $\mathcal{D}_n$ can be subdivided in a certain way ("*sufficiently refined*") then Stenger's method gives the exact value of topological degree. Otherwise, heuristic termination criteria have to be used and therefore one cannot be sure that the value of the topological degree is given correctly.

Definition 2.2 [14, 40, 43] *Let Π^n be an n–polyhedron. Suppose that*

$$F_n = (f_1, f_2, \dots, f_n) \colon \Pi^n \subset \mathbb{R}^n \to \mathbb{R}^n,$$

is continuous with $\Theta_n \notin F_n(\vartheta\Pi^n)$. If $n = 1$, $\vartheta\Pi^1$ is said to be ***sufficiently refined relative to*** $\operatorname{sgn} F_1$, *if* $0 \notin F_1(\vartheta\Pi^1)$. *If $n > 1$, $\vartheta\Pi^n$ is said to be* ***sufficiently refined relative to*** $\operatorname{sgn} F_n$, *if $\vartheta\Pi^n$ has been subdivided so that it may be written as a union of a finite number of $(n-1)$–dimensional regions, $Q_1^{n-1}, Q_2^{n-1}, \dots, Q_m^{n-1}$, each consisting of a union of a finite number of $(n-1)$–simplices with pairwise disjoint $(n-1)$–dimensional interiors and having the following properties:*

(a) the interiors of the Q_i^{n-1} are pairwise disjoint and each Q_i^{n-1} is connected;

(b) for each region Q_i^{n-1}, there exists at least one component of F_n, (for example f_{r_i}), that does not vanish on it;

(c) if $f_{r_i} \neq 0$ on Q_i^{n-1}, then ϑQ_i^{n-1} is sufficiently refined relative to $\operatorname{sgn} F_{r_i}$ *where:*

$$F_{n-1}^{r_i} = (f_1, f_2, \dots, f_{r_i-1}, f_{r_i+1}, \dots, f_n).$$

Next we will concentrate on the 2–dimensional case. This case is also very interesting since using the Picard's approach we are able to compute the exact number of roots of 1–dimensional functions in a given interval, by computing the topological degree in $\mathbb{R}^2$ on a Picard's extension [9, 10]. Furthermore, as we have already mentioned, the exact number of complex zeros within a specific region can be obtained by the value of the topological degree in $\mathbb{R}^2$ [52]. Thus, we can state the following:

Definition 2.3 *A segment $[p_i, p_j]$ is defined to be a* ***closed counterclockwise oriented portion*** *of $\vartheta\mathcal{D}_2$ with endpoints p_i and p_j and interior (p_i, p_j). A partition P of $\vartheta\mathcal{D}_2$ is either the empty set or a set $\{p_i\}_{i=1}^{g}$ of counterclockwise ordered points from $\vartheta\mathcal{D}_2$ such that*

$$\vartheta\mathcal{D}_2 = \sum_{i=1}^{g} [p_i, p_{i+1}], \qquad p_{g+1} = p_1. \quad (9)$$

Lemma 2.1 [23]. *A nonempty partition P forms a sufficient refinement of the boundary $\vartheta\mathcal{D}_2$ relative to the sign of a function $F_2 = (f_1, f_2)$ if and only if*

$$(p_i, p_{i+1}) \cap (p_j, p_{j+1}) = \emptyset, \quad \forall i \neq j,$$

and on each $[p_i, p_{i+1}]$, there exists a component of F_2, say f_j, that is of constant sign (i.e., $\neq 0$) on $[p_i, p_{i+1}]$, and the remaining component of F_2 is nonzero at p_i and p_{i+1}.

