ПРИКЛАДНА МАТЕМАТИКА

UDC 519.6 http://dx.doi.org/10.30970/vam.2024.33.12373

GENETIC PROGRAMMING WITH A METHOD OF FUNDAMENTAL SOLUTIONS FOR SOLVING THE STEADY-STATE INVERSE GEOMETRIC PROBLEM

I. Borachok, A. Marchenko

Ivan Franko National University of Lviv,
1, Universytetska str., 79000, Lviv, Ukraine
e-mails: ihor.borachok@lnu.edu.ua, arkadii.marchenko@lnu.edu.ua

We propose the use of genetic programming to reconstruct the interior boundary of a double-connected domain, based on known Cauchy data of a harmonic function given on the exterior boundary. The considered inverse problem is a nonlinear and ill-posed problem. Genetic programming is a branch of genetic algorithms in which individuals are represented as trees, and tree-specific crossover and mutation operators are used to generate offspring during the evolutionary process. We assume that the unknown inner boundary belongs to the class of star-like boundaries and is defined by an unknown radial function, which is represented by a tree individual.

To evaluate each individual's fitness, a nonlinear regularized functional is introduced. The computation of this functional requires solving the Dirichlet-Neumann problem for the Laplace equation. This boundary value problem is solved numerically using the method of fundamental solutions – a meshless numerical method in which the unknown function is approximated as a linear combination of fundamental solutions, and the collocation method is subsequently applied to determine the unknown coefficients. The proposed approach is straightforward to extend to higher dimensions, making it applicable to both two-dimensional and three-dimensional domains. For both cases, we explicitly provide the distribution of source and collocation points.

The effectiveness and robustness of the method are demonstrated through several numerical experiments using both exact data and data with added random noise. The underlying idea of the method is applicable to any boundary reconstruction problem involving star-like boundaries, provided the fundamental solution of the governing equation is known.

Key words: genetic programming, inverse geometric problem, Laplace equation, method of fundamental solutions, genetic algorithms.

1. Introduction

A method of fundamental solutions (MFS) is a meshless boundary collocation method, a popular choice for numerical solving of the direct and inverse problems for elliptic partial differential equations, in the case where the fundamental solution is known, see [2, 8, 13, 14]. The solution to the problem is sought as a linear combination of the fundamental solutions followed by the application of the collocation method to find the unknown coefficients in the MFS-approximation.

In [4], the application of genetic programming (GP) to the problem of boundary reconstruction is considered. The fitness function in this study is numerically computed using the boundary integral equations method (BIEM). This is a powerful method but requires complex calculations, especially in three dimensional domains, see e.g. [3, 6], therefore the previous study only considers the two dimensional case. We extend the results proposed in [4] by deriving the MFS approach and including three dimensional domains.

Let's consider the investigated problem in more detail. Let $D \subset \mathbb{R}^d$, $d = 2, 3$ be the double-connected domain, bounded with the two simple (without self intersections) closed curves (or surfaces, for $d = 3$) Γ_1 and Γ_2, that belongs to C^2 class. Moreover, the boundary Γ_1 lies in the interior of Γ_2. Firstly, let's consider the direct boundary value problem: find a bounded function u that satisfies the Laplace equation

$$\Delta u = 0 \text{ in } D, \tag{1}$$

and boundary Dirichlet and Neumann conditions

$$u = 0 \text{ on } \Gamma_1 \quad \text{and} \quad \frac{\partial u}{\partial \boldsymbol{\nu}} = g \text{ on } \Gamma_2, \tag{2}$$

where $g \in C^1(\Gamma_2)$ is a given smooth function, $\boldsymbol{\nu}$ is the outward unit normal to boundaries Γ_ℓ, $\ell = 1, 2$. The existence and uniqueness of classical or weak solutions of the problem (1)-(2) are well established, see [16, Chapter 6] or [19, Chapter 4].

Then the inverse problem consists in finding the form of the interior boundary Γ_1 by the additional measure on the boundary Γ_2

$$u = f \text{ on } \Gamma_2, \tag{3}$$

where $f \neq 0$ is a given smooth function. Well known that there exists a unique representation of the Γ_1, for more details we refer [16, Chapter 18]. To summarize, the inverse problem (1)-(3) is to reconstruct the unknown boundary Γ_1 from the given Cauchy data (f, g) on the outer boundary Γ_2, which, as is known, is a non-linear ill-posed problem, since the solution doesn't depend continuously on the data.

