TWMS J. App. and Eng. Math. V.13, N.1, 2023, pp. 86-97

NUMERICAL OPTIMIZATION ALGORITHM BASED ON GENETIC ALGORITHM FOR A DATA COMPLETION PROBLEM

B. JOUILIK[1], J. DAOUDI[2], C. TAJANI[2*], J. ABOUCHABAKA[1], §

Abstract. This work presents numerical optimization algorithm based on genetic algorithm to solve the data completion problem for Laplace's equation. It consists of covering the missing data on the inaccessible part of the boundary from measurements on the accessible part. This problem is known to be severely ill-posed in Hadamard sense; then, regularization methods must be exploited. Metaheuristics are methods inspired by natural phenomena and which have shown their effectiveness in solving several optimization problems in different domains. Thus, adapted genetic operators for real coded genetic algorithm is proposed by formulating the problem into an optimization one. Numerical results with irregular domain are presented showing the efficiency of the proposed algorithm.

Keywords: Inverse problem, cauchy problem, genetic algorithm, finite element method.

AMS Subject Classification: 65N30, 31A25, 65C35.

1. Introduction

A data completion problem is a class of inverse problems which consists of reconstructing the missing data on the inaccessible part of the boundary of the domain, that cannot be evaluated because of physical difficulties or geometric inaccessibility, from the overspecified boundary data on the remaining part. In other words, unlike the direct problem, in the data completion problem the geometry of the domain is determined, but the conditions on the boundary are not all known. The goal is to find the unknown boundary conditions based on the additional information provided on the boundary of domain [1].
This Cauchy problem arises in many areas of engineering and can be considered as a

[1] Faculty of sciences of Kenitra, Ibn Tofail University, Morocco.
e-mail: jouilik-bouchta@hotmail.com; ORCID: https://orcid.org/0000-0001-8077-4134.
e-mail: abouchabaka3@yahoo.fr; ORCID: https://orcid.org/0000-0002-7275-7999.
[2] Department of Mathematics, Polydisciplinary Faculty of Larache, Abdelmalek Essaadi University, Morocco.
e-mail: jamaldaouid5@gmail.com; ORCID: https://orcid.org/0000-0001-8434-2697.
e-mail: chakir_tajani@hotmail.fr; ORCID: https://orcid.org/0000-0002-5749-5193.
[*] Corresponding author.
§ Manuscript received: August 28, 2020; accepted: November 03, 2020.

challenge in many fields of industry such as detection of corrosion, medical imaging, structural mechanics, non-destructive testing of structure $\cdots$ We refer for example to [2,3,4]. However, it is known to be severely ill-posed in Hadamard sense [5]; indeed, experimental measurements are not sufficient to correctly determine the model parameters and a small perturbation of these measures influence the solution which makes its resolution by direct methods very difficult and leads to very unstable solutions. Hence, the investigation of many researchers to develop regularizing methods and efficient numerical approaches. Among them we mention the method of Quasi-reversibility [6], Thikhonov method [7] and the iterative method [8].

The metaheuristic methods as artificial bee colony [9], genetic algorithms (GAs) [10, 11], particle swarm optimization [12], ant colony optimization [13] and the Bat algorithm [14] are methods inspired by natural phenomena and which show their effectiveness in solving several problems in different areas. These techniques do not guarantee the best solution but it is to come as close as possible to the optimum value in a reasonable amount of time. However, the application of these methods, in particular GAs, requires an adaptation of these operators when solving each problem that can influence the quality of the solution and the time required to obtain the optimal solution, which explains the large number of genetic operators developed and adapted to each type of encoding.

Several researchers have used genetic algorithms to solve different inverse problems [15, 16, 17, 18, 19, 20]. Our goal is to adapt the genetic algorithm with a good choice of genetic operators with real coded genetic algorithm to approach the desired solutions for the data completion problem for the Laplace equation. Then; in this paper, the considered inverse problem is formulated as an optimization problem and we investigate the use of genetic algorithm with a real encoding, which showed its efficiency compared to the binary encoding, with adapted crossover and mutation operators, where the obtained direct problem for Laplace's equation is discretised using the finite element method.

The remainder of this paper is organized as follows: Section 2 gives the mathematical formulation of the data completion problem and its formulation on an optimization problem. Section 3 provides a brief review of genetic algorithms and the adapted genetic algorithm for the studied inverse problem. Section 4 presents a number of numerical results showing the effectiveness of the proposed algorithm.