Stenger [40] proved that, given a sufficient refinement of the boundary $\vartheta\mathcal{D}_2$ of $\mathcal{D}_2$, the topological degree can be computed as:

$$\deg[F_2, \mathcal{D}_2, \Theta_2] = \\ \frac{1}{4} \sum_{i=1}^{g} (-1)^{j_i - 1} \deg\left[f_{j_i+1}, [p_i, p_{i+1}], 0\right] \times \operatorname{sign} f_{j_i}(p_i), \quad (10)$$

where j_i is the index of the component of $F_2 = (f_1, f_2)$ that has constant sign on $[p_i, p_{i+1}]$, $f_3 = f_1$,

$$\deg[f_j, [p_i, p_{i+1}], 0] = \{\operatorname{sign} f_j(p_{i+1}) - \operatorname{sign} f_j(p_i)\}/2,$$

and

$$\operatorname{sign} f_j(p_i) = \begin{cases} 1, & \text{if} \quad f_j(p_i) > 0, \\ -1, & \text{if} \quad f_j(p_i) < 0. \end{cases}$$

Boult and Sikorski proposed an optimal complexity algorithm, in their paper [6], for computing with certainty the topological degree for any function from a class $\mathcal{F}$. For the 2–dimensional case, this class consists of functions

$$F_2 = (f_1, f_2) : \mathcal{B} \to \mathbb{R}^2,$$

defined on the unit square $\mathcal{B}$, which satisfy the Lipschitz condition with constant $K > 0$ and whose infinity norm along the boundary of $\mathcal{B}$ is at least $\delta > 0$. Then the following holds [39, p.194]:

(a) if $K/(4\delta) \geqslant 1$, then the function may have zeros in $\mathcal{B}$;

(b) the case $0.5 \leqslant K/(4\delta) < 1$ is still an open problem;

(c) if $K/(4\delta) < 0.5$, then the function does not have any zeros.

They also established a worst–case lower bound,

$$m^* = 4\left\lfloor \frac{K}{4\delta} \right\rfloor,$$

on the number of function evaluations required to compute the topological degree of any function in the class $\mathcal{F}$, using Stenger's method [40].

If the value of the Lipschitz constant K with respect to $\mathcal{D}_2$ and the infinity norm δ of F_2 along the boundary of $\mathcal{D}_2$ are known and we choose equally spaced points on the boundary of $\mathcal{D}_2$ separated by a distance $1/\lfloor K/(4\delta)\rfloor$, in the infinity norm, then Boult and Sikorski have shown that we are able to evaluate the topological degree with certainty using Stenger's method [6]. This is so because, in this case, a sufficient refinement is obtained. Thus, the values of main interest that have to be computed, are

$$\delta = \min_{x \in \vartheta \mathcal{D}_2} \|F_2(x)\|_\infty, \qquad (11)$$

and

$$K = \max_{\substack{x \neq y \\ x,y \in \mathcal{D}_2}} \frac{\|F_2(x) - F_2(y)\|_\infty}{\|x - y\|_\infty}. \qquad (12)$$

Notice that the value of δ is always positive since the topological degree is not defined in the case where a solution of the equation $F_2(x) = \Theta_2$ lies on the boundary of $\mathcal{D}_2$. This is also true in the Boult and Sikorski approach, since in this case the value of δ is zero and an infinite number of points has to be considered.

In general, the computation of δ and K is a hard task. In the present work, PSO has been employed to compute δ of F_2 along the boundary of $\mathcal{D}_2$, through subsequent minimizations on each side of the rectangle under consideration. Then, the smallest value obtained is considered as the value of δ. Moreover, K is estimated by repeatedly computing, through PSO, the maximum of the fraction of Eq. (12) on x, keeping y fixed, for an arbitrary large number of different $y \in \mathcal{D}_2$.

3 The Particle Swarm Optimization algorithm

Particle Swarm Optimization (PSO) is a stochastic optimization algorithm. More specifically, it belongs to the class of *Swarm Intelligence* algorithms, which are inspired from the social dynamics and emergent behavior that arise in socially organized colonies [4, 17, 18, 31]. The ideas that underlie PSO are not inspired by the evolutionary mechanisms encountered in natural selection, but rather by the social dynamics of flocking organisms, such as swarms and fish schools, which are governed by fundamental rules like nearest–neighbor velocity matching and acceleration by distance [18].

PSO is a population based algorithm, i.e., it exploits a population of individuals to probe promising regions of the search space. In this context, the population is called *swarm* and the individuals (i.e., the search points) are called *particles*. Each particle moves with an adaptable velocity within the search space, and retains a memory of the best position it ever encountered. In the *global* variant of PSO, the best position ever attained by all individuals of the swarm is communicated to all the particles. In the *local* variant, each particle is assigned to a topological neighborhood consisting of a prespecified number of particles. In this case, the best position ever attained by the particles that comprise the neighborhood is communicated among them [12, 18]. A thorough investigation of the convergence properties of PSO can be found in [7, 45].