The problem has important applications and is widely used in impedance tomography, nondestructive testing, and electrostatics [7, 17]. Therefore, there are already several works based, for example, on BIEM and Newton-type iterative regularization, see, [5, 9, 10, 16, 17], which makes it possible to obtain a sufficiently good reconstruction of the boundary. And also there are already several works where genetic algorithm (GA) is used, see [7, 20].

We suggest using GP instead. It is a GA, in the case when the individual is represented in the form of tree. For each tree-individual we define the fitness function that is used as a selection criteria for transition to the next population. In our case, the fitness is a least-squares penalty functional, which requires the numerical solution of the mixed boundary value problem using MFS. Nowadays, GP is a popular optimization technique that is mainly used to find analytical functions, see some researches [22, 23] and references therein.

For the outline of the work, in the section 2 we introduce GP as the main iterative algorithm for inverse geometric problem. Computation of the fitness function using MFS is given in the section 3. The configuration of the algorithm and results of the some numerical experiments are given in the section 4.

2. Application of GP

GP is a stochastic optimization technique that models the process of the evolution and manages the population of the tree-individuals. Following [11, 21], the algorithm starts with a randomly generated population. Next tree-specific genetic operators are applied to selected individuals to create a new offspring. The next population is created by the r-model [12] using the offspring, while bad candidates die. The process continues until the stop criteria are met. The stopping criteria include a maximum number of consecutive iterations t_{max} without fitness improvement and a chosen accuracy threshold $\delta_{max} > 0$.

We assume that the unknown interior boundary Γ_1 is a star-like curve (or star-like surface for $d = 3$) and has the following parametrization

$$\Gamma_1 = \begin{cases} r(s)(\cos s, \sin s),\ s \in [0, 2\pi], & d = 2, \\ r(\theta, \phi)(\sin\theta\cos\phi, \sin\theta\sin\phi, \cos\theta),\ \theta \in [0, \pi],\ \phi \in [0, 2\pi], & d = 3, \end{cases} \tag{4}$$

where $r : \mathbb{R}^{d-1} \to (0, \infty)$ is a periodic unknown function, representing the radial distance from the origin. By $v = v(r)$ we denote the individual that represents the radial function r in a tree form. For the individual we use the following fitness function

$$J(v(r)) = \|u^{(r)} - f\|^2_{L_2(\Gamma_2)} + \alpha\|u^{(r)}\|^2_{L_2(\Gamma_2)}, \tag{5}$$

where $u^{(r)}$ is a numerical solution of the boundary value problem (1)-(2) for the interior curve Γ_1, given by the radial function r. The penalty functional (5) is constructed from the additional measurement (3). Since the inverse problem is ill-posed, we add a regularization term $\alpha\|u^{(r)}\|^2_{L_2(\Gamma_2)}$, where $\alpha > 0$ is a regularization parameter. The calculation of (5) and numerical solution of the (1)-(2), when both boundaries are known, will be presented in the next section. The best individual is the one in which the fitness function is smaller.

Let F be a set of predefined functions, and T a set of predefined terminals, where $T_c \subset T$ represents constant values and $T_v \subset T$ represents variables. Tree nodes can be from both F (non-terminal nodes) and T (terminal nodes) sets.

Initially, a population of $pop_{size} > 0$ random individuals is generated. To generate the random tree we use the Full-Grow algorithm, see [18]. To obtain offspring, we use the tournament method for the selection of parents. For the tournament selection we use fitnesses computed by (5). New individuals are then created using crossover and mutation operators with selected probabilities $p_{cross}\%$ and $p_{mut}\%$, applied to parents. We use the following mutation operators, see [15].

Subtree mutation: replaces a randomly chosen node (except the root) with a newly generated subtree.

Uplift mutation: selects a subtree and treats it as a new one.

Node replace mutation: randomly replaces a node with another of the same type.

Shrink mutation: replaces a randomly selected subtree with a terminal node.

Terminal mutation: replaces a randomly chosen leaf with a new terminal.

Constant mutation: replaces a random terminal with a new random value from the range $[-1 - CJ(v(r)), 1 + CJ(v(r))]$, where C is a predefined constant and $J(v(r))$ the fitness of the current individual.