2. Data completion problem for Laplace's equation

2.1. **Mathematical formulation.** The goal in the data completion problem is to find the unknown boundary conditions based on the additional information provided on the accessible part of boundary of the domain.

We consider an open bounded domain $\Omega \subset \mathbb{R}^2$ of boundary $\partial\Omega = \Gamma = \Gamma_0 \cup \Gamma_1$ such that $\Gamma_0 \cap \Gamma_1 = \emptyset$ and $mes(\Gamma_1) \neq 0$.

The data completion problem is to construct the function u solution of the Cauchy problem for the Laplace equation:

$$\begin{cases} -\Delta u = 0 & in \quad \Omega \\ u = f & on \quad \Gamma_0 \\ \partial_n u = g & on \quad \Gamma_0 \end{cases} \tag{1}$$

where, u is the potential (or the temperature) at each point of the domain Ω,
$\partial_n u$ is the normal derivative of u,

f and g are respectively the known values of the function u and its flow on Γ_0.
The considered problem is severely ill-posed in the sense of Hadamard, since existence, unicity and stability are not always ensured. However, for compatible data $(f, g) \in H^{\frac{1}{2}}(\Gamma_0) \times H^{-\frac{1}{2}}(\Gamma_0)$, the problem (1) admits at most one solution [21].

2.2. **Formulation on optimization problem.** Since the function u on the boundary Γ_1 is to be determined, we consider it as a control $\chi \in L^2(\Gamma_1)$ in a direct problem formulation to fit the Cauchy data $g \in L^2(\Gamma_0)$.
Thus, we consider the direct problem:

$$\begin{cases} -\Delta u = 0 & in \quad \Omega \\ \partial_n u = g & on \quad \Gamma_0 \\ u = \chi & on \quad \Gamma_1 \end{cases} \tag{2}$$

We note that if $g \in L^2(\Gamma_0)$ and $\chi \in L^2(\Gamma_1)$, then there is a unique solution $u(g, \chi)$, of the direct problem (2), and we aim to find χ such that:

$$u(g, \chi)_{/\Gamma_0} = f$$

In doing so we attempt to minimize the functional:

$$J(\chi) = \|u(g, \chi)_{/\Gamma_0} - f\|_{L^2(\Gamma_0)}$$

by using Genetic Algorithm approach.

3. Approach genetic for the data completion problem

3.1. **Overview of Genetic algorithms.** Genetic Algorithms (GAs) are most famous Evolutionary Algorithms (EAs) which are inspired from natural evolution and selection. It is essentially a searching method based on the Darwinian principles of biological evolution. Genetic algorithms, primarily developed by Holland [22], have been successfully applied to various optimization problems [23, 24].
GAs search from a population of possible solutions instead of a single one. It uses random operators throughout the process including reproduction, crossover, and mutation. Thus, In a genetic algorithms, a population of individuals (possible solutions) is randomly selected. These individuals are subject to several genetic operators (selection, crossover, mutation, insertion, ..) to produce a new population containing in principle better individuals [10]. This population evolves more and more until a stopping criterion is satisfied and declaring obtaining optimal best solution. The performance of genetic algorithms depends on the choice of operators that will intervene in the production of the new populations.
The fitness or cost function are used to resolve the redundancy has no requirement for continuity in the derivatives, so virtually any fitness function can be selected for optimizing.
Irrespective of the problems treated, genetic algorithms are based on six principles:

- Creation of an initial population formed by a finite number of solutions. Each treated problem has a specific way to encode the individuals of the genetic population. A chromosome (a particular solution) has different ways of being coded: numeric, symbolic, matrix or alphanumeric;
- Definition of an evaluation function (fitness) to evaluate a solution;

- Selection mechanism to generate new solutions are used to identify individuals in a population. There are several methods in the literature, citing the method of selection by rank, roulette, by tournament, random selection, etc.;
- Reproduce the new individuals by using Genetic operators:
  - Crossover operator: It is a genetic operator that combines two chromosomes (parents) to produce a new chromosome (children) with crossover probability P_c ;
  - Mutation operator: It avoids establishing a uniform population unable to evolve. This operator is used to modify the genes of a chromosome selected with a mutation probability P_m;
- Insertion mechanism: to decide who should stay and who should disappear.
- Stopping test: to make sure about the optimality of the solution obtained by the genetic algorithm.