Assume an n–dimensional search space, $S \subset \mathbb{R}^n$, and a swarm consisting of N particles. The i–th particle is in effect an n–dimensional vector $X_i = (x_{i1}, x_{i2}, \ldots, x_{in})^\top \in S$. The velocity of this particle is also a D–dimensional vector, $V_i = (v_{i1}, v_{i2}, \ldots, v_{in})^\top \in S$. The best previous position encountered by the i–th particle is a point in S, denoted by $P_i = (p_{i1}, p_{i2}, \ldots, p_{in})^\top \in S$. Assume g_i to be the index of the particle that attained the best previous position among all the particles in the neighborhood of the i–th particle, and t to be the iteration counter. Then, the swarm is manipulated by the equations [7, 11, 37, 38]:

$$V_i(t+1) = \chi\Big[wV_i(t) + c_1 r_1\big(P_i(t) - X_i(t)\big) + c_2 r_2\big(P_{g_i}(t) - X_i(t)\big)\Big], \qquad (13)$$

$$X_i(t+1) = X_i(t) + V_i(t+1), \qquad (14)$$

where $i = 1, \ldots, N$; c_1 and c_2 are two parameters called *cognitive* and *social* parameters respectively; r_1, r_2, are random numbers uniformly distributed within $[0, 1]$; and g_i is the index of the particle that attained either the best position of the whole swarm (global version), or the best position in the neighborhood of the i–th particle (local version). The parameters χ and w are called *constriction factor* and *inertia weight* respectively, and they are used independently as mechanisms for the control of the velocity's magnitude, corresponding to two different PSO versions.

The value of the constriction factor is derived analytically [7, 45]. The inertia weight, w, is computed empirically, taking into account that it resolves the trade–off between the global (wide–ranging) and local (nearby) exploration ability of the swarm. A large inertia weight encourages global exploration (moving to previously not encountered areas of the search space), while a small one promotes local exploration, i.e., fine–tuning the current search area. A suitable value for w provides the desired balance between the global and local exploration ability of the swarm, and consequently improves the effectiveness of the algorithm. Experimental results suggest that it is preferable to initialize the inertia weight to a large value, giving priority to global

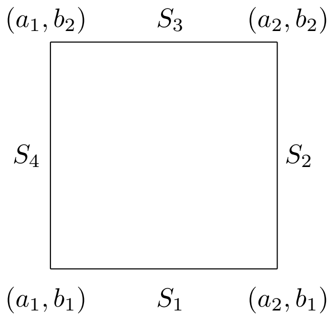

Figure 1: The domain over which the topological degree is computed. The sides $S_1, \ldots, S_4$, are considered counterclockwise.

exploration of the search space, and gradually decrease it, so as to obtain refined solutions [37, 38]. This finding is intuitively very appealing. In conclusion, an initial value of w around 1 and a gradual decline towards 0 is considered a proper choice for w.

In general, the constriction factor version of PSO is faster than the one with the inertia weight, although in some applications its global variant suffers from premature convergence. Proper fine–tuning of the parameters c_1 and c_2, results in faster convergence and alleviation of local minima [16]. As default values, $c_1 = c_2 = 2$ have been proposed, but experimental results indicate that alternative configurations, depending on the problem at hand, can produce superior performance [7, 27, 28, 31].

The initialization of the swarm and the velocities, is usually performed randomly and uniformly in the search space, although more sophisticated initialization techniques can enhance the overall performance of the algorithm [29].

4 Experimental results

In this section, the operation of the proposed technique on different test problems, is illustrated. The global version of the constriction factor PSO variant has been used. The parameters for each test problem were fixed: the size of the swarm was set equal to 10, the maximum number of iterations was 200, and the accuracy for detecting a peak was 10^{-4}. Moreover, the default values $\chi = 0.729$, $c_1 = c_2 = 2.05$ were used [7]. The particles have been constrained in the corresponding region for each test problem. For the estimation of the Lipschitz constant K, a sample of 2000 points has been used. The minimum of the infinity norm, (i.e., the value of δ), the estimation of the Lipschitz constant, K_{est}, as well as the actual value of the Lipschitz constant, K_{actual}, for each test problem are reported in Table 1. The sides of the search space are taken counterclockwise as illustrated in Fig. 1.