We use the following crossover operators.

One point crossover: a classic genetic operator that selects random subtrees from each parent and swaps them.

Uniform crossover: similar to one-point crossover, but swaps common nodes between parents based on a coin toss probability. If the node belongs to the common region and is a function, the entire subtree rooted at that node is inherited.

In addition to these genetic operators, geometric semantic operators, described in [22, 23], are also used.

To filter out the solutions that do not satisfy the basic conditions, such as closed boundary, real values, and interposition in exterior of Γ_2, the penalty values are set.

At the end of the process, the best chromosome r^* is selected as the numerical approximation of the unknown function r and interior boundary Γ_1 is given by (4). In the next section, we focus on computing individual's fitness.

3. The MFS for calculating the fitness function

In this section, we assume that both boundaries $\Gamma_\ell, \ell = 1, 2$ are known and we focus on the fitness (5) computation. Firstly, let's consider the numerical solution of the well-posed mixed boundary value problem (1)-(2). For two-dimensional domains, we assume that boundary curves have following parametrization

$$\Gamma_\ell = \left\{\boldsymbol{\gamma}_\ell(s) = (\gamma_{\ell 1}(s), \gamma_{\ell 2}(s)), s \in [0, 2\pi]\right\}, \ \ell = 1, 2 \tag{6}$$

and for three-dimensional domains

$$\Gamma_\ell = \left\{\boldsymbol{\gamma}_\ell(\theta,\phi) = (\gamma_{\ell 1}(\theta,\phi), \gamma_{\ell 2}(\theta,\phi), \gamma_{\ell 3}(\theta,\phi)),\ \theta \in [0,\pi], \phi \in [0, 2\pi]\right\}, \ \ell = 1, 2. \tag{7}$$

The solution u is approximated as a linear combination of fundamental solutions:

$$u(\boldsymbol{x}) \approx u_n(\boldsymbol{x}) = \sum_{j=1}^{n} \lambda_j \Phi(\boldsymbol{x}, \boldsymbol{y}_j),\ \boldsymbol{x} \in D, \tag{8}$$

where $n \in \mathbb{N},\ \lambda_j \in \mathbb{R},\ j = 1, \dots, N$ are unknown coefficients, Φ is a fundamental solution of the Laplace equation and $\boldsymbol{y}_j \notin \bar{D}$ are chosen source points. The fundamental solution has the following representation

$$\Phi(\boldsymbol{x}, \boldsymbol{y}) = \begin{cases} \dfrac{1}{2\pi} \ln \dfrac{1}{|\boldsymbol{x} - \boldsymbol{y}|},\ x \neq y, & d = 2, \\ \dfrac{1}{4\pi} \dfrac{1}{|\boldsymbol{x} - \boldsymbol{y}|},\ x \neq y, & d = 3, \end{cases} \tag{9}$$

with $|\boldsymbol{x} - \boldsymbol{y}| = \sqrt{\sum_{i=1}^{d} (x_i - y_i)^2}$, $\boldsymbol{x}$, $\boldsymbol{y} \in \mathbb{R}^d$. According to [1], the source points chosen according to the following rule:

$$\boldsymbol{y}_j = \begin{cases} 2\tilde{\boldsymbol{\gamma}}_{2j}, & \text{for even } j, \\ 0.5\tilde{\boldsymbol{\gamma}}_{1j}, & \text{for odd } j, \end{cases} \tag{10}$$

where

$$\tilde{\boldsymbol{\gamma}}_{\ell j} = \begin{cases} \boldsymbol{\gamma}_\ell(t_j),\ t_j = \dfrac{2\pi}{n} j, & d = 2, \\ \boldsymbol{\gamma}_\ell(\theta_j, \phi_j),\ \theta_j = \dfrac{\pi}{n-1} j,\ \phi_j = \dfrac{2\pi}{n} j, & d = 3, \end{cases}$$

for $\ell = 1, 2$, $j = 1, \dots, n$. By applying the collocation method and taking into account boundary conditions (2), we receive the following system of linear equations to determine the unknown coefficients λ_j

$$\begin{cases} \sum_{j=1}^{n} \lambda_j \Phi(\boldsymbol{x}_{1i}, \boldsymbol{y}_j) = 0, & i = 1, \dots, n, \\ \sum_{j=1}^{n} \lambda_j \dfrac{\partial \Phi(\boldsymbol{x}_{2i}, \boldsymbol{y}_j)}{\partial \boldsymbol{\nu}(\boldsymbol{x})} = g(\boldsymbol{x}_i), & i = 1, \dots, n, \end{cases} \tag{11}$$

with collocation points $\boldsymbol{x}_{\ell j} = \tilde{\gamma}_{\ell j}$. The system (11) is over-determined $2n \times n$ and is solved by the least squares method.