3.2. **Choice of Genetic operators for the optimization problem.** To solve the optimization problem using GAs, it is necessary to adapt the genetic operators, starting with the type of encoding, the crossover operator and the mutation operator.
To implement an GA, the decision variables must be encoded as strings of binary alphabets zero and one [25] or encoded as real numbers [26].

It is khown that the performance of real coded genetic algorithm is superior to binary coded genetic algorithm requiring huge computational time and memory, in particular, for high dimensional problems in which higher degree of precision is desired. Then, in this problem, a real coded GA are used where the decision variables are encoded as real numbers.

For the other operators, we have chosen the Arithmetic crossover [27] and the Power mutation [28], defined us follow:

- **Arithmetic crossover**
  Different types of crossover operators adapted to real coded GA have been developed by several authors that have shown their effectiveness in solving several optimization problems [29]. We opt for this proposed algorithm for the arithmetic crossover.
  In arithmetic crossover, two parents produce two offspring. The offspring are arithmetically represented by:

$$\begin{aligned} y_i^{(1)} &= \alpha_i x_i^{(1)} + (1-\alpha_i)\, x_i^{(2)} \\ y_i^{(2)} &= \alpha_i x_i^{(2)} + (1-\alpha_i)\, x_i^{(1)} \end{aligned} \tag{3}$$

  where α_i are uniform random numbers, say in $[-0.5, 1.5]$ [10].

- **Power Mutation (PM)**
  Several mutation operators are developed with real coded GA. However, we use the power mutation (PM) introduced for real coded genetic algorithms by [28]. The GA with (PM) outperforms other GAs which has shown its effectiveness in comparison with other operators in optimizing different problems.
  This mutation is based on power distribution. It is used to create a solution x' in the vicinity of a parent solution x. A uniform random number t between 0 and 1 is created. Also, a random number s is created which follows the power where the distribution function is given by: $f(x) = px^{p-1} \quad 0 \le x \le 1$

The following formula is used to form the mutated solution:

$$x' = \begin{cases} x - s(x - VarMin), & \text{if} \quad t < \alpha \\ x + s(VarMax - x), & \text{if} \quad t \geq \alpha \end{cases} \tag{4}$$

where $t = \frac{x - VarMin}{VarMax - VarMin}$,

$VarMin$ and $VarMax$ are lower and upper bounds of the decision variable and α is a uniformly distributed random number between 0 and 1 .

3.3. **Approach genetic for the data completion problem.** The approach genetic used to solve the inverse problem is defined following the steps:

- **Step 1:** Given an initial population χ_k^0 , $k = 1, 2, ...n$, with n number of individuals in the generation, solve the problems (P_k) in (5) by finite element method.

$$\begin{cases} -\Delta u = 0 & in \quad \Omega \\ \partial_n u = g & on \quad \Gamma_0 \\ u = \chi_k^0 & on \quad \Gamma_1 \end{cases} \tag{5}$$

- **Step 2:** Evaluate $J(\chi_k^0)$
- **Step 3:** Using $J(\chi_k^0)$ as indicator of fitness of each individual, the next generation χ_k^1 is created by GA with the following rule:
  $\chi_k^1 = M_u.C_r.S_e(\chi_k^0)$
  where S_e the selection, C_r the crossover and M_u the mutation.
- **Step 4:** Go to step 1 with χ_k^1 replacing χ_k^0 and continue.
- **Step 5:** The process continue for $\chi_k^1, i = 1, 2, \cdots, \textit{Max generation}$
  The genetic procedure is identified in this process (see Figure 1).

4. Numerical experiments

The solution domain considered is an example with non-smooth boundary, which is a square $\Omega = (0,1) \times (0,1)$ with a piecewise smooth boundary where $\Gamma = \Gamma_0 \cup \Gamma_1 \cup \Gamma_2 \cup \Gamma_3$; and $mes(\Gamma_0) \neq 0$ and $mes(\Gamma_2) \geq mes(\Gamma_0)$.
The problem is to construct a harmonic function u solution of the Cauchy problem for the Laplace equation and to find the unknown conditions on the inaccessible part Γ_0:

$$\begin{cases} -\triangle u = 0 & & \text{in} \ \ \Omega \\ u = f & & \text{on} \ \ \Gamma_1 \\ u = g, & \frac{\partial u}{\partial n} = h & \text{on} \ \ \Gamma_2 \\ \frac{\partial u}{\partial n} = k & & \text{on} \ \ \Gamma_3 \end{cases} \tag{6}$$

where; $\Gamma_0 = \{0\} \times (0,1)$
$\Gamma_1 = (0,1) \times \{0\}$
$\Gamma_2 = \{1\} \times (0,1)$
$\Gamma_3 = (0,1) \times \{1\}$