TEST PROBLEM 1 [39]. This test problem is defined by

$$F_1 = \begin{cases} x_1^2 - 4x_2 &= 0, \\ x_2^2 - 2x_1 + 4x_2 &= 0, \end{cases} \tag{15}$$

Test Problem	δ	K_{est}	K_{actual}
F_1	0.4375	7.9568	8
F_2	0.5	3.89	4
F_3	0.8284	3.8174	4

Table 1: Experimental results.

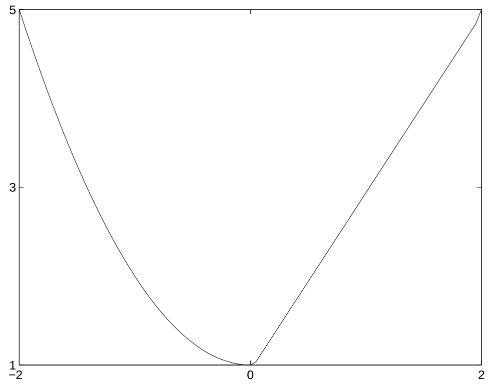

Figure 2: Plot of $\|F_1(x)\|_\infty$ on the side S_1

over the domain $[-2, 2] \times [-0.25, 0.25]$, and it has one zero. The minimum of the infinity norm on the boundary is $\delta = 0.4375$, and the estimation of K is 7.9568.

The plot of $\|F_1(x)\|_\infty$ at each side of the region's boundary is illustrated in Figs. 2–5. The form of the function is quite simple, although, not differentiable, supporting the choice of PSO for the minimization of the infinity norm. To ensure that the global minimizer will be found in cases where a multitude of minimizers are involved in the computation of the minimum of the infinity norm, techniques like *Deflection* and *Stretching* can be combined with PSO [27, 28, 31].

TEST PROBLEM 2 [39]. This test problem is defined by

$$F_2 = \begin{cases} x_1^2 + x_2^2 - 0.5 &= 0, \\ 2x_1x_2 - 0.5 &= 0, \end{cases} \tag{16}$$

over $[0, 1] \times [0, 1]$, and it has one zero of multiplicity 2. The minimum of the infinity norm on the boundary is $\delta = 0.5$, and the estimation of K is 3.89.

TEST PROBLEM 3 [39]. This test problem is defined by

$$F_3 = \begin{cases} x_1^2 - x_2^2 &= 0, \\ 2x_1x_2 &= 0, \end{cases} \tag{17}$$

over $[-1, 1] \times [-1, 1]$, and it has two zeros. The minimum of the infinity norm on the boundary is $\delta = 0.8284$, and the estimation of K is 3.8174.

PSO was able to detect accurately the minimum of the infinity norm on the boundary of the corresponding region for each test problem, as well as to provide an acceptable estimation of the Lipschitz constant of the function.