Finally, the fitness function (5) is represented in the discrete form as:

$$J(v(r)) = \sum_{i=1}^{n} \left[\sum_{j=1}^{n} \lambda_j^{(r)} \Phi(\boldsymbol{x}_{2i}, \boldsymbol{y}_j) - f(\boldsymbol{x}_{2i}) \right]^2 + \alpha \sum_{j=1}^{n} \left(\lambda_j^{(r)} \right)^2, \tag{12}$$

where $\lambda_j^{(r)}$ are solutions of the (11), for the given r.

4. Algorithm configuration and numerical examples

In this section, we describe the configuration of the GP algorithm and present the results of two numerical experiments for exact and noisy data.

The following parameters of the algorithm are used:

- $n = 64$;
- $\alpha = 1e - 10$ for exact data, or $\alpha = 1e - 5$ for noised data;
- $pop_{size} = 500$;
- in the r-model selection, r is 40% of pop_{size} (see [21]);
- maximum accuracy $\delta_{max} = 10^{-3}$;
- nondecreasing-fitness number of iterations $t_{max} = 100$;
- mutation probability $p_{mut} = 0.3$ and when it is applied, we use the following mutation operators based on the probabilities:
  - subtree mutation 10%;
  - uplift mutation 10%;
  - node replace mutation 25%;
  - shrink mutation 5%;
  - terminal mutation 20%;
  - constant mutation 50%;
  - geometric semantic mutation 40%;
- constant mutation scale $C_{scale} = 0.01$;
- crossover probability $p_{cross} = 0.7$ and when it is applied, we use the following crossover operators based on the probabilities:
  - one point crossover 10%;
  - uniform crossover 70%;
  - geometric semantic crossover 50%;
- predefined functions and terminals:
  - $T_v = \{s\}$ (or $T_v = \{\theta, \phi\}$, for $d = 3$);
  - $T_c = [-10, ..., 10]$;

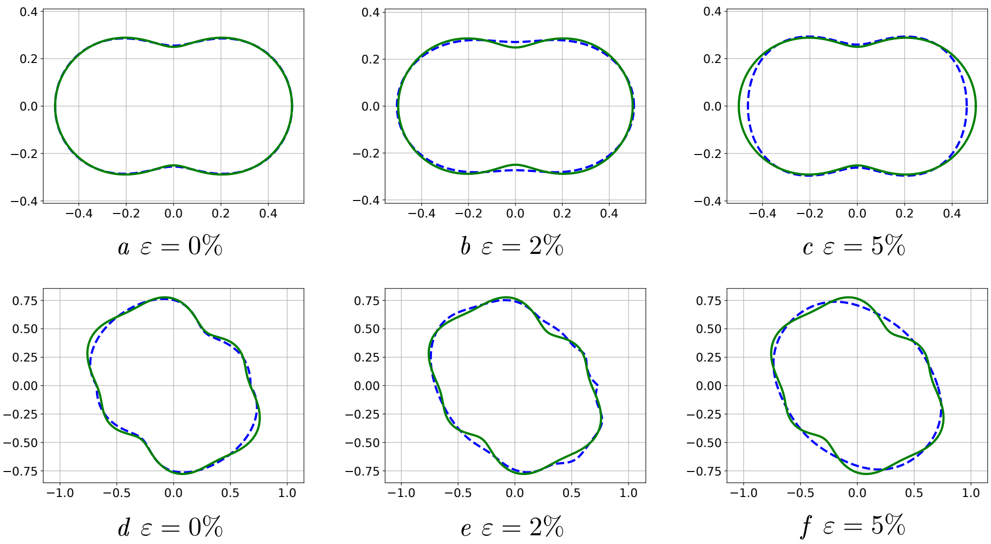


Fig. 1. Exact (green line) and reconstructed (dashed blue line) curves Γ_1 for different noise levels, for examples 1 and 2

– $F = \{+, -, *, protectedDiv, lf\}$,

where $protectedDiv(x,y) = \begin{cases} \dfrac{x}{y}, \ y \neq 0, \\ 1, \ y = 0, \end{cases}$ and $lf(x) = \dfrac{1}{1 + e^{(-x)}}$.