We note that f, g, h and k are known functions and can be calculated easily for each considered typical bench-mark test examples, namely the harmonic solution to be retrieved given by:

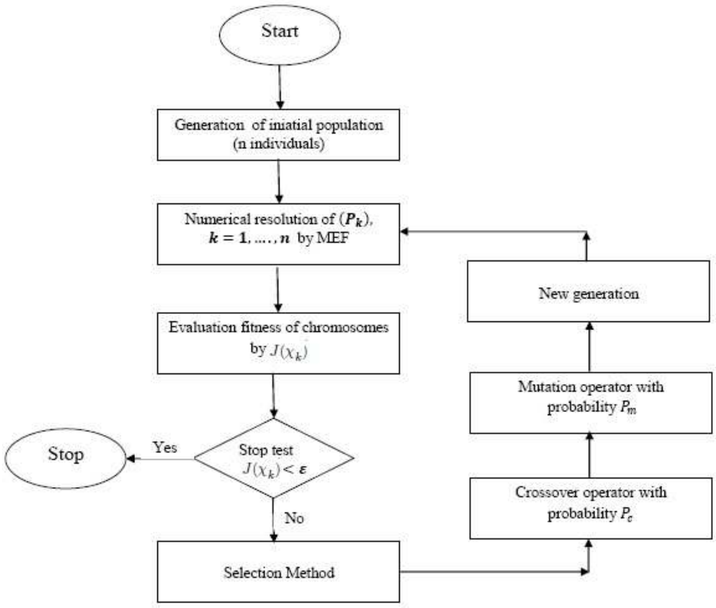


FIGURE 1. Approach genetic for the data completion problem

Example 1: $u(x, y) = x^2 - y^2$

Example 2: $u(x, y) = -yx^2 + y^3/3$

Example 3: $u(x, y) = \cos(x)\cosh(y) + \sin(x)\sinh(y)$

It should be noted that in many cases the form of the solution to be found is not known. Thus, we look for the approximation of the solution based on the polynomial interpolation of the function.

Setting the parameters to find a suitable combination occurring in a GA is a very important task and can be the most difficult. For this, a vast experiment is carried out in the case of our problem with various possible combinations of probability of crossover and probability of mutation.
The final values of the parameters for the considered GA are given as follows :

- Selection: Random
- $VarMin$ and $VarMax$, border of the research space,
- $MaxIt = 500$, maximum number of iterations,
- $nPop = 200$, maximum number of individuals population
- Probability of crossover $P_c = 0.85$
- Probability of mutation $P_m = 0.04$
- Insertion: Elitism

The experiments are done on a intel(R) Core(TM) i5-431OU CPU @ 2.6 GHz machine with 4.00 Go RAM.

Figures 2, 4 and 6 present the evolution of the numerical solution (Dirichlet condition on inaccessible part Γ_0) during the genetic process in comparison with the exact solution for different choices of research domains. These figures show that for all the choices, we can approach the exact solution either for reduced domains as $[-1, 1]$ or for fairly large domains like $[-5, 5]$ and $[-10, 10]$. This amounts to mentioned that even if in the initial population the best individual (the best solution) is quite far from the exact solution, the genetic process allows to achieve a good approximation to the exact solution.