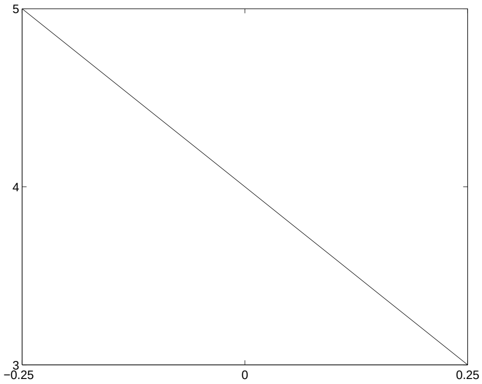

Figure 3: Plot of $\|F_1(x)\|_\infty$ on the side S_2

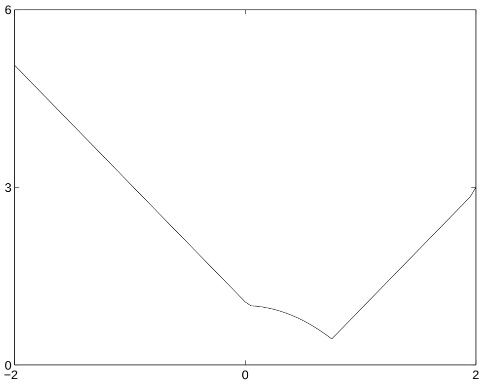

Figure 4: Plot of $\|F_1(x)\|_\infty$ on the side S_3

5 Conclusions

A technique for investigating the existence of function roots has been introduced. This technique exploits the PSO algorithm to compute the infinity norm on the boundary of the region, as well as to estimate the Lipschitz constant of the function under consideration. Then, a sufficient refinement of the boundary is obtained by applying Boult and Sikorski's algorithm, and finally, Stenger's theorem is used to compute the topological degree with certainty. In the case of complex roots, the topological degree is equal to the total number of the function's zeros.

The technique has been illustrated on several test problems with satisfactory results. Further work will involve an investigation of the theoretical properties and dynamics of the proposed technique as well as applications in higher dimensional problems.

Acknowledgment

We thank the anonymous reviewers for their helpful remarks and comments.

Bibliography

[1] O. Aberth. Computation of topological degree using interval arithmetic, and applications. *Math. Comp.*, 62:171–178, 1994.

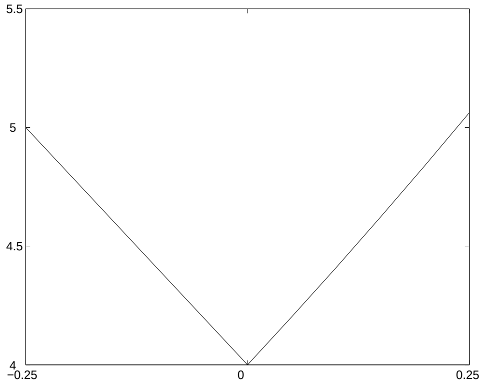

Figure 5: Plot of $\|F_1(x)\|_\infty$ on the side S_4

[2] M.A. Abido. Optimal design of power system stabilizers using particle swarm optimization. *IEEE Trans. Energy Conversion*, 17(3):406–413, 2002.

[3] D.K. Agrafiotis and W. Cedeno. Feature selection for structure–activity correlation using binary particle swarms. *Journal of Medicinal Chemistry*, 45(5):1098–1107, 2002.

[4] E. Bonabeau, M. Dorigo, and G. Théraulaz. *From Natural to Artificial Swarm Intelligence*. Oxford University Press, New York, 1999.

[5] T. Boult and K. Sikorski. Complexity of computing topological degree of Lipschitz functions in n dimensions. *J. Complexity*, 2:44–59, 1986.

[6] T. Boult and K. Sikorski. An optimal complexity algorithm for computing the topological degree in two dimensions. *SIAM J. Sci. Statist. Comput.*, 10:686–698, 1989.

[7] M. Clerc and J. Kennedy. The particle swarm–explosion, stability, and convergence in a multidimensional complex space. *IEEE Trans. Evol. Comput.*, 6(1):58–73, 2001.

[8] A.R. Cockshott and B.E. Hartman. Improving the fermentation medium for *Echinocandin B* production part ii: Particle swarm optimization. *Process Biochemistry*, 36:661–669, 2001.

[9] Kavvadias D.J., F.S. Makri, and M.N. Vrahatis. Locating and computing arbitrarily distributed zeros. *SIAM Journal on Scientific Computing*, 21(3):954–969, 1999.

[10] Kavvadias D.J. and M.N. Vrahatis. Locating and computing all the simple roots and extrema of a function. *SIAM Journal on Scientific Computing*, 17(5):1232–1248, 1996.

[11] R.C. Eberhart and Y. Shi. Comparison between genetic algorithms and particle swarm optimization. In V.W. Porto, N. Saravanan, D. Waagen, and A.E. Eiben, editors, *Evolutionary Programming*, volume VII, pages 611–616. Springer, 1998.

[12] R.C. Eberhart, P. Simpson, and R. Dobbins. *Computational Intelligence PC Tools*. Academic Press, 1996.