Having described the configuration of the GP, let's consider the results of numerical examples.

4.1. Numerical examples

For all examples we use the same outer boundary Γ_2, defined by

$$\Gamma_2 = \begin{cases} 1.5(\cos(s), \sin(s)), \ s \in [0, 2\pi], & d = 2, \\ 5(\sin\theta\cos\phi, \sin\theta\sin\phi, \cos\theta), \ \theta \in [0, \pi], \phi \in [0, 2\pi], & d = 3. \end{cases}$$

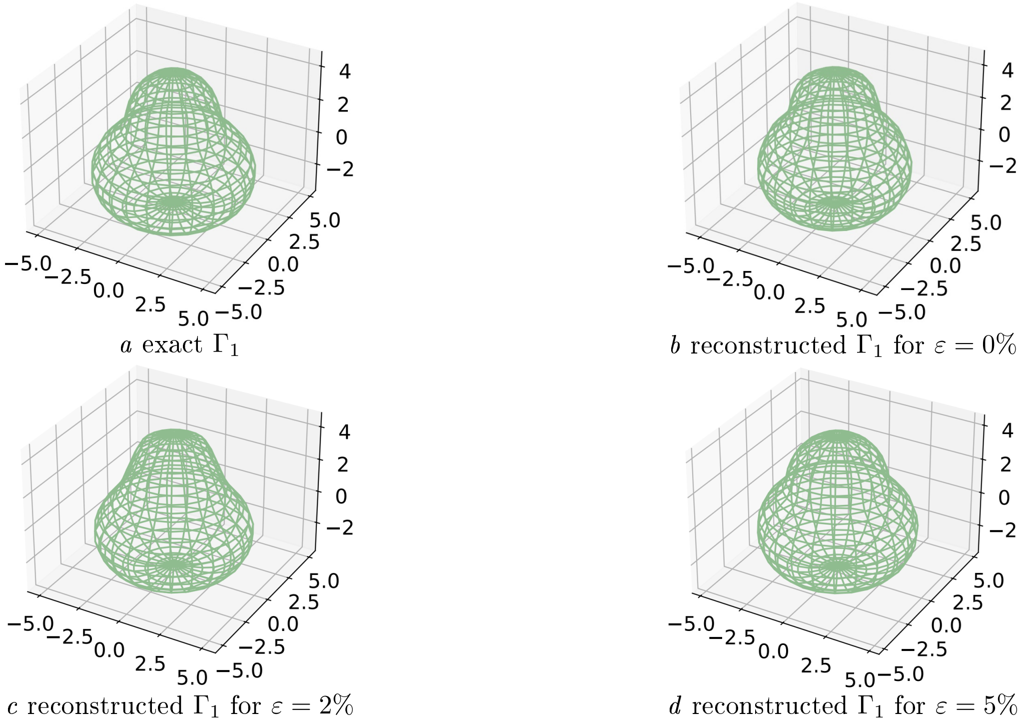
To generate the boundary function f, we solve the mixed boundary value problem (1)-(2) with $g(\boldsymbol{x}) = 2$, $\boldsymbol{x} \in \Gamma_2$ and exact boundary Γ_1, by the MFS and use double number of n, to avoid the inverse crime. The noised data f^ε is generated from the exact boundary function f by the following rule

$$f^\varepsilon = f + \varepsilon(2\eta - 1)\|f\|_{L_2},$$

where ε is a noise level and η is a random value in a range $(0, 1)$.

Firstly, we consider two-dimensional domains. Exact radial functions, the best fitness in the case of exact and noised data for two examples are provided in the table 1 and in the fig. 1 are given exact and reconstructed boundary curves: $1.a) - 1.c)$ for the example 1, $1.d) - 1.f)$ for the example 2, for different noise levels.