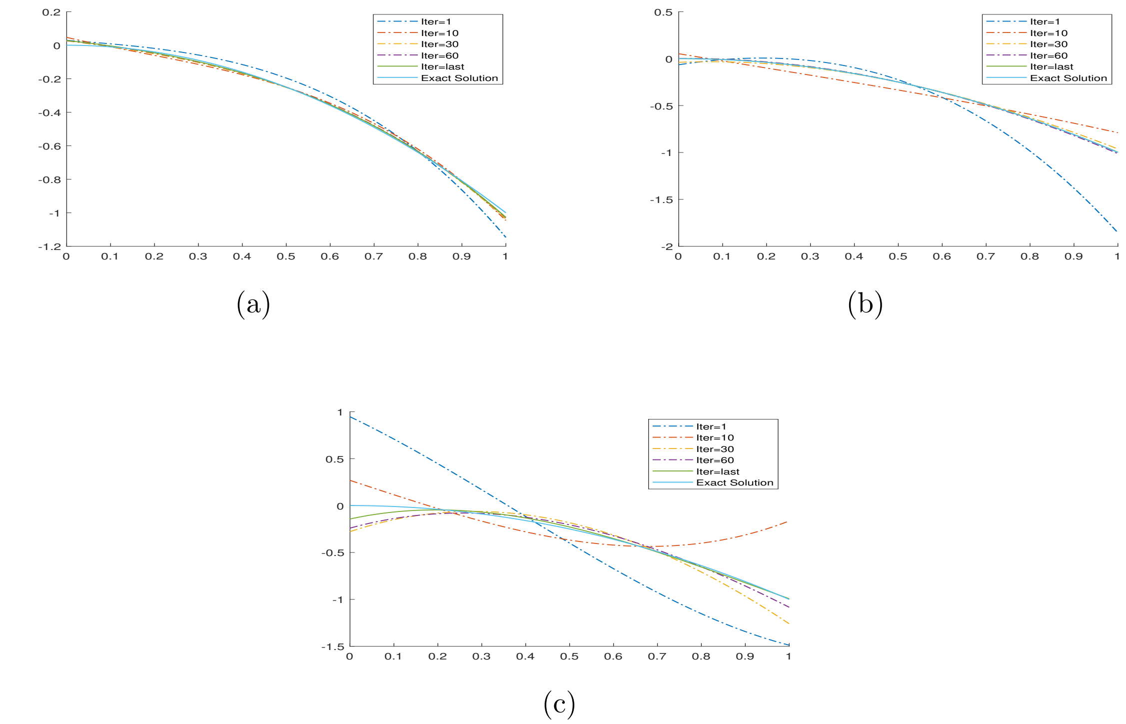
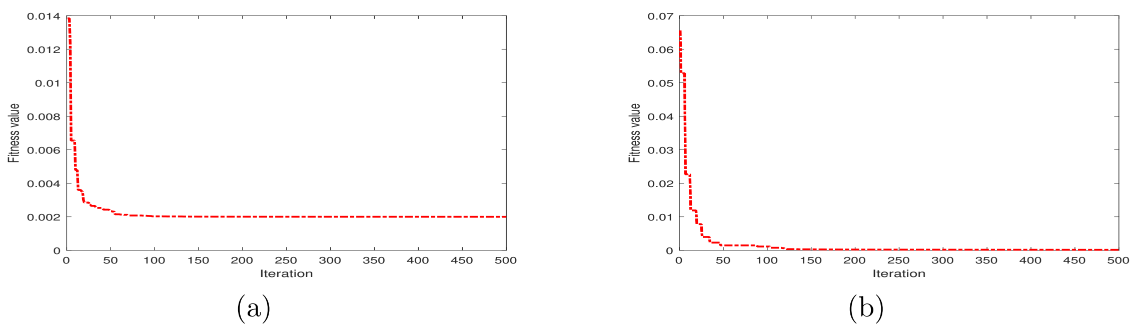
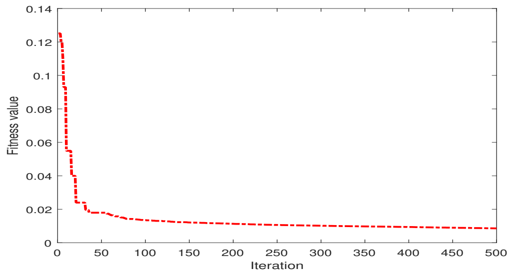


FIGURE 2. Numerical results during genetic process in $[-1, 1]$ (a), $[-5, 5]$ (b) and $[-10, 10]$ (c) for example 1

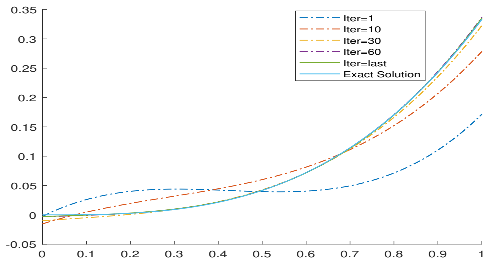


Concerning the figures 3, 5 and 7, they present evaluation of the functional $J(u) = ||u - u_{ex}||_{L^2(\Omega)}$ during genetic process until iteration 500, which decreases in all cases during the genetic process. However, when the choice of the interval is small enough, the algorithm requires less iterations to have a better solution, which amounts to signaling less time for a better solution. In particular, in the case of example 2 with the interval $[-1, 1]$ we obtain an error of 4×10^{-4} in less than 30 iterations, this same error with the