[13] P.C. Fourie and A.A. Groenwold. The particle swarm optimization algorithm in size and shape optimization. *Struct. Multidisc. Optim.*, 23:259–267, 2002.

[14] R.B. Kearfott. An efficient degree–computation method for a generalized method of bisection. *Numer. Math.*, 32:109–127, 1979.

[15] R.B. Kearfott, J. Dian, and A. Neumaier. Existence verification for singular zeros of complex nonlinear systems. *SIAM J. Numer. Anal.*, 38:360–379, 2000.

[16] J. Kennedy. The behavior of particles. In V.W. Porto, N. Saravanan, D. Waagen, and A.E. Eiben, editors, *Evolutionary Programming*, volume VII, pages 581–590. Springer, 1998.

[17] J. Kennedy and R.C. Eberhart. Particle swarm optimization. In *Proceedings IEEE International Conference on Neural Networks*, volume IV, pages 1942–1948, Piscataway, NJ, 1995. IEEE Service Center.

[18] J. Kennedy and R.C. Eberhart. *Swarm Intelligence*. Morgan Kaufmann Publishers, 2001.

[19] E.C. Laskari, K.E. Parsopoulos, and M.N. Vrahatis. Particle swarm optimization for integer programming. In *Proceedings of the IEEE 2002 Congress on Evolutionary Computation*, pages 1576–1581, Hawaii (HI), USA, 2002. IEEE Press.

[20] E.C. Laskari, K.E. Parsopoulos, and M.N. Vrahatis. Particle swarm optimization for minimax problems. In *Proceedings of the IEEE 2002 Congress on Evolutionary Computation*, pages 1582–1587, Hawaii (HI), USA, 2002. IEEE Press.

[21] N.G. Lloyd. *Degree Theory*. Cambridge University Press, Cambridge, 1978.

[22] W.Z. Lu, H.Y. Fan, A.Y.T. Leung, and J.C.K. Wong. Analysis of pollutant levels in central hong kong applying neural network method with particle swarm optimization. *Environmental Monitoring and Assessment*, 79:217–230, 2002.

[23] B. Mourrain, M.N. Vrahatis, and J.C. Yakoubsohn. On the complexity of isolating real roots and computing with certainty the topological degree. *J. Complexity*, 18:612–640, 2002.

[24] J.M. Ortega and W.C. Rheinbolt. *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, New York, 1970.

[25] C.O. Ourique, E.C. Biscaia, and J. Carlos Pinto. The use of particle swarm optimization for dynamical analysis in chemical processes. *Computers and Chemical Engineering*, 26:1783–1793, 2002.

[26] K.E. Parsopoulos, E.C. Laskari, and M.N. Vrahatis. Particle identification by light scattering through evolutionary algorithms. In *Proceedings of the 1st International Conference for Mathematics and Informatics for Industry*, pages 97–108, Thessaloniki, Greece, 2003.

[27] K.E. Parsopoulos, V.P. Plagianakos, G.D. Magoulas, and M.N. Vrahatis. Objective function "stretching" to alleviate convergence to local minima. *Nonlinear Analysis, Theory, Methods & Applications*, 47(5):3419–3424, 2001.

[28] K.E. Parsopoulos, V.P. Plagianakos, G.D. Magoulas, and M.N. Vrahatis. Stretching technique for obtaining global minimizers through particle swarm optimization. In *Proceedings of the Particle Swarm Optimization Workshop*, pages 22–29, Indianapolis (IN), USA, 2001.

[29] K.E. Parsopoulos and M.N. Vrahatis. Initializing the particle swarm optimizer using the nonlinear simplex method. In A. Grmela and N.E. Mastorakis, editors, *Advances in Intelligent Systems, Fuzzy Systems, Evolutionary Computation*, pages 216–221. WSEAS Press, 2002.

[30] K.E. Parsopoulos and M.N. Vrahatis. Particle swarm optimization method for constrained optimization problems. In P. Sincak, J. Vascak, V. Kvasnicka, and J. Pospichal, editors, *Intelligent Technologies–Theory and Application: New Trends in Intelligent Technologies*, volume 76 of *Frontiers in Artificial Intelligence and Applications*, pages 214–220. IOS Press, 2002.