Next we consider three-dimensional domains. Exact radial functions, the best fitness in the case of exact and noised data for two examples are provided in the table 2. For



a exact Γ_1 *b* reconstructed Γ_1 for $\varepsilon = 0\%$

c reconstructed Γ_1 for $\varepsilon = 2\%$ *d* reconstructed Γ_1 for $\varepsilon = 5\%$

Fig. 2. Exact and reconstructed surfaces Γ_1 for different noise levels, for example 3

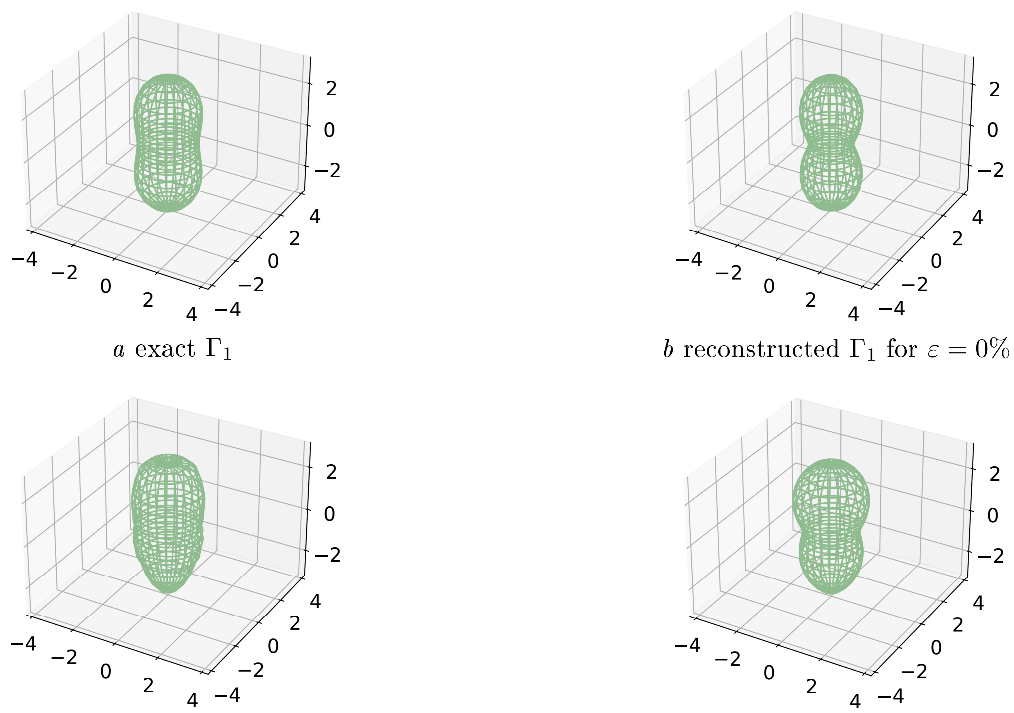
Table 1

Exact radial functions, the best fitness and number of iterations for exact and noisy data for examples 1 and 2 ($d = 2$)

	$r(s),\ s \in [0, 2\pi]$	ε	Iterations	$J(v(r^*))$
Example 1	$\sqrt{(0.5\cos s)^2 + (0.25\sin s)^2}$	0%	428	0.0002435263
		2%	219	0.0031541706
		5%	302	0.1199570379
Example 2	$\sqrt{\cos^3(2s+1) + \dfrac{5}{2+\cos^2 s}}$	0%	357	0.0019720625
		2%	431	0.0313415499
		5%	183	0.5381723422

the first three-dimensional example: in the fig. 2.*a*) the exact surface is presented and in the fig. 2.*b*)-2.*d*) are given reconstructed surfaces, for different noise levels. Similar results are presented in the fig. 3.*a*)-3.*d*), for another radial function, for the example 4.

As can be seen from the results of numerical examples, GP can be used as the algorithm for solving the non-linear ill-posed inverse geometric problem. For exact input data and noised data (up to 5%) the inner boundary is reconstructed, for higher noise levels the results are distorted. Provided results confirm the application of the proposed method. A combination of GP with some classical regularization algorithm can be further



a exact Γ_1 *b* reconstructed Γ_1 for $\varepsilon = 0\%$

c reconstructed Γ_1 for $\varepsilon = 2\%$ *d* reconstructed Γ_1 for $\varepsilon = 5\%$

Fig. 3. Exact and reconstructed surfaces Γ_1 for different noise levels, for example 4.