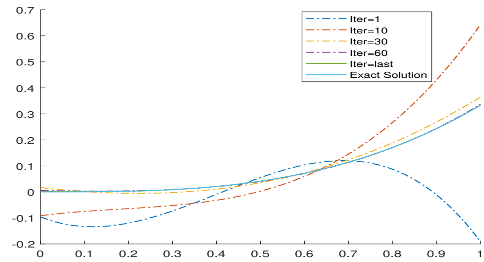


(c)

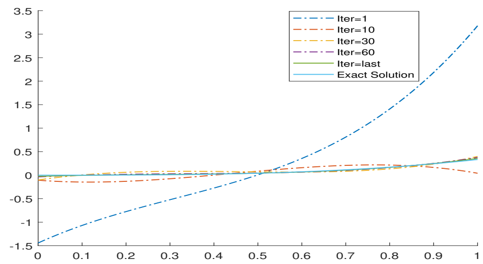
FIGURE 3. Evaluation function during genetic process in $[-1, 1]$ (a), $[-5, 5]$ (b) and $[-10, 10]$ (c) for example 1



(a)

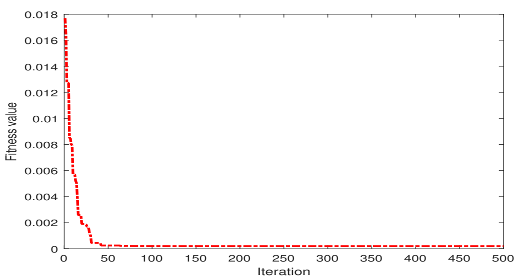


(b)

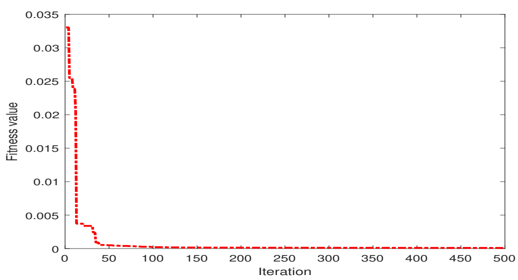


(c)

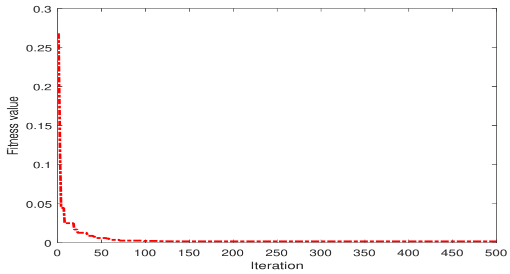
FIGURE 4. Numerical results during genetic process in $[-1, 1]$ (a), $[-5, 5]$ (b) and $[-10, 10]$ (c) for example 2



(a)



(b)

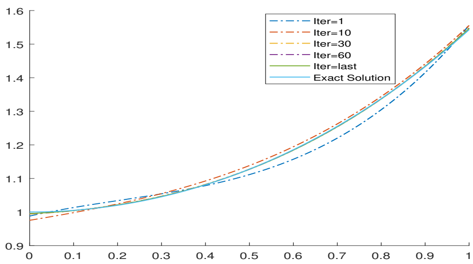


(c)

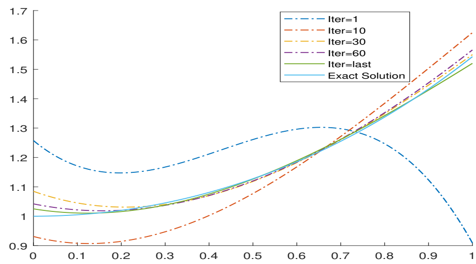
FIGURE 5. Evaluation function during genetic process in $[-1,1]$ (a), $[-5,5]$ (b) and $[-10,10]$ (c) for example 2

interval $[-5,5]$ requires more than 55 iterations. In the case where the considered interval is $[-10,10]$, we obtain just 1.8×10^{-3} after 124 iterations.

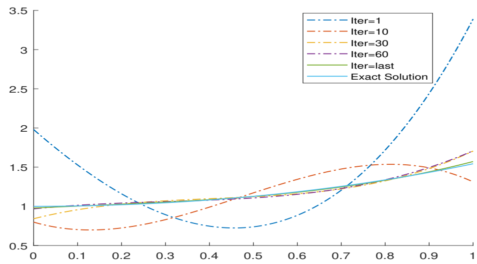
It should be noted that the same results were found with different domains showing the effectiveness of the proposed algorithm to produce good approximations.