[31] K.E. Parsopoulos and M.N. Vrahatis. Recent approaches to global optimization problems through particle swarm optimization. *Natural Computing*, 1(2–3):235–306, 2002.

[32] K.E. Parsopoulos and M.N. Vrahatis. Computing periodic orbits of nondifferentiable/discontinuous mappings through particle swarm optimization. In *Proceedings of the IEEE Swarm Intelligence Symposium*, pages 34–41, Indianapolis (IN), USA, 2003.

[33] E. Picard. Sur le nombre des racines communes à plusieurs équations simultanées. *Journ. de Math. Pure et Appl. (4^e série)*, 8:5–24, 1892.

[34] E. Picard. *Traité d'analyse*. Gauthier–Villars, Paris, 1922.

[35] T. Ray and K.M. Liew. A swarm metaphor for multiobjective design optimization. *Engineering Optimization*, 34(2):141–153, 2002.

[36] A. Saldam, I. Ahmad, and S. Al-Madani. Particle swarm optimization for task assignment problem. *Microprocessors and Microsystems*, 26:363–371, 2002.

[37] Y. Shi and R.C. Eberhart. A modified particle swarm optimizer. In *Proceedings IEEE Conference on Evolutionary Computation*, Anchorage, AK, 1998. IEEE Service Center.

[38] Y. Shi and R.C. Eberhart. Parameter selection in particle swarm optimization. In V.W. Porto, N. Saravanan, D. Waagen, and A.E. Eiben, editors, *Evolutionary Programming*, volume VII, pages 591–600. Springer, 1998.

[39] K.A. Sikorski. *Optimal Solution of Nonlinear Equations*. Oxford University Press, 2001.

[40] F. Stenger. Computing the topological degree of a mapping in $\mathbb{R}^n$. *Numer. Math.*, 25:23–38, 1975.

[41] M. Stynes. An algorithm for numerical calculation of topological degree. *Appl. Anal.*, 9:63–77, 1979.

[42] M. Stynes. A simplification of stenger's topological degree formula. *Numer. Math.*, 33:147–156, 1979.

[43] M. Stynes. On the construction of sufficient refinements for computation of topological degree. *Numer. Math.*, 37:453–462, 1981.

[44] V. Tandon, H. El-Mounayri, and H. Kishawy. Nc end milling optimization using evolutionary computation. *Int. J. Machine Tools & Manufacture*, 42:595–605, 2002.

[45] I.C. Trelea. The particle swarm optimization algorithm: Convergence analysis and parameter selection. *Information Processing Letters*, 85:317–325, 2003.

[46] M.N. Vrahatis. Solving systems of nonlinear equations using the nonzero value of the topological degree. *ACM Trans. Math. Software*, 14:312–329, 1988.

[47] M.N. Vrahatis. A short proof and a generalization of Miranda's existence theorem. *Proc. Amer. Math. Soc.*, 107:701–703, 1989.

[48] M.N. Vrahatis. An efficient method for locating and computing periodic orbits of nonlinear mappings. *J. Comp. Phys.*, 119:105–119, 1995.

[49] M.N. Vrahatis, T.C. Bountis, and M. Kollmann. Periodic orbits and invariant surfaces of 4–D nonlinear mappings. *Inter. J. Bifurc. Chaos*, 6:1425–1437, 1996.

[50] M.N. Vrahatis and K.I. Iordanidis. A rapid generalized method of bisection for solving systems of non-linear equations. *Numer. Math.*, 49:123–138, 1986.

[51] M.N. Vrahatis, H. Isliker, and T.C. Bountis. Structure and breakdown of invariant tori in a 4–D mapping model of accelerator dynamics. *Inter. J. Bifurc. Chaos*, 7:2707–2722, 1997.

[52] M.N. Vrahatis, O. Ragos, T. Skiniotis, F.A. Zafiropoulos, and T.N. Grapsa. The topological degree theory for the localization and computation of complex zeros of Bessel functions. *Numer. Funct. Anal. Optimiz.*, 18:227–234, 1997.

[53] M. Yokoyama. Computing the topological degree with noisy functions. *J. Complexity*, 13:272–278, 1997.