Table 2

Exact radial functions, the best fitness and number of iterations for exact and noisy data for examples 3 and 4 ($d = 3$)

	$r(\theta, \phi)$, $\theta \in [0; \pi]$, $\phi \in [0; 2\pi]$	ε	Iterations	$J(v(r^*))$
Example 3	$2 + \sqrt{(4.25 + 3\cos(3\theta))}$	0%	256	0.1826017664
		2%	188	0.2019086523
		5%	403	1.3745796331
Example 4	$2\sqrt{\cos(2\theta) + \sqrt{2 - \sin^2(2\theta)}}$	0%	294	0.1288771231
		2%	382	0.3286076847
		5%	236	1.5561072596

explored to obtain a better accuracy.

5. Conclusions

This study presents a novel application of GP with MFS for reconstructing the inner boundary of a double-connected domain from the given Cauchy data on the outer boundary. To solve the direct problem, the MFS method is used, which allows us to easily extend the algorithm to the case of three-dimensional domains, compared to our previous work [4], where the BIEM method was used (only in two-dimensional domains). In addition, the application of MFS results in reduced computational costs for two-dimensional domains compared to BIEM.

References

1. Alves C.J.S. On the choice of source points in the method of fundamental solutions / C.J.S. Alves // Eng. Anal. Bound. Elem. – 2009. – Vol. 33. – P. 1348–1361.
2. Bogomolny A. Fundamental solutions method for elliptic boundary value problems / A. Bogomolny // SIAM J. Numer. Anal. – 1985. – Vol. 22. – P. 644–669.
3. Borachok I. Numerical solution of a Cauchy problem for Laplace equation in 3-dimensional domains by integral equations / I. Borachok, R. Chapko, B.T. Johansson // Inverse Problems in Science and Engineering. – 2016. – Vol. 24 (9). – P. 1550–1568.
4. Borachok I. Genetic programming for the two-dimensional boundary reconstruction problem / I. Borachok, A. Marchenko // Visnyk of the Lviv University. – Ser. Applied Mathematics and Computer Science. – 2023. – Vol. 31. – P. 127–141.
5. Chapko R. On the non-linear integral equation approaches for the boundary reconstruction in double-connected planar domains / R. Chapko, O. Ivanyshyn Yaman, T. Kanafotskyi // Journal of Numerical and Applied Mathematics. – 2016. – Vol. 2 (122). – P. 7–20.
6. Chapko R. On a nonlinear integral equation approach for the surface reconstruction in semi-infinite-layered domains / R. Chapko, O. Ivanyshyn, O. Protsyuk // Inverse Problems in Science and Engineering. – 2013. – Vol. 21. – P. 547–561.
7. Eckel H. Non-linear integral equations for the complete electrode model in inverse impedance tomography / H. Eckel, R. Kress // Applicable Analysis. – 2008. – Vol. 87. – P. 1267–1288.
8. Fairweather G. The method of fundamental solutions for elliptic boundary value problems / G. Fairweather, A. Karageorghis // Adv. Comput. Math. – 1998. – Vol. 9. – P. 69–95.
9. Ivanyshyn O. Nonlinear integral equation methods for the reconstruction of an acoustically sound-soft obstacle / O. Ivanyshyn, B.T. Johansson // J. Integral Equations Appl. – 2007. – Vol. 19. – P. 289–308.
10. Ivanyshyn O. Nonlinear integral equations for solving inverse boundary value problems for inclusions and cracks / O. Ivanyshyn, R. Kress // J. Integral Equations Appl. – 2006. – Vol. 18. – P. 13–38.
11. Goldberg D.E. Genetic Algorithm in Search, Optimisation and Machine Learning / D.E. Goldberg. – Addison-Wesley, Reading, MA, 1989.
12. Holland J.H. Adaptation in Natural and Artificial System / J.H. Holland. – University of Michigan Press, Ann Arbor, 1975.
13. Karageorghis A. A meshless numerical identification of a sound-hard obstacle / A. Karageorghis, D. Lesnic // Engineering Analysis with Boundary Elements. – 2012. – Vol. 36 (7). – P. 1074–1081.
14. Karageorghis A. A survey of applications of the MFS to inverse problems / A. Karageorghis, D. Lesnic, L. Marin // Inv. Pr. Sci. Engn. – 2011. – Vol. 19. – P. 309–336.
15. Koza J.R. Genetic Programming / J.R. Koza, R. Poli. In: Burke E.K., Kendall G. (eds) Search Methodologies. Springer, Boston, MA, 2005.
16. Kress R. Linear Integral Equations, 2nd. ed. / R. Kress. – Springer-Verlag, Berlin Heidelberg New-York, 1999.
17. Kress R. Nonlinear integral equations and the iterative solution for an inverse boundary value problem / R. Kress, W. Rundell // Inv. Probl. – 2005. – Vol. 21. – P. 1207–1223.
18. Luke S. Issues in Scaling Genetic Programming: Breeding Strategies, Tree Generation, and Code Bloat / S. Luke. – [Dissertation] University of Maryland, 2000.