(a)



(b)

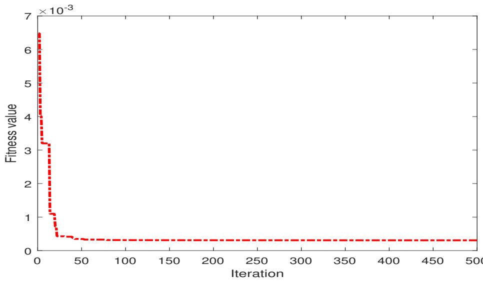


(c)

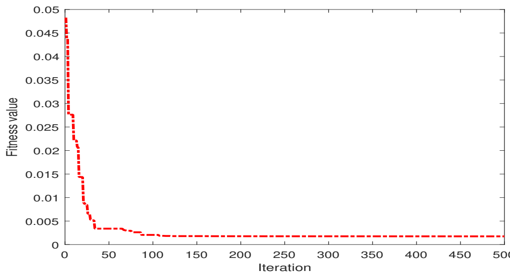
FIGURE 6. Numerical results during genetic process in $[-1,1]$ (a), $[-5,5]$ (b) and $[-10,10]$ (c) for example 3

5. Conclusion

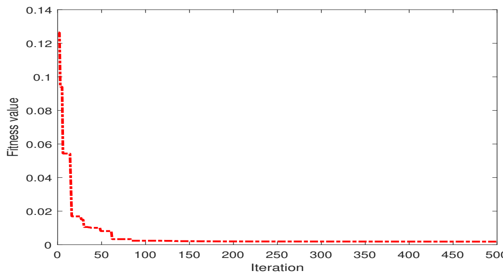
In this paper, a genetic approach is proposed to solve an important inverse problem known to be ill-posed. A formulation in an optimization problem is proposed. Adequate



(a)



(b)



(c)

FIGURE 7. Evaluation function during genetic process in $[-1,1]$ (a), $[-5,5]$ (b) and $[-10,10]$ (c) for example 3

choices of genetic operators adapted to the real-value encoding are made to optimize the functional; in addition to finite element method used to solve the direct problem. The numerical results presented in the case of an irregular domain gave satisfactory results showing the efficiency of the genetic algorithms with adaptation of these operators according to the studied problem to produce good results.
The objective of this study is to focus on the capacity of genetic algorithms to approach the solution of the data completion problem as an ill-posed problem. An improvement in genetic algorithm performance using other crossover and mutation operators is reported in another paper.

References

[1] V. A. Kozlov, V. G. Maz'ya, and A. V. Fomin, (1991), An iterative method for solving the Cauchy problem for elliptic equation, Comput. Maths. Math. Phys., 31(1), pp. 45-52.
[2] V. Isakov, Inverse problems for partial differential equations, applied mathematical sciences, 127, New York: Springer.
[3] G. Inglese, (1997), An inverse problem in corrosion detection, Inverse Problems 13, pp. 977-994.
[4] A. L. Rincon, S. Shimoda, (2016), The inverse problem in electroencephalography using the bidomainmodel of electrical activity, Journal of Neuroscience Methods, 274, pp. 94-105.
[5] J. Hadamard, (1923), Lectures on the Cauchy Problem in Linear Partial Diferential Equations, Yale University Press, New Haven.
[6] R. Lattès and J.L. Lions, (1967), Méthode de Quasi-réversibilité et Applications, Paris: Dunod.
[7] A. N. Tikhonov and V. Y. Arsenin, (1977), Solutions of Ill-Posed Problems, V.H. Winston and Sons, Washington DC.
[8] M. Jourhmane and A. Nachaoui, (1999), An alternating method for an inverse Cauchy problem, Num. Alg., 21, pp. 247-260.