19. McLean W. Strongly Elliptic Systems and Boundary Integral Equations, Cambridge University Press / W. McLean. – Cambridge, 2000.
20. Mera N.S. On the use of Genetic Algorithms for Solving Ill-Posed Problems / N.S. Mera, L. Elliott, D.B. Ingham // Inverse Problems in Engineering. – 2010. – Vol. 11. – P. 105–121.
21. Michalewicz Z. Genetic Algorithms + Data Structures = Evolution Programs, 3rd ed. / Z. Michalewicz. – Springer-Verlag, Berlin. – 1996.
22. Moraglio A. Geometric Semantic Genetic Programming / A. Moraglio, K. Krzysztof, J. Colin // Parallel Problem Solving from Nature – PPSN XII. Lecture Notes in Computer Science. – Vol. 7491. Springer, Berlin, Heidelberg. – 2012.
23. Nadizar G. Geometric semantic GP with linear scaling: Darwinian versus Lamarckian evolution / G. Nadizar, B. Sakallioglu, F. Garrow, S. Silva, L. Vanneschi // Genet Program Evolvable Mach. – 2024. – Vol. 25 (17).

Article: received 02.10.2024
revised 24.10.2024
printing adoption 14.11.2024

ГЕНЕТИЧНЕ ПРОГРАМУВАННЯ З МЕТОДОМ ФУНДАМЕНТАЛЬНИХ РОЗВ'ЯЗКІВ ДЛЯ РОЗВ'ЯЗУВАННЯ СТАЦІОНАРНОЇ ОБЕРНЕНОЇ ГЕОМЕТРИЧНОЇ ЗАДАЧІ

І. Борачок, А. Марченко

Львівський національний університет імені Івана Франка,
вул. Університетська 1, Львів, 79000, Україна
e-mails: ihor.borachok@lnu.edu.ua, arkadii.marchenko@lnu.edu.ua

Пропонуємо використання генетичного програмування для реконструкції внутрішньої межі двозв'язної області на основі відомих даних Коші гармонічної функції, заданих на зовнішній межі. Розглянута обернена задача є нелінійною і некоректною. Генетичне програмування – це розділ генетичних алгоритмів, у якому індивіди зображені у вигляді дерев, а специфічні для дерева оператори кросоверу та мутації використовують для створення нащадків під час еволюційного процесу. Припускаємо, що невідома внутрішня межа належить до класу зіркоподібних кривих (або поверхонь) і визначається невідомою радіальною функцією, яка зображена індивідом-деревом.

Для оцінки пристосовності кожного індивіда вводиться нелінійний регуляризований функціонал. Обчислення цього функціонала потребує розв'язання задачі Діріхле-Неймана для рівняння Лапласа. Ця крайова задача чисельно розв'язується за допомогою методу фундаментальних розв'язків – безсіткового методу, в якому невідома функція апроксимується як лінійна комбінація фундаментальних розв'язків, з послідовним застосуванням методу колокації для визначення невідомих коефіцієнтів. Запропонований підхід можна легко поширити на вищі виміри, що робить його застосовним до двовимірних і до тривимірних областей. Для обох випадків пропонуємо розподіл точок джерела та колокації.

Ефективність і стійкість методу продемонстровано кількома чисельними експериментами з використанням точних даних і даних з додаванням випадкового шуму. Основна ідея методу застосовна до будь-якої задачі реконструкції межі, з класу зіркоподібних меж, якщо відомий фундаментальний розв'язок рівняння задачі.

Ключові слова: генетичне програмування, обернена геометрична задача, рівняння Лапласа, метод фундаментальних розв'язків, генетичні алгоритми.