[9] D. Karaboga, and B. Gorkemli, (2011), A combinatorial artificial bee colony algorithm for traveling salesman problem, in: 2011 international symposium on innovations in intelligent systems and applications (INISTA), pp. 50-53.
[10] Z. Michalewicz, (1992), "Genetic algorithms + data structures = evolution programs". Berlin: Springer.
[11] D. Goldberg, (1989), Genetic Algorithm in Search, Optimization, and Machine Learning, Addison Wesley.
[12] J. L. Fernández-Martínez, E. García-Gonzalo, J. P. Fernández Álvarez, H. A. Kuzma, C. O. Menéndez Pérez, (2010), PSO: A powerful algorithm to solve geophysical inverse problems. Application to a 1D-DC resistivity case. J. Appl. Geophys, 71, pp. 13-25.
[13] M. Dorigo, and V. Maniezzo, A. Colorni, (1996), The ant system: optimization by a colony of cooperating agents, in: IEEE transactions on systems, man, and cybernetics– part B, 26, pp. 29-42.
[14] X. S. Yang, (2010), A new metaheuristic bat-inspired algorithm", in Nature inspired cooperative strategies for optimization (NICSO 2010). Springer, pp. 65-74.
[15] X. Yang, (2013), Metaheuristic algorithms for inverse problems, Int. J. Inno. Comp. and Appl., 5(2), pp. 76-84.
[16] N. S. Mera, L. Elliott, D. B. Ingham, (2004), Numerical solution of a boundary detection problerm using genetic algorithrms, Enginiering Analysis with Boundary Elements, 28(4), pp. 405-411.
[17] N. S. Mera, L. Elliott, D. B. Ingham, (2004), A multi-population genetic algorithm approach for solving ill-posed problems, Computational Mechanics, 33, pp. 254-262.
[18] N.S. Mera, L. Elliott and D.B. Ingham, (2003), On the use of Genetic Algorithms for Solving Ill-Posed Problems, Inv. Prob. in Engng, 11(2), pp. 105-121.
[19] A. L. Rincon, M. Bendahmane and B. Ainseba, (2014), Two-step genetic algorithm to solve the inverse problem in electrocardiography for cardiac sources, Comp. Meth. in Bio. and Bio. Eng.: Imaging and Visualization, 2(3), pp. 129-137.
[20] R. Pourgholi, H. Dana, S. H. Tabasi, (2014), Solving an inverse heat conduction problem using genetic algorithm: Sequential and multi-core parallelization approach, Applied Mathematical Modelling, 38, pp. 1948-1958.
[21] C. Tajani, J. Abouchabaka, (2018), On the data completion problem for Laplace's equation, Ann. of the Univ. of Crai., Math. and Comp. Scie.e Ser., 45(1), pp. 11-36.
[22] J. Holland, (1975), Adaptation in natural and artificial systems, University of Michigan Press.
[23] M. Gen, R. Cheng, (2000), Genetic Algorithms and Engineering Optimization, second ed., John Wiley and Sons, New York.
[24] A. I. Lahjouji, C. Tajani, M. Sabbane, (2017), New crossover operator to solve the fixed charge transportation problem, J. of Theor. and Appl. Inf. Tech., 95(8), pp. 1607-1617.
[25] D. E. Goldberg, (1989), Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, New York.
[26] C. Z. Janikow, Z. Michalewicz, (1991), An experimental comparison of binary and floating point representation in genetic algorithms, in: Proceedings of the Fourth International Conference on Genetic Algorithms, Morgan Kaufmann, San Francisco, pp. 31-36.
[27] P. Kaelo, M. M. Ali, (2007), Integrated crossover rules in real coded genetic algorithms, Eur. J. of Oper Res., 176, pp. 60-76.
[28] K. Deep, M. Thakur, (2007), A new mutation operator for real coded genetic algorithms, Appl. Math. and Comput., 193, pp. 211-230.
[29] K. Deep, M. Thakur, (2007), A new crossover operator for real coded genetic algorithms, Appl. Math. and Comput., 188, pp. 895-911.

Bouchta Jouilik is a PhD student in the Faculty of Sciences, University of Ibn Tofail, Morocco. He is interested in inverse problems and their applications.

Jamal Daoudi is PhD student in the Department of Mathematics, PolyDisciplinary Faculty of Larache, Abdelmalek Essaadi University, Morocco. His research interests are in the areas of applied mathematics (inverse problems), numerical analysis, optimization with the metaheuristic approaches.

Chakir Tajani received his PhD in 2012 from Ibn Tofail University in Applied Mathematics. Dr. Tajani is currently a professor in the department of Mathematics, Polydisciplinary Faculty, Abdelmalek Essaadi University, Larache, Morocco. His current research interests are in inverse problems and their applcations, numerical analysis and optimization.

Jaafar Abouchabaka received his MS and PhD. degrees from Mohamed V University in Applied Mathematics. Dr. Abouchabaka is currently a professor in the Department of Computer Science, Faculty of sciences, in Ibn Tofail University, Kenitra, Morocco.